Bull

Guide de gestion du système
Communications et réseaux

AIX

REFERENCE
86 F2 27EF 03

Bull

Guide de gestion du système
Communications et réseaux

AIX

Logiciel

Juin 2003

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
86 F2 27EF 03

Imprimé en France

Vos suggestions sur la forme et le fond de ce manuel seront les bienvenues. Une feuille destinée à recevoir vos remarques se trouve à la fin de ce document.

Pour commander d’autres exemplaires de ce manuel ou d’autres publications techniques Bull, veuillez utiliser le bon de commande également fourni en fin de manuel.

Marques déposées

Toutes les marques déposées sont la propriété de leurs titulaires respectifs.

AIX® est une marque déposée d’IBM Corp. et est utilisée sous licence.

UNIX est une marque déposée licenciée exclusivement par Open Group.

Linux est une marque déposée de Linus Torvalds.

A propos de ce manuel

Ce manuel s'adresse aux administrateurs système qui gèrent les connexions du système au réseau. Il est nécessaire de connaître le système d'exploitation de base et les sujets abordés dans les manuels *AIX 5L Version 5.2 System Management Guide: Operating System and Devices* et *AIX 5L Version 5.2 System User's Guide: Communications and Networks*.

En commençant par la bibliothèque de documentation AIX 5.2, toutes les informations que ce manuel contenait concernant la sécurité système AIX ou toutes les rubriques relatives à la sécurité ont été retirées. Pour consulter les informations de sécurité, reportez–vous au manuel *AIX 5L Version 5.2 Security Guide*.

Cette édition prend en charge l'édition d'AIX 5L Version 5.2 avec le module de maintenance recommandé 5200–01. Toutes les références spécifiques à ce module de maintenance sont signalées par l'indication *AIX 5.2 avec 5200–01*.

A qui s'adresse ce manuel ?

Ce manuel s'adresse aux administrateurs système effectuant les tâches de gestion du système impliquant des communications dans un réseau.

Conventions typographiques

Les conventions typographiques suivantes sont utilisées dans ce manuel :

Gras	Commandes, mots-clés, fichiers, répertoires et autres éléments dont le nom est prédéfini par le système.
Italique	Paramètres dont le nom ou la valeur est fourni par l'utilisateur.
`Espacement fixe`	Exemples (valeurs spécifiques, texte affiché, code programme), messages système ou données entrées par l'utilisateur.

Distinction majuscules/minuscules dans AIX

La distinction majuscules/minuscules s'applique à toutes les données entrées dans le système d'exploitation AIX. Vous pouvez, par exemple, utiliser la commande **ls** pour afficher la liste des fichiers. Si vous entrez `LS`, le système affiche un message d'erreur indiquant que la commande entrée est introuvable. De la même manière, **FICHEA**, **FiChea** et **fichea** sont trois noms de fichiers distincts, même s'ils se trouvent dans le même répertoire. Pour éviter toute action indésirable, vérifiez systématiquement que vous utilisez la casse appropriée.

ISO 9000

Ce produit a été développé et fabriqué conformément aux procédures de qualité ISO 9000.

Bibliographie

Les manuels suivants complètent la documentation sur les communications.

- *AIX 5L Version 5.2 System Management Guide: Operating System and Devices*
- *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*
- *AIX 5L Version 5.2 System User's Guide: Communications and Networks*
- *AIX 5L Version 5.2 General Programming Concepts: Writing and Debugging Programs*
- *AIX 5L Version 5.2 Network Information Service (NIS and NIS+) Guide*
- *AIX 5L Version 5.2 Commands Reference*
- *AIX 5L Version 5.2 Références et guide d'installation*
- *AIX 5L Version 5.2 Security Guide*

Table des matières

Chapitre 1. Procédures des tâches d'administration réseau

Le présent chapitre contient des instructions sur l'exécution des tâches d'administration réseau courantes :

Mise à niveau vers IPv6 à partir d'une configuration IPv4

Dans ce scénario, vous réalisez une mise à niveau manuelle d'une configuration IPv4 vers IPv6. Le réseau utilisé dans cet exemple est constitué d'un routeur et de deux sous–réseaux. Il y a deux hôtes sur chaque sous–réseau : le routeur et un autre hôte. Vous allez mettre à niveau chaque machine sur ce réseau vers la configuration IPv6. A la fin du scénario, le routeur annonce le préfixe `fec0:0:0:aaaa::/64` sur l'interface réseau `en0` et le préfixe `fec0:0:0:bbbb::/64` sur l'interface réseau `en1`. Vous allez d'abord configurer les machines pour qu'elles prennent en charge de façon temporaire IPv6 à des fins de test. Vous les configurerez ensuite pour qu'elles soient prêtes en configuration IPv6 au moment du démarrage.

Si vous exécutez AIX 5.2 et n'avez pas configuré vos paramètres IPv4, reportez–vous à Mise à niveau vers IPv6 sans configuration d'IPv4 dans AIX 5.2 et versions ultérieures page 1-5.

Etape 1. Configuration des hôtes pour IPv6

Sur les hôtes des deux sous–réseaux, effectuez les actions suivantes :

1. Vérifiez que IPv4 est configuré en tapant la commande suivante :

   ```
   netstat –ni
    La commande affiche un résultat semblable à ce qui suit :
   ```

   ```
   Name  Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs
   Coll
    en0   1500  link#2      0.6.29.4.55.ec      279393     0     2510     0
      0
    en0   1500  9.3.230.64  9.3.230.117         279393     0     2510     0
      0
    lo0   16896 link#1                            913     0      919     0
      0
    lo0   16896 127         127.0.0.1              913     0      919     0
      0
    lo0   16896 ::1                               913     0      919     0
      0
   ```

2. Avec les droits d'accès root, configurez vos paramètres IPv6 en spécifiant la commande suivante :

   ```
   autoconf6
   ```

3. Exécutez à nouveau la commande suivante :

   ```
   netstat –ni
    La commande affiche un résultat semblable à ce qui suit :
   ```

```
Name  Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
 en0   1500  link#2      0.6.29.4.55.ec      279679     0     2658     0
0
 en0   1500  9.3.230.64  9.3.230.117         279679     0     2658     0
0
 en0   1500  fe80::206:29ff:fe04:55ec        279679     0     2658     0
0
 sit0  1480  link#3      9.3.230.117              0     0        0     0
0
 sit0  1480  ::9.3.230.117                        0     0        0     0
0
 lo0   16896 link#1                            2343     0     2350     0
0
 lo0   16896 127         127.0.0.1              2343     0     2350     0
0
 lo0   16896 ::1                               2343     0     2350     0
0
```

4. Lancez les démons **ndpd–host** en tapant la commande suivante :

   ```
   startsrc –s ndpd–host
   ```

L'hôte est désormais prêt pour la configuration IPv6. Répétez cette procédure pour chaque hôte du sous–réseau.

Etape 2. Configuration du routeur pour IPv6

1. Vérifiez que les paramètres IPv4 sont configurés en entrant la commande suivante :

   ```
   netstat –ni
   ```

2. Avec les droits d'accès root, entrez la commande suivante :

   ```
   autoconf6
   ```

3. Configurez manuellement les adresses locales/du site sur les interfaces du routeur appartenant à chacun des deux sous–réseaux en entrant les commandes suivantes :

   ```
   # ifconfig en0 inet6 fec0:0:0:aaaa::/64 eui64 alias
    # ifconfig en1 inet6 fec0:0:0:bbbb::/64 eui64 alias
   ```

   Vous devrez réitérer cette action pour chaque sous–réseau auquel votre routeur envoie des paquets.

4. Pour activer le réacheminement IPv6, entrez la commande suivante :

   ```
   no –o ip6forwarding=1
   ```

5. Pour démarrer le démon **ndpd–router**, saisissez :

   ```
   startsrc –s ndpd-router
   ```

   Le démon **ndpd–router** annonce des préfixes correspondant aux adresses locales que vous avez configurées sur le routeur. Dans ce cas, ndpd–router annonce le préfixe ec0:0:0:aaaa::/64 on en0 et le préfixe fec0:0:0:bbbb::/64 on en1.

Etape 3. Définition d'IPv6 pour une configuration sur les hôtes au démarrage

Votre nouvelle configuration IPv6 est supprimée lorsque vous redémarrez la machine. Pour activer la fonctionnalité d'hôte IPv6 à chaque redémarrage du système, procédez comme suit :

1. Ouvrez le fichier **/etc/rc.tcpip** avec votre éditeur favori.
2. Supprimez la mise en commentaire des lignes suivantes :

   ```
   # Start up autoconf6 process
    start /usr/sbin/autoconf6 ""

   # Start up ndpd-host daemon
    start /usr/sbin/ndpd-host "$src_running"
   ```

Au redémarrage, la configuration IPv6 est définie. Répétez ce processus pour chaque hôte.

Etape 4 : Définition d'IPv6 pour une configuration sur le routeur au démarrage

Votre nouvelle configuration IPv6 est supprimée lorsque vous redémarrez la machine. Pour activer la fonctionnalité de routeur IPv6 à chaque redémarrage du système, procédez comme suit :

1. Ouvrez le fichier **/etc/rc.tcpip** avec votre éditeur favori.
2. Supprimez la mise en commentaire de la ligne suivante :

   ```
   # Start up autoconf6 process
    start /usr/sbin/autoconf6 ""
   ```

3. Ajoutez les lignes suivantes immédiatement à la suite de la ligne dont vous venez de supprimer la mise en commentaire :

   ```
   # Configure site-local addresses for router
    ifconfig en0 inet6 fec0:0:0:aaaa::/ eui64 alias
    ifconfig en1 inet6 fec0:0:0:bbbb::/ eui64 alias
   ```

Dans ce scénario, le réseau comporte uniquement deux sous–réseaux, en0 et en1. Vous devrez ajouter une ligne dans ce fichier pour chaque sous–réseau auquel votre routeur envoie des paquets.

4. Supprimez la mise en commentaire de la ligne suivante :

```
# Start up ndpd-router daemon
 start /usr/sbin/ndpd-router "$src_running"
```

Au redémarrage, la configuration IPv6 est automatiquement lancée.

Mise à niveau vers IPv6 sans configuration d'IPv4 dans AIX 5.2 et versions ultérieures

Le présent scénario indique comment configurer des hôtes et un routeur pour IPv6 sans que les paramètres IPv4 soient configurés. Le réseau utilisé dans cet exemple est constitué d'un routeur et de deux sous–réseaux. Il y a deux hôtes sur chaque sous–réseau : le routeur et un autre hôte. A la fin du scénario, le routeur annonce le préfixe `fec0:0:0:aaaa::/64` sur l'interface réseau `en0` et le préfixe `fec0:0:0:bbbb::/64` sur l'interface réseau `en1`. Vous allez d'abord configurer les machines pour qu'elles prennent en charge de façon temporaire IPv6 à des fins de test. Vous les configurerez ensuite pour qu'elles soient prêtes en configuration IPv6 au moment du démarrage.

Ce scénario part du principe que le fichier **bos.net.tcp.client** est installé.

Pour effectuer une mise à niveau IPv6 avec IPv4 déjà configuré, reportez–vous à Upgrade to IPv6 with IPv4 Configured page 1-2.

Etape 1 : Configuration des hôtes pour IPv6

1. Avec les droits d'accès root, entrez la commande suivante sur chaque hôte du sous–réseau :

   ```
   autoconf6 -A
    Toutes les interfaces adaptées à IPv6 sur le système sont affichées.
   ```

   Remarque : Pour afficher un sous–ensemble d'interfaces, utilisez l'indicateur `-i`. Par exemple, `autoconf6 -i en0 en1` affiche les interfaces `en0` et `en1`.

2. Entrez la commande suivante pour afficher vos interfaces :

   ```
   netstat -ni
    La commande affiche un résultat semblable à ce qui suit :

   Name  Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs
   Coll
    en0   1500  link#3      0.4.ac.17.b4.11        7     0       17     0
      0
    en0   1500  fe80::204:acff:fe17:b411            7     0       17     0
      0
    lo0   16896 link#1                           436     0      481     0
      0
    lo0   16896 127         127.0.0.1            436     0      481     0
      0
    lo0   16896 ::1                              436     0      481     0
      0
   ```

3. Lancez les démons **ndpd–host** en tapant la commande suivante :

   ```
   startsrc -s ndpd-host
   ```

Etape 2 : Configuration du routeur pour IPv6

1. Avec les droits d'accès root, entrez la commande suivante sur l'hôte du routeur :

   ```
   autoconf6 -A
    Toutes les interfaces adaptées à IPv6 sur le système sont affichées.
   ```

   Remarque : Pour afficher un sous–ensemble d'interfaces, utilisez l'indicateur `-i` . Par exemple, `autoconf6 -i en0 en1` affiche les interfaces `en0` et `en1`.

   La commande affiche un résultat semblable à ce qui suit :

```
Name  Mtu   Network     Address             Ipkts Ierrs    Opkts Oerrs
Coll
 en1   1500  link#2      0.6.29.dc.15.45         0     0        7     0
   0
 en1   1500  fe80::206:29ff:fedc:1545             0     0        7     0
   0
 en0   1500  link#3      0.4.ac.17.b4.11         7     0       17     0
   0
 en0   1500  fe80::204:acff:fe17:b411             7     0       17     0
   0
 lo0   16896 link#1                            436     0      481     0
   0
 lo0   16896 127         127.0.0.1             436     0      481     0
   0
 lo0   16896 ::1                               436     0      481     0
   0
```

2. Configurez manuellement les adresses locales/du site sur les interfaces du routeur appartenant à chacun des deux sous–réseaux en entrant les commandes suivantes :

```
# ifconfig en0 inet6 fec0:0:0:aaaa::/64 eui64 alias
 # ifconfig en1 inet6 fec0:0:0:bbbb::/64 eui64 alias
```

   Remarque : Vous devrez réitérer cette action pour chaque sous–réseau auquel votre routeur envoie des paquets.

3. Pour activer le réacheminement IPv6, entrez la commande suivante :

```
no –o ip6forwarding=1
```

4. Pour démarrer le démon **ndpd–router**, saisissez :

```
startsrc –s ndpd–router
```

   Le démon **ndpd–router** annonce des préfixes correspondant aux adresses locales que vous avez configurées sur le routeur. Dans ce cas, ndpd–router annonce le préfixe ec0:0:0:aaaa::/64 on en0 et le préfixe fec0:0:0:bbbb::/64 on en1.

Etape 3. Définition d'IPv6 pour une configuration sur les hôtes au démarrage

Lorsque vous aurez exécuté l'étape 1 pour chaque hôte, la configuration IPv6 sera supprimée lorsque vous redémarrez la machine. Pour activer la fonctionnalité d'hôte IPv6 à chaque redémarrage du système, procédez comme suit :

1. Ouvrez le fichier **/etc/rc.tcpip** avec votre éditeur favori.
2. Supprimez la mise en commentaire des lignes suivantes :

```
# Start up autoconf6 process
 start /usr/sbin/autoconf6 ""

# Start up ndpd-host daemon
 start /usr/sbin/ndpd-host "$src_running"
```

3. Ajoutez l'indicateur `–A` à `start /usr/sbin/autoconf6 ""`:

```
start /usr/sbin/autoconf6 "" –A
```

4. Répétez ce processus pour chaque hôte.

Au redémarrage, la configuration IPv6 est automatiquement lancée.

Etape 4 : Définition d'IPv6 pour une configuration sur le routeur au démarrage

Lorsque vous aurez exécuté l'étape 2 pour le routeur, la configuration IPv6 sera supprimée lorsque vous redémarrez la machine. Pour activer la fonctionnalité de routeur IPv6 à chaque redémarrage du système, procédez comme suit :

1. Ouvrez le fichier **/etc/rc.tcpip** avec votre éditeur favori.
2. Supprimez la mise en commentaire de la ligne suivante :

```
# Start up autoconf6 process
 start /usr/sbin/autoconf6 ""
```

3. Ajoutez l'indicateur `-A` à cette ligne :

   ```
   start /usr/sbin/autoconf6 "" -A
   ```

4. Ajoutez les lignes suivantes immédiatement à la suite de la ligne dont vous venez de supprimer la mise en commentaire :

   ```
   # Configure site-local addresses for router
    ifconfig en0 inet6 fec0:0:0:aaaa::/ eui64 alias
    ifconfig en1 inet6 fec0:0:0:bbbb::/ eui64 alias
   ```

   Dans ce scénario, le réseau comporte uniquement deux sous–réseaux, en0 et en1. Vous devrez ajouter une ligne dans ce fichier pour chaque sous–réseau auquel votre routeur envoie des paquets.

5. Supprimez la mise en commentaire de la ligne suivante :

   ```
   # Start up ndpd-router daemon
    start /usr/sbin/ndpd-router "$src_running"
   ```

6. Exécutez la commande ci–après pour activer le réacheminement IP au moment du démarrage :

   ```
   no -r -o ip6forwarding=1
   ```

Au redémarrage, la configuration IPv6 est automatiquement lancée.

Migration de SNMPv1 vers SNMPv3

Le présent scénario présente une migration classique de SNMPv1 vers SNMPv3.

Dans AIX 5.2, l'agent SNMP par défaut s'exécutant au moment du démarrage du système est la version non chiffrée de SNMPv3. SNMPv3 utilise le fichier **/etc/snmpdv3.conf** comme fichier de configuration. Vous devez faire migrer tous les paramètres que vous avez configurés dans le fichier **/etc/snmpd.conf**, qui est utilisé par SNMPv1 dans AIX 5.1 et versions antérieures, dans le fichier **/etc/snmpdv3.conf**.

Dans ce scénario, les communautés et les interruption configurées dans le fichier **/etc/snmpd.conf** seront migrées vers le fichier **/etc/snmpdv3.conf**. A la fin du scénario, SNMPv3 fournira une fonctionnalité identique à celle proposée par SNMPv1. Si vous n'avez configuré aucun de vos signaux d'interruption ou communautés SNMPv1, vous n'avez pas besoin de suivre cette procédure.

Ce fichier ne contient aucune information sur les fonctions disponibles dans SNMPv3. Pour plus d'informations sur la création d'utilisateurs avec des fonctions de SNMPv3 non disponibles dans SNMPv1, reportez–vous à Création d'utilisateurs dans SNMPv3 page 1-13.

Le fichier suivant est l'exemple de fichier **/etc/snmpd.conf** qui va faire l'objet de la migration. Les communautés suivantes sont configurées : `daniel`, `vasu`, et `david`. Vous devez les faire migrer manuellement.

```
logging         file=/usr/tmp/snmpd.log         enabled
 logging         size=0                          level=0

 community       daniel       0.0.0.0     0.0.0.0         readWrite
1.17.35
 community       vasu         9.3.149.49  255.255.255.255 readOnly   10.3.5
 community       david        9.53.150.67 255.255.255.255 readWrite
1.17.35

 view 1.17.35    udp  icmp  snmp 1.3.6.1.2.1.25
 view 10.3.5     system interfaces tcp icmp

 trap            daniel          9.3.149.49      1.17.35   fe
 trap            vasu            9.3.149.49      10.3.5   fe
 trap            david           9.53.150.67     1.17.35   fe

 smux            1.3.6.1.4.1.2.3.1.2.3.1.1         sampled_password  #
sampled
```

Pour exécuter les étapes de ce scénario, reportez–vous au fichier **/etc/snmpd.conf**. Gardez à votre portée une copie de ce fichier lorsque vous démarrez cette procédure.

Etape 1. Migration des informations de communauté

Les noms de communauté du fichier **/etc/snmpd.conf** sont intégrés aux entrées `VACM_GROUP` du fichier **/etc/snmpdv3.conf**. Chaque communauté doit être placée dans un groupe. Vous accordez ensuite aux groupes les droits d'accès et d'affichage nécessaires.

1. Avec les droits de l'utilisateur racine, ouvrez le fichier **/etc/snmpdv3.conf** avec votre éditeur favori. Recherchez les entrées `VACM_GROUP` dans le fichier.
2. Créez une entrée `VACM_GROUP` pour chaque communauté que vous voulez faire migrer. Si plusieurs communautés doivent partager les mêmes droits d'accès et d'affichage, vous ne devez créer qu'un seul groupe pour elles. Les noms de communauté du fichier **/etc/snmpd.conf** deviennent les valeurs *`securityName`* des entrées `VACM_GROUP`. Dans ce scénario, les entrées suivantes ont été ajoutées pour `vasu`, `daniel`, et `david`:

```
#-------------------------------------------------------------------
# entrées VACM_GROUP
#    Définit un groupe de sécurité (composé d'utilisateurs ou de
communautés)
#    pour le modèle VACM (View-based Access Control Model).
# Format :
#  groupName securityModel securityName storageType
VACM_GROUP group2 SNMPv1  vasu -
VACM_GROUP group3 SNMPv1  daniel -
VACM_GROUP group3 SNMPv1  david -
#-------------------------------------------------------------------
```

- *groupName* peut être la valeur de vote choix, sauf group1.
- *securityModel* reste SNMPv1 car nous migrons les communautés SNMPv1.
- Dans ce scénario, daniel et david partagent les mêmes droits d'affichage et d'accès au fichier **/etc/snmpd.conf**. Par conséquent, ils sont tous les deux membres de group3 dans le fichier **/etc/snmpdv3.conf**. La communauté vasu est placée dans un groupe différent parce ses droits d'accès et d'affichage sont différents de ceux de david et daniel.

Les communautés sont désormais placées dans des groupes.

Etape 2. Migration des informations d'affichage

Les informations d'affichage du fichier **/etc/snmpd.conf** deviennent les entrées COMMUNITY, VACM_VIEW, et VACM_ACCESS dans le fichier **/etc/snmpdv3.conf**. Ces entrées déterminent les droits d'accès et d'affichage pour chaque groupe.

1. Créez les entrées COMMUNITY pour daniel, vasu, et david, en conservant les mêmes adresses IP pour netAddr et netMask que celles indiquées dans le fichier **/etc/snmpd.conf**.

```
#-------------------------------------------------------------------------
# COMMUNITY
#    Définit une communauté pour une sécurité basée sur la communauté.
# Format :
#  communityName securityName securityLevel netAddr netMask storageType
COMMUNITY public    public     noAuthNoPriv 0.0.0.0     0.0.0.0         -
COMMUNITY daniel    daniel     noAuthNoPriv 0.0.0.0     0.0.0.0         -
COMMUNITY vasu      vasu       noAuthNoPriv 9.3.149.49  255.255.255.255 -
COMMUNITY david     david      noAuthNoPriv 9.53.150.67 255.255.255.255 -
#-------------------------------------------------------------------------
```

2. Créez une entrée VACM_VIEW pour chaque objet ou variable MIB auquel chaque groupe a accès. D'après le fichier **/etc/snmpd.conf**, daniel et david ont accès à udp, icmp, snmp, et 1.3.6.1.2.1.25 (sous–arborescence hôte définie dans RFC 1514), et vasu a accès aux interfaces system, interfaces, tcp, et icmp. La migration de ces entrées d'affichage dans le fichier **/etc/snmpdv3.conf** s'effectue comme suit :

```
#-------------------------------------------------------------------
# Entrées VACM_VIEW
#    Définit un ensemble particulier de données MIB, appelé une vue, pour
le
#    Modèle VACM (View-based Access Control Model).
# Format :
#  viewName viewSubtree viewMask viewType storageType

VACM_VIEW group2View        system                    - included -
VACM_VIEW group2View        interfaces                - included -
VACM_VIEW group2View        tcp                       - included -
VACM_VIEW group2View        icmp                      - included -

VACM_VIEW group3View        udp                       - included -
VACM_VIEW group3View        icmp                      - included -
VACM_VIEW group3View        snmp                      - included -
VACM_VIEW group3View        1.3.6.1.2.1.25            - included -
#-------------------------------------------------------------------
```

3. Définissez les droits d'accès aux variables MIB définies dans les entrées VACM_VIEW en ajoutant les entrées VACM_ACCESS. Dans le fichier **/etc/snmpd.conf**, daniel et david ont tous les deux un droit readWrite sur leurs variables MIB, tandis que vasu a le droit readOnly.

   Définissez ces droits en ajoutant les entrées VACM_ACCESS . Dans ce scénario, nous avons donné à group2 (vasu) group2View pour readView, mais lui avons donné – pour writeView car vasu avait le droit readOnly dans le fichier **/etc/snmpd.conf**. Nous avons donné à group3 (daniel et david) group3View pour à la fois readView et writeView car ces groupes avaient un accès readWrite dans **/etc/snmpd.conf**. Voir l'exemple ci–après.

```
#--------------------------------------------------------------------
# Entrées VACM_ACCESS
#    Identifie les droits d'accès accordés aux différents groupe de
sécurité
#    pour le modèle VACM (View-based Access Control Model).
# Format :
# groupName contextPrefix contextMatch securityLevel securityModel
readView writeView notifyView storageType
 VACM_ACCESS  group1 - - noAuthNoPriv SNMPv1  defaultView - defaultView -
VACM_ACCESS  group2 - - noAuthNoPriv SNMPv1  group2View - group2View -
VACM_ACCESS  group3 - - noAuthNoPriv SNMPv1  group3View group3View
group3View -
#--------------------------------------------------------------------
```

Etape 3. Migration des informations d'interruption

Les informations d'interruption du fichier **/etc/snmpd.conf** deviennent les entrées NOTIFY, TARGET_ADDRESS, et TARGET_PARAMETERS dans le fichier **/etc/snmpdv3.conf**. Toutefois, seuls TARGET_ADDRESS et TARGET_PARAMETERS ont besoin d'être migrés.

1. Les adresses IP figurant dans les entrées d'interruption du fichier **/etc/snmpd.conf** sont intégrés aux entrées TARGET_ADDRESS du fichier **/etc/snmpdv3.conf**. Cette ligne spécifie l'hôte vers lequel le signal d'interruption sera envoyé. Vous pouvez définir les entrées targetParams. Dans ce scénario, nous utilisons trapparms1, trapparms2, trapparms3, et trapparms4, qui seront définis dans les entrées TARGET_PARAMETERS .

```
#--------------------------------------------------------------------
# TARGET_ADDRESS
#    Définit l'adresse et les paramètres d'une application de gestion
#    à utiliser pour envoyer des notifications.
# Format :
#  targetAddrName tDomain tAddress tagList targetParams timeout
retryCount storageType
 TARGET_ADDRESS Target1 UDP 127.0.0.1       traptag trapparms1 - - -
TARGET_ADDRESS Target2 UDP 9.3.149.49   traptag trapparms2 - - -
TARGET_ADDRESS Target3 UDP 9.3.149.49   traptag trapparms3 - - -
TARGET_ADDRESS Target4 UDP 9.53.150.67  traptag trapparms4 - - -
#--------------------------------------------------------------------
```

2. Les noms de communauté figurant dans les entrées d'interruption du fichier **/etc/snmpd.conf** sont intégrés aux entrées TARGET_ PARAMETERS du fichier **/etc/snmpdv3.conf**. Les noms de communauté doivent être mappés avec une entrée TARGET_ADDRESS spécifique avec les valeurs targetParams. Par exemple, la communauté daniel est mappée avec trapparms2, qui, sous l'entrée TARGET_ADDRESS est mappée avec l'adresse IP 9.3.149.49. La communauté daniel et l'adresse IP 9.3.149.49 étaient à l'origine une entrée d'interruption dans le fichier **/etc/snmpd.conf**. Voir l'exemple ci–après :

```
#--------------------------------------------------------------------
# TARGET_PARAMETERS
#    Définit les paramètres de traitement des messages et de sécurité
#    à utiliser pour envoyer les notifications à une cible de gestion
particulière.
# Format :
#  paramsName mpModel securityModel securityName securityLevel
storageType
 TARGET_PARAMETERS trapparms1 SNMPv1  SNMPv1  public  noAuthNoPriv -
TARGET_PARAMETERS trapparms2 SNMPv1  SNMPv1  daniel  noAuthNoPriv -
TARGET_PARAMETERS trapparms3 SNMPv1  SNMPv1  vasu    noAuthNoPriv -
TARGET_PARAMETERS trapparms4 SNMPv1  SNMPv1  david   noAuthNoPriv -
#--------------------------------------------------------------------
```

3. Les informations `trapmask` du fichier **/etc/snmpd.conf** ne migrent pas vers le fichier **/etc/snmpdv3.conf**.

Etape 4. Migration des informations smux

Si vous disposez des informations smux que vous devez faire migrer, vous pouvez copier ces lignes directement dans le nouveau fichier. Dans ce scénario, l'entrée smux `sampled` a été configurée dans le fichier **/etc/snmpd.conf**. Cette ligne doit être copiée dans le fichier **/etc/snmpdv3.conf**.

```
#--------------------------------------------------------------------
#       smux <client OIdentifier> <password> <address> <netmask>
 smux            1.3.6.1.4.1.2.3.1.2.3.1.1         sampled_password  #
sampled
#--------------------------------------------------------------------
```

Etape 5. Arrêt et démarrage du démon snmpd

Une fois terminée la migration du démon **/etc/snmpd.conf** vers le fichier **/etc/snmpdv3.conf**, arrêtez et démarrez le démon **snmpd**. Vous devez arrêter et démarrer le démon **snmpd** chaque fois que vous modifiez le fichier **/etc/snmpdv3.conf**.

1. Entrez la commande suivante pour arrêter le démon :

   ```
   stopsrc -s snmpd
   ```

2. Entrez la commande suivante pour redémarrer le démon :

   ```
   startsrc -s snmpd
   ```

Remarque : La simple actualisation de l'agent SNMPv3 ne fonctionne pas comme sous SNMPv1. Si vous modifiez le fichier **/etc/snmpdv3.conf**, vous devez arrêter et redémarrer le démon, comme indiqué dans la procédure précédente. La fonction de configuration dynamique prise en charge dans SNMPv3 ne vous permet pas d'actualisation.

Création d'utilisateurs dans SNMPv3

Ce scénario montre comment créer un utilisateur dans SNMPv3 en éditant manuellement les fichiers **/etc/snmpdv3.conf** et **/etc/clsnmp.conf**.

L'utilisateur `u1` sera créé dans ce scénario. L'utilisateur `u1` reçoit des clés d'autorisation mais pas des clés de confidentialité (qui sont disponibles uniquement si vous avez installé le fichier **snmp.crypto**). Le protocole HMAC–MD5 servira à créer les clés d'autorisation de u1. Une fois qu'u1 aura été configuré, il sera placé dans un groupe, puis les droits d'accès d'affichage de ce groupe seront définis. Enfin, des entrées de signaux d'interruption seront créées pour u1.

Chaque valeur individuelle utilisée dans les fichiers **/etc/snmpdv3.conf** et **/etc/clsnmp.conf** ne doit pas excéder 32 octets.

Etape 1. Création de l'utilisateur

1. Choisissez le protocole de sécurité à utiliser, HMAC–MD5 ou HMAC–SHA. Dans le présent scénario, HMAC–MD5 est utilisé.
2. Générez les clés d'authentification à l'aide de la commande **pwtokey**. Votre sortie peut différer, selon le protocole d'authentification utilisé et l'utilisation ou non de clés de confidentialité. Ces clés sont utilisées dans les fichiers **/etc/snmpdv3.conf** et **/etc/clsnmp.conf**. La commande employée pour l'utilisateur `u1` est la suivante :

   ```
   pwtokey -p HMAC-MD5 -u auth   anypassword   9.3.230.119
   ```

   L'adresse IP spécifiée est celle où s'exécute l'agent. Le mot de passe est quelconque, mais veillez à le conserver en un lieu sûr en vue d'une utilisation ultérieure. Le résultat est présenté de la manière suivante :

   ```
   Affichage de la clé d'authentification 16 octets HMAC-MD5 authKey :
      63960c12520dc8829d27f7fbaf5a0470

   Affichage de la clé d'authentification 16 octets HMAC-MD5 localized
   authKey :   b3b6c6306d67e9c6f8e7e664a47ef9a0
   ```

3. Avec les droits de l'utilisateur racine, ouvrez le fichier **/etc/snmpdv3.conf** avec votre éditeur favori.
4. Créez un utilisateur en ajoutant une entrée `USM_USER` suivant le format donné dans le fichier. La valeur *`authKey`* est la clé d'authentification localisée générée avec la commande **pwtokey**. L'entrée associée à l'utilisateur `u1` est la suivante :

   ```
   #--------------------------------------------------------------------
   # Entrées USM_USER
   #   Définit l'utilisateur pour le modèle USM (User-based Security Model).
   # Format :
   #  userName engineID authProto authKey privProto privKey keyType
   storageType
   #
   USM_USER u1 - HMAC-MD5 b3b6c6306d67e9c6f8e7e664a47ef9a0 - - L -
   #--------------------------------------------------------------------
   ```

   – *`userName`* est le nom de l'utilisateur. Dans ce cas, il s'agit de `u1`.

   – *`authProto`* doit être le protocole que vous avez utilisé lors de la création des clés. Dans ce cas, il s'agit de `HMAC-MD5`.

   – *`authKey`* est la clé d'authentification localisée générée avec la commande **pwtokey**.

   – *`privProto`* et *`privkey`* ,e sont pas précisés car nous n'utilisons pas les clés de confidentialité dans ce scénario.

   – *`keyType`* est `L` parce que nous utilisons la clé d'authentification localisée.

5. Sauvegardez puis fermez le fichier **/etc/snmpdv3.conf**.
6. Ouvrez le fichier **/etc/clsnmp.conf** du gestionnaire SNMP sous votre éditeur favori.

7. Ajoutez le nouvel utilisateur selon le format donné dans le fichier. L'entrée associée à l'utilisateur `u1` est la suivante :

```
#--------------------------------------------------------------------
#
# Format des entrées :
# winSnmpName targetAgent admin secName password context secLevel
authProto authKey privProto privKey
#
user1  9.3.230.119  SNMPv3  u1  -  -  AuthNoPriv  HMAC-MD5
63960c12520dc8829d27f7fbaf5a0470  -  -
#--------------------------------------------------------------------
```

   - *winSnmpName* peut être la valeur de votre choix. Cette valeur est utilisée lors des requêtes SNMP à l'aide de la commande **clsnmp**.
   - *targetAgent* représente l'adresse IP d'exécution de l'agent, qui a également servi lors de la création des clés d'authentification.
   - *admin* est défini comme `SNMPv3` car nous allons envoyer des requêtes SNMPv3.
   - *secName* est le nom de l'utilisateur que vous créez. Dans ce cas, il s'agit de `u1`.
   - *seclevel* est défini comme `AuthNoPriv` car il est configuré pour utiliser l'authentification mais pas la confidentialité (par conséquent, il n'y a pas de valeurs pour *privProto* et *privKey*).
   - *authproto* est défini comme le protocole d'authentification utilisé pour créer les clés d'authentification.
   - *authKey* est la clé d'authentification non localisée générée avec la commande **pwtokey**.

8. Sauvegardez et fermez le fichier **/etc/clsnmp.conf**.

Etape 2. Configuration du groupe

L'utilisateur doit maintenant être placé dans un groupe. Si vous disposez déjà d'un groupe configuré avec tous les droits d'accès et d'affichage que vous voulez accorder à cet utilisateur, vous pouvez y placer l'utilisateur. Si vous voulez doter cet utilisateur de droits d'accès et d'affichage n'existant dans aucun groupe existant, créez un groupe dans lequel vous ajoutez l'utilisateur.

Pour ajouter l'utilisateur à un nouveau groupe, créez une nouvelle entrée `VACM_GROUP` dans le fichier **/etc/snmpdv3.conf**. L'entrée de groupe associée à l'utilisateur `u1` est la suivante :

```
#-------------------------------------------------------------
# entrées VACM_GROUP
#    Définit un groupe de sécurité (composé d'utilisateurs ou de
communautés)
#    pour le modèle VACM (View-based Access Control Model).
# Format :
#  groupName securityModel securityName storageType
VACM_GROUP group1 USM u1 -
#-------------------------------------------------------------
```

- *groupName* peut être le nom de votre choix. Il devient le nom du groupe. Dans ce cas, il s'agit de `group1`.
- *securityModel* est défini comme `USM`, qui tire parti des fonctionnalités de sécurité SNMPv3.
- *securityName* est le nom de l'utilisateur. Dans ce cas, il s'agit de `u1`.

Etape 3. Configuration des permissions d'accès et d'affichage

Les droits d'accès et d'affichage doivent être définis pour le nouveau groupe que vous venez de créer. Ces droits sont définis en ajoutant les entrées `VACM_VIEW` et `VACM_ACCESS` au fichier **/etc/snmpdv3.conf**.

1. Choisissez les droits d'accès et d'affichage à accorder au groupe.
2. Ajoutez les entrées `VACM_VIEW` au fichier **/etc/snmpdv3.conf** pour définir les objets MIB auquel le groupe peut accéder. Dans ce scénario, `group1` aura accès aux sous–arborescences MIB `interfaces`, `tcp`, `icmp`, et `system`. Nous allons toutefois limiter l'accès de `group1` à la variable MIB `sysObjectID` au sein de la sous–arborescence MIB système.

```
#-------------------------------------------------------------------
# Entrées VACM_VIEW
# Définit un ensemble particulier de données MIB, appelé une vue, pour le
#    Modèle VACM (View-based Access Control Model).
# Format :
#  viewName viewSubtree viewMask viewType storageType
VACM_VIEW group1View        interfaces     - included -
VACM_VIEW group1View        tcp           - included -
VACM_VIEW group1View        icmp             - included -
VACM_VIEW group1View        system         - included -
VACM_VIEW group1View        sysObjectID    - excluded -
#-------------------------------------------------------------------
```

   – *viewName* est le nom de la vue. Dans ce cas, il s'agit de `group1View`.
   – *viewSubtree* est la sous–arborescence MIB à laquelle vous voulez donner accès.
   – *viewType* détermine si les sous–arborescences MIB définies sont incluses dans la vue. Dans le cas présent, toutes les sous–arborescences sont incluses, à l'exception de la variable MIB `sysObjectID` qui fait partie de la sous–arborescence `system` .

3. Ajoutez une entrée `VACM_ACCESS` au fichier **/etc/snmpdv3.conf** afin de définir les droits accordés au groupe sur les objets MIB indiqués ci–dessus. Pour `group1`, un accès en lecture seule est accordé.

```
#-------------------------------------------------------------------------
# Entrées VACM_ACCESS
#    Identifie les droits d'accès accordés aux différents groupe de
sécurité
#    pour le modèle VACM (View-based Access Control Model).
# Format :
# groupName contextPrefix contextMatch securityLevel securityModel
readView writeView notifyView storageType
VACM_ACCESS  group1 - - AuthNoPriv USM group1View - group1View -
#-------------------------------------------------------------------------
```

   – *groupName* est le nom du groupe. Dans ce cas, il s'agit de `group1`.
   – *securityLevel* est le niveau de sécurité utilisé. Dans le présent scénario, des clés d'authentification sont utilisées, mais pas des clés de confidentialité. La valeur est donc définie sur `AuthNoPriv`.
   – *securityModel* correspond au modèle de sécurité que vous utilisez (SNMPv1, SNMPv2c ou USM). Dans le présent scénario, il est défini sur `USM` pour permettre l'utilisation des dispositifs de sécurité SNMPv3.
   – *readView* détermine les VACM_VIEWs auxquels le groupe a un accès en lecture. Dans ce scénario, `group1View` est accordé, ce qui donne à `group1` un accès en lecture aux entrées `group1View VACM_VIEW` .
   – *writeView* détermine les VACM_VIEWs auxquels le groupe a un accès en écriture. Dans le présent scénario, aucun accès en écriture n'est accordé au `group1`.

– *notifyView* spécifie le nom de la vue à appliquer en cas de signal d'interruption exécuté sous le contrôle de l'entrée dans la table d'accès.

Remarque : Il arrive que plusieurs entrées VACM_ACCESS soient nécessaires pour un groupe. Si des utilisateurs du groupe ont des paramètres d'authentification et de confidentialité différents (noAuthNoPriv, AuthNoPriv, ou AuthPriv), plusieurs entrées VACM_ACCESS sont requises et le paramètre securityLevel doit être défini en conséquence.

Etape 4. Configuration des entrées d'interruption pour l'utilisateur

Les entrées d'interruption dans SNMPv3 sont créées en ajoutant les entrées NOTIFY, TARGET_ADDRESS et TARGET_PARAMETERS au fichier **/etc/snmpdv3.conf**. L'entrée TARGET_ADDRESS indique la destination des interruptions et l'entrée TARGET_PARAMETERS établit une équivalence entre les informations de TARGET_ADDRESS et group1.

L'entrée NOTIFY a été configurée par défaut. Voici l'entrée NOTIFY par défaut :

```
NOTIFY notify1 traptag trap –
```

Dans le présent scénario, nous utilisons la valeur qui est spécifiée dans l'entrée par défaut, traptag.

1. Ajoutez une entrée TARGET_ADDRESS pour spécifier la destination des signaux d'interruption.

```
#-------------------------------------------------------------------------
# TARGET_ADDRESS
#    Définit l'adresse et les paramètres d'une application de gestion
#    à utiliser pour envoyer des notifications.
# Format :
#  targetAddrName tDomain tAddress tagList targetParams timeout
retryCount storageType
#-------------------------------------------------------------------------
TARGET_ADDRESS Target1 UDP 9.3.207.107     traptag trapparms1 – – –
```

   – *targetAddrName* peut être le nom de votre choix. Dans ce scénario, nous avons utilisé Target1.

   – *tAddress* est l'adresse IP à laquelle doivent être envoyés les signaux d'interruption du groupe.

   – *tagList* est le nom configuré dans l'entrée NOTIFY. Dans ce cas, il s'agit de traptag.

   – *targetParams* peut être la valeur de votre choix. Nous avons utilisé trapparms1, qui sera employé dans l'entrée TARGET_PARAMETERS .

2. Ajouter une entrée TARGET_PARAMETERS .

```
#-------------------------------------------------------------------------
# TARGET_PARAMETERS
#    Définit les paramètre de traitement des messages et de sécurité
#    à utiliser pour envoyer les notifications à une cible de gestion
particulière.
# Format :
#  paramsName mpModel securityModel securityName securityLevel
storageType
#-------------------------------------------------------------------------
TARGET_PARAMETERS trapparms1 SNMPv3  USM     u1        AuthNoPriv
```

   – *paramsName* est identique à la valeur targetParams dans l'entrée TARGET_ADDRESS qui, dans ce cas, est trapparms1.

   – *mpModel* est la version de SNMP qui est utilisée.

- *securityModel* correspond au modèle de sécurité que vous utilisez (SNMPv1, SNMPv3 ou USM). Dans le présent scénario, il est défini sur `USM` pour permettre l'utilisation des dispositifs de sécurité SNMPv3.
- *securityName* est le nom d'utilisateur indiquée dans l'entrée `USM_USER` qui, dans ce cas, est `u1`.
- *securityLeve* est défini comme `AuthNoPriv` car nous utilisons des clés d'authentification mais pas des clés de confidentialité.

Etape 5. Arrêt et démarrage du démon snmpd

Après avoir modifié le fichier **/etc/snmpdv3.conf**, arrêtez et démarrez le démon **snmpd**.

1. Entrez la commande suivante pour arrêter le démon **snmpd** :

   ```
   stopsrc –s snmpd
   ```

2. Entrez la commande suivante pour arrêter le démon **snmpd** :

   ```
   startsrc –s snmpd
   ```

Les nouveaux paramètres sont désormais validés.

Remarque : La simple actualisation de l'agent SNMPv3 utilisant **refresh –s snmpd** ne fonctionne pas comme sous SNMPv1. Si vous modifiez le fichier **/etc/snmpdv3.conf**, vous devez arrêter et redémarrer le démon, comme indiqué dans la procédure précédente. La fonction de configuration dynamique prise en charge dans SNMPv3 ne vous permet pas d'actualisation.

Etape 6. Test de votre configuration

Pour vérifier que votre configuration est correcte, vous pouvez lancer la commande suivante dans le gestionnaire SNMP.

```
clsnmp –h user1 walk   mib
```

où *mib* est une sous–arborescence MIB à laquelle l'utilisateur a accès. Dans ce scénario, il peut s'agir de `interfaces`, `tcp`, `icmp`, ou `system`. Si la configuration est correcte, vous visualisez les informations dans la sous–arborescence spécifiée.

Si la sortie obtenue n'est pas correcte, révisez les étapes présentées dans ce document et vérifiez que vous avez entré correctement toutes les informations.

Mise à jour dynamique des clés d'authentification et de confidentialité dans SNMPv3

Le présent scénario démontre comment mettre à jour dynamiquement des clés pour un utilisateur dans SNMPv3. Dans ce scénario, l'utilisateur `u4` met à jour les clés d'authentification pour l'utilisateur `u8`. Les deux utilisateurs `u4` et `u8` ont déjà créé des clés d'authentification basées sur le mot de passe `defaultpassword` et l'adresse IP `9.3.149.49`, et le tout fonctionne correctement.

Dans ce scénario, de nouvelles clés sont créées pour l'utilisateur `u8` et le fichier **/etc/snmpdv3.conf** est mis à jour dynamiquement. La clé d'authentification de l'utilisateur `u8` se trouvant dans le fichier **/etc/clsnmp.conf** côté gestionnaire doit être modifiée manuellement pour correspondre aux nouvelles clés.

Effectuez une sauvegarde du fichier **/etc/snmpdv3.conf** dans l'agent SNMP et une sauvegarde du fichier **/etc/clsnmp.conf** dans le gestionnaire SNMP avant de démarrer la procédure.

Ci–dsous se trouve le fichier **/etc/snmpdv3.conf** qui va être mis à jour dynamiquement :

```
USM_USER u4 - HMAC-MD5 18a2c7b78f3df552367383eef9db2e9f - - N -
 USM_USER u8 - HMAC-SHA 754ebf6ab740556be9f0930b2a2256ca40e76ef9 - - N -

 VACM_GROUP group1 SNMPv1  public  -
 VACM_GROUP group2 USM u4 -
 VACM_GROUP group2 USM u8 -

 VACM_VIEW defaultView        internet                    - included -

 VACM_ACCESS  group1 - - noAuthNoPriv SNMPv1  defaultView - defaultView -
 VACM_ACCESS  group2 - - noAuthNoPriv USM defaultView defaultView
defaultView -
 VACM_ACCESS  group2 - - AuthNoPriv USM defaultView defaultView defaultView
-
 VACM_ACCESS  group2 - - AuthPriv USM defaultView defaultView defaultView -

 NOTIFY notify1 traptag trap -

 TARGET_ADDRESS Target1 UDP 127.0.0.1       traptag trapparms1 - - -
 TARGET_ADDRESS Target2 UDP 9.3.149.49       traptag trapparms2 - - -
 TARGET_ADDRESS Target3 UDP 9.3.149.49       traptag trapparms3 - - -
 TARGET_ADDRESS Target4 UDP 9.3.149.49       traptag trapparms4 - - -

 TARGET_PARAMETERS trapparms1 SNMPv1  SNMPv1  public  noAuthNoPriv -
 TARGET_PARAMETERS trapparms3 SNMPv2c  SNMPv2c  publicv2c  noAuthNoPriv -
 TARGET_PARAMETERS trapparms4 SNMPv3  USM     u4 AuthNoPriv -
```

Ci–dessous se trouve le fichier **/etc/clsnmp.conf** qui va être mis à jour pour l'utilisateur `u8` :

```
testu4 9.3.149.49 snmpv3 u4 - - AuthNoPriv HMAC-MD5
18a2c7b78f3df552367383eef9db2e9f - -
 testu8 9.3.149.49 snmpv3 u8 - - AuthNoPriv HMAC-SHA
754ebf6ab740556be9f0930b2a2256ca40e76ef9 - -
```

Pour mettre à jour votre mot de passe et vos clés d'authentification, procédez comme suit :

1. Du côté du gestionnaire SNMP, exécutez la commande **pwchange**. Au cours de ce scénario, nous avons lancé la commande suivante :

   ```
   pwchange -u auth -p HMAC-SHA defaultpassword newpassword 9.3.149.49
   ```

   Cette commande génère une nouvelle clé d'authentification.

   – *`-u auth`* précise que seule une clé d'authentification sera créée. Si vous mettez à jour les clés de confidentialité également, indiquez `-u all`.

– *-p HMAC-SHA* indique le protocole qui sera utilisé pour créer la clé d'authentification. Si vous mettez à jour les clés de confidentialité également, indiquez `-p all`.

– *defaultpassword* est le mot de passe utilisé pour créer la dernière clé d'authentification (par exemple si `bluepen` aurait été utilisé pour créer la dernière clé d'authentification, `bluepen` serait aussi utilisé ici)

– *newpassword* est le nouveau mot de passe qui sera utilisé pour générer le clé d'authentification. Conservez–le pour référence ultérieure.

– *9.3.149.49* est l'adresse IP où s'exécute l'agent SNMP.

Cette commande a généré le résultat suivant :

```
Cliché de la valeur de 40 octets HMAC-SHA authKey keyChange :
   8173701d7c00913af002a3379d4b150a
   f9566f56a4dbde21dd778bb166a86249
   4aa3a477e3b96e7d
```

Vous utiliserez cette clé d'authentification à l'étape suivante.

Remarque : Conservez en lieu sûr les nouveaux mots de passe que vous utilisez. Vous devrez les réutiliser lorsque vous effectuerez des modifications ultérieurement.

2. Sur le gestionnaire SNMP, l'utilisateur `u4` modifiera la clé d'authentification pour l'utilisateur `u8` en entrant la commande suivante :

```
clsnmp -h testu4 set
usmUserAuthKeyChange.12.0.0.0.2.0.0.0.0.9.3.149.49.2.117.56

\'8173701d7c00913af002a3379d4b150af9566f56a4dbde21dd778bb166a862494aa3a47
7e3b96e7d\'h
```

– `testu4` est utilisé car il est mappé avec l'utilisateur `u4` dans le fichier **/etc/clsnmp.conf**.

– L'ID d'instance de *usmUserAuthKeyChange* inclut, en valeurs décimales, l'ID de moteur de l'agent SNMP où se produit la mise à jour et le nom d'utilisateur dont la clé d'authentification est mise à jour. L'ID moteur peut être trouvé dans le fichier **/etc/snmpd.boots** (le fichier **/etc/snmpd.boots** contient deux chaînes de chiffres. L'ID moteur correspond à la première chaîne. Ignorez la deuxième chaîne.

  Il est nécessaire de convertir les valeurs hexadécimales de cet ID moteur pour pouvoir l'utiliser. Tous les deux chiffres de l'ID moteur hexadécimal se convertissent en une seule valeur décimale. Par exemple, l'ID moteur `000000020000000009039531` sera lu sous la forme `00 00 00 02 00 00 00 00 09 03 95 31`. Chacun de ces nombres doit être converti en valeurs décimales, donnant `0.0.0.2.0.0.0.0.9.3.149.49` (Vous trouverez une table de conversion dans Annexe C. Table de conversion page C-1 .). Le premier nombre de la chaîne est le nombre d'octets de la chaîne décimale. Dans ce cas, il s'agit de `12`, dont le résultat est `12.0.0.0.2.0.0.0.0.9.3.149.49`.

  Le nombre suivant est le nombre d'octets du nom d'utilisateur, suivi des valeurs décimales du nom d'utilisateur lui–même. Dans ce cas, le nom d'utilisateur est `u8`. Lorsqu'i est converti en valeurs décimales, `u8` devient `117.56`. Parce que le nom d'utilisateur a une longueur de 2 octets, la valeur représentant le nom d'utilisateur devient `2.117.56`. Vous pouvez l'ajouter à la fin de l'ID moteur décimal (Vous trouverez une table de conversion dans Annexe C. Table de conversion page C-1 .).

  Dans ce cas, le résultat est `12.0.0.0.2.0.0.0.0.9.3.149.49.2.117.56`.

– La valeur suivante de la commande est la nouvelle clé d'authentification générée à l'aide de la commande **pwchange** dans l'étape précédente.

Remarque : Si des clés de confidentialité sont également configurées pour l'utilisateur, cette procédure soit être réitérée pour leur mise à jour. Lors de la mise à jour des clés de confidentialité, utilisez la valeur `usmUserPrivKeyChange` à la place de la valeur `usmUserAuthKeyChange` .

L'utilisation de `usmUserOwnAuthKeyChange` au lieu de `usmUserAuthKeyChange` permet à l'utilisateur de modifier sa propre clé d'authentification. Par exemple, l'utilisateur `u4` peut modifier sa propre clé d'authentification avec `usmUserOwnAuthKeyChange`.

La sortie de la commande est la suivante :

```
1.3.6.1.6.3.15.1.2.2.1.6.12.0.0.0.2.0.0.0.0.9.3.149.49.2.117.56 =
'8173701d7c00913af002a3379
 d4b150af9566f56a4dbde21dd778bb166a862494aa3a477e3b96e7d'h
```

Une fois la commande terminée, le fichier **/etc/snmpdv3.conf** est automatiquement mis à jour, au bout de 5 minutes, côté agent SNMP. Vous pouvez également arrêter puis démarrer le démon SNMP pour mettre à jour le fichier. L'entrée suivant pour l'utilisateur `u8` est mise à jour dynamiquement dans le fichier **/etc/snmpdv3.conf** :

```
USM_USER u8 0000000200000000009039531 HMAC-SHA
4be657b3ae92beee322ee5eaeef665b338caf2d9
 None - L nonVolatile
```

3. Côté gestionnaire SNMP, lancez la commande **pwtokey** pour générer la nouvelle clé d'authentification sur la base du nouveau mot de passe à placer dans le fichier **/etc/clsnmp.conf**. Au cours de ce scénario, nous avons lancé la commande suivante :

```
pwtokey -u auth -p HMAC-SHA  newpassword 9.3.149.49
```

   - *`-u auth`* précise que seule une clé d'authentification sera créée. Si vous mettez à jour les clés de confidentialité également, indiquez `-u all`.
   - *`-p HMAC-SHA`* indique le protocole qui sera utilisé pour créer la clé d'authentification. Si vous mettez à jour les clés de confidentialité également, indiquez `-p all`.
   - Le mot de passe utilisé (dans ce cas *`newpassword`*) doit être le même que le mot de passe utilisé lors de la génération de nouvelles clés d'authentification avec la commande **pwchange**.
   - L'adresse IP utilisée (dans le cas présent, *`9.3.149.49`*) doit être celle où s'exécute l'agent.

   Le résultat donne les clés d'authentification localisées et non localisées :

```
Affichage de la clé d'authentification 20 octets HMAC-SHA authKey :

   79ce23370c820332a7f2c7840c3439d12826c10d

 Affichage de la clé d'authentification 20 octets HMAC-SHA localized
authKey :
   b07086b278163a4b873aace53a1a9ca250913f91
```

4. Ouvrez le fichier **/etc/clsnmp.conf** avec l'éditeur de votre choix et placez la clé d'authentification non localisée sur la ligne correspondant à l'utilisateur dont les clés sont mises à jour. Dans le présent scénario, l'entrée est la suivante :

```
testu8 9.3.149.49 snmpv3 u8 - - AuthNoPriv HMAC-SHA
79ce23370c820332a7f2c7840c3439d12826c10d - -
 Enregistrez, puis fermez le fichier.
```

5. Testez la configuration mise à jour en exécutant la commande suivante :

```
clsnmp -v -h testu8 walk   mib
```

   où *mib* est une variable MIB à laquelle l'utilisateur u8 a un accès en lecture. Dans ce cas, l'utilisateur u8 a accès à Internet.

Création d'un alias local pour la messagerie électronique

La création d'alias locaux pour la messagerie électronique permet de créer des groupes ou des listes de distribution auxquels du courrier peut être envoyé.

Dans ce scénario, geo@medussa, mark@zeus, ctw@athena, and dsf@plato sont ajoutés à l'alias de messagerie `testers`. Une fois l'alias `testers` créé, glenda@hera reçoit la propriété de l'alias.

Une fois l'alias `testers` ajouté au fichier **/etc/mail/aliases**, la base de données des alias est recompilée à l'aide de la commande **sendmail**. Une fois la base de données recompilée, des messages électroniques peuvent être envoyés à l'alias `testers`.

1. Ouvrez le fichier **/etc/mail/aliases** sous votre éditeur favori.
2. Sur une ligne vierge, ajoutez un nom d'alias suivi de deux points (:) et d'une liste de destinataires séparés par une virgule. Par exemple, l'entrée suivante définit l'alias `testers` :

   ```
   testers: geo@medussa, mark@zeus, ctw@athena, dsf@plato
   ```

3. Créez le propriétaire de l'alias. Si la commande **sendmail** ne parvient pas à envoyer du courrier à l'alias, elle envoie un message d'erreur au propriétaire.

   Ajoutez une ligne dans **/etc/mail/aliases** pour indiquer le propriétaire. Le format de cette ligne est `owner-` *`groupname`*`:` *`owner`*, où *`groupname`* est le nom de l'alias et *`owner`* l'adresse électronique du propriétaire. Dans cet exemple, `glenda@hera` est défini comme le propriétaire de l'alias `testers` :

   ```
   testers: geo@medussa, mark@zeus, ctw@athena, dsf@plato
    owner-testers: glenda@hera
   ```

4. Une fois l'alias créé, exécutez la commande `sendmail -bi` pour recompiler la base de données des alias. Vous devez exécuter cette commande à chaque fois que vous mettez à jour le fichier **/etc/mail/aliases**.

Vous pouvez maintenant envoyer des messages à l'alias `testers`.

Configuration des serveurs de nom de domaine

Dans ce scénario, un serveur de noms maître, un serveur de noms esclave et un serveur de noms d'indice est configuré pour effectuer une résolution de noms. Chaque serveur de noms est une machine distincte et pour chacun, un fichier **/etc/named.conf** est configuré, même si les informations de chacun sont différentes. Le fichier **/etc/named.conf** est lu chaque fois que le démon **named** est démarré. Il indique le type du serveur (maître, esclave, indice) et l'endroit où il obtient ses données de résolution de noms. Chacun de ces serveurs de noms exécute BIND 8.

Le serveur de noms maître est configuré pour fournir une résolution de noms pour la zone `abc.aus.century.com`. Dans ce scénario, l'adresse IP du serveur de noms maître est `192.9.201.1`, et son nom hôte est `venus.abc.aus.century.com`. Il fournit une résolution de noms pour les noms hôte venus, earth, mars, et jupiter. Le fichier **/etc/named.conf** est configuré pour indiquer que le démon **named** doit rechercher les fichiers de données dans le répertoire **/usr/local/domain**. Les fichiers de données qui seront configurés pour le serveur de noms maître sont **named.ca**, **named.abc.local**, **named.abc.data**, et **named.abc.rev**.

Un serveur de noms esclave est alors configuré. Le nom hôte du serveur de noms esclave est `earth.abc.aus.century.com`, et son adresse IP est `192.9.201.5`. Dans le fichier **/etc/named.conf** du serveur de noms esclave, nous indiquons l'adresse du serveur de noms maître de façon à ce que le serveur de noms esclave puisse répliquer les fichiers **named.abc.data** et **named.abc.rev** du serveur de noms maître. En outre, les fichiers de données **named.ca** et **named.abc.local** sont configurés pour ce serveur.

Un serveur de noms d'indices est alors configuré. Le serveur de noms d'indices stocke une cache locale contenant le nom hôte et les équivalences d'adresses. Si une adresse ou un nom hôte demandé ne se trouve pas dans sa cache, le serveur d'indices contacte le serveur de noms maître, obtient les informations de résolution et les ajoute à sa cache. En outre, les fichiers de données **named.ca** et **named.abc.local** sont configurés pour ce serveur.

Toutes les informations des fichiers de données **named** (pas le fichier **/etc/named.conf**) des serveurs de noms doivent avoir le format Standard Resource Record Format. Pour plus de détails sur les informations contenues dans les fichiers de données **named**, reportez–vous à Standard Resource Record Format for TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*

L'administrateur de chacun des serveurs de noms est `gail.zeus.abc.aus.century.com`. Ceci est indiqué dans les fichiers de données locaux de chaque serveur de noms. En outre, dans ce scénario, le serveur de noms racine est `relay.century.com`, avec l'adresse IP `129.114.1.2`.

A la fin de ce scénario, la résolution de noms est fournie pour les hôtes venus, earth, mars, et jupiter. En outre, une résolution de noms inverse (adresse IP en nom hôte) est fournie. Lorsqu'une demande impossible à résoudre est reçue, le serveur de noms maître contacte `relay.century.com` pour trouver les informations requises.

Etape 1. Configuration du serveur de noms maître

1. Sur le serveur de noms maître, ouvrez le fichier **/etc/named.conf**. S'il n'existe pas de fichiers **/etc/named.conf** dans le répertoire **/etc**, créez–en un en exécutant la commande suivante :

   ```
   touch /etc/named.conf
   ```

   Procédez comme suit pour configurer le fichier **/etc/named.conf** :

   a. Indiquez une clause de répertoire dans la strophe des options. Ceci permet aux fichiers de données **named** d'utiliser des chemins relatifs au répertoire **/usr/local/domain**. Dans ce scénario, ce qui suit a été ajouté :

      ```
      options {
           directory "/usr/local/domain";
       };
      ```

      Si vous choisissez de ne pas indiquer de répertoire, les fichiers de données requis sont recherchés dans le répertoire **/etc**.

   b. Pour stocker des données d'enregistrement en dehors des zones définies, spécifiez le nom du fichier de zone d'indices. Dans ce scénario, ce qui suit a été ajouté :

      ```
      zone "." IN {
           type hint;
           file "named.ca";
       };
      ```

   c. Ajoutez les strophes suivantes pour spécifier chaque zone, le type de serveur de noms que vous configurez et le fichier de données du domaine de votre serveur de noms. Dans ce scénario, le serveur maître des zones avant et inverse est le suivant :

      ```
      zone "abc.aus.century.com" in {
           type master;
           file "named.abc.data";
       };
       zone "201.9.192.in-addr.arpa" in {
           type master;
           file "named.abc.rev";
       };
      ```

   d. Définissez le nom du fichier **named** local. Par exemple :

      ```
      zone "0.0.127.in-addr.arpa" in {
           type master;
           file "named.abc.local";
       };
      ```

      Après avoir édité le fichier, enregistrez–le et fermez–le.

2. Ouvrez le fichier **/usr/local/domain/named.ca**. Ajoutez les adresses des serveurs de noms racine du domaine. Ce qui suit a été ajouté dans ce scénario :

   ```
   ; root name servers.
   .          IN    NS    relay.century.com.
   relay.century.com.   3600000    IN    A     129.114.1.2
   ```

   Après avoir édité le fichier, enregistrez–le et fermez–le.

3. Ouvrez le fichier **/usr/local/domain/named.abc.local**. Ajoutez les informations suivantes :

– La valeur de SOA (Start Of Authority) de la zone et les délais TTL (time–to–live) par défaut. Ce qui suit a été ajouté dans ce scénario :

```
$TTL 3h    ;3 hour

 @ IN SOA venus.abc.aus.century.com. gail.zeus.abc.aus.century.com.  (

                            1       ;serial
                            3600    ;refresh
                            600     ;retry
                            3600000 ;expire
                            3600    ;negative caching TTL
 )
```

– L'enregistrement du serveur de noms (NS). Insérez une tabulation au début de la ligne. Le démon **named** remplace la tabulation par le nom de zone :

```
 <tab>       IN    NS     venus.abc.aus.century.com.
```

– L'enregistrement PTR (pointeur).

```
1       IN    PTR    localhost.
```

Après avoir édité le fichier, enregistrez–le et fermez–le.

4. Ouvrez le fichier **/usr/local/domain/named.abc.data**. Ajoutez les informations suivantes :

– La valeur de SOA (Start Of Authority) et les délais TTL (timetolive) par défaut de la zone. Cet article indique le début de la zone. Un seul article de SOA par zone est autorisé. Dans ce scénario, ce qui suit a été ajouté :

```
$TTL 3h    ;3 hour

 @ IN    SOA     venus.abc.aus.century.com.
gail.zeus.abc.aus.century.com. (
                1       ;serial
                3600    ;refresh
                600     ;retry
                3600000 ;expire
                3600    ;negative caching TTL
 )
```

– Les enregistrements du serveur de noms de tous les serveurs de noms maîtres de la zone. Insérez une tabulation au début de la ligne. Le démon **named** remplace la tabulation par le nom de zone :

```
 <tab>       IN    NS       venus.abc.aus.century.com.
```

– Les informations de résolution de noms en adresses pour tous les hôtes dans la zone d'autorité du serveur de noms.

```
venus         IN    A        192.9.201.1
 earth         IN    A        192.9.201.5
 mars          IN    A        192.9.201.3
 jupiter       IN    A        192.9.201.7
```

Insérez d'autres types d'entrée : articles de nom canonique ou de serveur de noms (facultatif). Après avoir édité le fichier, enregistrez–le et fermez–le.

5. Ouvrez le fichier **/usr/local/domain/named.abc.rev**. Ajoutez les informations suivantes :

– La valeur de SOA (Start Of Authority) de la zone et les délais TTL (time–to–live) par défaut. Cet article indique le début de la zone. Un seul article de SOA par zone est autorisé :

```
$TTL 3h    ;3 hour

 @  IN  SOA  venus.abc.aus.century.com. gail.zeus.abc.aus.century.com.
(
                              1       ;serial
                              3600    ;refresh
                              600     ;retry
                              3600000 ;expire
                              3600    ;negative caching TTL
 )
```

– Les autres types d'entrées, tels que les articles de serveur de noms. Si vous incluez ces articles, insérez une tabulation au début de la ligne. Le démon **named** remplace la tabulation par le nom de zone. Dans ce scénario, ce qui suit a été ajouté :

```
 <tab>      IN      NS      venus.abc.aus.century.com.
```

– Des informations de résolution adresse–nom sur tous les hôtes à placer dans la zone d'autorité du serveur de noms.

```
1                       IN    PTR    venus.abc.aus.century.com.
5                       IN    PTR    earth.abc.aus.century.com.
3                       IN    PTR    mars.abc.aus.century.com.
7                       IN    PTR    jupiter.abc.aus.century.com.
```

Après avoir édité le fichier, enregistrez–le et fermez–le.

6. Créez un fichier **/etc/resolv.conf** via la commande :

   ```
   touch /etc/resolv.conf
   ```

   La présence de ce fichier indique que l'hôte doit utiliser un serveur de noms pour la résolution de noms.

7. Ajoutez l'entrée suivante dans le fichier **/etc/resolv.conf** :

   ```
   nameserver 127.0.0.1
   ```

   `127.0.0.1` est l'adresse de bouclage qui, pour l'accès au serveur de noms, dirige l'hôte vers lui–même. Ce fichier **/etc/resolv.conf** peut également comporter une ligne du type :

   ```
   domain abc.aus.century.com
   ```

   Dans ce cas, `abc.aus.century.com` est le nom du domaine. Après avoir édité le fichier, enregistrez–le et fermez–le.

8. Utilisez le raccourci SMIT **smit stnamed** pour activer le démon **named**. Cette commande initialise le démon à chaque lancement du système. Indiquez quand vous souhaitez lancer le démon **named** : immédiatement, au prochain lancement du système ou les deux.

Etape 2. Configuration du serveur de noms esclave

Pour configurer un serveur de noms esclave, utilisez la procédure suivante. Editez une série de fichiers puis utilisez SMIT pour démarrer le démon **named**.

1. Sur le serveur de noms esclave, ouvrez le fichier **/etc/named.conf**. S'il n'existe pas de fichiers **/etc/named.conf** dans le répertoire **/etc**, créez–en un en exécutant la commande suivante :

   ```
   touch /etc/named.conf
   ```

   Procédez comme suit pour configurer le fichier **/etc/named.conf** :

a. Indiquez une clause de répertoire dans la strophe des options. Ceci permet aux fichiers de données **named** d'utiliser des chemins relatifs au répertoire **/usr/local/domain**. Dans ce scénario, ce qui suit a été ajouté :

```
options {
     directory "/usr/local/domain";
 };
```

Si vous choisissez de ne pas indiquer de répertoire, le démon **named** recherche les fichiers de données requis dans le répertoire **/etc**.

b. Pour stocker des données d'enregistrement en dehors des zones définies, spécifiez le nom du fichier de zone d'indices pour le serveur de noms.

```
zone "." IN {
     type hint;
     file "named.ca";
 };
```

c. Spécifiez les clauses de zone esclave. Chaque strophe comprend le type de zone, un nom de fichier dans lequel le serveur de noms peut sauvegarder ses données et l'adresse IP du serveur de noms maître, à partir duquel le serveur de noms maître peut répliquer ses fichiers de données. Dans ce scénario, nous avons ajouté les clauses de zone esclave suivantes :

```
zone "abc.aus.century.com" IN {
    type slave;
     file "named.abc.data.bak";
     masters { 192.9.201.1; };
 };

zone "201.9.192.in-addr.arpa" IN {
     type slave;
     file "named.abc.rev.bak";
     masters { 192.9.201.1; };
 };
```

d. Pour supporter l'adressage en boucle, indiquez une zone de type *maître* avec comme source le fichier `named.abc.local`, ainsi que le domaine dont le serveur de noms est responsable.

```
zone "0.0.127.in-addr.arpa" in {
     type master;
     file "named.abc.local";
 };
```

Après avoir édité le fichier, enregistrez–le et fermez–le.

2. Editez le fichier **/usr/local/domain/named.ca**

Ce fichier contient le serveur d'adresses qui est le serveur de domaine racine du réseau. Dans ce scénario, ce qui suit a été ajouté :

```
; root name servers.
.          IN    NS    relay.century.com.
relay.century.com.   3600000    IN    A     129.114.1.2
```

Après avoir édité le fichier, enregistrez–le et fermez–le.

3. Ouvrez le fichier **/usr/local/domain/named.abc.local**. Dans ce scénario, ce qui suit a été ajouté :

– La valeur de SOA (Start Of Authority) de la zone et les délais TTL (time–to–live) par défaut :

```
$TTL 3h    ;3 hour

 @ IN SOA earth.abc.aus.century.com. gail.zeus.abc.aus.century.com.  (

                              1       ;serial
                              3600    ;refresh
                              600     ;retry
                              3600000 ;expire
                              3600    ;negative caching TTL
 )
```

– L'enregistrement du serveur de noms (NS). Insérez une tabulation au début de la ligne. Le démon **named** remplace la tabulation par le nom de zone. Par exemple :

```
 <tab>      IN    NS     earth.abc.aus.century.com.
```

– L'enregistrement PTR (pointeur).

```
1       IN    PTR    localhost.
```

Après avoir édité le fichier, enregistrez–le et fermez–le.

4. Créez un fichier **/etc/resolv.conf** via la commande :

```
touch /etc/resolv.conf
```

5. Ajoutez les lignes suivantes au fichier :

```
nameserver 127.0.0.1
 domain abc.aus.century.com
```

Après avoir édité le fichier, enregistrez–le et fermez–le.

6. Utilisez le raccourci SMIT **smit stnamed** pour activer le démon **named**. Cette commande initialise le démon à chaque lancement du système. Indiquez quand vous souhaitez lancer le démon **named** : immédiatement, au prochain lancement du système ou les deux.

Etape 3. Configuration du serveur de noms d'indices

Pour configurer un serveur de noms d'indices ou de *mémoire cache uniquement*, suivez la procédure ci–dessous, qui édite une série de fichiers puis a recours à SMIT ou à la ligne de commande pour démarrer le démon **named**.

1. Sur le serveur de noms d'indices, ouvrez le fichier **/etc/named.conf**. S'il n'existe pas de fichiers **/etc/named.conf** dans le répertoire **/etc**, créez–en un en exécutant la commande suivante :

```
touch /etc/named.conf
```

Procédez comme suit pour configurer le fichier **/etc/named.conf** :

a. Indiquez une clause de répertoire dans la strophe des options. Ceci permet aux fichiers de données **named** d'utiliser des chemins relatifs au répertoire **/usr/local/domain**. Dans ce scénario, ce qui suit a été ajouté :

```
options {
     directory "/usr/local/domain";
 };
```

b. Pour supporter l'adressage en boucle, indiquez une zone de type *maître* avec comme source le fichier **named.abc.local**, ainsi que le domaine dont le serveur de noms est responsable. Dans cet exemple, le mot–clé du répertoire d'options a été indiqué dans le fichier **/etc/named.conf**.

```
zone "0.0.127.in-addr.arpa" IN {
     type master;
       file "named.abc.local";
 };
```

c. Spécifiez le nom du fichier de zone de cache. Par exemple :

```
zone "." IN {
     type hint;
     file "named.ca";
 };
```

Après avoir édité le fichier, enregistrez–le et fermez–le.

2. Editez le fichier **/usr/local/domain/named.ca**

   Ce fichier contient l'adresse des serveurs de noms "experts" pour le domaine racine (root) du réseau. Par exemple :

```
; root name servers.
.          IN    NS    relay.century.com.
relay.century.com.   3600000    IN    A     129.114.1.2
```

   Après avoir édité le fichier, enregistrez–le et fermez–le.

3. Editez le fichier **/usr/local/domain/named.local**. Dans ce scénario, les informations suivantes ont été ajoutées :

   – La valeur de SOA (Start Of Authority) de la zone et les délais TTL (time–to–live) par défaut :

```
$TTL 3h    ;3 hour

 @  IN  SOA  venus.abc.aus.century.com. gail.zeus.abc.aus.century.com.
(
                                1       ;serial
                                3600    ;refresh
                                600     ;retry
                                3600000 ;expire
                                3600    ;negative caching TTL
 )
```

   – L'enregistrement du serveur de noms (NS). Insérez une tabulation au début de la ligne. Le démon **named** remplace la tabulation par le nom de zone :

```
 <tab>       IN    NS     venus.abc.aus.century.com.
```

   – L'enregistrement PTR (pointeur).

```
1       IN    PTR    localhost.
```

   Après avoir édité le fichier, enregistrez–le et fermez–le.

4. Créez un fichier **/etc/resolv.conf** via la commande :

```
touch /etc/resolv.conf
```

5. Ajoutez les lignes suivantes au fichier :

```
nameserver 127.0.0.1
 domain abc.aus.century.com
```

   Après avoir édité le fichier, enregistrez–le et fermez–le.

6. Utilisez le raccourci SMIT **smit stnamed** pour activer le démon **named**. Cette commande initialise le démon à chaque lancement du système. Indiquez quand vous souhaitez lancer le démon **named** : immédiatement, au prochain lancement du système ou les deux.

Chapitre 2. Communications et réseaux : généralités

Ce chapitre présente les concepts de base pour la compréhension des systèmes en réseau. Il est destiné à l'administrateur système peu familiarisé avec les réseaux. Ceux qui maîtrisent déjà ces concepts sous UNIX peuvent passer directement au chapitre suivant.

Un réseau est la combinaison d'un ou plusieurs ordinateurs interconnectés. Le réseau *physique* regroupe les éléments matériels du réseau (cartes, câbles, lignes téléphoniques, etc). Quant au réseau *logique*, il comporte les éléments logiciels et le modèle conceptuel du réseau.

Les concepts présentés sont les suivants :

Fonctions de communication

Les réseaux offrent diverses fonctions de communication dédiées aux applications et aux utilisateurs. Ils permettent par exemple aux utilisateurs d'effectuer les tâches suivantes :

- Envoi de courrier électronique (e–mail)
- Emulation d'un terminal ou connexion à un autre ordinateur.
- Transfert de données.
- Exécution de programmes résidant sur un nœud distant.

L'application de réseau la plus répandue est la messagerie électronique, qui permet aux utilisateurs d'échanger des messages. Les utilisateurs peuvent se trouver sur le même système (dans ce cas un réseau n'est pas nécessaire), sur des systèmes différents situés dans des immeubles différents, voire des pays différents.

Un réseau de communication permet également à un système d'en simuler un autre de façon à accéder aux informations, comme s'il était un autre type d'ordinateur ou de terminal. La connexion par le réseau à un système distant donne accès aux mêmes programmes et fichiers qu'avec une connexion locale sans réseau.

La connexion par le réseau à un système distant donne accès aux mêmes programmes et fichiers qu'avec une connexion locale sans réseau. Les données sont transférables par le réseau d'un système à un autre, qu'il s'agisse de fichiers, de répertoires ou de systèmes de fichiers complets. Elles peuvent être sauvegardées à distance et dupliquées sur plusieurs machines pour parer aux défaillances matérielles.

Il existe plusieurs protocoles permettant aux applications et aux utilisateurs d'un système d'appeler des procédures et des applications d'un autre système, ce qui est utile pour répartir la charge des routines particulièrement lourdes.

Présentation des réseaux

La complexité des réseaux informatiques modernes a donné lieu à plusieurs modèles conceptuels de réseaux. Le plus connu est le modèle de référence pour l'interconnexion des systèmes ouverts (OSI) proposé par l'organisation internationale de normalisation (ISO), aussi appelé modèle OSI en sept couches. Les sept couches du modèle OSI sont numérotées comme suit :

7 Application
6 Présentation
5 Session
4 Transport
3 Réseau
2 Liaison
1 Physique

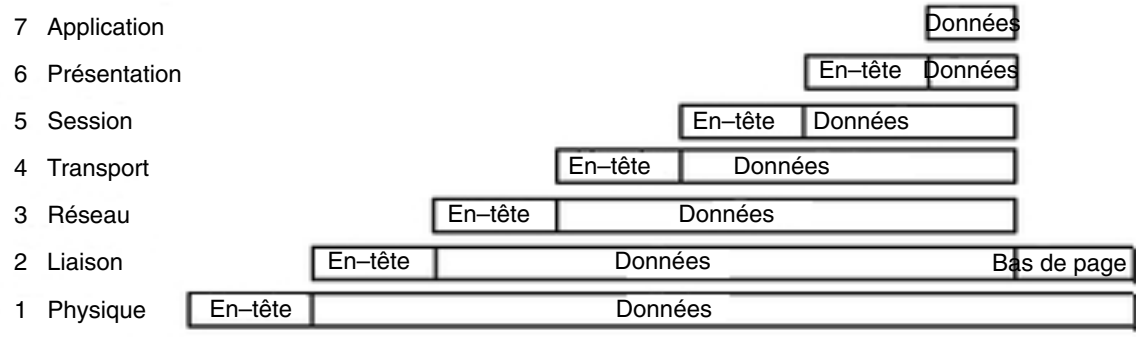
Les niveaux 1 à 3 sont propres aux réseaux et varient en fonction du réseau physique utilisé. Les niveaux 4 à 7 couvrent les fonctions de haut niveau, indépendantes du réseau. Chacune de ces couches décrit une fonction de communication spécifique et non un protocole donné. Les sept couches fonctionnent du niveau le plus bas (niveau machine) au niveau le plus haut (le niveau auquel la plupart des échanges humains interviennent), comme suit :

Application	Englobe les applications qui utilisent le réseau.
Présentation	Met en forme les données pour les rendre cohérentes pour les applications.
Session	Gère les connexions entre les applications.
Transport	Assure l'acheminement des données sans erreur.
Réseau	Gère les connexions aux autres machines du réseau.
Liaison	Assure la transmission des données à travers la couche physique (qui, par nature, n'est pas fiable).
Physique	Décrit les supports physiques du réseau. Par exemple, le câble à fibre optique requis pour un réseau FDDI (Fiber Distributed Data Interface) fait partie de la couche physique.

Remarque : Le modèle de référence OSI, utile pour la présentation conceptuelle, n'est en pratique pas toujours scrupuleusement suivi par les protocoles de réseau. Par exemple, lors de la discussion du protocole TCP/IP, les fonctions des couches Application et Présentation peuvent être combinées, de même que les couches Session et Transport, ainsi que les couches Liaison et Physique.

Chaque couche du modèle OSI communique avec la couche équivalente sur la machine distante (cf. figure Modèle de référence OSI). Elle transmet les données uniquement aux couches situées immédiatement au-dessus ou au-dessous d'elle. Chaque couche encapsule les informations héritées des couches supérieures et ajoute ses propres informations d'en-tête (et de fin pour la couche Liaison).

Figure 1. Modèle de référence OSI Cette illustration montre les différents niveaux de communication du Modèle OSI décrits dans le texte précédent.



Les réseaux offrent nombre de possibilités aux entreprises et aux particuliers :

- Entrée de données,
- Recherche de données,
- Soumission par lots à distance,
- Partage des ressources
- Partage des données,
- Courrier électronique.

Une entrée de données correspond à une importation de données directement dans des fichiers de données locaux ou distants. Par là-même, les risques d'échec ou d'erreur liés à un transfert en plusieurs étapes sont réduits. La recherche des données consiste à examiner des fichiers de données pour obtenir des informations particulières. Leur mise à jour consiste à modifier, ajouter ou supprimer des informations stockées dans des fichiers locaux ou distants. La soumission par lots à distance consiste à entrer à distance des trains de données traités le plus souvent durant la nuit ou une période de faible activité. Pour toutes ces fonctions, les communications et réseaux se révèlent non seulement souhaitables mais aussi indispensables.

Les réseaux autorisent également le partage des ressources : données, programmes, espace de stockage et périphériques (tels que les imprimantes, les modems, les terminaux et les disques inamovibles). Cette particularité accroît à la fois la rentabilité du système (périphériques partagés) et sa fonctionnalité (une seule copie des programmes et fichiers, évitant ainsi tout problème de cohérence inhérent aux copies multiples).

Réseaux physiques

Le réseau physique est constitué par l'ensemble des câbles (coaxiaux, à paire torsadée, optiques ou téléphoniques) qui relient les unités matérielles, les cartes des systèmes hôtes raccordés et les éventuels concentrateurs, répéteurs, routeurs et ponts utilisés sur le réseau. (Le terme *hôte* est employé dans le sens d'ordinateur connecté au réseau.)

Les réseaux physiques varient en fonction de leur taille et du type de matériel qui les composent. On distingue généralement les *réseaux locaux* (LAN) des *réseaux longue distance* (WAN). Un réseau local couvre une zone géographiquement réduite (1 à 10 km), comme par exemple un immeuble de bureaux, un entrepôt, un campus, par opposition au réseau longue distance (WAN) qui dessert une zone plus vaste (pays, continent, etc.) Un réseau longue distance (WAN) fournit des communications de données dans une zone plus vaste (pays, continent, etc.) que celle desservie par un réseau local (LAN). Un modèle intermédiaire de réseau existe également, appelé *metropolitan area networks* (MAN). Dans ce guide, les réseaux MAN sont généralement englobés dans les réseaux WAN.

Les réseaux locaux utilisent généralement des équipements Ethernet standard, IEEE 802.3 ou en anneau à jeton, et les réseaux longue distance et asynchrones utilisent les moyens de communication fournis par les entreprises de télécommunications. Dans les deux cas, les opérations effectuées sur le réseau physique sont généralement soumises à des normes de communications réseau telles que l'EIA (Electronics Industry Association) ou l'ITU (International Telecommunication Union).

Systèmes réseau et protocoles

Toute communication sur un réseau requiert un support matériel et logiciel. *Le matériel* est l'équipement physique connecté au réseau physique. *Le logiciel* regroupe les programmes et les pilotes de périphérique utilisés pour l'exploitation d'un système.

L'équipement matériel d'un système comprend les cartes et autres dispositifs qui donnent accès ou font office d'interface entre la partie logicielle du système et le réseau physique. Chacune de ces cartes doit être installée à un emplacement de carte d'entrée/sortie (E/S) sur le système. D'autres dispositifs, tels que les modems, peuvent être raccordés à un port standard de l'ordinateur.

Ces cartes sont compatibles avec les normes du réseau physique (par exemple, EIA 232D, Smartmodem, V.25 bis, EIA 422A, X.21 ou V.35) et avec les *protocoles* utilisés (par exemple, les protocoles SDLC, HDLC et bisynchrones). Le support logiciel, s'il n'est pas intégré à la carte, est fourni par le pilote de la carte.

Protocoles

Tout logiciel de communication fait appel à un *protocole* (ou plusieurs), ensemble de règles sémantiques et syntaxiques qui définissent comment les unités fonctionnelles assurent la communication : livraison de l'information, conditionnement des données pour en assurer l'intégrité jusqu'à destination, et chemin d'accès. Les protocoles se chargent également de coordonner le flux de messages et leur acquittement.

Les protocoles interviennent à différents niveaux du noyau et ne peuvent être manipulés directement. Leur activation s'effectue en fonction des programmes sollicités par l'utilisateur au niveau de l'interface de programmation d'application (API) lors de l'exécution des tâches (transfert de fichiers, connexion à distance, émulation de terminal, etc.).

Adresses

Les *adresses* associées à la fois au logiciel et au matériel, indiquent à la station expéditrice ou au poste de contrôle comment identifier la station du destinataire : elles permettent de localiser les emplacements de stockage et de réception. Une adresse physique est un code unique attribué à chaque unité ou station connectée à un réseau.

Par exemple, sur un réseau en anneau à jeton, la commande **netstat –iv** affiche l'adresse de la carte dédiée à ce type de réseau. Il s'agit de l'adresse physique. La commande **netstat –iv** procure également des informations d'adressage au niveau de l'utilisateur et de la classe. Les adresses sont souvent définies par le logiciel, mais il arrive qu'elles soient créées également par l'utilisateur.

Domaines

Liée au concept d'adresse, la notion de *domaine* est commune à un grand nombre de réseaux de communication. La structure d'Internet, par exemple, illustre comment les domaines définissent l'adresse IP (Internet Protocol). Internet est un réseau extensif qui regroupe de nombreux réseaux de moindre envergure. Les adresses Internet sont structurées hiérarchiquement en domaines pour faciliter le routage et l'adressage. Au sommet de la structure se trouvent les catégories les plus générales, par exemple `com` pour le secteur commercial, `edu` pour le secteur de l'enseignement et `gov`.

Le domaine `com` est divisé en domaines plus restreints correspondant aux entreprises individuelles, `ibm`, par exemple. Ce domaine `ibm.com` est à son tour divisé en sous-domaines qui, cette fois, correspondent aux adresses Internet des divers sites, par exemple `austin.ibm.com` ou `raleigh.ibm.com`. C'est à ce niveau que commencent à apparaître le nom des *hôtes*. Dans ce contexte, les hôtes sont les ordinateurs connectés au réseau. Par exemple, le domaine `austin.ibm.com` peut comporter les systèmes `hamlet` et `lear`, aux adresses respectives `hamlet.austin.ibm.com` et `lear.austin.ibm.com`.

Passerelles et ponts

Le réseau Internet regroupe une grande variété de réseaux faisant intervenir divers matériels et logiciels. La communication entre ces réseaux hétérogènes s'effectue par le biais de *passerelles* et de *ponts*. Un pont est une unité fonctionnelle qui relie deux réseaux locaux pouvant utiliser la même procédure de contrôle de liaison logique (LLC), Ethernet par exemple, mais des procédures de contrôle d'accès au support (MAC) différentes. La passerelle couvre un champ plus large : elle intervient au–dessus de la couche Liaison et assure, s'il y a lieu, la conversion des protocoles et des interfaces pour permettre à deux protocoles de communiquer entre eux. Elle permet le transfert des données à travers les divers réseaux qui composent Internet.

Routage

L'utilisation de noms de domaines pour l'adressage et de passerelles pour le transfert facilite grandement le *routage*, opération qui consiste à définir le parcours d'un message jusqu'à sa destination. En effet, c'est le nom du domaine qui définit la destination : dans un réseau étendu comme Internet, l'information est acheminée d'un réseau de communication au suivant jusqu'à destination. Chacun des réseaux vérifie le nom du domaine en fonction de ceux qu'il connaît et achemine l'information, jusqu'à l'extrémité logique suivante. Ainsi, chaque réseau par lequel les données transitent participe au processus de routage.

Noeud local et noeud distant

Un réseau physique est utilisé par les systèmes hôtes qui y résident. Chacun de ces systèmes hôtes est un *noeud* sur le réseau. C'est-à-dire un point adressable du réseau qui offre des services de traitement hôte. L'intercommunication entre ces différents nœuds engendre les notions de *local* et de *distant*. *Local* s'applique aux unités, fichiers ou systèmes directement accessibles à partir de votre système, sans recourir à une ligne de communication. *Distant* s'applique aux unités, fichiers ou systèmes accessibles à partir de votre système via une ligne de communication. En effet, les fichiers locaux sont implantés sur votre système alors que les fichiers distants résident sur un serveur de fichiers ou un autre noeud accessible via un réseau physique, par exemple, un réseau Ethernet, un réseau en anneau à jeton ou des lignes téléphoniques.

Client et serveur

Les concepts de client et de serveur sont liés aux notions de "local" et de "distant". Un serveur est un ordinateur qui contient des données ou fournit des services accessibles aux autres ordinateurs du réseau. Les types de serveur les plus courants sont les serveurs de fichiers dans lesquels sont stockés des fichiers, les serveurs de noms qui stockent les noms et adresses, les serveurs d'application qui stockent les programmes et applications et les serveurs d'impression, qui planifient et dirigent les travaux d'impression vers leur destination.

Un client est un ordinateur qui sollicite des services ou des données auprès d'un serveur. Par exemple, un client peut demander un code de programme mis à jour ou une application auprès d'un serveur de code. Pour obtenir un nom ou une adresse, le client contacte un serveur de noms. Un client peut également interroger un serveur de fichiers pour retrouver des fichiers et des données, et effectuer des opérations (saisie de données, recherches, mise à jour d'articles).

Communication avec d’autres systèmes d’exploitation

Un réseau peut relier divers types d’ordinateurs (modèles ou constructeurs hétérogènes). Des programmes de communication sont alors utilisés pour pallier les disparités entre les systèmes d’exploitation installés sur ces machines.

Parfois, ces programmes nécessitent l’installation préalable d’un autre programme sur le réseau. Certains programmes peuvent nécessiter la présence sur le réseau d’un autre programme ou de protocoles de connexion (tels que TCP/IP ou SNA).

Avec les versions AIX 4.3.2 et ultérieures, par exemple, AIX Fast Connect for Windows permet aux clients PC d’accéder aux fichiers et aux imprimantes à l’aide du logiciel réseau du client PC natif. Les utilisateurs PC peuvent accéder directement aux systèmes de fichiers distants à partir de leurs machines comme si ces fichiers étaient sauvegardés en local. De plus, ils peuvent lancer des tâches d’impression sur des imprimantes utilisant le système de spoulage, visualiser celles qui sont disponibles et configurer une imprimante en réseau. Pour plus d’informations sur AIX Fast Connect, reportez–vous au *AIX Fast Connect for Windows Version 3.1 Guide*.

Chapitre 3. Messagerie électronique

La messagerie fournit un outil d'échange du courrier électronique (e–mail) entre les utilisateurs d'un même système ou de systèmes distincts connectés via un réseau. Ce chapitre décrit le système de messagerie, l'interface utilisateur standard, et les protocoles IMAP (Internet Message Access Protocol) et POP (Post Office Protocol).

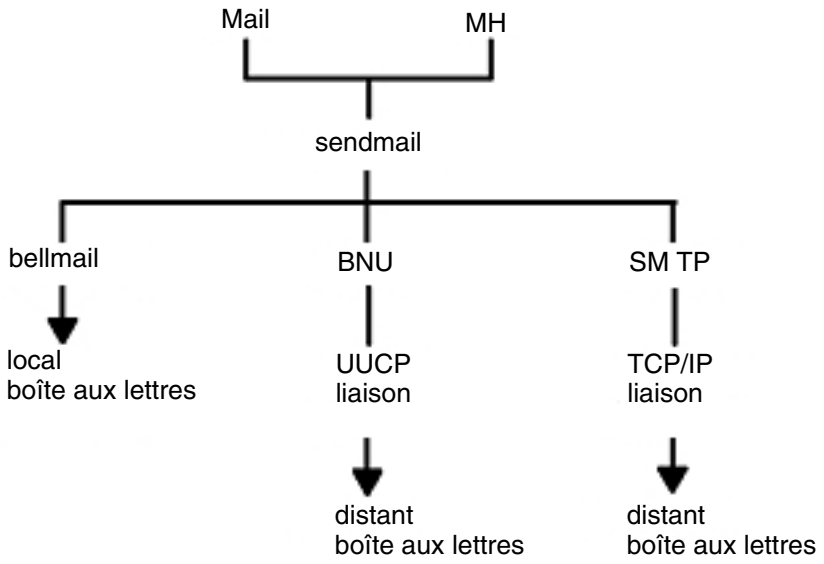
La messagerie, outil de livraison de messages interréseau, comprend une interface utilisateur, un programme de routage et un programme de livraison des messages (aussi appelé programme facteur). L'acheminement des messages est assuré entre deux utilisateurs d'un même système hôte ou de systèmes hôtes ou réseaux différents. L'outil comporte également une fonction d'édition limitée pour présenter les en-têtes dans un format reconnu par l'hôte récepteur.

Une *interface utilisateur* permet aux utilisateurs de créer, envoyer et recevoir des messages. La messagerie propose deux interfaces : **mail** et **mhmail**. La commande **mail** est l'interface utilisateur standard des systèmes UNIX. La commande **mhmail** est l'interface utilisateur du gestionnaire de message (MH). Plus évoluée, cette dernière s'adresse aux utilisateurs chevronnés.

Un *programme de routage des messages* sert à acheminer les messages jusqu'à destination. Dans la messagerie présentée ici, il s'agit du programme **sendmail**. Ce programme, intégré au système d'exploitation de base (BOS), est installé avec ce dernier. **Sendmail** est un démon qui utilise les informations des fichiers **/etc/mail/sendmail.cf** et **/etc/mail/aliases** pour effectuer le routage nécessaire.

Remarque : Dans les versions antérieures à AIX 5.1, les fichiers **sendmail.cf** et **aliases** sont respectivement situés dans **/etc/sendmail.cf** et **/etc/mail/aliases**.

En fonction de la route, la commande **sendmail** fait appel à différents *programmes facteur* pour livrer les messages.



Comme l'illustre la figure :

- Pour acheminer un courrier local, le programme **sendmail** achemine les messages au programme **bellmail**. Le programme **bellmail** transmet le courrier au système local, dans la boîte aux lettres système de l'utilisateur, située dans le répertoire **/var/spool/mail**.
- Pour acheminer le courrier via une liaison réseau UUCP, le programme **sendmail** achemine les messages à l'aide de BNU (Basic Network Utilities).
- Pour transmettre un courrier via TCP/IP, la commande **sendmail** établit une connexion TCP/IP au système distant et utilise le protocole SMTP (Simple Mail Transfer Protocol) pour effectuer le transfert.

Gestion du courrier

L'administrateur du courrier est responsable de l'exécution des tâches suivantes :

1. Pour que **sendmail** soit exécuté à l'amorçage du système, configurez le fichier **/etc/rc.tcpip** comme suit. Reportez–vous à Configuration du fichier /etc/rc.tcpip pour lancer le démon sendmail
2. Personnaliser le fichier de configuration **/etc/mail/sendmail.cf**. Par défaut, **/etc/mail/sendmail.cf** est défini pour permettre la livraison du courrier local et du courrier TCP/IP. Le fichier **/etc/mail/sendmail.cf** doit être modifié pour pouvoir acheminer le courrier via une liaison BNU. Pour en savoir plus, reportez-vous à "sendmail.cf File" dans le document *AIX Version 5.2 Files Reference*.
3. Définir les alias aux niveaux système et domaine dans le fichier **/etc/mail/aliases**. Pour en savoir plus, reportez-vous à "Gestion des alias".
4. Gérer les files d'attente de messages. Pour plus de détails, reportez vous à "Gestion des fichiers et répertoires de file d'attente courrier"e.
5. Gérer le journal des messages. Pour en savoir plus, reportez–vous à "Gestion de la journalisation".

Configuration du fichier /etc/rc.tcpip pour lancer le démon sendmail

Pour que **sendmail** soit exécuté à l'amorçage du système, configurez le fichier **/etc/rc.tcpip** comme suit :

1. Modifiez le fichier **/etc/rc.tcpip** avec l'éditeur de votre choix.
2. Recherchez la ligne introduite par **start /usr/lib/sendmail**. Par défaut, cette ligne ne doit pas être en commentaire, c'est-à-dire précédée du signe #. Si ce signe figure en début de ligne, supprimez-le.
3. Sauvegardez le fichier.

Le démon **sendmail** sera exécuté à l'amorçage du système.

Gestion des alias

Les alias mettent en correspondance des noms et des listes d'adresses par le biais de fichiers personnels, système ou domaine. Il existe trois types d'alias :

personnel	Défini par l'utilisateur dans son fichier **$HOME/.mailrc**.
système local	Défini par l'administrateur du système de messagerie dans le fichier **/etc/mail/aliases**. Les alias de ce type s'appliquent au courrier traité par le programme **sendmail** sur le système local. Ils ont rarement besoin d'être modifiés.
domaine	Par défaut, le programme **sendmail** lit **/etc/alias** pour convertir les alias. Pour effacer les paramètres par défaut et utiliser NIS, modifiez ou créez la commande **/etc/netsvc.conf** et ajoutez la ligne : `aliases=nis`

Fichier /etc/mail/aliases

Le fichier **/etc/mail/aliases** se compose d'une série d'entrées au format suivant :

Alias: *Nom1, Nom2, ... NomX*

Alias étant une chaîne alphanumérique de votre choix (sans caractères spéciaux, tels que @ et !). Les variables *Nom1* à *NomX* représentent une liste de noms de destinataires. Cette liste de noms peut s'étendre sur plusieurs lignes. Chaque ligne de suite doit commencer par un espace ou une tabulation. Les lignes blanches ou précédées d'un dièse (#) sont des commentaires.

Le fichier **/etc/mail/aliases** doit comporter les trois alias suivants :

MAILER-DAEMON	ID de l'utilisateur destinataire des messages adressés au démon du programme facteur. Ce nom est attribué initialement à l'utilisateur racine : `MAILER-DAEMON: root`
postmaster	ID de l'utilisateur chargé de l'exploitation de la messagerie locale. L'alias **postmaster** définit une adresse de boîte aux lettres unique valable sur chaque système du réseau. Cette adresse permet d'envoyer des requêtes à l'alias **postmaster** à partir de n'importe quel système, sans connaître l'adresse exacte de l'utilisateur sur ce système. Ce nom est attribué initialement à l'utilisateur racine : `postmaster: root`
nobody	ID destinataire des messages adressés aux programmes tels que **news** et **msgs**. Ce nom est attribué initialement à **/dev/null** : `nobody: /dev/null` Pour recevoir ces messages, déclarez l'alias comme utilisateur valide.

A chaque modification du fichier, vous devez le recompiler dans un format de base de données exploitable par la commande **sendmail**. Reportez-vous à Création d'une base de données d'alias.

Création d'alias de système local

Pour obtenir des instructions pas à pas sur la création d'un local system alias, reportez–vous à Créer un alias local de messagerie page 1-21.

Création d'une base de données d'alias

La commande **sendmail** n'utilise pas directement les définitions d'alias dans le fichier **/etc/mail/aliases** du système local. Elle fait appel à une version de ce fichier générée par le gestionnaire de base de données. Pour compiler la base de données d'alias, vous avez le choix entre les méthodes suivantes :

- Lancez la commande **/usr/sbin/sendmail** assortie de l'indicateur **-bi**.
- Exécutez la commande **newaliases**. Cette commande provoque la lecture, par **sendmail**, du fichier **/etc/mail/aliases** du système local et la création d'un nouveau fichier contenant les informations de la base d'alias. Ce fichier est au format Berkeley, plus efficace :

  /etc/mail/aliases.db

  (Les versions antérieures à AIX 5.1 créaient deux fichiers de bases de données, **/etc/aliases.dir** et **/etc/aliases.pag**.)
- Lancez la commande **sendmail** assortie de l'indicateur **Rebuild Aliases**. Cette commande reconstruit automatiquement la base de données d'alias lorsqu'elle est périmée. Le reconstruction automatique peut être dangereuse sur des machines très chargées, contenant de gros fichiers d'alias. Si la reconstruction dure plus longtemps que le délai imparti (normalement, 5 minutes), il y a des chances que plusieurs processus la lancent simultanément.

1. Sans ces fichiers, la commande **sendmail** ne peut pas traiter le courrier et génère un message d'erreur.
2. Si plusieurs bases de données d'alias sont spécifiées, l'indicateur **-bi** reconstruit tous les types qu'il peut interpréter (il peut, par exemple, reconstruire les bases de données NDBM, et non les bases NIS).

Le fichier **/etc/netsvc.conf** contient l'ordonnancement des services système. Pour spécifier l'ordonnancement des services des alias, ajoutez la ligne suivante :

```
aliases=service, service
```

`service` pouvant être `files` ou `nis`. Par exemple:

```
aliases=files, nis
```

Indique à la commande **sendmail** de tenter d'abord le fichier d'alias local puis, en cas d'échec, d'essayer `nis`. Si `nis` est défini comme un service, il doit être actif.

Pour plus d'informations sur le fichier **/etc/ netsvc.conf**, reportez–vous à *AIX 5L Version 5.2 Files Reference*.

Le fichier **/etc/netsvc.conf** contient l'ordonnancement des services système. Pour spécifier l'ordonnancement des services des alias, ajoutez la ligne suivante :

```
aliases=service, service
```

`service` pouvant être `files` ou `nis`. Par exemple :

```
aliases=files, nis
```

indique à la commande **sendmail** de tenter d'abord le fichier d'alias local puis, en cas d'échec, d'essayer `nis`. Si `nis` est défini comme un service, il doit être actif.

Pour en savoir plus sur le fichier **/etc/netsvc.conf**, reportez-vous à *AIX Files Reference*.

Gestion des fichiers et répertoires de file d'attente courrier

La file d'attente courrier est un répertoire qui stocke des données et gère les files d'attente de messages distribués par la commande **sendmail**. Son nom par défaut est **/var/spool/mqueue**.

Les messages peuvent être mis en attente pour diverses raisons : si la commande **sendmail** est configurée pour exécuter la file d'attente à intervalles réguliers et non immédiatement, les messages y sont stockés temporairement. Par ailleurs, si un système hôte distant ne répond pas à une demande de connexion courrier, la messagerie met les messages en attente en vue d'une tentative ultérieure.

Impression de la file d'attente courrier

Pour imprimer le contenu de la file d'attente, lancez la commande **mailq** (ou spécifiez l'indicateur **-bp** avec la commande **sendmail**).

Une liste des ID de file d'attente est générée, indiquant la taille de chaque message, la date de son insertion dans la file et les noms d'expéditeur et de destinataire.

Fichiers de file d'attente courrier

Chaque message en attente est associé à un certain nombre de fichiers, désignés par :

`TypefID`

`ID` est l'ID unique de file d'attente et *`Type`*, le type du fichier symbolisé par une lettre :

- **d** Fichier de données contenant le corps du texte du message sans l'en-tête.
- **q** Fichier de contrôle de file d'attente contenant les informations utiles au traitement du travail.
- **t** Fichier temporaire correspondant à l'image du fichier **q** lors de sa reconstitution. Très vite renommé **q**.
- **x** Fichier de transcription créé pour la durée d'une session, dans lequel sont consignés tous les événements de la session.

Par exemple, soit le message portant l'ID de file d'attente AA00269, les fichiers suivants sont générés et supprimés du répertoire de file d'attente courrier pendant que **sendmail** tente de livrer ce message :

- **dfAA00269** Fichier de données
- **qfAA00269** Fichier de contrôle
- **tfAA00269** Fichier temporaire
- **xfAA00269** Fichier de transcription

Fichier de contrôle q

Ce fichier contient une série de lignes commençant par les lettres suivantes :

B Spécifie le `body type`. Le reste de la ligne est une chaîne de texte définissant le `body type`. En l'absence de ce champ, le `body type` est de 7 bits par défaut et aucun traitement particulier n'est entrepris. Valeurs possibles : *7BIT* et *8BITMIME*.

C Contient l'adresse de contrôle. Pour les adresses de destinataires constituées d'un fichier ou d'un programme, **sendmail** se comporte comme le propriétaire du fichier ou du programme. L'utilisateur de contrôle devient propriétaire du fichier ou du programme. Les adresses de destinataires sont lues à partir d'un fichier **.forward** ou **:include:** comportant aussi un utilisateur de contrôle propriétaire du fichier. **sendmail** délivre les messages à ces destinataires en tant qu'utilisateur de contrôle, puis retourne à la racine.

F Contient des bits d'indicateur. Les indicateurs sont une combinaison de **w**, indiquant le message d'avertissement **EF_WARNING**, **r**, indiquant le message de réponse **EF_RESPONSE**, **8**, définissant l'indicateur **EF_HAS8BIT** et **b**, définissant l'indicateur **EF_DELETE_BCC**. Les autres lettres sont ignorées.

H Ligne(s) contenant la définition de l'en-tête. Le nombre de lignes est indifférent. L'ordre d'apparition des lignes `H` détermine leur disposition dans le message final. Elles utilisent la syntaxe de définition des en-têtes appliquée dans le fichier **/etc/mail/sendmail.cf**. (Pour les versions antérieures à AIX 5.1, ce fichier est **/etc/sendmail.cf**.)

I Définit l'information sur le i–node et l'unité pour le fichier **df**. Utile pour recouvrer la file d'attente courrier après un crash de disque.

K Heure (en secondes) de la dernière tentative de distribution.

M Lorsqu'un message est mis en file d'attente suite à une erreur lors d'une tentative de livraison, le type d'erreur est stocké sur la ligne **M**.

N Nombre total de tentatives de distribution.

O Spécifie la valeur MTS originale de la transaction ESMTP. Utilisé exclusivement pour les Notifications d'état de distribution.

P Ligne précisant le niveau de priorité du message courant, lequel détermine l'ordre d'exécution des messages en file d'attente. Plus le numéro est élevé, plus la priorité est basse, autrement dit, la priorité croît à mesure que l'on descend dans la liste des messages. Autrement dit, la priorité croît à mesure que l'on descend dans la liste des messages. Le niveau de priorité initial est fonction de la classe et de la taille du message.

Q Destinataire initial tel que spécifié par le champ `ORCPT=` dans une transaction ESMTP. Utilisé exclusivement pour les Notifications d'état de distribution. Il ne s'applique qu'à la ligne "R" figurant immédiatement après.

R Lignes comportant chacune une adresse de destinataire.

S Lignes comportant chacune une adresse d'expéditeur.

T Ligne indiquant l'heure de création, qui sert à calculer le délai de rétention du message en file d'attente.

V Numéro de version du format de fichier de file d'attente utilisé pour que les nouveaux fichiers binaires **sendmail** puissent lire les fichiers créés sous les versions antérieures. Valeur par défaut : **zero**. Si présent, doit figurer sur la première ligne du fichier.

Z ID enveloppe initiale (issu de la transaction SMTP). Utilisé exclusivement pour les Notifications d'état de distribution.

$** Contient une définition de macro. Les valeurs de certaines macros (**$r et **$s**) sont passées au cours de la phase d'exécution de la file.

Le fichier **q** associé au message adressé à `amy@zeus` se présenterait comme suit :

```
P217031
 T566755281
 MDeferred: Connection timed out during user open with zeus
 Sgeo
 Ramy@zeus
 H?P?return-path: <geo>
 Hreceived: by george (0.13 (NL support)/0.01)
    id AA00269; Thu, 17 Dec 87 10:01:21 CST
 H?D?date: Thu, 17 Dec 87 10:01:21 CST
 H?F?From: geo
 Hmessage-id: <8712171601.AA00269@george>
 HTo: amy@zeus
 Hsubject: test
```

où :

`P217031`	Priorité du message
`T566755281`	Temps de soumission en secondes
`MDeferred: Connection timed out during user open with zeus`	Message d'état
`Sgeo`	ID de l'expéditeur
`Ramy@zeus`	ID du destinataire
`HLines`	Informations d'en-tête du message.

Spécification des délais au démon sendmail

Un format horaire spécial est prévu pour spécifier les délais associés au message et les intervalles de traitement des files d'attente. Ce format est le suivant :

`-q`*`NombreUnité`*

Nombre est un entier et *Unité* est une des lettres symbolisant l'unité utilisée. *Unité* peut avoir l'une des valeurs suivantes :

s	secondes
m	minutes
h	heures
d	jours
w	semaines

L'unité de temps par défaut est la minute (**m**). Voici trois exemples :

```
/usr/sbin/sendmail -q15d
```

Avec cette commande, **sendmail** traite la file d'attente tous les 15 jours.

```
/usr/sbin/sendmail -q15h
```

Avec cette commande, **sendmail** traite la file d'attente toutes les 15 heures.

```
/usr/sbin/sendmail -q15
```

Avec cette commande, **sendmail** traite la file d'attente toutes les 15 minutes.

Exécution forcée de la file d'attente courrier

Si vous trouvez qu'une file d'attente commence à saturer, vous pouvez forcer son exécution par le biais de l'indicateur **–q** (sans autre valeur). Vous pouvez également spécifier l'indicateur **–v** (verbose) pour voir ce qui se passe :

```
/usr/sbin/sendmail -q-v
```

Vous pouvez également limiter les travaux à ceux dotés d'un identificateur de file, d'un expéditeur ou d'un destinataire donné, via l'un des modificateurs de file d'attente. Par exemple, **-qRsally** limite l'exécution de la file d'attente aux travaux dont l'adresse d'un des destinataires contient la chaîne **sally**. De même, **-qS** *chaîne* limite l'exécution à quelques expéditeurs et **-ql** *chaîne,* à quelques identificateurs de file d'attente.

Intervalle de traitement de la file d'attente

L'intervalle de traitement de la file d'attente courrier par le démon **sendmail** est déterminé par l'indicateur **-q**, qui est pris en compte au lancement du démon.

Généralement, **sendmail** est lancé par le fichier **/etc/rc.tcpip** au démarrage du système. Ce fichier contient la variable QPI (Queue Processing Interval), qui sert à attribuer une valeur à l'indicateur **-q** à l'exécution du démon **sendmail**. Par défaut, la valeur de **qpi** est 30 minutes. Pour la modifier :

1. Modifiez le fichier **/etc/rc.tcpip** avec l'éditeur de votre choix.
2. Recherchez la ligne qui définit cette valeur, par exemple :

   ```
   qpi=30m
   ```

3. Changez la valeur de *qpi* en fonction de vos besoins.

Ces modifications prendront effet au prochain lancement du système. Pour une prise en compte immédiate, arrêtez puis relancez le démon **sendmail** en spécifiant la nouvelle valeur pour l'indicateur **–q**. Pour plus d'informations, reportez-vous à "Arrêt du démon sendmail", page3-10 et "Lancement du démon sendmail", page3-10.

Transfert de file d'attente courrier

Si un système hôte est hors service pendant quelques temps, de nombreux messages envoyés ou en transit sur ce système sont peut-être stockés dans votre file d'attente courrier. Ce phénomène alourdit le traitement de la file d'attente au détriment des performances de votre système. Dans ce cas, vous avez la possibilité de transférer temporairement la file d'attente vers un autre emplacement et d'en créer une nouvelle. Vous pourrez ainsi traiter l'ancienne file une fois le système hôte remis en service. Pour effectuer ces opérations :

1. Arrêtez le démon **sendmail** comme indiqué à "Arrêt du démon sendmail", page 3-10.
2. Déplacez la totalité du répertoire de file d'attente :

   ```
   cd/var/spool
    mv mqueue omqueue
   ```

3. Relancez **sendmail** comme indiqué à "Lancement du démon sendmail", page 3-10.
4. Pour traiter l'ancienne file d'attente, entrez :

   ```
   /usr/sbin/sendmail -oQ/var/spool/omqueue -q.
   ```

   L'indicateur **–oQ** désigne le répertoire temporaire de la file transférée, et l'indicateur **–q** demande l'exécution de tous les travaux de la file. Pour obtenir un compte rendu du déroulement des opérations, précisez **-v**.

   Remarque : Cette opération peut durer un certain temps.

5. Supprimez fichiers journaux et répertoire temporaire une fois la file d'attente vidée :

```
rm /var/spool/omqueue/*
 rmdir /var/spool/omqueue
```

Lancement du démon sendmail

Pour lancer le démon **sendmail**, entrez :

```
startsrc –s sendmail –a "–bd –q15"
```

```
/usr/lib/sendmail –bd –q15
```

Si **sendmail** est déjà activé à l'exécution de ces commandes, un message vous indique que le démon ne peut être lancé plusieurs fois :

```
The sendmail subsystem is already active. Multiple instances are no
t supported.
```

Sinon, un message vous confirme le lancement du démon.

Arrêt du démon sendmail

Pour arrêter le démon **sendmail**, exécutez la commande **stopsrc -s sendmail**. Sinon :

- Recherchez l'ID de processus de **sendmail**.
- Saisissez la commande **kill** *sendmail_pid*
  (où *sendmail_pid* est l'ID de processus du processus **sendmail**).

Gestion de la journalisation

La commande **sendmail** consigne dans un journal les activités de la messagerie en faisant appel au démon **syslogd**. Le démon **syslogd** doit être configuré et exécuté pour permettre la journalisation. Dans le fichier **/etc/syslog.conf** notamment, la ligne ci-après doit être activée (et non mise en commentaire) :

```
mail.debug              /var/spool/mqueue/log
```

Si elle est désactivée, modifiez-la à l'aide de l'éditeur de votre choix, en prenant soin d'indiquer le chemin d'accès correct. Si vous modifiez le fichier **/etc/syslog.conf** au cours de l'exécution du démon **syslogd**, vous devez régénérer le démon comme suit :

```
refresh -s syslogd
```

Si le fichier **/var/spool/mqueue/log** n'existe pas, vous devez le créer via la commande :

```
touch /var/spool/mqueue/log
```

Les messages sont consignés dans le fichier journal au format suivant :

Chaque ligne d'un journal système comporte un horodateur, le nom de la machine qui l'a généré (pour les journaux concernant plusieurs machines d'un réseau local), le mot "`sendmail:`," et un message. La plupart des messages sont constitués d'une série de paires *nom=valeur*.

Deux lignes communes sont consignées lorsqu'un message est traité. La première indique la réception d'un message : il y en a une par message. Certains champs peuvent être omis. Les champs du message sont les suivants :

`from`	Adresse de l'expéditeur de l'enveloppe.
`size`	Taille du message (en octets).
`class`	Classe (priorité numérique) du message.
`pri`	Priorité initiale du message (pour le tri des files d'attente).
`nrcpts`	Nombre de destinataires de l'enveloppe pour ce message (après définition d'alias et transmission).
`proto`	Protocole utilisé pour la réception du message (par exemple, ESMTP ou UUCP).
`relay`	Machine d'où provient le message.

Une autre ligne est consignée à chaque **tentative de livraison** (il peut donc y en avoir plusieurs par message - si le message est différé ou qu'il y a plusieurs destinataires). Les champs du message sont les suivants :

`to`	Liste des destinataires, séparés par une virgule.
`ctladdr`	*"Utilisateur contrôleur"*, c'est-à-dire nom de l'utilisateur dont les références sont utilisées pour la livraison.
`delay`	Délai total entre le moment où le message a été reçu et le moment où il a été délivré.
`xdelay`	Durée nécessaire pour cette tentative de livraison.
`mailer`	Nom du programme facteur utilisé pour délivrer à ce destinataire.
`relay`	Nom de l'hôte qui a effectivement accepté (ou rejeté) ce destinataire.
`stat`	Etat de la livraison.

Les informations qui peuvent être consignées sont nombreuses. Le journal est structuré en niveaux. Au niveau le plus bas, seules les situations très inhabituelles sont consignées. Au niveau le plus élevé, même les événements insignifiants le sont. Par convention, les niveaux inférieurs à 10 sont considérés utiles. Les niveaux supérieurs à 64 sont réservés à la mise au point et les niveaux intermédiaires (11–64), dédiés aux informations détaillées.

Les types d'activité consignés par la commande **sendmail** dans le journal sont spécifiés via l'option **L** dans le fichier **/etc/mail/sendmail.cf**. (Pour les versions antérieures à AIX 5.1, ce fichier est **/etc/sendmail.cf**.)

Gestion du journal

Sans cesse alimenté par de nouvelles données, le journal peut prendre des proportions non négligeables. Par ailleurs, il arrive que certains incidents génèrent des entrées inattendues dans la file d'attente courrier. Pour limiter l'encombrement du journal et de la file d'attente, exécutez le script **shell /usr/lib/smdemon.cleanu.** Ce script force la commande **sendmail** à traiter la file d'attente et tient à jour quatre copies des fichiers journaux à des niveaux de mise à jour croissants **log.0**, **log.1**, **log.2** et **log.3**. A chaque exécution du script, le contenu des fichiers est transféré comme suit :

- **log.2** à **log.3**
- **log.1** à **log.2**
- **log.0** à **log.1**
- **log** à **log.0**.

Ces transferts permettent de reprendre la journalisation sur un nouveau fichier. Exécutez le script manuellement ou à intervalle régulier à l'aide du démon **cron.**

Journalisation du trafic

De nombreuses versions de SMTP n'implémentent pas complètement le protocole. Par exemple, certains SMTP basés sur PC ne savent pas interpréter les lignes de suite dans les codes de réponse. Ceci peut être très difficile à déceler. Si vous suspectez un problème de cet ordre, vous pouvez activer la journalisation du trafic via l'indicateur **-X**. Par exemple :

```
/usr/sbin/sendmail-X /tmp/traffic -bd
```

Cette commande consigne l'intégralité du trafic dans le fichier **/tmp/traffic.**

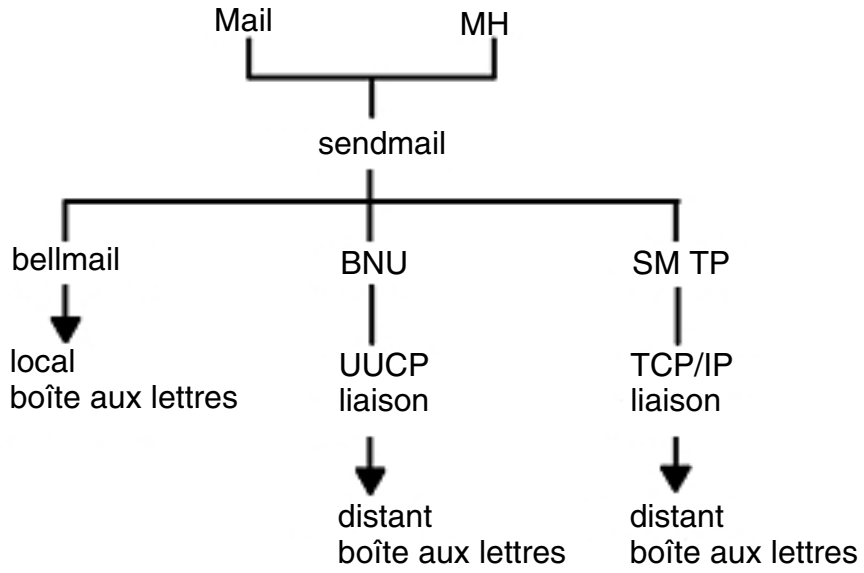
Cette opération consigne une énorme quantité de données en très peu de temps et ne doit jamais être effectuée dans le cadre de l'exploitation normale. Après avoir lancé un démon de ce type, forcez l'implémentation **errant** à envoyer un message à votre hôte. Tout le trafic entrant et sortant de **sendmail**, trafic SMTP entrant compris, sera consigné dans ce fichier.

Via **sendmail**, vous pouvez consigner un cliché des fichiers ouverts et du cache de connexion en lui envoyant un signal **SIGUSR1**. Les résultats sont consignés avec la priorité **LOG_DEBUG.**

Journalisation des données statistiques

La commande **sendmail** assure le suivi du volume de courrier traité par chaque programme facteur qui communique avec la commande. Ces programmes sont définis dans le fichier **/etc/sendmail.cf**. (Pour les versions antérieures à AIX 5.1, ce fichier est **/etc/sendmail.cf**.)

Figure 2. Programmes facteur utilisés par la commande Sendmail Cette illustration représente un type d'organigramme structuré du haut vers le bas avec la messagerie et MH en haut. Il en part des branches correspondant à bellmail, BNU et SMTP. Sous le niveau précédent se trouvent respectivement la boîte aux lettres locale, la liaison UUCP et la liaison TCP/IP. Sous la liaison UUCP et sous la liaison TCP/IP se trouvent des boîtes aux lettres distantes.



Pour lancer la collecte des données statistiques, créez le fichier **/var/tmp/sendmail.st** comme suit :

```
touch /var/tmp/sendmail.st
```

Si la commande **sendmail** rencontre des erreurs pendant l'enregistrement des données statistiques, elle inscrit un message via la sous-routine **syslog**. Ces erreurs n'entravent pas les autres opérations de **sendmail**.

La commande **sendmail** met les informations à jour chaque fois qu'un courrier est traité. La taille du fichier reste égale, mais les nombres dans le fichier augmentent.

Ils représentent le volume de courrier depuis que vous avez créé ou réinitialisé le fichier **/etc/mail/statistics**.

Remarque : Dans les versions antérieures à AIX 5.1, les statistiques étaient conservées dans le fichier **/var/tmp/sendmail.st**.

Affichage des informations des programmes facteurs

Les données statistiques conservées dans le fichier **/etc/mail/statistics** sont sauvegardées sous un format de base de données, et ne peuvent donc être consultées comme un fichier texte. Pour les afficher, tapez ceci à une invite de commande :

```
/usr/sbin/mailstats
```

Cette commande lit les données du fichier **/etc/mail/statistics**, et les formate avant de les envoyer vers la sortie standard. Pour obtenir de plus amples informations sur la sortie de la commande **/usr/sbin/mailstats**, lisez sa description dans la *AIX Version 5.2 Commands Reference*.

Les champs proposés ont la signification suivante :

`msgs_from`	Nombre de messages reçus du programme facteur par la machine locale.
`bytes_from`	Nombre d'octets des messages reçus du programme facteur par la machine locale.
`msgs_to`	Nombre de messages émis par la machine locale à l'aide du programme facteur.
`bytes_to`	Nombre d'octets des messages émis par la machine locale à l'aide du programme facteur.

Si la commande **sendmail** envoie le courrier directement dans un fichier de type **$HOME/dead.letter** ou alias, le décompte des messages et des octets est imputé au programme facteur prog.

Mise au point de sendmail

Il existe de nombreux indicateurs de mise au point, intégrés à la commande **sendmail**. A chaque indicateur sont associés un numéro et un niveau, les niveaux supérieurs indiquant un accroissement des informations. Par convention, les niveaux supérieurs à 9 fournissent tellement d'informations que vous ne souhaiterez pas les consulter - sauf pour mettre au point un module particulier de code source. Les indicateurs de mise au point sont définis via l'indicateur **–d**, comme illustré dans l'exemple ci-dessous :

```
debug-flag:     -d debug-list
debug-list:     debug-flag[.debug-flag]*
debug-flag:     debug-range[.debug-level]
debug-range:    integer|integer-integer
debug-level:    integer
```

Par exemple :

```
-d12            Set flag 12 to level 1
-d12.3          Set flag 12 to level 3
-d3-17          Set flags 3 through 17 to level 1
-d3-17.4        Set flags 3 through 17 to level 4
```

Les indicateurs de mise au point disponibles sont les suivants :

–d0 Mise au point générale.

–d1 Affiche les informations d'envoi.

–d2 Prend fin avec *fini* ( ).

–d3 Indique la charge moyenne.

–d4 Espace disque suffisant.

–d5 Affiche les événements.

–d6 Affiche le courrier non parvenu.

–d7 Nom du fichier de file d'attente.

–d8 Résolution de noms DNS.

–d9 Effectue un suivi des requêtes RFC1413.

–d9.1 Met le nom d'hôte sous forme canonique.

–d10 Affiche le courrier reçu par le destinataire.

–d11 Effectue un suivi des livraisons.

–d12 Affiche le mappage de l'hôte relatif.

–d13 Affiche les livraisons.

–d14 Affiche les virgules du champ d'en-tête.

–d15 Affiche l'activité des requêtes d'obtention (get) du réseau.

–d16 Connexions sortantes.

–d17 Affiche la liste des hôtes MX.

Remarque : Il existe désormais près de 200 indicateurs de mise au point définis dans le programme **sendmail**.

Protocoles IMAP (Internet Message Access Protocol) et POP (Post Office Protocol)

Pour l'accès à distance à la messagerie, il existe deux serveurs de protocole de messagerie électronique basés sur Internet :

- POP (Post Office Protocol),
- IMAP (Internet Message Access Protocol).

Ces deux types de serveur stockent le courrier électronique et y donnent accès. Grâce à ces protocoles, l'ordinateur n'a plus besoin d'être allumé pour la réception du courrier.

Le serveur POP fournissant un système de courrier électronique hors ligne, par le biais du logiciel client POP, le client a accès à distance au serveur de messagerie pour réceptionner son courrier. Il peut télécharger son courrier et, ensuite, soit le supprimer immédiatement du serveur, soit le conserver sur le serveur POP. Le courrier, une fois chargé sur la machine cliente, est traité localement sur cette machine. Le serveur POP autorise l'accès à une boîte aux lettres utilisateur à un seul client à la fois.

Le serveur IMAP propose un "super-ensemble" de fonctions POP, mais avec une autre interface. Le serveur IMAP fournit un service hors ligne, un service en ligne et un service déconnecté. Le protocole IMAP permet de manipuler des boîtes aux lettres à distance comme si elles étaient locales. Par exemple, les clients peuvent faire des recherches dans les messages et y insérer des indicateurs d'état tels que "deleted" ou "answered" ("supprimé" ou "répondu"). En outre, les messages peuvent être conservés dans la base de données du serveur tant qu'ils ne sont pas supprimés explicitement. Le serveur IMAP permet à plusieurs clients d'accéder de façon interactive et simultanée aux boîtes aux lettres utilisateur.

Les serveurs IMAP et POP sont exclusivement des serveurs d'accès au courrier. Pour l'envoi du courrier, ils utilisent le protocole SMTP (Simple Mail Transfer Protocol).

IMAP et POP sont tous deux des protocoles ouverts, qui reposent sur les normes décrites dans les RFC (Request for Comments). Le serveur IMAP repose sur le RFC 1730 et le serveur POP sur le RFC 1725. Les deux serveurs sont " orientés connexion " et utilisent des sockets TCP. L'écoute IMAP et POP a respectivement lieu sur les ports identifiés 143 et 110. En outre, le démon **inetd** gère les deux serveurs.

Configuration des serveurs IMAP et POP

Prérequis

Vous devez être utilisateur racine (root).

Procédure

1. Désactivez le commentaire des entrées **imapd** et **pop3d** dans le fichier **/etc/inetd.conf**.
2. Rafraîchissez le démon **inetd** avec la commande :

   ```
   refresh -s inetd
   ```

Tests de configuration

Vous pouvez lancer quelques tests pour vous assurer que les serveurs imapd et pop3d sont opérationnels.

Tout d'abord, vérifiez que leur écoute a lieu sur les ports identifiés : Pour cela, procédez comme suit :

```
netstat –a | grep imap
 netstat –a | grep pop
```

En principe, le résultat de la commande **netstat** donne :

```
tcp      0       0      *.imap2          *.*         LISTEN
tcp      0       0      *.pop3           *.*         LISTEN
```

Si vous n'obtenez pas ce résultat, vérifiez à nouveau les entrées dans le fichier **/etc/inetd.conf**, puis relancez la commande **refresh –s inetd**.

Testez la configuration sur le serveur imapd, via telnet, au niveau du port imap2, `143`. Vous obtenez l'invite imapd. Vous pouvez entrer les commandes IMAP version 4 définies dans la RFC 1730. Pour ce faire, tapez un point (.) puis un espace suivi du nom de la commande. Par exemple :

```
. NomCommande
```

Notez l'écho des mots de passe quand telnet est utilisé vers le serveur imapd.

Dans l'exemple `telnet` suivant, vous devez indiquer votre propre mot de passe à la place de *id_password* dans la commande **login**.

```
telnet e-xbelize 143
 Trying...
 Connected to e-xbelize.austin.ibm.com.
 Escape character is '^]'.
 * OK e-xbelize.austin.ibm.com IMAP4 server ready
 . login id id_password
 . OK
 . examine /usr/spool/mail/root
 * FLAGS (\Answered \Flagged \Draft \Deleted \Seen)

* OK [PERMANENTFLAGS (\Answered \Flagged \Draft \Deleted \Seen \*)]

 * 0 EXISTS
 * 0 RECENT
 * OK [UIDVALIDITY 823888143]
 . OK [READ-ONLY] Examine completed
 . logout
 * BYE Server terminating connection
 . OK Logout completed
 Connection closed.
```

Testez la configuration du serveur pop3d, via telnet, au niveau du port pop3, `110`. Vous obtenez l'invite pop3d. Vous pouvez entrer les commandes POP définies dans la RFC 1725. Pour ce faire, tapez un point (.) puis un espace suivi du nom de la commande. Par exemple :

```
. CommandName
```

Notez l'écho des mots de passe quand telnet est utilisé vers le serveur pop3d.

Dans l'exemple `telnet` suivant, vous devez indiquer votre propre mot de passe à la place de *id_password* dans la commande **pass**.

```
telnet e-xbelize 110
 Trying...
 Connected to e-xbelize.austin.ibm.com.
```

```
 Escape character is '^]'.
 +OK e-xbelize.austin.ibm.com POP3 server ready
 user id
 +OK Name is a valid mailbox
pass  id_password
 +OK Maildrop locked and ready
 list
 +OK scan listing follows
 .
 stat
 +OK 0 0
 quit
 +OK
 Connection closed.
```

syslog

Le logiciel serveur IMAP et POP adresse des journaux à l'outil **syslog**.

Pour configurer la journalisation IMAP et POP sur votre système par le biais de **syslog**, vous devez être un utilisateur racine. Editez le fichier de configuration **/etc/syslog.conf** pour y ajouter une entrée pour ***.debug** comme suit :

```
*.debug /usr/adm/imapd.log
```

Le fichier `usr/adm/imapd.log` doit être existant avant la relecture par le démon **syslogd** du fichier **/etc/syslog.conf**. Pour créer **usr/adm/imapd.log**, utilisez la commande :

```
touch /usr/adm/imapd.log
```

Ensuite, rafraîchissez **syslogd** avec la commande suivante pour la relecture de son fichier de configuration :

```
refresh -s syslogd
```

Informations de référence du courrier

Cette section fournit un bref récapitulatif des commandes, fichiers et répertoires intervenant dans la messagerie.

Liste des commandes

Cette liste répertorie les commandes d'exploitation et de gestion de la messagerie.

bugfiler	Enregistre les comptes rendus d'anomalies dans des répertoires courrier spécifiques.
comsat	Avertit les utilisateurs de l'arrivée d'un courrier (démon).
mailq	Imprime le contenu de la file d'attente courrier.
mailstats	Affiche les statistiques relatives au trafic du courrier.
newaliases	Crée une copie de la base de données d'alias à partir du fichier **/etc/aliases**.
rmail	Gère le courrier distant reçu via la commande **uucp** de BNU.
sendbug	Envoie un compte rendu d'anomalies à une adresse spécifique.
sendmail	Délivre le courrier en local ou sur le réseau.
smdemon.cleanu	Epure la file d'attente **sendmail** pour les tâches de routine.

Liste des fichiers et répertoires courrier

Les fichiers et répertoires sont présentés par fonction.

Remarque : Dans les versions antérieures à AIX 5.1, les fichiers **sendmail.cf** et **aliases** sont respectivement situés dans **/etc/sendmail.cf** et **/etc/aliases**.

Messagerie

/usr/share/lib/Mail.rc	Définit les valeurs par défaut du système local pour tous les utilisateurs de la messagerie. Fichier de texte modifiable pour définir les caractéristiques par défaut de la commande **mail**.
$HOME/.mailrc	Permet de modifier les valeurs par défaut du système local pour la messagerie.
$HOME/mbox	Stocke le courrier traité d'un utilisateur.
/usr/bin/Mail, /usr/bin/mail, ou **/usr/bin/mailx**	Indique trois noms associés au même programme. La messagerie est l'une des interfaces entre l'utilisateur et le système de messagerie.
/var/spool/mail	Indique le répertoire par défaut de dépôt du courrier. Le courrier est stocké par défaut dans le fichier **/var/spool/mail/nom-utilisateur**.
/usr/bin/bellmail	Prend en charge la livraison du courrier local.
/usr/bin/rmail	Assure l'interface courrier distant pour BNU.
/var/spool/mqueue	Contient le fichier journal et les fichiers temporaires associés aux messages de la file d'attente courrier.

Commande sendmail

/usr/sbin/sendmail	Commande **sendmail**.
/usr/ucb/mailq	Pointe sur le fichier **/usr/sbin/sendmail**. Equivaut à **/usr/sbin/sendmail –bp**.
/usr/ucb/newaliases	Pointe sur le fichier **/usr/sbin/sendmail**. Equivaut à **/usr/sbin/sendmail –bi**.
/etc/netsvc.conf	Spécifie l'ordre de certains services de résolution de noms.
/usr/sbin/mailstats	Formate et affiche les données statistiques **sendmail** recueillies dans le fichier par défaut **/etc/sendmail.st**, s'il existe. Vous pouvez spécifier un autre fichier.
/etc/aliases	Décrit une version texte du fichier d'alias pour la commande **sendmail**. Vous pouvez éditer ce fichier pour créer, modifier ou supprimer des alias de votre système.
/etc/aliasesDB	Décrit un répertoire contenant les fichiers de base de données d'alias, **DB.dir** et **DB.pag**, créés à partir du fichier **/etc/aliases** à l'exécution de la commande **sendmail –bi**.
/etc/sendmail.cf	Contient les informations de configuration de **sendmail** dans un format texte. Editez ce fichier pour modifier les informations.
/usr/lib/smdemon.cleanu	Spécifie un fichier shell qui exécute la file d'attente courrier et tient à jour des fichiers journaux **sendmail** dans le répertoire **/var/spool/mqueue**.
/var/tmp/sendmail.st	Rassemble les statistiques relatives au trafic du courrier. Ce fichier a une taille fixe. Utilisez la commande **/usr/sbin/mailstats** pour afficher son contenu. Supprimez-le si vous ne voulez pas recueillir ce type d'informations.
/var/spool/mqueue	Désigne le répertoire contenant les fichiers temporaires associés à chaque message en file d'attente. Ce répertoire peut contenir le fichier journal.
/var/spool/cron/crontabs	Désigne le répertoire contenant les fichiers lus par le démon **cron** pour déterminer le travail à exécuter. Le fichier **root** comporte une ligne d'exécution du script shell **smdemon.cleanu**.

Liste des commandes IMAP et POP

/usr/sbin/imapd	Process serveur IMAP (Internet Message Access Protocol).
/usr/sbin/pop3d	Process serveur POP3 (Post Office Protocol version 3.

Chapitre 4. Protocole TCP/IP

Ce chapitre décrit la suite de logiciels réseau TCP/IP (Transmission Control Protocol/Internet Protocol). TCP/IP est un protocole normalisé souple et puissant, permettant de connecter plusieurs ordinateurs à d'autres machines.

Ce chapitre traite des points suivants :

Remarque : La plupart des tâches abordées dans ce chapitre nécessitent les droits d'utilisateur racine.

Préparation du réseau TCP/IP

TCP/IP étant un protocole réseau très souple, vous pouvez le personnaliser et l'adapter aux besoins spécifiques de votre organisation. Les principaux points à prendre en compte pour préparer votre réseau sont les suivants. Chaque point fait l'objet d'un étude détaillée dans la suite de ce manuel. Cette liste vous permettra simplement d'évaluer la portée des actions possibles.

1. Choisissez le type de matériel réseau que vous souhaitez utiliser : anneau à jeton (token–ring), Ethernet Version 2, IEEE 802.3, interface FDDI (Fiber Distributed Data Interface), canal optique série (Serial Optical Channel, SOC) ou protocole SLIP (Serial Line Interface Protocol).
2. Tracez l'implantation physique du réseau.

   Réfléchissez aux fonctions que devra assurer chaque machine hôte. Par exemple, vous devez choisir à ce stade les machines qui serviront de passerelles avant de passer au câblage du réseau.
3. Optez selon vos besoins pour un réseau *plat* ou une structure de réseau *hiérarchisée*.

   Si votre réseau est de petite taille, concentré sur un seul site, et ne comprend qu'un réseau physique, un réseau plat peut parfaitement convenir.Si votre réseau est très étendu, complexe, avec de nombreux sites ou plusieurs réseaux physiques, il sera peut–être plus pratique d'opter pour un réseau hiérarchisé.
4. Si votre réseau doit être raccordé à d'autres réseaux, vous devez réfléchir à l'installation et à la configuration des passerelles qui seront nécessaires. Les éléments à prendre en compte sont :

   a. choisir les machines qui serviront de passerelles ;

   b. décider si vous utiliserez le routage statique ou le routage dynamique, à moins que vous ne choisissiez une combinaison des deux. Si vous optez pour le routage dynamique, choisissez les démons de routage que devra utiliser chaque passerelle, en tenant compte des différents types de protocoles de communication à prendre en charge.
5. préparez un schéma d'adressage.

   Si votre réseau n'est pas destiné à faire partie d'un interréseau plus étendu, choisissez le schéma d'adressage convenant le mieux à vos besoins. Si vous souhaitez intégrer votre réseau au sein d'un interréseau plus étendu tel qu'Internet, vous devrez vous procurer un jeu officiel d'adresses auprès de votre fournisseur d'accès à Internet (FAI).
6. Voyez s'il convient d'envisager la division de votre système en plusieurs sous–réseaux. Si oui, décidez du mode d'attribution des masques de sous–réseau.
7. Choisissez des conventions de noms. Chaque machine du réseau doit posséder un nom d'hôte unique.
8. Décidez si votre réseau requiert un serveur de noms pour la résolution des noms ou si le recours au fichier **/etc/hosts** est suffisant.

   Si vous décidez d'utiliser des serveurs de noms, choisissez le type de serveur nécessaire et combien de serveurs de noms vous devez prévoir pour être efficace.
9. Décidez des types de services que le réseau doit, selon vous, proposer aux utilisateurs distants : services de messagerie, d'impression, partage de fichiers, connexion à distance, exécution de commandes à distance, etc.

Installation et configuration pour TCP/IP

Pour plus d'informations sur l'installation de TCP/IP, reportez–vous au manuel *AIX 5L Version 5.2 Installation Guide and Reference*.

Configuration de TCP/IP

Une fois TCP/IP installé, la configuration du système peut être effectuée.

Pour configurer TCP/IP, vous pouvez :

- utiliser l'application Web-based System Manager Network (raccourci **wsm network**),
- utiliser SMIT (System Management Interface System),
- éditer un format de fichier,
- lancer une commande à partir de l'invite du shell.

Par exemple, le script shell **rc.net** effectue la configuration minimale du système hôte pour TCP/IP au démarrage du système (ce script est lancé à la seconde phase de l'amorçage par le gestionnaire de configuration). Si vous utilisez SMIT ou Web-based System Manager pour configurer le système hôte, le fichier **rc.net** est automatiquement configuré.

Vous pouvez également reconfigurer le fichier **/etc/rc.bsdnet** à l'aide d'un éditeur standard et ainsi utiliser les commandes traditionnelles de configuration de TCP/IP sous UNIX, telles que **ifconfig**, **hostname** et **route**. Pour en savoir plus, reportez-vous à "Liste des commandes TCP/IP". Si vous utilisez cette méthode, entrez le raccourci **smit configtcp**, puis sélectionnez une configuration rc de type **BSD**.

Certaines tâches, telles que la configuration d'un serveur de noms, ne peuvent être accomplies via SMIT ou via Web-based System Manager.

Configuration des systèmes hôte

Chaque système hôte du réseau doit être adapté aux besoins des utilisateurs et aux contraintes du réseau. Pour chaque hôte, vous devez configurer l'interface de réseau, définir l'adresse Internet, le nom d'hôte et les routes statiques vers les passerelles ou les autres systèmes hôte. Il faut également spécifier les démons à lancer par défaut et configurer le fichier **/etc/hosts** pour la résolution des noms (ou configurer l'hôte de telle sorte qu'il utilise le serveur de noms).

Configuration des hôtes en tant que serveurs

Si la machine hôte joue un rôle spécifique (passerelle, serveur de fichiers ou serveur de noms), la configuration de base doit être complétée.

Par exemple, si le réseau est organisé hiérarchiquement et que vous utilisez le protocole **DOMAIN** pour la résolution des noms dans les adresses Internet, vous devez configurer au moins un serveur de noms.

N'oubliez pas qu'un hôte serveur n'a pas besoin d'être une machine dédiée : elle peut également servir à d'autres fonctions. Par exemple, si la fonction de serveur de noms est relativement limitée, la machine peut également être utilisée comme station de travail ou serveur de fichiers sur le réseau.

> **Remarque :** Si NIS ou NIS+ est installé sur votre système, ces services peuvent également vous aider à la résolution des noms. Pour plus d'informations, reportez–vous à *NIS/NIS+ (Network Information Services) AIX 5L Version 5.2 Guide*.

Configuration des passerelles

Si vous envisagez de connecter votre réseau à d'autres réseaux, il vous faut configurer au moins une machine hôte passerelle. Pour cela, vous devez déterminer les protocoles de communication nécessaires et les démons de routage (**routed** ou **gated**) correspondants.

Commandes de gestion système TCP/IP

Voici la liste des commandes utiles pour configurer et gérer le réseau TCP/IP :

arp	Affichage/modification des tables de traduction d'adresse Internet en adresse matérielle, utilisées par **ARP** (Address Resolution Protocol).
finger	Retour d'informations concernant les utilisateurs sur un hôte spécifique.
host	Affichage de l'adresse Internet d'un hôte spécifique ou d'un nom d'hôte figurant dans une adresse Internet spécifique.
hostname	Affichage ou définition du nom et de l'adresse Internet d'un système hôte local.
ifconfig	Configuration des interfaces de réseau.
netstat	Affichage des adresses locales et distantes, des tables de routage, des données statistiques sur le matériel et du compte rendu des paquets transférés.
no	Affichage ou définition des options courantes du noyau de réseau.
ping	Détermination de l'accessibilité d'un système hôte.
route	Manipulation des tables de routage.
ruptime	Affichage des informations d'état sur les hôtes connectés aux réseaux physiques locaux et exécutant le serveur **rwhod**.
rwho	Affichage des informations d'état sur les utilisateurs des hôtes connectés aux réseaux physiques locaux et exécutant le serveur **rwhod**.
setclock	Calage de l'heure et de la date de l'hôte local sur celles du service horaire du réseau.
timedc	Informations sur le démon **timed**.
trpt	Compte rendu de suivi du protocole sur les prises TCP.
whois	Service du répertoire de noms Internet.

Configuration d'une liste de contrôle du réseau TCP/IP

Utilisez la procédure suivante comme guide de configuration de votre réseau. Prenez le temps nécessaire pour rassembler les informations et comprendre les instructions.

Une fois le réseau installé et opérationnel, cette liste de contrôle vous servira à résoudre des incidents.

Prérequis

1. Les matériels réseau sont installés et câblés. Reportez–vous aux *AIX 5L Version 5.2* Cartes réseau LAN TCP/IP.
2. Le logiciel TCP/IP doit être installé (voir le manuel *AIX Installation Guide*).

Procédure

1. Consultez ”Protocoles TCP/IP”, page 4-6, pour la structure de base de TCP/IP. Vous devez comprendre :
   - la structure en couches de TCP/IP (différents protocoles résidant sur différentes couches),
   - le mécanisme de flux des données à travers les couches.
2. Effectuez la configuration minimale de chaque machine hôte du réseau : ajout d’une interface réseau, affectation d’une adresse IP, attribution d’un nom d’hôte à chaque machine hôte et définition d’une route par défaut d’accès au réseau. Consultez tout d’abord les sections ”Interfaces de réseau TCP/IP”, page 4-49 , ”Adressage TCP/IP”, page 4-57 et ”Définition des noms d’hôte”, page 4-65.
3. Adressage TCP/IP page 4-57, et Choix des noms pour les hôtes de votre réseau page 4-65.

   Remarque : Chaque machine du réseau doit subir cette configuration minimale, qu’il s’agisse d’un hôte utilisateur, d’un serveur de fichiers, d’une passerelle ou d’un serveur de noms.
4. Configurez et lancez le démon **inetd** sur chaque machine hôte du réseau. Consultez la section ”Démons TCP/IP”, page 4-161 , et procédez comme indiqué à ”Configuration du démon inetd”, page 4-164 .
5. Configurez chaque machine hôte pour effectuer la résolution des noms en local ou utiliser le serveur de noms. Si vous installez un système hiérarchique de type DOMAIN, vous devez configurer au moins une machine hôte en tant que serveur de noms. Reportez-vous à ”Résolution de noms sous TCP/IP”, page 4-63 .
6. Si votre réseau doit être connecté à d’autres réseaux distants, configurez au moins une machine hôte comme passerelle. Pour l’acheminement interréseau, la passerelle peut utiliser des routes statiques ou un démon de routage. Reportez-vous à ”Routage TCP/IP”, page 4-168 .
7. Déterminez pour chaque machine hôte du réseau, les services accessibles. Par défaut, ils le sont tous. Pour changer cette configuration, procédez comme indiqué à ”Services réseau client”, page 4-165 .
8. Désignez, parmi les machines hôtes, celles qui joueront le rôle de serveurs et définissez leurs services respectifs. Pour lancer les démons de serveur de votre choix, reportez-vous à ”Services réseau serveur”, page 4-166 .
9. Configurez les serveurs d’impression à distance nécessaires. Pour en savoir plus, reportez-vous aux généralités sur les imprimantes dans *AIX Guide to Printers and Printing*.
10. Si vous le souhaitez, configurez une machine à utiliser comme serveur horaire maître pour le réseau. Pour en savoir plus, reportez-vous au démon **timed** dans le manuel *AIX 5L Version 5.2 Commands Reference*.

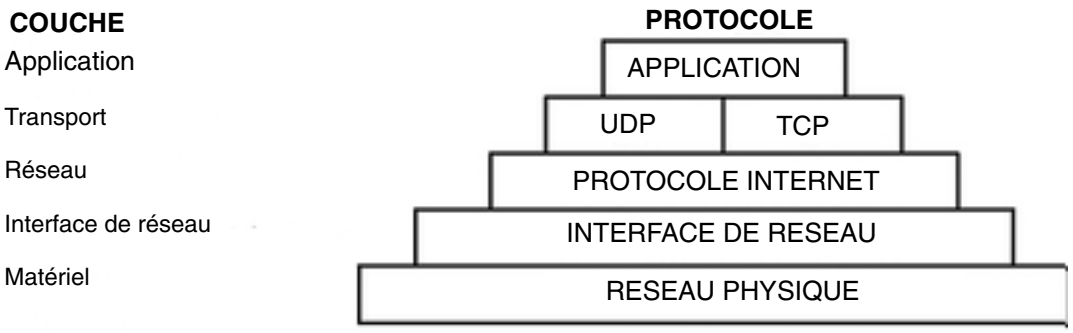
Protocoles TCP/IP

Cette section traite des points suivants :

Les protocoles sont des ensembles de règles de formats de message et de procédures qui permettent aux machines et aux applications d'échanger des informations. Ces règles doivent être observées par chaque machine impliquée dans la communication pour que le message puisse être interprété par le système destinataire.

La *suite* de protocoles TCP/IP (voir figure) peut être représentée en couches (ou niveaux).

Figure 3. Suite de protocoles TCP/IP Cette illustration représente les couches du protocole TCP/IP. Ce sont, à partir du haut : couche d'application, couche de transport, couche réseau, couche d'interface réseau et matériel.

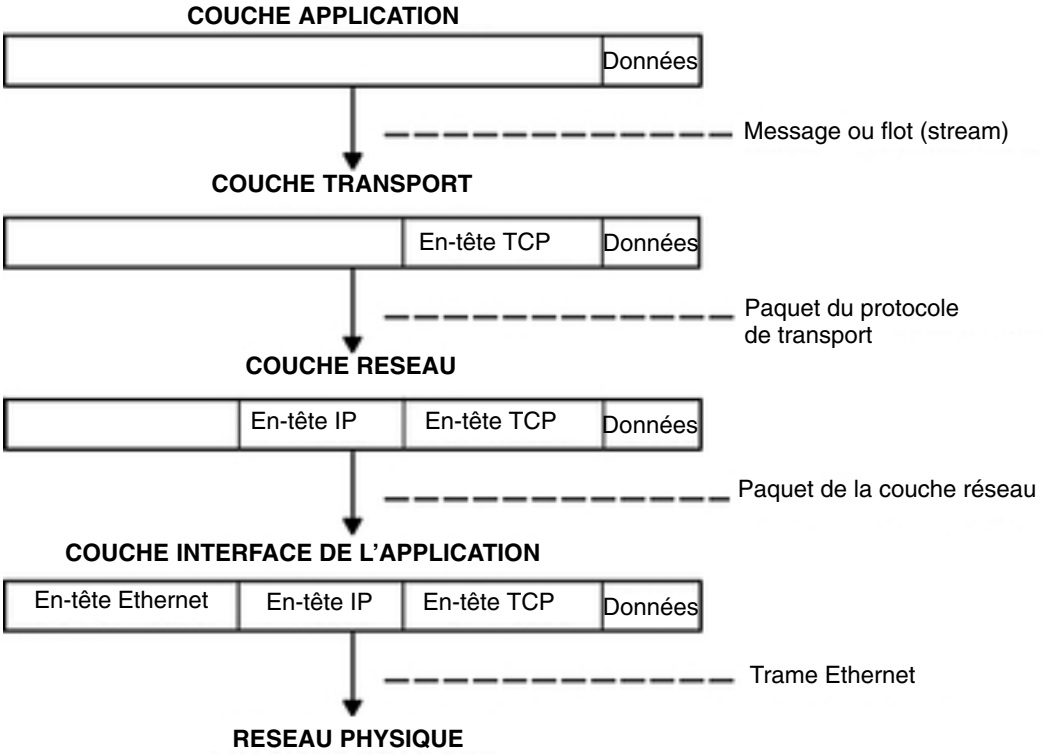


TCP/IP définit précisément l'acheminement de l'information de l'émetteur au destinataire. Les messages ou trains de données sont envoyés par les programmes d'application à l'un des deux protocoles Internet de niveau transport : UDP (User Datagram Protocol) ou TCP (Transmission Control Protocol). A la réception des données, ces protocoles les divisent en *paquets*, y ajoutent une adresse de destination et les transmettent à la couche de protocole suivante, la couche Réseau Internet.

La couche Réseau Internet encapsule le paquet dans un datagramme IP (Internet Protocol), insère les données d'en-tête et de fin, décide de la destination du datagramme (directement à destination ou via une passerelle) et transmet le datagramme à la couche Interface réseau.

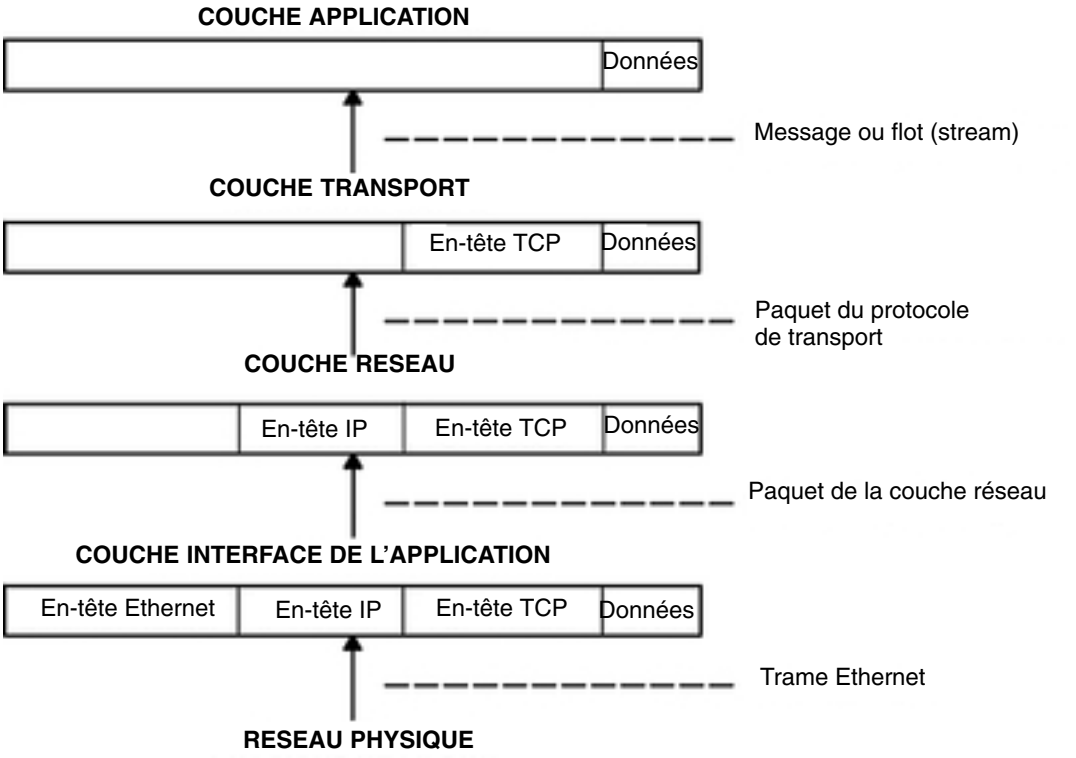
La couche Interface réseau réceptionne les datagrammes IP et les transmet sous forme de *trames* à travers un réseau spécifique, tel que Ethernet ou anneau à jeton (voir figure).

Figure 4. Mouvement des informations de l'application émettrice vers l'hôte récepteur Cette illustration représente le flux d'informations descendant dans les couches de protocole TCP/IP de l'émetteur vers l'hôte.



Les trames reçues par une machine hôte sont réexpédiées à travers les couches de protocoles dans le sens inverse. Chaque couche supprime l'information d'en-tête correspondante jusqu'à ce que les données atteignent de nouveau la couche Application (reportez–vous à la figure). Les trames arrivent dans la couche Interface réseau (dans le cas présent, une carte Ethernet). L'en–tête Ethernet est supprimé et le datagramme renvoyé vers la couche Réseau. Dans la couche Réseau, le protocole Internet supprime l'en-tête IP et envoie à son tour le paquet vers la couche Transport. A ce niveau, l'en-tête TCP est supprimé par le protocole TCP et les données sont envoyées vers la couche Application.

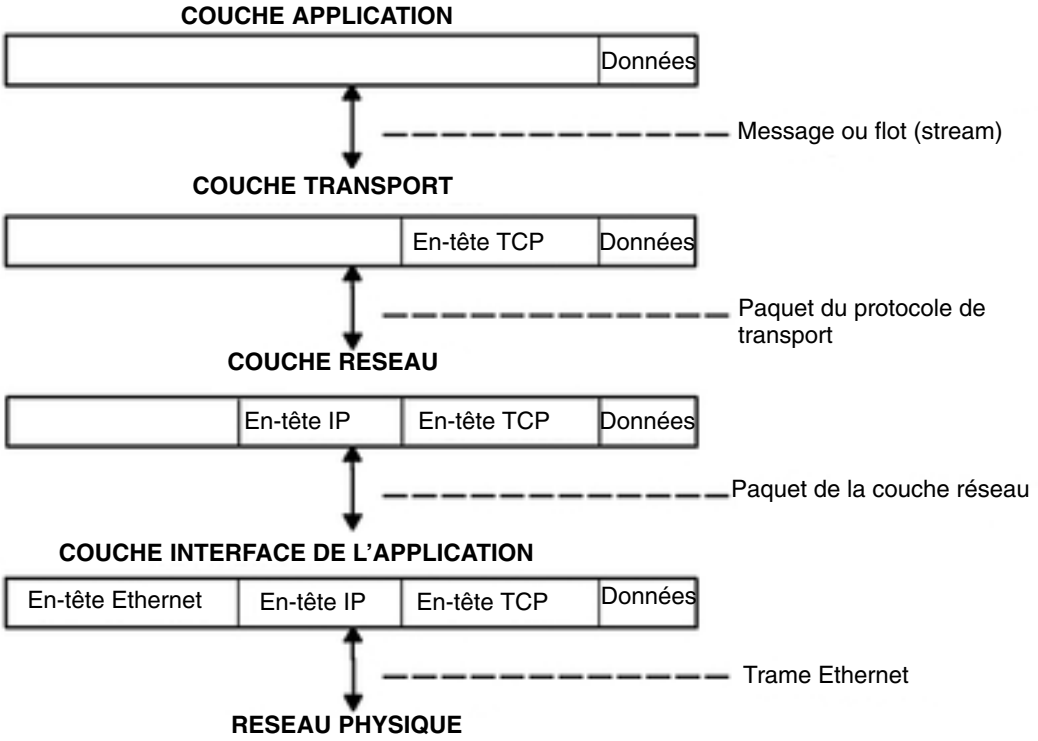
Figure 5. Mouvement des informations de l'hôte vers l'application Cette illustration représente le flux d'informations remontant les couches de protocole TCP/IP de l'hôte vers l'émetteur.



Les machines hôtes envoient et reçoivent des informations simultanément. En ce sens, la figure Transmission et réception des données hôtes représente avec plus d'exactitude le mode de communication de l'hôte.

Les hôtes d'un réseau envoient et reçoivent les informations simultanément. La Figure 6 représente avec davantage de précision un hôte en cours de communication.

Figure 6. Transmissions et réceptions des données hôte Cette illustration représente les flux de données dans les deux sens dans les couches TCP/IP.



Remarque : Les en–têtes sont ajoutés et supprimés dans chaque couche de protocole au fur et à mesure que les données sont transmises et reçues par l'hôte.

IP version 6 - Généralités

IP (Internet Protocol) version 6 (IPv6 ou IP *ng*) est la prochaine génération IP, conçue comme une évolution d'IP version 4 (IPv4). Si IPv4 a permis le développement d'un Internet global, il n'est cependant pas capable de progresser davantage à cause de deux facteurs fondamentaux : espace d'adressage limité et complexité du routage. Les adresses 32 bits IPv4 ne fournissent pas suffisamment de flexibilité pour le routage global Internet. Le déploiement de CIDR (Classless InterDomain Routing) a étendu la durée de vie du routage IPv4 d'un certain nombre d'années, mais l'effort de gestion du routage continue toutefois à augmenter. Même si le routage IPv4 pouvait être augmenté, Internet finirait par être à court de numéros de réseau.

L'IETF (Internet Engineering Task Force) ayant reconnu qu'IPv4 ne serait pas capable d'assumer la croissance phénoménale d'Internet, le groupe de travail IETF IP *ng* a été formé. Parmi les propositions effectuées, SIPP (Simple Internet Protocol Plus) a été choisi comme étape dans le développement d'IP. Il a été renommé IP *ng*, et RFC1883 a été finalisé en décembre 1995.

IPv6 étend le nombre maximal d'adresses Internet de façon à gérer la croissance de la population utilisatrice d'Internet. Par rapport à IPv4, IPv6 présente l'avantage de permettre la coexistence des nouveautés et des éléments existants. Ceci permet une migration ordonnée d'IPv4 (adressage 35 bits) à IPv6 (adressage 128 bits) sur un réseau opérationnel.

Cette présentation est destinée à donner au lecteur une compréhension générale du protocole IP*ng*. Pour plus d'informations, veuillez vous reporter à RFC 2460, 2373, 2465, 1886, 2461, 2462 et 2553.

Routage et adressage étendus

IPv6 augmente la taille de l'adresse IP de 32 bits à 128 bits, prenant ainsi en charge davantage de niveaux dans la hiérarchie d'adressage, un nombre beaucoup plus grand de nœuds adressables et une configuration automatique plus simple des adresses.

Dans IPv6, il existe trois types d'adresses :

unicast — Un paquet envoyé à une adresse unicast est livré à l'interface identifiée par cette adresse. Une adresse unicast a une portée particulière : local–liaison, local–site, global. Il existe également deux adresses unicast spéciales :

- ::/128 (adresse non spécifiée)
- ::1/128 (adresse en boucle)

multicast — Un paquet envoyé à une adresse multicast est livré à l'interface identifiée par cette adresse. Une adresse multicast est identifiée par le préfixe ff::/8. Les adresses multicast ont une portée semblable à celle des adresses unicast : local–nœud, local–liaison, local–site et local–organisation.

anycast — Une adresse anycast a un seul expéditeur, plusieurs auditeurs et un seul interlocuteur (normalement le "plus proche", conformément à la mesure de distance des protocoles de routage). Par exemple, il peut y avoir plusieurs serveurs Web à l'écoute d'une adresse anycast. Lorsqu'une requête est envoyée à cette adresse, un seul serveur répond.

Une adresse anycast ne se distingue pas d'une adresse unicast. Une adresse unicast devient une adresse anycast lorsque plus d'une interface est configurée avec cette adresse.

Remarque : Il n'existe pas d'adresse de diffusion dans IPv6. Cette fonction est remplacée par l'adresse multicast.

Configuration automatique

Les principaux mécanismes disponibles, permettant à un nœud de s'initialiser et de commencer à communiquer avec d'autres nœuds sur un réseau IPv4 sont le codage "hard–coding", **BOOTP** et **DHCP**.

IPv6 introduit le concept de *portée* aux adresses **IP**, dont l'une est local–liaison. Ceci permet à un hôte à créer une adresse valide à partir du préfixe local–liaison prédéfini et son identificateur local. Cet identificateur local est en général extrait de l'adresse MAC (medium access control) de l'interface à configurer. A l'aide de cette adresse, le nœud peut communiquer avec les autres hôtes sur le même sous–réseau et, pour un sous–réseau entièrement isolé, peut ne pas avoir besoin d'une autre configuration d'adresse.

Adresses significatives

Avec IPv4, la seule signification généralement identifiable dans les adresses est la diffusion (en général tout 1 ou tout 0), et les classes (par exemple, une classe D est multicast). Avec IPv6, il est possible d'examiner rapidement le préfixe pour déterminer la *portée* (par exemple, local-liaison), multicast ou unicast, et un mécanisme d'affectation (basé sur le fournisseur, sur l'implantation géographique, etc.).

Les informations de routage peuvent également être chargées explicitement dans les bits supérieurs des adresses, bien que l'IETF n'ait pas encore finalisé ce point (pour les adresses basées sur le fournisseur, les informations de routage sont implicitement présentes dans l'adresse).

Détection d'adresse en double

Lorsqu'une interface est initialisée, ou réinitialisée, elle se sert de la configuration automatique pour essayer d'associer une adresse de type local-liaison à cette interface (l'adresse n'est pas encore affectée à cette interface dans le sens traditionnel). A ce stade, l'interface rejoint les groupes multicast tous nœuds et nœuds sollicités, et leur envoie un message de découverte de voisinage. Avec l'adresse multicast, le noeud peut déterminer si cette adresse local-liaison particulière a été préalablement affectée, puis choisir une autre adresse. Ceci évite une des erreurs communes de gestion de réseau, c'est-à-dire l'affectation de la même adresse à deux interfaces différentes sur le même lien.
(Il est encore possible de créer des adresses en double de portée globale pour les nœuds ne se trouvant pas sur le même lien.)

Configuration automatique de découverte voisinage/adresse sans état

Le protocole **NDP (Neighbor Discovery Protocol)** pour IPv6 est utilisé par des nœuds (hôtes et routeurs) pour déterminer les adresses de couche liaison pour les voisins connus sur des liens rattachés, et maintient les tables de routage par destination pour les connexions actives. Les hôtes utilisent également **NDP** pour découvrir des routeurs de voisinage désireux d'acheminer des paquets pour leur compte et détectent les adresses de couche liaison modifiées. **NDP** (Neighbor Discovery Protocol) utilise ICMP (Internet Control Message Protocol) version 6 avec ses propres types de messages uniques. D'une façon générale, le protocole **NDP** IPv6 correspond à la combinaison du protocole ARP IPv4, RDISC (ICMP Router Discovery) et ICMP Redirect (ICMPv4), avec beaucoup d'améliorations.

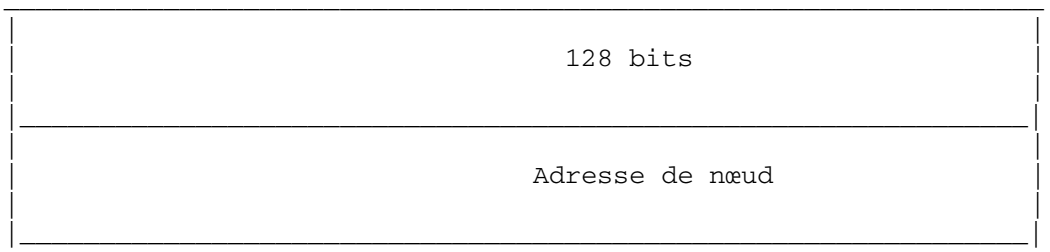
IPv6 définit le mécanisme de configuration automatique d'une adresse avec et sans état. *La configuration automatique sans état* n'exige pas de configuration manuelle des hôtes, une configuration, éventuelle, minimale des routeurs; et pas de serveur supplémentaire. Le mécanisme sans état permet à un hôte de générer ses propres adresses à l'aide d'une combinaison d'informations disponibles localement et présentées par les routeurs. Les routeurs annoncent les préfixes qui identifient le(s) sous–réseau(x) associés à un lien, tandis que les hôtes génèrent un jeton d'interface qui identifie de façon unique une interface sur un sous–réseau. Une adresse est formée par la combinaison des deux éléments.
En l'absence de routeurs, un hôte ne peut générer que des adresses de type local-liaison. Ces adresses sont toutefois suffisantes pour la communication entre nœuds rattachés au même lien.

Simplification de routage

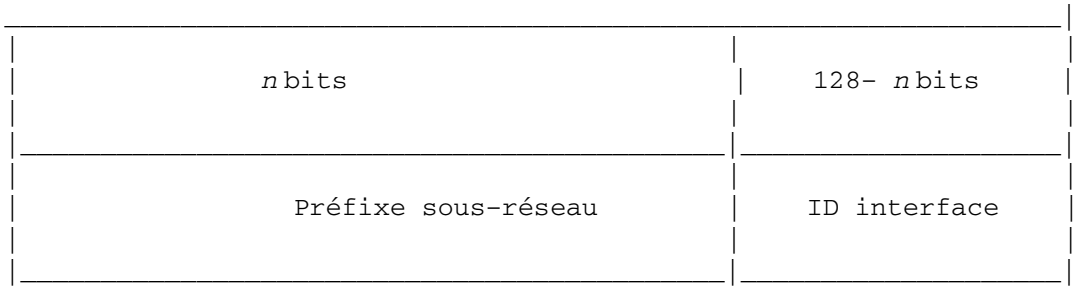
Pour simplifier les problèmes de routage, les adresses IPv6 sont décomposées en deux parties : un préfixe et un ID. La différence avec le découpage des adresses IPv4 n'est pas très sensible, mais présente deux avantages :

absence de classe	Il n'y a pas de nombre fixe de bits pour le préfixe ou l'ID, ce qui permet de réduire les pertes dues à une suraffectation.
imbrication	Il est possible d'utiliser un nombre arbitraire de divisions si l'on considère différents nombres de bits comme préfixe.

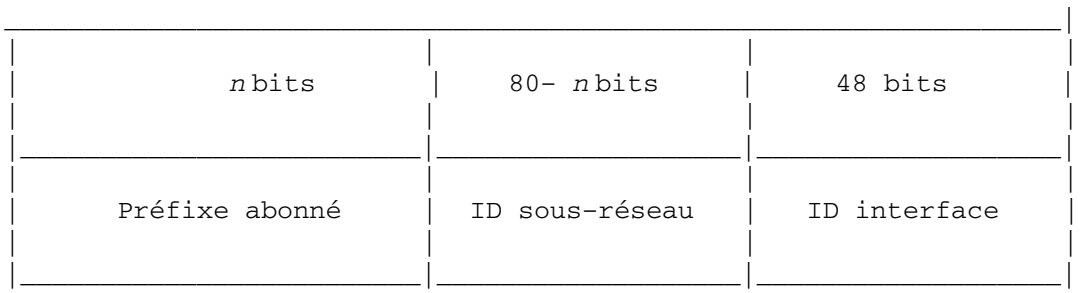
Cas 1 :



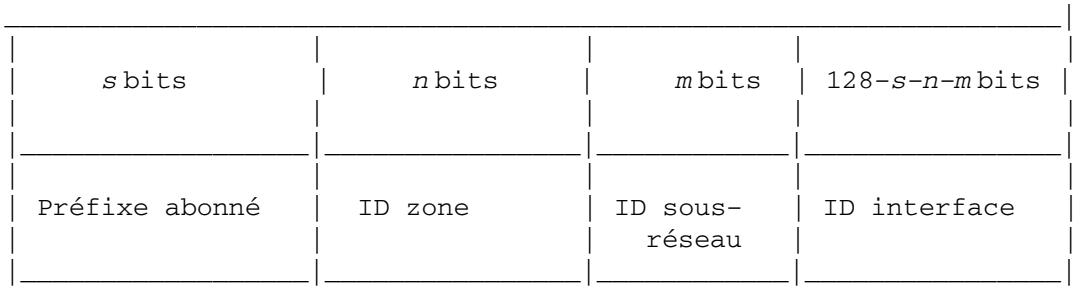
Cas 2 :



Cas 3 :



Cas 4 :



En général, IPv4 ne peut aller au delà du cas 3, même avec VLSM (Variable Length Subnet Mask, masque de sous–réseau de longueur variable). (VLSM est un moyen d'allouer des ressources d'adresses **IP** à des sous–réseaux selon leurs besoins plutôt qu'en respectant des règles générales à l'échelle du réseau). Il s'agit autant d'un artefact de la longueur d'adresse la plus courte que de la définition des préfixes de longueur variable, mais cela mérite cependant d'être noté.

Simplification du format d'en-tête

IPv6 simplifie l'en-tête IP, soit par suppression complète soit par déplacement sur un en-tête d'extension de certains champs trouvés dans l'en-tête IPV4, et il définit un format plus souple pour les informations facultatives (en-têtes d'extension). Spécifiquement, notez l'absence de :

- longueur d'en-tête (la longueur est constante)
- identification
- indicateurs
- décalage de fragment (déplacé dans les en-têtes d'extension de fragmentation)
- total de contrôle d'en-tête (l'en-tête de protocole de couche supérieure ou d'extension de sécurité gère l'intégrité de*s* données*)*

Tableau 1. En–tête IPv4 :

Version	IHL	Type de service	Longueur totale	
Identification			Identifica teurs	Décalage fragment (Offset)
Durée de vie		Protocole	Total de contrôle d'en-tête (checksum)	
Adresse source				
Adresse de destination				
Options				Remplissage

Tableau 3–2. En–tête Ipv6 :

Version	Prio	Libellé du flux	
Longueur bloc		En-tête suivant	Limite de tronçon
Adresse source			
Adresse de destination			

IP *ng* inclut un mécanisme d'options amélioré par rapport à IPv4. Les options IPv6 sont placées dans des en-têtes d'extension séparés qui résident entre l'en-tête IPv6 et l'en-tête de couche transport dans un paquet. La plupart des en-têtes d'extension ne sont pas examinés ou traités par un routeur le long du chemin de livraison de paquets.
Ce mécanisme apporte une grande amélioration aux performance du routeur pour les paquets contenant des options. Dans IPv4, la présence d'options requiert l'examen de toutes les options par le routeur.

Une autre amélioration provient du fait que, contrairement aux options IPv4, les en-têtes d'extension IPv6 peuvent être d'une longueur arbitraire et le nombre total d'options transmises dans un paquet n'est pas limité à 40 octets. Cette fonction, ainsi que la façon dont elle est traitée, permet aux options IPv6 d'être utilisées pour les fonctions qui n'étaient pas pratiques dans IPv4, comme les options d'authentification et d'encapsulage de sécurité IPv6.

Pour améliorer les performances de gestion des en-têtes d'option suivants et du protocole de transport qui suit, les options IPv6 sont toujours un multiple entier de huit octets, pour conserver cet alignement pour les en–têtes suivants.

En utilisant des en-têtes d'extension au lieu d'un spécificateur de protocole et de champs d'options, l'intégration des extensions nouvellement définies est plus facile.

Les spécifications actuelles définissent les en-têtes d'extension comme suit :

- Options bond par bond s'appliquant à chaque bond (routeur) sur le chemin

- En-tête de routage pour un routage de source strict ou non (rarement utilisé)
- Un fragment définit le paquet comme un fragment et contient des informations à ce sujet (les routeur Ipv6 ne fragmentent pas les paquets)
- Authentification (reportez–vous à Sécurité IP dans *AIX 5L Version 5.2 Security Guide*
- Chiffrement (reportez–vous à Sécurité IP dans *AIX 5L Version 5.2 Security Guide*
- Options de destination pour le nœud de destination (ignoré par les routeurs)

Amélioration du contrôle trafic/qualité du service

La qualité du service peut être contrôlée à l'aide d'un protocole de contrôle comme RSVP, et IPv6 fournit une définition de priorité explicite pour les paquets en utilisant le champ de priorité dans l'en-tête IP. Un nœud peut définir cette valeur pour indiquer la priorité relative d'un paquet ou d'un ensemble de paquets, pouvant être alors utilisés par le nœud, un ou plusieurs routeurs, ou la destination pour indiquer que faire du paquet (l'abandonner ou non).

IPv6 spécifie deux types de priorités, une pour le trafic contrôle en cas de congestion, et une pour le trafic non contrôlé en cas de congestion. Il n'y a aucun ordre relatif entre ces deux types.

Le trafic contrôle en cas de congestion est un trafic répondant aux embouteillages par un algorithme de limitation. Dans ce cas, les priorités sont :

0	trafic non caractérisé
1	trafic "de remplissage" (par exemple, informations sur le réseau)
2	transfert de données non assisté (par exemple, messagerie automatique)
3	(réservé)
4	transfert de lot assisté (par exemple, FTP)
5	(réservé)
6	trafic interactif (par exemple, Telnet)
7	trafic de contrôle (par exemple, protocoles de routage)

Le trafic non contrôlé en cas de congestion est un trafic répondant à des situations d'embouteillage par l'abandon (ou simplement la non réexpédition) des paquets, par exemple le trafic vidéo, audio ou autre trafic en temps réel. Les niveaux explicites ne sont pas définis avec des exemples, mais l'ordre est semblable à celui utilisé pour le trafic contrôlé en cas de congestion.

- La valeur la plus basse que la source cherche le plus à rejeter doit être utilisée pour le trafic.
- La valeur la plus haute que la source cherche le moins à rejeter doit être utilisée pour le trafic.

Ce contrôle de priorité ne s'applique qu'au trafic provenant d'une adresse source particulière. Le contrôle de trafic à partir d'une adresse ne constitue pas une priorité explicitement supérieure à un transfert de lot assisté à partir d'une autre adresse.

Libellé du flux

En-dehors de la définition de priorité de base pour le trafic, IPv6 définit un mécanisme de spécification d'un flux particulier de paquets. En termes IPv6, un *flux* est une suite de paquets envoyés à partir d'une source spécifique vers une destination spécifique (unicast ou multicast), pour laquelle la source recherche un traitement spécial par les routeurs intervenants.

Cette identification de flux peut servir pour le contrôle de priorité, mais peut également être utilisée pour un certain nombre de contrôles.

Le libellé de flux est choisi de façon aléatoire, et ne doit pas être utilisé pour identifier une caractéristique du trafic différente du flux correspondant. Un routeur ne peut donc pas déterminer qu'un paquet est d'un type particulier (par exemple, FTP) par le seul examen du libellé de flux. Il pourra cependant déterminer qu'il s'agit d'une partie de la même suite de paquets que le dernier paquet portant ce libellé.

Remarque : Jusqu'à généralisation de l'utilisation d'IPv6, le libellé de flux est principalement expérimental. Les utilisations et les contrôles impliquant des libellés de flux n'ont pas encore été définis ni standardisés.

Utilisation de tunnel

La clé d'une transition IPv6 réussie est la compatibilité avec la base installée existante d'hôtes IPv4 et de routeurs. Le maintien de cette compatibilité permet un passage en douceur d'Internet sur IPv6.

Dans la plupart des cas, l'infrastructure de routage IPv6 évolue dans le temps. Pendant le déploiement de l'infrastructure IPv6, l'infrastructure de routage IPv4 existante peut rester fonctionnelle et peut servir à acheminer le trafic IPv6. L'utilisation de tunnels permet d'utiliser une infrastructure de routage IPv4 existante pour acheminer le trafic IPv6.

Les hôtes et routeurs IPv6/IPv4 peuvent utiliser des tunnels pour les datagrammes IPv6 sur des zones de la topologie de routage IPv4 en les encapsulant dans des paquets IPv4. Le tunnel peut être utilisé d'une multitude de façons.

Routeur-routeur	Les routeurs IPv6/IPv4 interconnectés par une infrastructure IPv4 peuvent faire passer dans un tunnel les reliant des paquets IPv6. Dans ce cas, le tunnel fractionne un segment du chemin complet qu'emprunte le paquet IPv6.
Hôte-routeur	Les hôtes IPv6/IPv4 peuvent faire passer dans un tunnel des paquets IPv6 vers un routeur intermédiaire IPv6/IPv4 accessible via une infrastructure IPv4. Ce type de tunnel fractionne le premier segment du chemin complet du paquet.
Hôte-hôte	Les hôtes IPv6/IPv4 interconnectés par une infrastructure IPv4 peuvent faire passer des paquets IPv6 dans un tunnel les reliant. Dans ce cas, le tunnel fractionne tout le chemin qu'emprunte le paquet.
Routeur-hôte	Les routeurs IPv6/IPv4 peuvent faire passer dans un tunnel des paquets IPv6 jusqu'à leur hôte final IPv6/IPv4. Dans ce cas, le tunnel ne fractionne que le dernier segment du chemin complet.

Les techniques de tunnel sont généralement classées en fonction du mécanisme par lequel le nœud d'encapsulage détermine l'adresse du nœud en fin de tunnel. Dans les méthodes routeur-routeur ou hôte-routeur, le paquet IPv6 est acheminé par tunnel vers un routeur. Dans les méthodes hôte-hôte ou routeur-hôte, le paquet IPv6 passe dans un tunnel tout le long jusqu'à sa destination.

Le noeud d'entrée du tunnel (noeud d'encapsulage) crée un en-tête IPv4 d'encapsulage et transmet le paquet encapsulé. Le noeud de sortie du tunnel (noeud de décapsulage) reçoit le paquet encapsulé, supprime l'en-tête IPv4, met à jour l'en-tête IPv6 et traite le paquet IPv6 reçu. Toutefois, le nœud d'encapsulage doit mettre à jour les informations sur l'état du logiciel pour chaque tunnel, par exemple MTU pour chaque tunnel, pour traiter les paquets IPv6 acheminés dans le tunnel.

Sécurité IPv6

Pour plus d'informations sur la sécurité IP, versions 4 et 6, reportez–vous à Sécurité IP dans *AIX 5L Version 5.2 Security Guide*.

Support IPv6 des adresses locales du site et des liens Multihomed

Plusieurs interfaces peuvent être définies pour un hôte. Un hôte comportant deux ou plusieurs interfaces interactives est dit multihomed. Chaque interface est associée à une adresse de type local. Ces adresses sont suffisantes pour la communication entre nœuds rattachés à un même lien.

Un hôte multihomed est associés à deux ou plusieurs adresses de type local. Dans cette implémentation IPv6, 4 options permettent de déterminer comment la résolution des adresses de couche liaison s'effectue sur les hôtes multihomed. L'option 1 est activée par défaut.

Option 0	Aucune action multihomed n'est effectuée. Les transmissions sortent par la première interface de type local. Lorsque le protocole **NDP (Neighbor Discovery Protocol)** doit résoudre les adresses, il envoie (multicast) un message de découverte de voisinage sur chaque interface pour laquelle est définie cette adresse de type local. **NDP** met le paquet de données en attente jusqu'à ce qu'il reçoive le premier message d'avis de voisinage (Neighbor Advertisement). Le paquet de données est alors transmis par cette liaison.
Option 1	Lorsque le protocole **NDP** doit résoudre une adresse (lorsqu'il envoie un paquet de données vers une destination et que les informations relatives à la liaison pour le tronçon suivant ne sont pas dans le cache de voisinage (Neighbor Cache), il envoie (multicast) un message de découverte de voisinage sur chaque interface pour laquelle est définie cette adresse de type local. **NDP** met alors le paquet de données en attente jusqu'à ce qu'il reçoive les informations concernant la liaison. **NDP** attend de recevoir la réponse de chaque interface. Ceci permet de garantir que les paquets de données sont envoyés par l'intermédiaire des interfaces sortantes appropriées. Si **NDP** répondait au premier avis de voisinage sans attendre les autres réponses, il pourrait arriver qu'un paquet de données soit envoyé sur une liaison non associée à l'adresse source du paquet. Comme **NDP** doit attendre toutes les réponses, on constate un certain délai avant l'envoi du premier paquet. De toute façon, un délai est également à prévoir lors de l'attente de la première réponse.

Option 2	Le fonctionnement multihomed est autorisé mais l'expédition d'un paquet de données est limitée à l'interface spécifiée par main_if6. Lorsque le protocole **NDP** doit résoudre les adresses, il envoie (multicast) un message de découverte de voisinage sur chaque interface pour laquelle est définie cette adresse de type local. Il attend alors le message d'avis de voisinage en provenance de l'interface spécifiée par main_if6 (voir la commande **no**). Dès qu'il reçoit la réponse de cette interface, le paquet de données est envoyé sur cette liaison.
Option 3	Le fonctionnement multihomed est autorisé mais l'expédition d'un paquet de données est limitée à l'interface spécifiée par main_if6 et les adresses de type local ne sont acheminées que pour l'interface spécifiée par main_site6 (voir la commande **no**). Le protocole **NDP** fonctionne comme avec l'option 2. Pour les applications qui acheminent des paquets de données en utilisant des adresses de type local, sur un hôte multihomed, seule l'adresse locale spécifiée par main_site6 est utilisée.

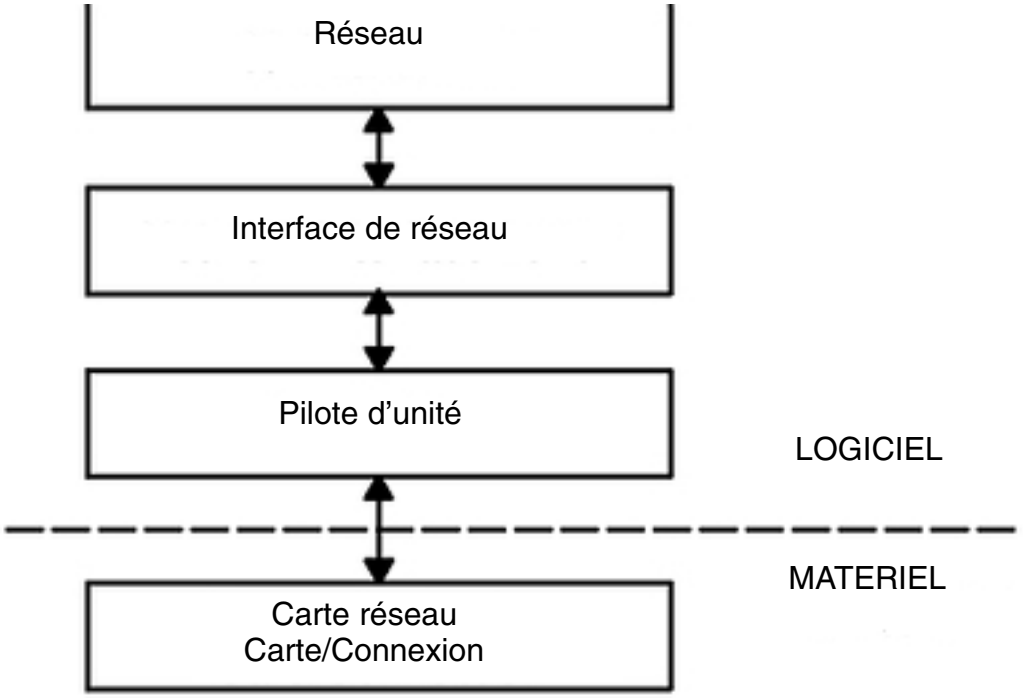
Suivi de paquet

Le suivi de paquet consiste à contrôler le parcours d'un paquet à travers les couches jusqu'à destination. La commande **iptrace** permet d'effectuer ce contrôle au niveau de la couche Interface de réseau. La commande **ipreport** génère en sortie un compte rendu de suivi aux formats hexadécimal et ASCII. La commande **trpt** effectue le contrôle au niveau de la couche transport pour le protocole TCP. La sortie de la commande **trpt** est plus détaillée : elle comprend des informations sur la date et l'heure, l'état **TCP** et la mise en séquence des paquets.

En-têtes de paquet au niveau interface de réseau

Au niveau de la couche Interface de réseau, des en-têtes sont associés aux données sortantes.

Figure 7. Flux de paquet dans la structure de l'interface réseau Cette illustration représente un flux de données bidirectionnel dans les couches de la structure de l'interface réseau. Ce sont, en partant du haut : (logiciel) couche réseau, couche d'interface réseau, pilote de périphérique, et (matériel) carte adaptateur réseau ou connexion.



Les paquets transitent alors par la carte de réseau vers le réseau correspondant. Ils traversent parfois plusieurs passerelles avant d'atteindre leur destination. Une fois arrivés au réseau de destination, ces en-têtes sont supprimés et les données envoyées à l'hôte concerné.

Ce processus s'applique aux informations d'en-tête de plusieurs interfaces de réseau courantes.

En-têtes de trame pour carte Ethernet

Le tableau ci-après représente un en-tête de trame IP (Internet Protocol) ou ARP (Address Resolution Protocol) pour la carte Ethernet.

En-tête de trame de carte Ethernet		
Zone	**Longueur**	**Définition**
DA	6 octets	Adresse de destination.
SA	6 octets	Adresse source. Si le bit 0 de cette zone est positionné à 1, l'information de routage (RI) est présente.
Type	2 octets	Type du paquet : IP ou ARP. IP ou ARP (le type est représenté par des numéros, comme indiqué ci-dessous).

Numéros de la zone Type :

IP	0800
ARP	0806

En-tête de trame pour réseau en anneau à jeton

L'en-tête MAC (Medium Access Control) pour carte anneau à jeton se compose des cinq zones ci-dessous :

En-tête MAC pour réseau en anneau à jeton		
Zone	**Longueur**	**Définition**
AC	1 octet	Contrôle d'accès. La valeur x'00' confère à l'en-tête la priorité 0.
FC	1 octet	Contrôle de la zone. La valeur x'40' indique une trame LLC (Logical Link Control).
DA	6 octets	Adresse de destination.
SA	6 octets	Adresse source. Si le bit 0 de cette zone est positionné à 1, l'information de routage (RI) est présente.
RI	18 octets	Information de routage. Les valeurs possibles sont fournies plus loin.

L'en-tête MAC comprend deux zones d'informations sur le routage, de 2 octets chacune : le contrôle de routage (RC) et les numéros de segment. Huit numéros de segment au maximum peuvent être utilisés pour désigner les destinataires d'une diffusion limitée. Les informations RC sont fournies aux octets 0 et 1 de la zone RI. Les deux premiers bits de la zone RC peuvent prendre les valeurs suivantes :

bit (0) = 0 Utilisation de la route de non–diffusion, spécifiée dans la zone RI.

bit (0) = 1 Création de la zone RI et diffusion vers tous les anneaux.

bit (0) = 1 Diffusion via tous les ponts.

bit (1) = 1 Diffusion via certains ponts.

L'en-tête LLC (contrôle de liaison logique) comporte les cinq zones suivantes :

En-tête LLC 802.3		
Zone	**Longueur**	**Définition**
DSAP	1 octet	Point d'accès au service de destination. La valeur est x'aa'.
SSAP	1 octet	Point d'accès au service source. La valeur est x'aa'.
CONTROL	1 octet	Commandes et réponses LLC (contrôle de liaison logique). Trois valeurs possibles (présentées plus loin).
PROT_ID	3 octets	ID de protocole. Cette zone est réservée. Sa valeur est de x'0'.
TYPE	2 octets	Type du paquet : **IP** ou **ARP**.

Valeurs de la zone CONTROL

x'03' Trame d'information non numérotée (UI). Mode de transmission normale ou non séquentielle des données de la carte anneau à jeton sur le réseau. Les données sont mises en séquence par **TCP/IP**.

x'AF' Trame XID (Exchange Identification). Elle transmet les caractéristiques de l'hôte émetteur.

x'E3' Trame de test. Cette trame teste la route de transmission, et renvoie les données reçues.

En-têtes de trame 802.3

L'en-tête MAC (Medium Access Control) pour la carte 802.3 est composé de deux zones, comme vous pouvez le constater dans le tableau d'en–têtes suivant.

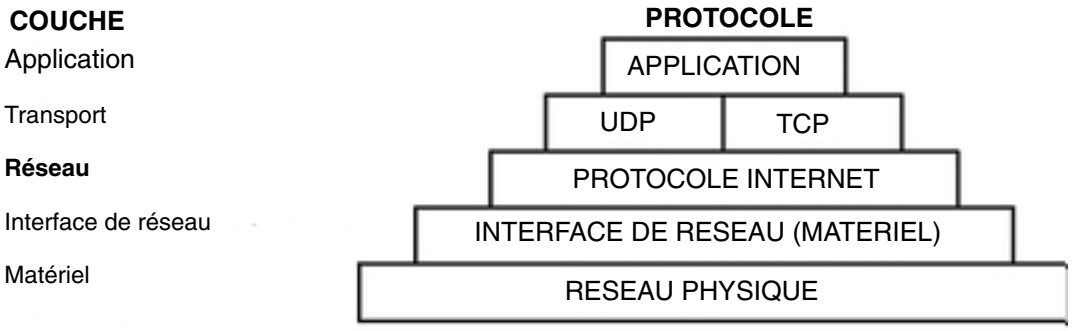
En-tête MAC 802.3		
Zone	**Longueur**	**Définition**
DA	6 octets	Adresse de destination.
SA	6 octets	Adresse source. Si le bit 0 de cette zone est positionné à 1, l'information de routage (RI) est présente.

L'en-tête LLC (Logical Link Control) pour la carte 802.3 est identique à l'en-tête MAC de l'anneau à jeton.

Protocoles Internet de niveau réseau

Les protocoles Internet de niveau réseau gèrent la communication entre les machines. Autrement dit, c'est la couche qui assure le routage **TCP/IP**. Ces protocoles réceptionnent les demandes de transmission de paquets (dotés de l'adresse réseau de la machine destinataire) issues de la couche Transport, convertissent les paquets en datagrammes et les communiquent à la couche Interface de réseau (voir figure).

Figure 8. Couche réseau de la suite de protocoles TCP/IP Cette illustration représente les différentes couches de la suite de protocoles TCP/IP. En partant du haut, la couche d'application comprend l'application. La couche de transport comprend UDP et TCP. La couche réseau comprend l'interface réseau (matériel). La couche matérielle contient le réseau physique.



TCP/IP fournit les protocoles requis pour être conforme à RFC 1100 *(Official Internet Protocols)*, ainsi que d'autres protocoles couramment utilisés par les machines hôtes en environnement Internet.

> **Remarque :** Sous **TCP/IP**, l'utilisation des numéros de réseau, version, prise, service et protocole Internet est également conforme à RFC 1010 *(Assigned Numbers)*.

Protocole de résolution d'adresse

Le premier protocole intervenant au niveau réseau est le **protocole de résolution d'adresse (ARP)**. Ce protocole est chargé de traduire dynamiquement les adresses Internet en adresses matérielles uniques sur les réseaux locaux.

Pour illustrer le fonctionnement d'**ARP**, prenons le cas de deux noeuds, `jim` et `fred`. Si le nœud `jim` désire communiquer avec `fred`, et que `jim` et `fred` ne résident pas sur le même réseau local (LAN), `jim` et `fred` doivent utiliser des *ponts*, *routeurs* ou *passerelles* et des adresses IP. Au sein d'un réseau local, les nœuds requièrent en outre les adresses matérielles (niveau inférieur).

Les noeuds implantés sur le même segment d'un réseau local font appel au protocole **ARP** pour déterminer l'adresse matérielle d'autres noeuds. Tout d'abord, le noeud `jim` diffuse

une demande **ARP** pour connaître l'adresse matérielle de `fred`. Cette demande comporte les adresses **IP** et matérielle de `jim` et l'adresse **IP** de `fred`. Lorsque `fred` reçoit la requête **ARP**, il place une entrée destinée à `jim` dans sa mémoire cache **ARP** (utilisée pour établir rapidement l'équivalence entre l'adresse **IP** et l'adresse matérielle). Ensuite, `fred` renvoie directement à `jim` une réponse **ARP** avec l'adresse IP et l'adresse matérielle de `fred`. Lorsque le nœud `jim` reçoit cette réponse, il place à son tour une entrée destinée à `fred` dans sa mémoire cache **ARP**.

Dès lors, `jim` peut correspondre directement avec `fred` sans recours au protocole **ARP** (à moins que l'entrée en mémoire cache **ARP** destinée à `fred` ne soit supprimée).

Contrairement à la plupart des protocoles, les en-têtes de paquet ARP n'ont pas un format fixe. Le message est conçu pour s'adapter à diverses technologies de réseau, telles que :

- Carte de réseau local Ethernet (qui prend en charge les protocoles Ethernet et 802.3)
- Carte de réseau en anneau à jeton
- Carte de réseau FDDI (Fiber Distributed Data Interface)

En revanche, **ARP** ne traduit pas les adresses pour **SLIP** ou convertisseur optique série **(SOC)**, car il s'agit de connexions point à point.

Les tables de traduction sont tenues à jour par le noyau et les utilisateurs ou les applications n'ont pas d'accès direct à **ARP**. Lorsqu'une application envoie un paquet Internet à l'un des pilotes d'interface, le pilote demande l'équivalence d'adresse. Si cette équivalence ne figure pas dans la table, un paquet ARP de diffusion est envoyé aux hôtes du réseau local via le pilote d'interface demandeur.

Les entrées de la table d'équivalence (mappage) **ARP** sont supprimées au bout de 20 minutes et les entrées incomplètes au bout de 3 minutes. Pour insérer une entrée permanente dans la table, lancez la commande **arp** assortie du paramètre *pub* :

```
arp –s 802.3 host2 0:dd:0:a:8s:0 pub
```

Lorsqu'un hôte prenant en charge **ARP** reçoit un paquet de demande **ARP**, il note l'adresse IP et l'adresse matérielle du système demandeur et met à jour sa table d'équivalence. Si son adresse **IP** ne correspond pas à l'adresse demandée, il rejette le paquet. Sinon, il envoie un paquet de réponse au système demandeur. Le système demandeur enregistre la nouvelle équivalence pour l'appliquer aux paquets Internet similaires en attente.

Protocole ICMP

Le deuxième protocole intervenant au niveau réseau est le protocole de message de contrôle interréseau (**ICMP**). Ce protocole, partie intégrante de toute implémentation **IP** gère les messages d'erreur et de contrôle pour **IP**. Il est utilisé par les passerelles et les systèmes hôtes pour transmettre les comptes rendus d'incidents aux machines émettrices d'un paquet. Il est chargé de :

- tester l'accessibilité d'une destination,
- signaler les erreurs de paramètres dans un en-tête de datagramme,
- effectuer la synchronisation horaire et évaluer le temps de transit,
- obtenir les adresses Internet et les masques de sous-réseau.

Remarque : **ICMP** utilise la prise en charge de base d'IP comme s'il s'agissait d'un protocole de niveau supérieur. **ICMP** fait partie intégrante du protocole IP et doit être mis en œuvre par tout module **IP**.

ICMP rend compte des anomalies de l'environnement de communications sans garantir pour autant la fiabilité du protocole **IP**. Autrement dit, il ne garantit pas la livraison d'un paquet IP ni l'envoi d'un message **ICMP** à l'hôte source en cas d'échec ou d'erreur de livraison.

Les messages **ICMP** sont émis dans les cas suivants :

- destination d'un paquet inaccessible,
- capacité tampon insuffisante sur l'hôte passerelle pour la réexpédition d'un paquet,
- passerelle capable d'obtenir que l'hôte achemine le courrier via un chemin plus court.

TCP/IP envoie et reçoit plusieurs types de message ICMP (reportez–vous à Types de messages ICMP page 4-22). Le protocole **ICMP** intégré au noyau, ne dispose d'aucune interface API.

Types de messages ICMP

ICMP peut envoyer ou recevoir des messages du type :

echo request	Demande d'écho envoyée par les hôtes et les passerelles pour tester l'accessibilité de la destination.
information request	Demande d'information envoyée par les hôtes et les passerelles pour obtenir l'adresse Internet d'un réseau auquel ils sont connectés. Avec ce type de message, la portion réseau de l'adresse de destination IP est positionnée à 0.
timestamp request	Demande de l'heure courante à la machine de destination.
address mask request	Demande de masque d'adresse envoyée par l'hôte pour identifier son masque de sous-réseau. Cette demande est envoyée à une passerelle s'il en connaît l'adresse ou sous forme de message de diffusion.
destination unreachable	Message envoyé lorsqu'une passerelle ne parvient pas à livrer un datagramme IP.
source quench	Demande effectuée auprès de l'émetteur de datagrammes lorsque son débit d'émission est trop élevé pour que les passerelles ou hôtes puissent traiter les datagrammes entrants.
redirect message	Message de redirection envoyé lorsqu'une passerelle détecte qu'un hôte n'utilise pas une route optimale.
echo reply	Réponse d'écho renvoyée, par la machine réceptrice, à l'émetteur d'une demande d'écho.
information reply	Message envoyé par les passerelles en réponse aux demandes d'adresse (avec les zones source et destination du datagramme IP renseignées).
timestamp reply	Réponse indiquant l'heure courante.
address mask reply	Réponse de masque d'adresse envoyée aux machines qui requièrent des masques de sous-réseau.
parameter problem	Message envoyé lorsqu'un hôte ou une passerelle relève une anomalie dans un en-tête de datagramme.
time exceeded	Message envoyé lorsque les conditions ci–dessous sont réunies : • A chaque datagramme IP est associée une durée de vie (nombre de bonds), décrémentée par chaque passerelle. • Un datagramme est rejeté par une passerelle, sa durée de vie ayant atteint la valeur 0.
Internet Timestamp	Horodateur Internet utilisé pour enregistrer les dates et heures durant le parcours.

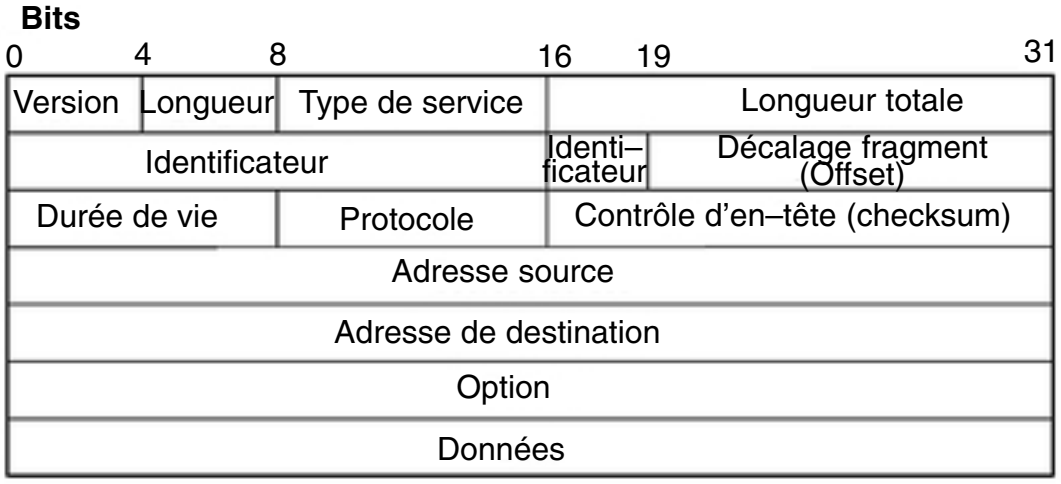
Protocole Internet

Le troisième protocole intervenant au niveau réseau est le **protocole Internet (IP)**. **IP** est un protocole sans connexion car il traite chaque paquet d'informations séparément. Il effectue la livraison des paquets pour Internet, sans garantie de livraison (aucun acquittement de message n'est exigé auprès des hôtes émetteur, récepteur et intermédiaires) et sans connexion (chaque paquet d'informations est traité séparément).

IP assure l'interface avec les protocoles de niveau Interface de réseau. Les connexions physiques d'un réseau transmettent l'information sous forme d'une trame composée d'un en-tête et de données. L'en-tête contient les adresses source et destination. **IP** utilise un datagramme Internet contenant des informations similaires à celles de la trame physique. Son en–tête comporte également les adresses Internet source et de destination des données.

IP définit le format des données acheminées sur le réseau Internet (voir figure).

Figure 9. En–tête de paquet IP (Internet Protocol) Cette illustration représente les premiers 32 bits d'un en–tête de paquet IP typique. Le tableau ci–dessous dresse la liste des différentes entités.



Définitions des zones d'en-tête IP

Version	Version **IP** utilisée. La version courante du protocole IP est 4.
Longueur	Longueur de l'en-tête du datagramme, en nombre de mots de 32 bits.
Type de service	Zone comprenant cinq champs qui définissent pour le paquet concerné le type de priorité, le délai, le débit et le niveau de fiabilité souhaités. Cette demande n'est pas garantie par Internet. Les paramètres par défaut de ces cinq champs sont normaux. Actuellement, cette zone n'est pas utilisée par Internet de façon généralisée. La mise en œuvre d'**IP** est conforme à la spécification **IP** RFC 791, *Internet Protocol*.

Longueur totale	Longueur du datagramme, en octets, incluant l'en-tête et les données. La fragmentation en paquets au niveau des passerelles et le réassemblage à destination sont assurés. La longueur totale du paquet IP peut être configurée par interface individuelle à l'aide de **wsm** de Web-based System Manager, de la commande **ifconfig** ou via le raccourci **smit chinet** de SMIT. Pour déclarer ces valeurs comme permanentes dans la base de données de configuration, utilisez Web-based System Manager ou SMIT, et pour les définir ou les modifier dans le système en exécution, utilisez la commande **ifconfig**.
Identificateur	Nombre entier unique identifiant le datagramme.
Indicateurs (flags) de fragment	Contrôle la fragmentation du datagramme ainsi que le champ Identification. Indique si le datagramme doit être fragmenté et si le fragment courant est le dernier.
Décalage fragment (Offset)	Décalage du fragment dans le datagramme d'origine, en unités de 8 octets.
Durée de vie	Durée de rétention du datagramme sur Internet. Ce paramètre évite de conserver indéfiniment sur Internet les datagrammes qui n'ont pas abouti. La durée de rétention par défaut est de 255 secondes.
Protocole	Type de protocole de niveau supérieur.
Total de contrôle d'en–tête (checksum)	Nombre calculé pour assurer l'intégrité des valeurs d'en-tête.
Adresse source	Adresse Internet de l'hôte émetteur.

Adresse de destination
: Adresse Internet de l'hôte récepteur.

Options
: Options de test et de mise au point du réseau. Zone facultative pour certains datagrammes.

  End of Option List
  : Indique la fin de la liste des options. Utilisé à la fin de la liste des options (et non de chaque option) Cette option doit être utilisée uniquement si la fin de la liste ne coïncide par avec la fin de l'en–tête **IP**. Cette option n'est utilisée que si les options excèdent la longueur du datagramme.

  No Operation
  : Permet l'alignement avec d'autres options. Par exemple, alignement à 32 bits du début de l'option suivante.

  Loose Source and record Route
  : Informations de routage fournies par la source du datagramme Internet aux passerelles, qui les utilisent pour expédier le datagramme à destination et les enregistrent. Il s'agit d'une route source *libre* : la passerelle ou l'**IP** hôte peut utiliser n'importe quelle route via un nombre quelconque de passerelles intermédiaires pour atteindre l'adresse suivante dans la route.

  Strict Source and record Route
  : Informations de routage fournies par la source du datagramme Internet aux passerelles, qui les utilisent pour expédier le datagramme à destination et les enregistrent. Il s'agit d'une route source *imposée* : la passerelle ou l'**IP** hôte doit envoyer le datagramme directement à l'adresse suivante spécifiée par la route source en passant par le réseau direct indiqué dans l'adresse, pour atteindre la passerelle ou l'hôte suivant spécifié dans la route.

  Record Route
  : Cette option permet d'enregistrer le parcours suivi par le datagramme Internet.

  Stream Identifier
  : Cette option véhicule un identificateur de flot (stream) à travers des réseaux qui ne prennent pas en charge le concept de flot.

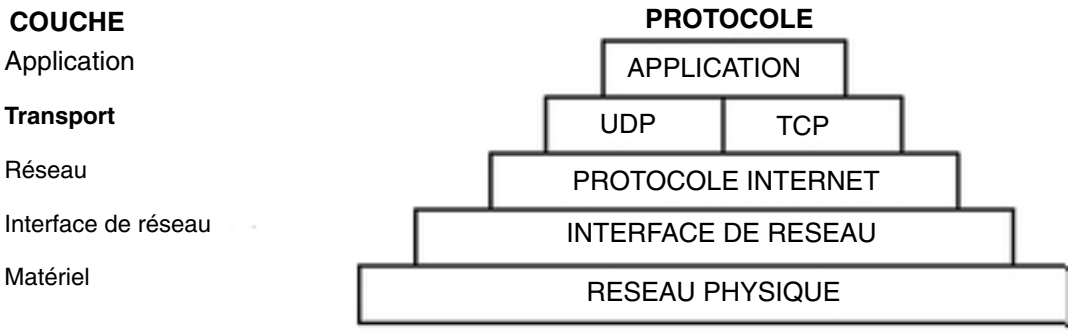
  Internet Timestamp
  : Enregistre la date et l'heure le long du parcours du datagramme.

L'en-tête **IP** est automatiquement préfixé aux paquets sortants, et supprimé des paquets entrants qui vont être envoyés aux protocoles de niveau supérieur. Le protocole IP procure un système d'adressage universel des hôtes sur le réseau Internet.

Protocoles Internet de niveau transport

Les protocoles **TCP/IP** de niveau transport (voir figure) permettent aux programmes d'application de communiquer entre eux.

Figure 10. Couche de transport de la suite de protocoles TCP/IP Cette illustration représente les différentes couches de la suite de protocoles TCP/IP. En partant du haut, la couche d'application comprend l'application. La couche de transport comprend UDP et TCP. La couche réseau comprend l'interface réseau (matériel). La couche matérielle contient le réseau physique.



Les protocoles UDP (User Datagram Protocol) et **TCP** en sont les principaux : ils autorisent l'interconnexion d'hôtes Internet et l'échange de messages entre applications implantées sur des hôtes différents. Le mécanisme est le suivant : lorsqu'une application envoie à la couche Transport une demande d'expédition de message, les protocoles **UDP** et **TCP** fragmentent l'information en paquets qu'ils dotent d'un en-tête portant l'adresse de destination. Ces paquets sont alors soumis par les protocoles à la couche réseau. Pour déterminer la destination exacte du message, les protocoles **TCP** et **UDP** se servent des ports de protocole de l'hôte.

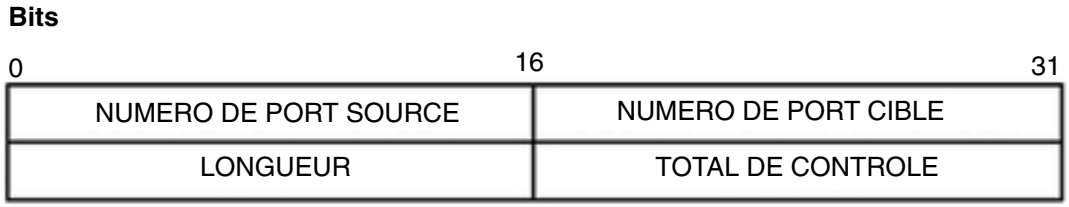
Les protocoles et applications de niveau supérieur utilisent **UDP** pour les connexions datagramme et **TCP** pour les connexion Stream (trains de données). Ces protocoles sont mis en œuvre par l'interface Sockets du système d'exploitation.

Protocole UDP

Le protocole **UDP** intervient lorsqu'une application de réseau doit envoyer des messages à une application ou un process d'un autre réseau : il fournit aux applications d'hôtes Internet le moyen de communiquer par datagramme. L'émetteur d'un message ne connaît pas les process actifs au moment de l'envoi, c'est pourquoi le protocole **UDP** utilise les ports de protocole de destination (ou sur un hôte, points de destination abstraits dans une machine), identifiés par des nombres entiers positifs, pour envoyer les messages à un ou plusieurs points de destination. A la réception des messages, les ports de protocole placent les messages dans des files d'attente, où ils seront récupérés en temps voulu par les applications du réseau récepteur.

UDP fait appel à l'**IP** sous–jacent pour envoyer ses datagrammes, il assure donc la livraison des messages sans connexion comme le fait le protocole IP, mais sans garantie de livraison ni de protection contre la duplication. **UDP** présente cependant deux particularités : il autorise l'émetteur à spécifier le numéro des ports source et cible et calcule le total de contrôle de l'en-tête et des données. Il offre ainsi aux applications émettrices et réceptrices un moyen de fiabiliser la livraison d'un message (voir figure).

Figure 11. En–tête de paquet UDP (User Datagram Protocol) Cette illustration représente les premiers 32 bits d'un en–tête de paquet UDP typique. Les 16 premiers bits contiennent le numéro de port source et la longueur. Les 16 bits suivants contiennent le numéro de port de destination et le total de contrôle.



Les applications qui exigent une garantie de livraison des datagrammes doivent exercer elles-mêmes un contrôle si elles utilisent **UDP**. Les applications qui exigent une garantie de livraison des flots de données doivent recourir à **TCP**.

Définitions des zones d'en-tête UDP

Numéro de port source	Adresse du port de protocole émetteur de l'information.
Numéro de port cible	Adresse du port de protocole récepteur de l'information.
Longueur	Longueur en octets du datagramme UDP.
Total de contrôle (Checksum)	Contrôle du datagramme UDP sur la base du même algorithme que le protocole IP.

L'interface de programmation d'applications (API) avec UDP est constituée d'un ensemble de sous-routines de bibliothèque fourni par l'interface Sockets.

Protocole TCP

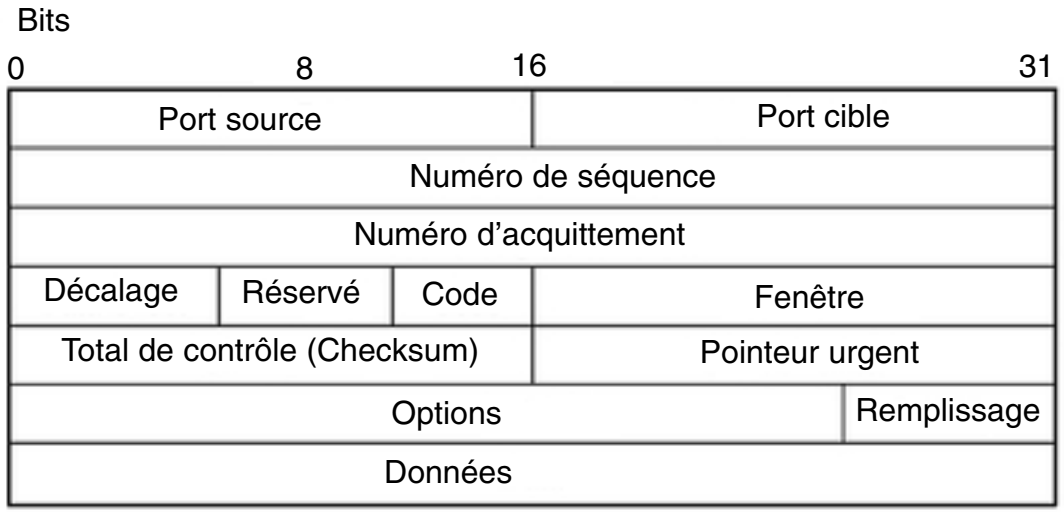
Le protocole **TCP** (Transmission Control Protocol) assure le transfert fiable des flots entre les hôtes Internet. Comme **UDP**, il fait appel au protocole sous-jacent **IP** pour véhiculer les datagrammes et en assurer la transmission par bloc en flot continu d'un port de process à l'autre. Contrairement à **UDP**, **TCP** garantit une livraison fiable des messages. Il garantit que les données ne seront livrées au process destinataire sans que les données soient altérées, perdues, dupliquées ou restituées dans le désordre. Ainsi, les programmeurs d'applications ne sont pas contraints de gérer ce type d'erreurs dans leur logiciel.

TCP présente les caractéristiques suivantes :

Transfert de données de base	**TCP** peut véhiculer entre ses utilisateurs un flot continu d'octets 8 bits en regroupant des octets en segments pour les transmettre par Internet. Avec **TCP**, la taille des segments atteint au moins 1024 octets. En général, c'est **TCP** qui détermine le moment propice pour assembler et expédier les paquets.
Fiabilité	**TCP** doit récupérer les données altérées, perdues, dupliquées ou désorganisées par Internet. Pour ce faire, il affecte un numéro de séquence à chaque octet transmis et exige un accusé de réception positif (ACK) de la part du **TCP** récepteur. S'il ne reçoit pas cet accusé après un certain délai, les données sont retransmises. Ce délai est fixé dynamiquement pour chaque connexion, en fonction du temps de transmission aller-retour. Côté destinataire, les numéros de séquence servent à réordonner les segments et à éliminer les doublons. Les données altérées sont traitées grâce au total de contrôle ajouté à chaque segment : ce total est vérifié à la réception des segments et les segments altérés sont rejetés.
Contrôle de flux	**TCP** permet de réguler le débit des données émises, en associant à chaque accusé de réception une fenêtre indiquant l'intervalle de numéros de séquence admis au-delà du dernier segment reçu. La fenêtre précise le nombre d'octets que l'émetteur est autorisé à envoyer avant de recevoir la prochaine autorisation.
Multiplexage	**TCP** permet à un grand nombre de process d'un même hôte d'utiliser simultanément les fonctions de communication **TCP**. **TCP** reçoit un ensemble d'adresses de ports pour chaque hôte et combine le numéro de port à l'adresse réseau et à l'adresse hôte pour pouvoir identifier chaque prise de façon unique. Une paire de prises identifie à son tour chaque connexion de façon unique.
Connexions	**TCP** doit initialiser et tenir à jour certaines informations d'état pour chaque flot de données. La combinaison de ces informations (prises, numéros de séquence, tailles de fenêtre) est appelée connexion. Chaque connexion est identifiée par une paire de prises uniques, une pour chaque extrémité.
Priorité et protection	Les utilisateurs de **TCP** peuvent spécifier un niveau de priorité et de protection pour leurs communications. Sinon, des valeurs par défaut sont prévues.

La figure d'un **en–tête de paquet TCP** illustre ces caractéristiques.

Figure 12. En–tête de paquet TCP (Transmission Control Protocol) Cette illustration représente le contenu de l'en–tête du paquet TCP. Les entités individuelles sont répertoriées dans le texte ci–dessous.



Définitions de zones d'en-tête TCP

Port source — Numéro de port du programme d'application source.

Port cible — Numéro de port du programme d'application cible.

Numéro de séquence — Numéro d'ordre du premier octet de données dans le segment.

Numéro d'acquittement — Numéro identifiant la position du plus grand octet reçu.

Décalage — Décalage (Offset) de la portion de données du segment.

Réservé — Zone réservée à un usage ultérieur.

Code — Bits de contrôle servant à identifier l'objet d'un segment :

URG
La zone Pointeur urgent est valide.

ACK
La zone Acquittement est valide.

PSH
Le segment requiert un PUSH.

RTS
Réinitialise la connexion.

SYN
Synchronise les numéros de séquence.

FIN
Fin du flot d'octets.

Fenêtre — Volume de données admissible par la destination.

Total de contrôle (Checksum) — Vérifie l'intégrité des données et de l'en-tête du segment.

Pointeur urgent
Indique les données à livrer dès que possible. Le pointeur marque la fin des données urgentes.

Options

End of Option List
Utilisé à la fin de la liste des options options (et non de chaque option) uniquement si la fin de la liste ne coïncide par avec la fin de l'en–tête **TCP**.

No Operation
Indique la limite entre deux options. Par exemple, alignement du début d'une option suivante sur un mot. L'émetteur n'étant pas obligé d'utiliser cette option, le destinataire doit être prêt à traiter les options même ne commençant pas sur un mot.

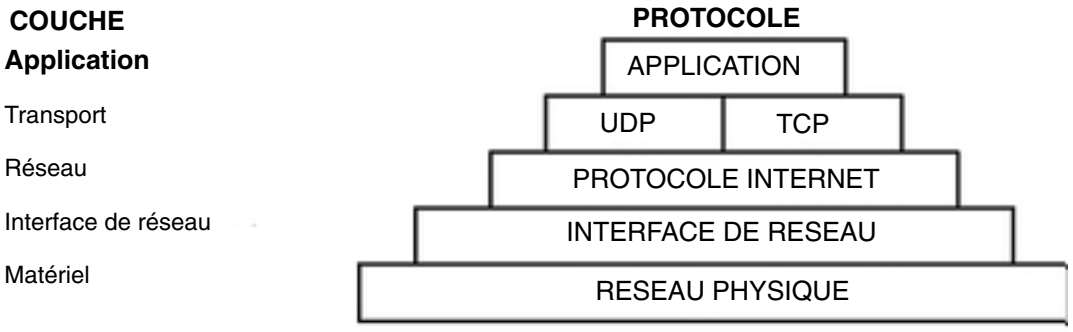
Maximum Segment Size
Taille maximale de segment acceptable par **TCP** (indiquée dans la demande de connexion initiale). Cette option doit être envoyée uniquement dans la demande de connexion initiale.

L'interface de programmation d'applications (API) avec **TCP** est constituée d'un ensemble de sous-routines de bibliothèque fourni par l'interface Sockets.

Protocoles Internet de niveau application

Au niveau du programme d'application, **TCP/IP** met en œuvre des protocoles Internet de niveau supérieur (voir figure).

Figure 13. Couche d'application de la suite de protocoles TCP/IP Cette illustration représente les différentes couches de la suite de protocoles TCP/IP. En partant du haut, la couche d'application comprend l'application. La couche de transport comprend UDP et TCP. La couche réseau comprend l'interface réseau (matériel). La couche matérielle contient le réseau physique.



Lorsqu'une application doit envoyer des données à une application sur un hôte différent, les informations sont envoyées aux protocoles de niveau transport pour être préparées à la transmission.

Les protocoles Internet de niveau application officiels englobent :

- **Domain Name Protocol** (Domain Name Protocol)
- **Exterior Gateway Protoco** l (Exterior Gateway Protocol)
- **File Transfer Protocol** (File Transfer Protocol)
- **Name/Finger Protocol** (Name/Finger Protocol)
- **Telnet Protocol** (Telnet Protocol)
- **Trivial File Transfer Protocol** (Trivial File Transfer Protocol)

TCP/IP met en œuvre d'autres protocoles de niveau supérieur, non officiels, mais couramment utilisés par la communauté Internet pour les programmes d'application. A savoir :

- Distributed Computer Network (DCN) **Local–Network Protocol** (Distributed Computer Network Local–Network Protocol)
- **Remote Command Execution Protocol** (Remote Command Execution Protocol)
- **Remote Login Protocol** (Remote Login Protocol)
- **Remote Shell Protocol** (Remote Shell Protocol)
- **Routing Information Protocol** (Routing Information Protocol)
- **Time Server Protocol** (Time Server Protocol).

TCP/IP ne fournit pas d'interface API à ces protocoles.

Protocole DOMAIN

Le protocole **DOMAIN** permet à un système hôte membre d'un domaine de jouer le rôle de *serveur de noms* auprès des autres systèmes hôtes de son domaine. Il utilise comme protocole sous-jacent le protocole **UDP** ou **TCP** et permet à un réseau local d'affecter des noms d'hôte dans son domaine indépendamment des autres domaines. Normalement, le protocole **DOMAIN** utilise **UDP**. Toutefois, si la réponse **UDP** est tronquée, **DOMAIN** fait appel au protocole **TCP**. Le protocole **DOMAIN** de **TCP/IP** prend en charge les deux.

Pour résoudre les noms et adresses Internet, les routines de résolution locales du système d'appellation hiérarchique **DOMAIN** peuvent recourir à la base de résolution de noms locale tenue par le démon **named**. Si le nom demandé par l'hôte ne figure pas dans cette base, la routine de résolution interroge un serveur de noms **DOMAIN** distant. Dans tous les cas, en cas d'échec, la routine tente d'utiliser le fichier **/etc/hosts.**

Remarque : TCP/IP configure les routines de résolution locales pour le protocole **DOMAIN**, si le fichier local **/etc/resolv.conf** existe. Sinon, **TCP/IP** les configure pour qu'elles utilisent la base de données **/etc/hosts**.

TCP/IP implémente le protocole **DOMAIN** dans le démon **named** et les routines de résolution, mais ne lui fournit pas d'interface API.

Protocole EGP

Le protocole **EGP (Exterior Gateway Protocol)** est le mécanisme qui permet à la passerelle extérieure d'un *système autonome* de partager les informations de routage avec des passerelles extérieures d'autres systèmes autonomes.

Systèmes autonomes

Un système autonome est un groupe de réseaux et de passerelles sous la responsabilité d'une autorité administrative. Les passerelles sont dites *intérieures limitrophes* si elles résident sur le même système autonome et *extérieures limitrophes* si elles résident sur des systèmes autonomes différents. Les passerelles qui échangent des informations de routage via le protocole **EGP** sont appelées *passerelles limitrophes* ou *homologues **EGP**.*
Le protocole **EGP** permet aux passerelles de systèmes autonomes d'accéder aux informations de leurs homologues **EGP**.

Via **EGP**, une passerelle extérieure peut demander à échanger des informations d'accès avec une autre passerelle extérieure. **EGP** vérifie en permanence que ses passerelles homologues répondent aux demandes, et les aident dans ces échanges par des messages de mise à jour de routage.

EGP limite la portée d'une passerelle extérieure aux réseaux de destination accessibles en tous points dans le système autonome de cette passerelle. Autrement dit, une passerelle extérieure utilisant **EGP** peut transmettre les informations aux passerelles **EGP** limitrophes, mais ne peut fournir des informations concernant ses passerelles limitrophes hors de son système autonome.

EGP n'interprète aucune distance métrique spécifiée dans les messages de mise à jour de routage issus d'autres protocoles. **EGP** utilise la zone de distance pour indiquer si un chemin existe (la valeur 255 signifiant qu'un réseau est inaccessible). La valeur spécifiée ne peut pas servir à déterminer le chemin le plus court entre deux routes, sauf si ces dernières sont situées dans un seul système autonome. C'est pourquoi **EGP** n'est pas utilisé comme algorithme de routage. De ce fait, un seul chemin peut être emprunté entre la passerelle extérieure et un réseau.

Contrairement au protocole **RIP (Routing Information Protocol)**, qui peut être appliqué à un système autonome de réseaux Internet qui reconfigurent dynamiquement les routes, les routes **EGP** sont prédéterminées dans le fichier **/etc/gated.conf**. **EGP** considère que IP est le protocole sous-jacent implicite.

Types de messages EGP

Neighbor Acquisition Request	Demande émise par les passerelles extérieures pour devenir limitrophes.
Neighbor Acquisition Reply	Réponse favorable des passerelles extérieures pour devenir limitrophes.
Neighbor Acquisition Refusal	Réponse défavorable des passerelles extérieures pour devenir limitrophes. Les raisons du refus sont indiquées dans le message, par exemple `out of table space`.
Neighbor Cease	Demande émise par les passerelles extérieures pour mettre fin à une relation limitrophe. Les raisons sont indiquées dans le message, par exemple, `going down`.
Neighbor Cease Acknowledgment	Acceptation par les passerelles extérieures de la demande d'interruption d'une relation limitrophe.
Neighbor Hello	Message émis par une passerelle limitrophe pour vérifier qu'une connexion est active. Une passerelle émet un message `Hello` et la passerelle interrogée confirme la connexion en émettant la réponse `I Heard You`.
I Heard You	Réponse d'une passerelle extérieure au message `Hello`. Le message `I Heard You` s'accompagne des informations d'accès à la passerelle qui émet la réponse et, si la passerelle est inaccessible, d'un message d'explication, par exemple `You are unreachable due to problems wi th my network interface`.
NR Poll	Interrogation émise par les passerelles extérieures auprès des passerelles limitrophes pour déterminer leur capacité d'accès aux autres passerelles.
Network Reachability	Réponse des passerelles extérieures au message `NR Poll`. Pour chaque passerelle interrogée, le message `Network Reachability` indique les adresses auxquelles la passerelle limitrophe lui donne accès.
EGP Error	Réponse émise par les passerelles extérieures aux messages EGP qui présentent des totaux de contrôle ou des valeurs de zones erronés.

TCP/IP implémente le protocole EGP dans le démon **gated** mais ne lui fournit pas d'interface de programmation d'applications (API).

Protocole FTP

Le protocole **FTP (File Transfer Protocol)** permet le transfert des données entre hôtes hétérogènes et le transfert indirect de fichiers entre deux hôtes étrangers. Il donne accès à la liste des répertoires distants, permet de changer de répertoire distant courant, de créer ou de supprimer des répertoires distants et de transférer plusieurs fichiers en une seule demande. Un système de protection par mot de passe et numéro de compte utilisateur est assuré au niveau de l'hôte étranger. Conçu à l'origine pour des applications, **FTP** est également utilisé pour les sessions interactives orientées utilisateur.

FTP a recours au transfert fiable de flot **(TCP/IP)** pour l'envoi des fichiers, et à une connexion Telnet pour le transfert des commandes et des réponses. **FTP** reconnaît plusieurs formats de fichiers de base, notamment NETASCII, IMAGE et Local 8.

TCP/IP implémente **FTP** dans les commandes **ftpet** (utilisateur) et **ftpd** (serveur) mais ne fournit pas d'interface de programmation d'applications (API) avec ce protocole.

Si vous créez des répertoires et utilisateurs ftp anonymes, veillez à ce que le répertoire personnel des utilisateurs ftp et anonymes (par exemple, **/u/ftp**) appartienne à un utilisateur racine mais ne soit pas accessible en écriture (par exemple, `dr-xr-xr-x`). Vous pouvez utiliser le script **/usr/samples/tcpip/anon.ftp** pour créer ces comptes, fichiers et répertoires.

Protocole Telnet

Le protocole **TELNET** fournit une méthode de communication standard pour les terminaux et process orientés terminal. **TELNET** est utilisé couramment par les programmes d'émulation de terminal pour la connexion à un hôte distant. Il sert à la communication de terminal à terminal et inter–process, et est sollicité par d'autres protocoles (par exemple, **FTP**) pour l'établissement d'un canal de contrôle de protocole.

TCP/IP implémente **TELNET** dans les commandes utilisateur **tn, telnet,** ou **tn3270**. Le démon **telnetd** ne fournit pas d'interface API pour **TELNET**.

TCP/IP accepte les options Telnet négociées entre le client et le serveur :

BINARY TRANSMISSION (pour sessions **tn3270**)	Transmet les caractères sous forme de données binaires.
SUPPRESS GO_AHEAD (Le système d'exploitation supprime les options GO–AHEAD.)	Lors de la transmission de données, à la demande de l'expéditeur des données, ne transmet pas au destinataire d'option GO_AHEAD. Si cette option n'est pas acceptée, les interlocuteurs suppriment la connexion dans les deux directions. Cette action doit être exécutée de manière autonome dans les deux directions.
TIMING MARK (Reconnue mais reçoit une réponse négative)	Vérifie que les données transmises ont été entièrement traitées.
EXTENDED OPTIONS LIST	Fournit la possibilité de 256 options supplémentaires à la liste des options **TELNET**. Sans cette option, **TELNET** admet un maximum de 256 options.
ECHO (Commande modifiable par l'utilisateur)	Transmet les caractères d'écho déjà renvoyés à l'expéditeur d'origine.
TERM TYPE	Permet au serveur de déterminer le type de terminal connecté à un programme utilisateur **TELNET**.

SAK (Secure Attention Key)	Sécurise la communication entre vous et le système.
NAWS (Negotiate About Window Size)	Dans une relation client–serveur, permet aux deux parties de négocier la taille de la fenêtre (si les applications l'autorisent).

Remarque : **Telnet** doit autoriser la transmission de caractères 8 bits en mode non binaire pour l'implémentation de la page de code ISO 8859 Latin. Cette condition est nécessaire pour l'internationalisation des commandes **TCP/IP**.

Protocole TFTP

Le protocole **TFTP (Trivial File Transfer Protocol)** peut lire et enregistrer des fichiers issus de ou destinés à un hôte distant. **TFTP** est généralement plus rapide que **FTP** car, pour acheminer les fichiers, il fait appel au protocole **UDP** qui ne garantit pas la livraison des fichiers. Comme **FTP**, **TFTP** peut traiter les fichiers sous forme de données NETASCII ou binaires 8 bits. En revanche, il ne permet pas de lister ou de modifier les répertoires d'un hôte distant et ne prévoit pas de protection de type mot de passe. De plus, sous **TFTP**, l'écriture et la recherche des données sont limitées aux répertoires publics.

TCP/IP implémente **TFTP** dans les commandes utilisateur **tftp** et **utftp,** et dans la commande serveur **tftpd**. La commande **utftp** est une variante de la commande **tftp** utilisable dans les chaînages (pipes). **TCP/IP** ne fournit pas d'interface API pour le protocole FINGER.

Protocole FINGER

FINGER est un protocole Internet de niveau application qui joue le rôle d'interface entre la commande **finger** et le démon **fingerd**. Le démon **fingerd** renvoie les informations sur les utilisateurs connectés à un hôte distant spécifique. Pour limiter la commande à un utilisateur donné, spécifiez-le (commande finger). **FINGER** utilise **Transmission Control Protocol** (Transmission Control Protocol) comme protocole sous–jacent.

Remarque : TCP/IP ne fournit pas d'interface API pour le protocole FINGER.

Protocole HELLO

Le **protocole de réseau local distribué HELLO** s'applique aux passerelles intérieures et doit être utilisé dans des systèmes autonomes. (Pour plus de détails, reportez-vous à Systèmes autonomes, page 4-31.) **HELLO** est chargé de tenir à jour les informations de connectivité, de routage et d'horloge. Il permet à chaque machine de trouver le chemin le plus rapide vers la destination et met à jour dynamiquement l'information de routage vers cette destination.

Ce protocole est fourni par le démon **gated**.

Protocole REXEC

Le protocole d'exécution à distance, fourni par la commande utilisateur **rexec** et le démon **rexecd**, permet de lancer des commandes sur un hôte distant compatible.

Protocole LOGIN

Le protocole de connexion à distance LOGIN, fourni par la commande utilisateur **rlogin** et le démon **rlogind**, permet aux utilisateurs de se connecter à un hôte distant et d'utiliser leur terminal comme s'ils étaient connectés directement à cet hôte.

Protocole SHELL

Le protocole de commande à distance SHELL, fourni par la commande utilisateur **rsh** et le démon **rshd**, permet d'ouvrir un shell sur un hôte étranger compatible pour y exécuter des commandes.

Protocole RIP

Le protocole de routage **RIP (Routing Information protocol)** et les démons **routed** et **gated** qui le mettent en œuvre, sont chargés de suivre les informations de routage (en fonction du nombre de bonds effectués) et de tenir à jour les entrées de la table de routage du noyau.

Protocole TIMED

Le démon **timed** est chargé de la synchronisation horaire des hôtes. Il est fondé sur le concept de client/serveur.

Nombres réservés

Dans un souci de compatibilité avec l'environnement de réseau général, des nombres connus sont attribués aux versions, réseaux, ports, protocoles et options de protocoles Internet, de même qu'aux machines, réseaux, systèmes d'exploitation, protocoles, services et terminaux. TCP/IP applique les numéros et noms définis par la norme RFC 1010, *Nombres réservés.*

Une zone de 4 bits est prévue dans l'en-tête **IP** pour identifier la version du protocole interréseau utilisé. Le numéro de version d'**IP** en décimal est 4. Pour plus d'informations sur les nombres et noms réservés de **TCP/IP**, reportez–vous aux fichiers **/etc/protocols** et **/etc/services** inclus dans TCP/IP. Pour les noms et nombres réservés en général, reportez-vous à la norme RFC 1010 et au fichier **/etc/services**.

Cartes de réseau local (LAN) TCP/IP

Cette section traite des points suivants :

La carte réseau est le dispositif matériel raccordé physiquement aux câbles du réseau. Elle est chargée de recevoir et de transmettre les données au niveau physique. Elle est contrôlée par le pilote de carte.

Chaque machine doit être équipée d'autant de cartes réseau (ou connexions) que de réseaux auxquels elle est connectée. Par exemple, si un hôte est raccordé à deux réseaux en anneau à jeton, il doit être équipé de deux cartes réseau.

TCP/IP utilise les cartes réseau et connexions suivantes :

- Ethernet standard version 2
- IEEE 802.3
- Anneau à jeton
- Cartes asynchrones et ports série natifs (décrit dans *AIX 5L Version 5.2 Asynchronous Communication Guide*)
- Interface FDDI (Fiber Distributed Data Interface)
- Convertisseur de canal optique série (décrit dans *AIX 5L Version 5.2 Kernel Extensions and Device Support Programming Concepts*)
- 100 and 155 ATM
- ATM (mode de transfert asynchrone)

Les technologies de réseau Ethernet et 802.3 utilisent le même type de carte.

Chaque machine offre un nombre limité d'emplacements d'extension, que vous pouvez utiliser pour les cartes de communication. En outre, chaque machine ne prend en charge qu'un nombre limité de cartes de communication d'un type donné : 8 cartes Ethernet/802.3 maximum, 8 cartes en anneau à jeton maximum et une seule carte asynchrone de 64 ports maximum. Dès lors, vous pouvez installer sur votre machine n'importe quelle combinaison de ces cartes en respectant les contraintes logicielles (nombre et type de carte) et matérielles (nombre total d'emplacements d'extension disponibles).

Une seule interface TCP/IP doit être configurée, quel que soit le nombre de convertisseurs optiques série pris en charge par le système. Le pilote d'unité **Optique série** exploite les deux convertisseurs de canal même si une seule interface logique TCP/IP est configurée.

Installation d'une carte réseau

Pour installer une carte réseau :

1. Arrêtez l'ordinateur. Pour l'arrêt système, reportez-vous à la commande **shutdown**.
2. Mettez la machine hors tension.
3. Déposez le capot de l'ordinateur.
4. Recherchez un connecteur libre et insérez la carte réseau.
   Veillez à enclencher correctement la carte dans le connecteur.
5. Remettez le capot de l'ordinateur.
6. Relancez l'ordinateur.

Configuration et gestion des cartes

Pour configurer et gérer les cartes pour réseau en anneau à jeton ou Ethernet, suivez les procédures du tableau suivant.

Configuration et gestion des tâches relatives aux cartes			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment[5]

Configuration d'une carte	**smit chgtok** (anneau à jeton)
	smit chgenet (Ethernet)
		1Recherchez le nom de la carte :[1]
		`lsdev -C -c adapter -t tokenring -H`
		ou
		`lsdev -C -c adapter -t ethernet -H`
		2Redéfinissez la vitesse de l'anneau (anneau à jeton) ou le type de connecteur (Ethernet), si nécessaire. Par exemple :
		`chdev -l tok0 -a ring_speed=16 -P`
		ou
		`chdev -l ent0 -a bnc_select=dix -P`	

Détermination de l'adresse matérielle de la carte réseau	**smit chgtok** (anneau à jeton) **smit chgenet** (Ethernet)	`lscfg -l tok0 -v` (token ring)[2] `lscfg -l ent0 -v` (Ethernet)[2]	

Définition d'une adresse matérielle secondaire	**smit chgtok** (anneau à jeton)
	smit chgenet (Ethernet)
		1Définissez l'adresse matérielle secondaire. Par exemple, pour anneau à jeton : [2,3]
		`chdev -l tok0 -a alt_addr=0X10005A4F1B7F`
		Pour Ethernet : [2,3]
		`chdev -l ent0 -a alt_addr=0X10005A4F1B7F -p`
		2Commencez à utiliser l'adresse secondaire, pour anneau à jeton : [4]
		`chdev -l tok0 -a use_alt_addr=yes`
		Pour Ethernet : [4]
		`chdev -l ent0 -a use_alt_addr=yes`	

Remarques :

1. Le nom de la carte réseau peut changer si vous l'installez à un autre emplacement ou que vous la retirez du système. Dans ce cas, veillez à mettre à jour la base de données de configuration via la commande **diag –a.**
2. Indiquez le nom de votre carte à la place de `tok0` et `ent0`.
3. Remplacez par l'adresse matérielle `0X10005A4F1B7F`.
4. Une interruption de communication peut se produire après cette opération jusqu'à ce que les hôtes vident leur mémoire cache ARP et enregistrent cette nouvelle adresse matérielle.
5. Ces tâches ne sont pas disponibles dans Web-based System Manager Management Environment.

Configuration et utilisation des réseaux locaux virtuels (VLAN)

Les réseaux VLAN (Virtual Local Area Networks) ont une structure de type domaine de diffusion logique. Un réseau VLAN divise les groupes d'utilisateurs d'un réseau physique réel en segments de réseaux logiques. Cette mise en œuvre prend en charge la norme de repérage VLAN IEEE 802.1Q, ainsi que la fonction de prise en charge de plusieurs ID VLAN exécutés sur des cartes Ethernet. Chaque ID VLAN est associé aux couches supérieures (IP, etc.) grâce à une interface Ethernet distincte, et crée une entité logique de carte Ethernet par réseau VLAN, par exemple `ent1`, `ent2` et ainsi de suite.

IEEE 802.1Q de VLAN peut être configurée pour n'importe quelle carte Ethernet prise en charge. Les cartes doivent être connectées à un commutateur qui prend en charge la norme IEEE 802.1Q de VLAN.

Vous pouvez configurer plusieurs unités logiques VLAN sur un seul système. Chaque unité logique VLAN constitue une entité de carte Ethernet supplémentaire. Ces unités logiques peuvent être utilisées pour configurer les mêmes interfaces IP Ethernet telles qu'elles sont utilisées avec les cartes physiques Ethernet. Par conséquent, vous devez augmenter la valeur de l'option *ifsize* de la commande **no** (dont la valeur par défaut est 8), pour inclure les interfaces Ethernet pour chaque carte, mais également toutes les unités logiques VLAN configurées. Reportez–vous à la documentation de la commande **no**.

A chaque VLAN, vous pouvez associer une valeur différente de MTU (unité de transmission maximum) même si vous partagez une carte Ethernet physique individuelle.

La prise en charge de VLAN est gérée par SMIT. A partir de la ligne de commande, tapez le raccourci **smit vlan** et sélectionnez les éléments appropriés dans le menu principal de VLAN. Vous pouvez également recourir à l'aide en ligne.

Après la configuration de VLAN, configurez l'interface IP (par exemple, `en1` pour Ethernet standard ou `et1` pour IEEE 802.3), à l'aide de Web-based System Manager, de SMIT ou des commandes.

Remarques :

1. Si vous tentez de configurer une valeur ID de VLAN alors qu'il est en cours d'utilisation par la carte spécifiée, la configuration échoue et le message d'erreur suivant s'affiche :

```
Method error (/usr/lib/methods/chgvlan):
  0514-018 The values specified for the following attributes
      are not valid:
      vlan_tag_id  ID indicateur VLAN
```

2. Si un utilisateur (l'interface IP par exemple) utilise actuellement l'unité logique VLAN, toute tentative de retirer l'unité logique VLAN échoue. Un message similaire à l'exemple suivant s'affiche :

```
Method error (/usr/lib/methods/ucfgcommo):
    0514-062 Cannot perform the requested function because the
        specified device is busy.
```

Pour retirer l'unité VLAN logique, détachez d'abord l'utilisateur. Par exemple, si l'utilisateur est l'interface IP `en1`, vous pouvez utiliser la commande suivante :

```
ifconfig en1 detach
```

Ensuite, retirez l'interface de réseau à l'aide des menus TCP/IP de SMIT.

3. Si un utilisateur (l'interface IP par exemple) utilise actuellement l'unité logique VLAN, toute tentative de modifier les caractéristiques VLAN (ID indicateur VLAN ou carte de base) échoue. Un message similaire à l'exemple suivant s'affiche :

```
Method error (/usr/lib/methods/chgvlan):
    0514-062 Cannot perform the requested function because the
        specified device is busy.
```

Pour modifier l'unité VLAN logique, détachez d'abord l'utilisateur. Par exemple, si l'utilisateur est l'interface IP `en1`, vous pourriez utiliser la commande suivante :

```
ifconfig en1 detach
```

Ensuite, modifiez le VLAN et ajoutez de nouveau l'interface de réseau à l'aide des menus TCP/IP de SMIT.

Identification des incidents

Pour identifier les incidents relatifs à VLAN, vous pouvez utiliser **tcpdump** et **trace**. Le tableau suivant décrit l'ID du suivi d'erreur pour chaque type de paquet de transmission :

paquets de transmission	3FD
paquets de réception	3FE
autres événements	3FF

La commande **entstat** affiche les valeurs totales des statistiques de la carte physique pour laquelle VLAN est configuré. Elle *ne* fournit pas les statistiques individuelles de l'unité logique VLAN spécifique.

Restrictions

Le cliché à distance n'est pas pris en charge avec un VLAN. De plus, vous ne pouvez pas utiliser les unités logiques VLAN pour créer un Etherchannel Cisco Systems.

Utilisation de cartes ATM

La norme internationale ATM (Asynchronous Transfer Mode) définit une méthode de transmission à grande vitesse pour le transport d'éléments mixtes composés d'audio, vidéo, et de données informatiques ordinaires sur des réseaux locaux, métropolitains et longue distance (LAN, MAN et WAN). Les cartes ATM offrent une connectivité en duplex intégral pour les serveurs ESCALA ou pour les clients qui utilisent des circuits virtuels permanents (PVC) et des circuits virtuels commutés (SVC). Les mises en œuvre PVC et SVC sont conformes aux spécifications ATM Forum. Le nombre maximum de circuits virtuels pris en charge varie selon la carte. La plupart des cartes prennent en charge un minimum de 1024 circuits virtuels.

Technologie ATM

ATM (Asynchronous Transfer Mode) est une technologie de commutation de cellules, orientée connexion. Sur un réseau ATM, les terminaux sont raccordés au réseau via des connexions en duplex intégral dédiées. Les réseaux ATM sont construits sur la base de commutateurs interconnectés par des connexions physiques dédiées. Pour que le transfert de données puisse avoir lieu, des connexions de bout en bout doivent être établies. Une interface physique unique peut assurer des connexions multiples. Les stations émettrices transmettent les données en segmentant les unités PDU (Protocol Data Unit) en cellules de 53-octets. La charge est maintenue sous forme de cellules lors du transport sur le réseau. C'est au niveau des stations réceptrices que les cellules sont réassemblées en PDU. Les connexions sont identifiées par un identificateur de chemin virtuel (VPI) et un identificateur

de canal virtuel (VCI). Le champ VPI occupe 1 octet dans l'en–tête de 5 octets de la cellule ATM ; tandis que le champ VCI occupe 2 octets dans l'en–tête de 5 octets de la cellule ATM. En d'autres termes, une paire VPI:VCI identifie la source de la cellule ATM. Le commutateur ATM a pour fonction d'identifier l'origine de la cellule, de déterminer le saut suivant et de diriger la cellule vers un port. La paire VPI:VCI change sur une base saut par saut. Aussi les valeurs VPI:VCI ne sont-elles pas universelles. Un circuit virtuel est décrit par la concaténation de valeurs VPI:VCI à travers le réseau.

Connexions ATM

Dans l'architecture ATM, il existe deux types de circuits virtuels : permanent (PVC) et commuté (SVC).

Circuits virtuels permanents (PVC)	La configuration des PVC est statique et manuelle. Les commutateurs composant le réseau ATM doivent être configurés au préalable de façon à reconnaître la combinaison VPI:VCI de chaque terminal et à acheminer les cellules ATM de ces points via le réseau ATM. Après l'établissement d'une liaison de réseau entre deux points d'extrémité, les cellules ATM peuvent être transmises à travers le réseau ATM et les commutateurs ATM. Pour pouvoir acheminer la cellule vers sa destination, les commutateurs de réseau doivent convertir les valeurs VPI:VCI de manière appropriée.
Circuits virtuels commutés (SVC)	Les SVC sont configurés dynamiquement, sur la base des besoins. Les terminaux ATM sont affectés d'adresses de 20-octets. Deux concepts entrent en jeu : le panneau de contrôle et le panneau de données. Le panneau de contrôle utilise une paire de canaux de signalisation VPI:VCI 0:5. Les SVC initient sur demande une configuration d'appel, permettant à une station ATM d'envoyer des éléments d'information spécifiant l'adresse ATM de destination (et, éventuellement, l'adresse ATM source). Généralement, la station appelante, le réseau et la station appelée interviennent dans la négociation. Finalement, un appel est soit accepté, soit rejeté. S'il est accepté, le réseau affecte des valeurs VPI:VCI au panneau de données des deux stations (appelante et appelée). Sur le panneau de contrôle, le réseau ATM achemine (ou commute) les paquets de signaux sur la base des adresses ATM. Pendant le routage de ces paquets, les commutateurs définissent les tables de routage des cellules du panneau de données. Sur le panneau de données, les réseaux ATM commutent les cellules sur la base des VPI:VCI, presque comme dans le cas des PVC. A la fin du transfert, la connexion est close.

L'adresse ATM est construite par enregistrement sur le réseau ATM et acquisition des 13 octets les plus significatifs. Les 6 octets suivants correspondent à l'adresse matérielle "gravée" dans la carte. L'octet le moins important est le sélecteur ; son utilisation est laissée à la discrétion de l'utilisateur. Les réseaux ATM n'interprètent pas cet octet.

TCP/IP sur ATM

Les normes de *Internet Engineering Task Force RFC1577, Classical IP et ARP,* spécifient le mécanisme d'implémentation d'IP sur ATM. ATM étant une technologie orientée connexion et IP une technologie orientée datagramme, le mappage IP-ATM n'est pas évident.

Un réseau ATM est le plus souvent réparti en sous–réseaux IP logiques (LIS). Chacun est composé d'un certain nombre de stations ATM. Les LIS présentent des analogies avec les segments LAN classiques. Les LIS sont interconnectés par des routeurs. Une carte donnée (sur une station ATM) peut faire partie de plusieurs LIS. Cette fonction est très utile pour la mise en œuvre de routeurs.

RFC1577 spécifie RFC1483 qui spécifie LLC/SNAP Encapsulation comme valeur par défaut. Dans les réseaux PVC, pour chaque station IP, tous les PVC doivent être définis manuellement, par configuration des valeurs VPI:VCI. Si l'encapsulage LLC/SNAP n'est pas utilisé, l'adresse IP de destination associée à chaque VPI:VCI doit être définie.
Si l'encapsulage LLC/SNAP est utilisé, la station IP peut connaître l'adresse IP distante par le biais d'un mécanisme InARP. Pour les réseaux SVC, RFC1577 spécifie un serveur ARP pour chaque LIS. L'objet de ce serveur est de convertir les adresses IP en adresses ATM sans utiliser de messages de diffusion. Chaque station IP est configurée avec l'adresse ATM du serveur ARP. Les stations IP configurent les SVC avec le serveur ARP, lequel, à son tour, envoie les demandes InARP aux stations IP. Sur la base de la réponse InARP, un serveur ARP configure IP en mappes d'adresses ATM. Les stations IP envoient les paquets ARP au serveur ARP pour convertir les adresses, lequel renvoie les adresses ATM.
Les stations IP configurent ensuite un SVC vers la station de destination, et le transfert des données démarre. Les entrées ARP dans les stations IP et le serveur ARP sont fondées sur un mécanisme bien défini. Pour l'environnement PVC comme pour l'environnement SVC, chaque station IP dispose d'au moins un circuit virtuel par adresse de destination.

La norme Internet Engineering Task Force RFC2225 remplace la norme RFC1577 et s'attache principalement au support de la liste des adresses de requêtes ATM ARP. Cette liste contient une ou plusieurs adresses ATM de serveurs ATM ARP situés au sein du LIS. Le client RFC2225 supprime le point d'échec individuel associé aux services ATM ARP du client 1577. Les clients 2225 ont la possibilité de commuter sur les serveurs ARP de secours en cas d'échec du serveur actuel ATM ARP.

ESCALA définit la première entrée de la liste d'adresses des requêtes ATM ARP comme le serveur ATM ARP principal, les autres étant définies comme des serveurs ATM ARP secondaires.

Le client essaie toujours d'utiliser le serveur ATM ARP principal. En cas d'échec de connexion à ce serveur, le client essaie de se connecter au premier serveur secondaire (la position dans la liste d'adresses des requêtes ATM ARP détermine l'ordre de celui–ci). En cas d'échec de la connexion au premier serveur ATM ARP secondaire, le client essaie de se connecter au serveur ATM ARP secondaire suivant de la liste, et ainsi de suite. Ce processus continue jusqu'à ce que la connexion aboutisse.

En cas d'échec de connexion au serveur ATM ARP principal, indépendamment du serveur ATM ARP secondaire auquel il est connecté ou tente de se connecter, le client fait, toutes les 15 minutes, une nouvelle tentative de connexion au serveur principal ATM ARP. En cas de réussite, la connexion au serveur ATM ARP secondaire est abandonnée.

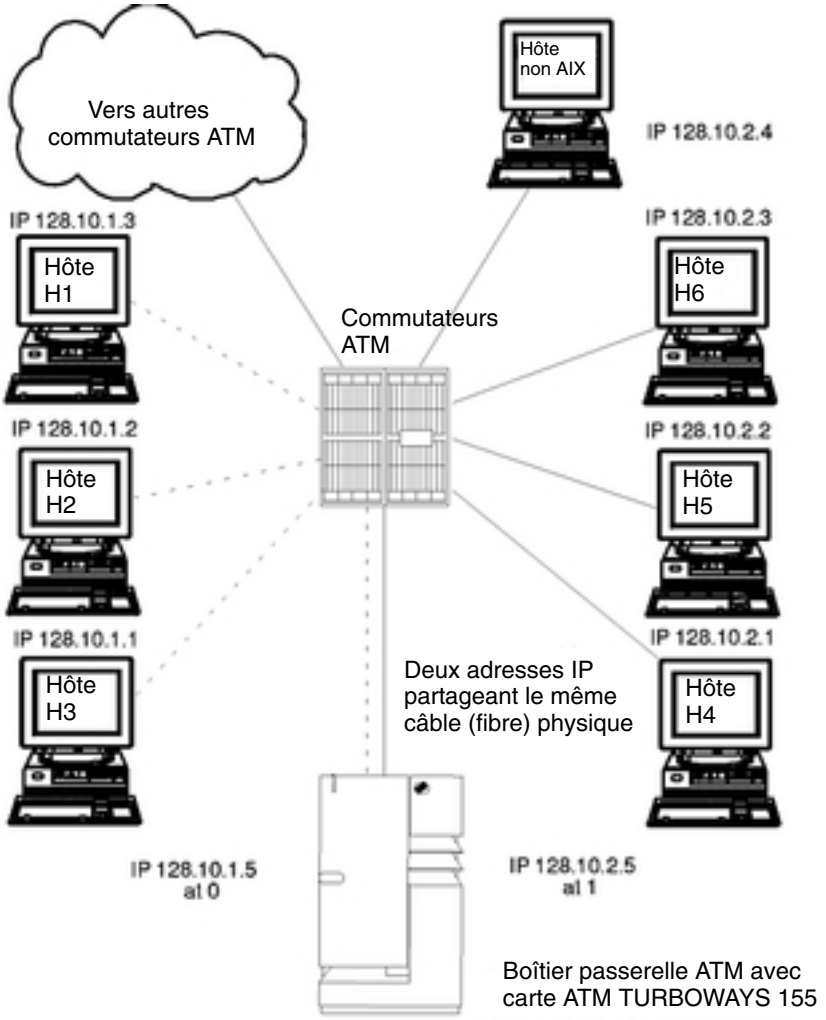
La liste d'adresses des requêtes ATM ARP est saisie manuellement à l'aide du menu SMIT ou de la commande **ifconfig**. Cette liste ne peut pas être configurée avec la MIB (Management Information Base).

Réseau PVC

Utilisez la Figure 14 pour configurer votre réseau.

Dans la figure Réseau ATM type, un sous–réseau IP logique est représenté par des lignes de pointillés, reliant chaque hôte au commutateur. L'autre sous-réseau IP est représenté par des traits pleins.

Figure 14. Réseau ATM représentatif Cette illustration représente un réseau ATM typique en étoile. Au centre de l'étoile se trouve le commutateur ATM. Des hôtes IP numérotés partent du commutateur ainsi que des liaisons vers d'autres commutateurs ATM et un boîtier de passerelle et un adaptateur ATM.



Le tableau suivant indique comment configurer les hôtes H3 et H4 pour qu'ils puissent communiquer avec une passerelle et avec chaque hôte sur leur propre réseau IP logique.

Configuration type d'un hôte		
Pilote d'interface réseau	**VPI:VCI**	**Observations**
Hôte H3		
at0	0:40	Connexion à 128.10.1.5 (passerelle)
at0	0:42	Connexion à 128.10.1.2
at0	0:43	Connexion à 128.10.1.3
Hôte H4		
at0	0:50	Connexion à 128.10.2.5 (passerelle)
at0	0:52	Connexion à 128.10.2.2
at0	0:53	Connexion à 128.10.2.3
at0	0:54	Connexion à 128.10.2.4

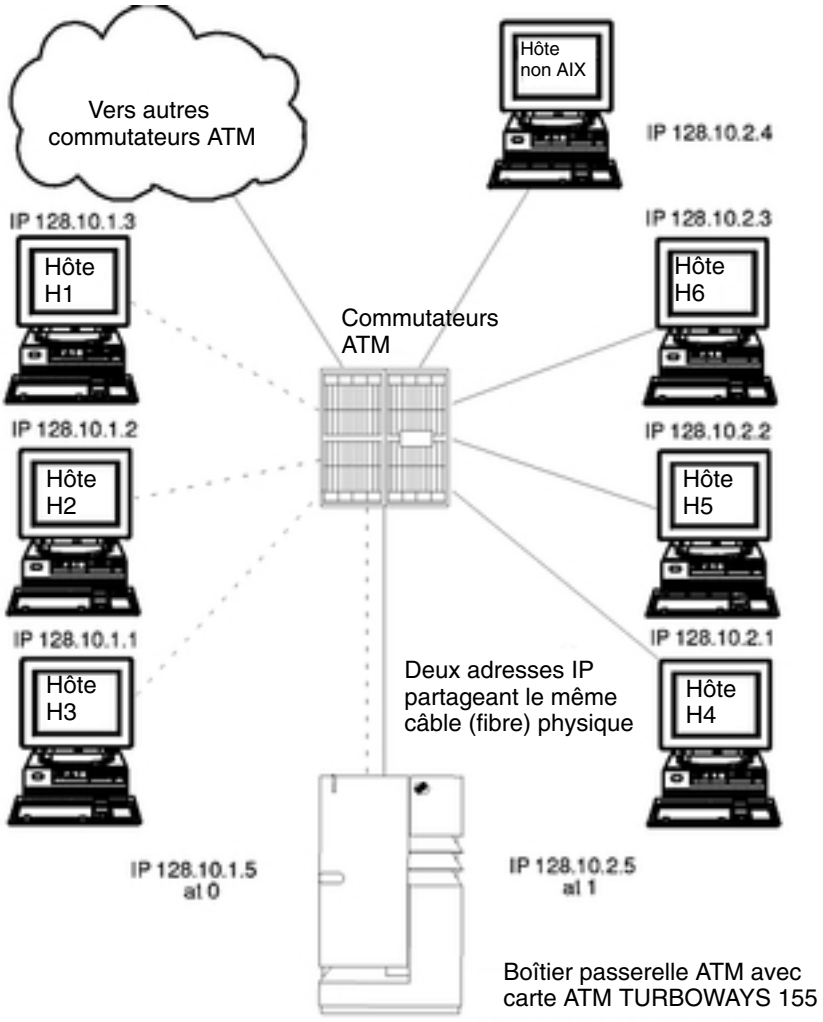
Pour atteindre les hôtes d'un autre sous–réseau IP logique, il suffit de créer une connexion VPI:VCI à la passerelle (les VPI:VCI indiqués sont de simples exemples).

Le boîtier de la passerelle ATM est équipé d'un ATM avec deux adresses IP partageant le même câble physique.

Réseau SVC

En vous aidant de la Figure 15, imaginez que l'hôte H3 veut appeler H4. H1 est le serveur ARP du sous–réseau 1 et H6, celui du sous–réseau 2. En supposant que le masque de sous–réseau est 255.255.255.0, les stations ayant les adresses 128.10.1.X sont membres d'un sous–réseau, tandis que les stations ayant les adresses 128.10.2.X sont membres d'un autre sous–réseau. Reportez–vous à la liste des configurations hôte représentatives à l'aide des SVC.

Figure 15. Réseau ATM représentatif Cette illustration représente un réseau ATM typique en étoile. Au centre de l'étoile se trouve le commutateur ATM. Des hôtes IP numérotés partent du commutateur ainsi que des liaisons vers d'autres commutateurs ATM et un boîtier de passerelle et un adaptateur ATM.



Liste de configurations type d'hôte				
Pilote d'interface réseau	**Adresse IP**	**Serveur ARP**	**Adresse du serveur ARP**	**Adresse passerelle**
Hôte H1				
at0	128.10.1.3	Oui		128.10.1.5
Hôte H3				
at0	128.10.1.1	Non	Adresse ATM de H1	128.10.1.5
Passerelle				
at0	128.10.1.5	Non	Adresse ATM de H1	
at1	128.10.2.5	Non	Adresse ATM de H6	
Hôte H4				
at0	128.10.2.1	Non	Adresse ATM de H6	128.10.2.5
Hôte H6				
at0	128.10.2.3	Oui		128.10.2.5

Remarque : Chaque sous–réseau requiert un et un seul serveur ARP.

H3 identifiant que l'adresse 128.10.2.1 ne se trouve pas sur son sous-réseau, consulte H1 pour convertir l'adresse IP de la passerelle par défaut en adresse ATM. H3 lance ensuite un appel à la passerelle. La passerelle identifie que les données sont associées au second sous-réseau et consulte H6 pour convertir effectivement l'adresse IP de H4 en adresse ATM. Des connexions sont ensuite établies entre H3 et la passerelle, et entre la passerelle et H4.

Configuration d'une carte ATM

Pour configurer la carte ATM, utilisez Web-based System Manager, la commande **wsm** ou le raccourci SMIT **smit chg_atm**. Sélectionnez un nom de carte, puis avec l'aide en ligne et les listes à choix multiples, décidez des modifications à apporter à votre configuration.

Statistiques sur la carte ATM

La commande **atmstat** permet d'obtenir des statistiques sur la carte ATM. Assortie de l'indicateur **–r**, elle remet les statistiques à zéro. Son format est **atmstat** *NomUnité*. Elle renvoie les ensembles de statistiques suivants :

Statistiques de transmission

`Packets :` Nombre de paquets (ou de PDU) transmis.

`Bytes :` Décompte des octets transmis. Ils représentent les octets de l'utilisateur. La charge ATM (par exemple, en-tête de cellule ATM, en-queue AAL5 PDU, etc.) est exclue.

`Interrupts :`
Champ non utilisé.

`Transmit Errors :`
Nombre d'erreurs de transmission pour l'unité.

`Packets Dropped :`
Nombre de paquets de transmission abandonnés, suite, par exemple, à un incident sur le tampon.

`Max Packets on S/W Transmit Queue :`
Champ non applicable à ATM.

`S/W Transmit Queue Overflow :`
Champ non applicable à ATM.

`Current S/W + H/W Transmit Queue Length :`
Longueur de la file d'attente de transmission courante.

`Cells Transmitted :`
Nombre de cellules transmises par cette unité.

`Out of Xmit Buffers :`
Nombre de paquets de transmission abandonnés, suite à un incident sur les tampons Xmit.

`Current HW Transmit Queue Length :`
Nombre courant de paquets de transmission sur la file d'attente matérielle.

`Current SW Transmit Queue Length :`
Champ non applicable à ATM.

Statistiques de réception

`Packets :` Nombre de paquets (ou de PDU) reçus.

`Bytes :` Décompte des octets reçus. Ils représentent les octets de l'utilisateur. La charge ATM (par exemple, en-tête de cellule ATM, en-queue AAL5 PDU, etc.) est exclue.

`Interrupts :`
Nombre d'interruptions effectuées par le système pour les indications carte-vers-système. Parmi les événements susceptibles de provoquer ces interruptions, citons des paquets reçus, des indications de transmission effectuée, etc.

`Receive Errors :`
Nombre d'erreurs de réception pour cette unité.

`Packets Dropped :`
Nombre de paquets de réception abandonnés, suite par exemple à un incident sur les tampons.

`Bad Packets :`
Champ non applicable à ATM.

`Cells Received :`
Nombre de cellules reçues par cette unité.

`Out of Rcv Buffers :`
Nombre de paquets abandonnés, suite à un incident sur les tampons de réception.

`CRC Errors :`
Nombre de paquets reçus ayant rencontré des erreurs CRC.

`Packets Too Long :`
Nombre de paquets reçus, qui excédaient la taille maximale du PDU.

`Incomplete Packets :`
Nombre de paquets incomplets reçus.

`Cells Dropped :`
Nombre de cellules abandonnées. Les raisons de l'abandon des cellules sont diverses ; incident au niveau de l'en–tête (HEC), tampon saturé, etc.

Statistiques générales

`No mbuf Errors :`
Nombre de requêtes mbuf refusées.

`Adapter Loss of Signals :`
Nombre de pertes de signal rencontrées par la carte.

`Adapter Reset Count :`
Nombre de réinitialisations effectuées sur la carte.

`Driver Flags: Up Running Simplex :`
Indicateurs NDD.

`Virtual Connections in use :`
Nombre de VC actuellement alloués ou en cours d'utilisation.

`Max Virtual Connections in use :`
Nombre maximal de VC alloués depuis la dernière remise à zéro des statistiques.

`Virtual Connections Overflow :`
Nombre de demandes d'allocation de VC refusées.

`SVC UNI Version :`
Version UNI courante du protocole de signalisation utilisé.

Statistiques ATM Micro Channel complémentaires

Pour des statistiques détaillées, lancez la commande **atmstat** assortie de l'indicateur **–d.**

Turboways ATM Adapter Specific Statistics:

`Packets Dropped - No small DMA buffer :`
Nombre de paquets de réception abandonnés suite à l'absence de petits tampons système pour DMA sur la carte.

`Packets Dropped - No medium DMA buffer :`
Nombre de paquets de réception abandonnés suite à l'absence de tampons système moyens pour DMA sur la carte.

`Packets Dropped – No large DMA buffer :`
Nombre de paquets de réception abandonnés suite à l'absence de grands tampons système pour DMA sur la carte.

`Receive Aborted – No Adapter Receive buffer :`
Nombre de paquets de réception abandonnés suite à l'absence de tampons de réception sur la carte.

`Transmit Aborted – No small DMA buffer :`
Nombre de paquets de transmission abandonnés suite à l'absence de petits tampons système pour DMA.

`Transmit Aborted – No medium DMA buffer :`
Nombre de paquets de transmission abandonnés suite à l'absence de tampons système moyens pour DMA.

`Transmit Aborted – No large DMA buffer :`
Nombre de paquets de transmission abandonnés suite à l'absence de grands tampons système pour DMA.

`Transmit Aborted – No MTB DMA buffer :`
Nombre de paquets de transmission abandonnés suite à l'absence de grands tampons système pour DMA.

`Transmit Aborted – No Adapter Transmit buffer :`
Nombre de paquets de transmission abandonnés suite à l'absence de tampons de transmission sur la carte.

`Max Hardware Transmit Queue Length :`
Nombre maximal de paquets de transmission en attente dans la file matérielle.

`Small Mbufs in Use :`
Nombre de petits tampons en cours d'utilisation. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Medium Mbufs in Use :`
Nombre de tampons moyens en cours d'utilisation. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Large Mbufs in Use :`
Nombre de grands tampons en cours d'utilisation. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Huge Mbufs in Use :`
Nombre de très grands tampons en cours d'utilisation. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`MTB Mbufs in Use :`
Nombre de tampons MTB en cours d'utilisation. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Max Small Mbufs in Use :`
Nombre maximal de petits tampons qui ont été utilisés. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Max Medium Mbufs in Use :`
Nombre maximal de tampons moyens qui ont été utilisés. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Max Large Mbufs in Use :`
Nombre maximal de grands tampons qui ont été utilisés. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Max Huge Mbufs in Use :`
Nombre maximal de très grands tampons qui ont été utilisés. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`MTB Mbufs in Use :`
Nombre maximal de tampons MTB qui ont été utilisés. Le pilote d'unité de carte alloue ces tampons en fonction des données de configuration fournies par les administrateurs système. Cette information peut servir à affiner les données de configuration.

`Small Mbufs overflow :`
Nombre de fois qu'un petit tampon n'a pu être alloué. Cette information peut servir à affiner les données de configuration.

`Medium Mbufs overflow :`
Nombre de fois qu'un moyen tampon n'a pu être alloué. Cette information peut servir à affiner les données de configuration.

`Large Mbufs overflow :`
Nombre de fois qu'un grand tampon n'a pu être alloué. Cette information peut servir à affiner les données de configuration.

`Huge Mbufs overflow :`
Nombre de fois qu'un très grand tampon n'a pu être alloué. Cette information peut servir à affiner les données de configuration.

`MTB Mbufs overflow :`
Nombre de fois qu'un tampon MTB n'a pu être alloué. Cette information peut servir à affiner les données de configuration.

Statistiques propres à la carte ATM PCI

`Total 4K byte Receive Buffers : 768 Using : 512`
Nombre de tampons de réception alloués ainsi que le nombre de tampons actuellement en cours d'utilisation.

Max 4K byte Receive Buffers limit : 1228 max_used : 514
Nombre maximum de tampons de réception pouvant âtre alloués ainsi que le nombre de tampons qui ont été utilisés depuis la dernière configuration ou ouverture de la carte.

Interfaces de réseau TCP/IP

La couche interface de réseau TCP/IP convertit les datagrammes IP de la couche réseau en paquets interprétables et transmissibles par les technologies de réseau. Une interface de réseau est un logiciel spécifique d'un réseau qui permet la communication entre le pilote d'unité du réseau et la couche IP. Ainsi, la couche IP dispose d'une interface fiable pour communiquer avec toutes les cartes réseau en place.

La couche IP sélectionne l'interface de réseau correspondant à l'adresse de destination du paquet à transmettre. Chaque interface est dotée d'une adresse. La couche interface de réseau est chargée d'ajouter ou de supprimer l'en-tête appliqué par la couche liaison pour assurer la livraison du message. Le pilote de **carte réseau** contrôle la carte réseau.

Une interface de réseau est généralement associée à une carte réseau, mais ce n'est pas obligatoire (l'interface de bouclage (loopback), par exemple, ne l'est pas). Chaque machine doit être équipée d'autant de cartes que de réseaux (et non de types de réseau) auxquels elle est connectée. Cependant, la machine requiert seulement une copie du logiciel d'interface de réseau et une copie du pilote d'unité de réseau. Par exemple, si un hôte est raccordé à deux réseaux en anneau à jeton, il doit être équipé de deux cartes réseau. Cependant, il requiert seulement une copie du logiciel d'interface de réseau et une copie du pilote de réseaux en anneau à jeton.

TCP/IP accepte plusieurs types d'interface de réseau :

- Ethernet standard version 2 (en)
- IEEE 802.3 (et)
- Anneau à jeton (tr)
- SLIP (sl)
- Bouclage (lo)
- FDDI
- Optique série (so)
- ATM (at)
- Protocole point à point (ppp)
- Adresse IP virtuelle (vi)

Les interfaces Ethernet, 802.3 et anneau à jeton sont destinées aux réseaux locaux (LAN) et l'interface SLIP (Serial Line Internet Protocol) aux connexions série. L'interface de bouclage (loopback) est utilisée par les hôtes pour que les messages qu'ils envoient leur soient réexpédiés. L'interface Optique série s'applique aux réseaux optiques point à point exploitant le gestionnaire d'unité de liaison optique série. L'interface ATM est utilisée pour les connexions ATM 100 Mbits/sec et 155 Mbits/sec. Le protocole PPP (Point to Point protocol) est généralement utilisé lors de la connexion à un autre ordinateur ou réseau via un modem. L'interface d'adresse IP virtuelle (également connue sous le nom d'*interface virtuelle*) n'est pas associée à une carte réseau donnée. Plusieurs instances d'une interface virtuelle peuvent être configurées sur un hôte. Lorsque des interfaces virtuelles sont configurées, l'adresse de la première d'entre elles devient l'adresse source sauf si une application a choisi une autre interface. Les processus qui utilisent une adresse IP virtuelle comme adresse source peuvent envoyer des paquets sur toute interface de réseau qui fournit la meilleure route vers cette destination. Les paquets entrants destinés à une adresse IP virtuelle sont livrés au processus quelle que soit l'interface via laquelle ils arrivent.

Configuration automatique des interfaces de réseau

A l'installation d'une nouvelle carte réseau (physique), le système d'exploitation ajoute automatiquement l'interface correspondante. Par exemple, si vous installez une carte réseau en anneau à jeton, le système la nomme `tok0` et ajoute l'interface de réseau en anneau à jeton `tr0`. De même, si vous installez une carte Ethernet, le système la nomme `ent0` et ajoute une interface Ethernet version 2 et une interface IEEE 802.3 (respectivement nommées `en0` et `et0`).

Dans la plupart des cas, il existe une correspondance unique entre un nom de carte et un nom d'interface de réseau. Par exemple, la carte réseau en anneau à jeton `tok0` correspond à l'interface `tr0`, la carte tok1, à l'interface `tr1`, etc. De même, la carte Ethernet `ent0` correspond aux interfaces `en0` (Ethernet version 2) et `et0` (IEEE 802.3), la carte `ent1`, aux interfaces `en1` (Ethernet version 2) et `et1` (IEEE 802.3).

Conformément à RFC1577, une station ATM peut faire partie de plusieurs sous-réseaux IP logiques. Dans ce cas, plusieurs interfaces sont associées à une unité, ce qui suppose d'ajouter une interface spécifique et de lui affecter un nom d'unité.

Remarque : En circonstances normales d'exploitation, vous n'aurez jamais à supprimer ou ajouter manuellement une interface de réseau. Mais vous pouvez être amené à le faire au cours d'une procédure de résolution d'incident. Dans ce cas, utilisez **wsm** (Web-based System Manager) ou le raccourci SMIT **smit inet** pour supprimer et réajouter l'interface appropriée.

A chaque lancement du système, l'interface de réseau est automatiquement configurée en fonction des informations de la base de données ODM, avec des valeurs par défaut. La communication n'est possible que si une adresse Internet lui a été attribuée. C'est le seul attribut que vous ayez à définir. Les autres attributs peuvent conserver leur valeur par défaut. Le détail de ces valeurs est donné dans les paragraphes qui suivent.

Configuration Ethernet par défaut

Voici les attributs de carte réseau Ethernet et leurs valeurs par défaut qui peuvent être modifiées dans le menu SMIT Sélection d'une interface de réseau ou avec **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
netaddr		
state	down	up, down, detach
arp	yes	yes, no
netmask		
broadcast		

Voici l'attribut de pilote d'unité réseau Ethernet et sa valeur par défaut qui peut être modifiée dans le menu SMIT Sélection d'une interface de réseau ou avec **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
mtu	1500	60 à 1500

Configuration 802.3 par défaut

Voici les attributs de carte réseau 802.3 et leurs valeurs par défaut qui peuvent être modifiées dans le menu SMIT Sélection d'une interface de réseau ou avec **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
netaddr		
state	down	up, down, detach
arp	yes	yes, no
netmask		
broadcast		

Voici l'attribut de pilote d'unité réseau 802.3 et sa valeur par défaut qui peut être modifiée dans le menu SMIT Sélection d'une interface de réseau ou avec **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
mtu	1492	60 à 1492

Valeurs de configuration par défaut de l'anneau à jeton

Voici les attributs de carte réseau en anneau à jeton et leurs valeurs par défaut qui peuvent être modifiées dans le menu SMIT Sélection d'une interface de réseau ou avec **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
netaddr		
netmask		
state	down	up, down, detach
arp	yes	yes, no
hwloop	no	yes, no
netmask		
broadcast		
allcast	no	yes, no

Voici les attributs de pilote d'unité réseau en anneau à jeton et leurs valeurs par défaut qui peuvent être modifiées dans le menu SMIT Sélection d'une interface de réseau ou avec **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
mtu (4Mbps)	1500	60 à 4056
mtu (16Mbps)	1500	60 à 17960

Remarque : Lorsque la communication transite par un pont, la valeur MTU par défaut (de 1500 octets) doit être ramenée à 8 octets en dessous de la valeur maximum I-frame déclarée par le pont dans le champ de contrôle de routage. Par exemple, si la valeur de "maximum I-frame" est 1500 dans le champ de contrôle de routage, celle de MTU doit être fixée à 1492 (pour les interfaces anneau à jeton seulement). Pour en savoir plus, reportez-vous à Incidents sur un pont reliant deux réseaux en anneau à jeton, page 4-234.

Avec la carte en anneau à jeton IBM 16/4 PowerPC (ISA), le mtu est limité à 2000.

Configuration SLIP par défaut

Voici les attributs de carte réseau SLIP et leurs valeurs par défaut telles qu'elles s'affichent dans le menu SMIT Sélection d'une interface de réseau ou dans **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
netaddr		
dest		
state	up	up, down, detach
netmask		

Voici l'attribut de pilote d'unité réseau SLIP et sa valeur par défaut telle qu'elle s'affiche dans le menu SMIT Sélection d'une interface de réseau ou dans **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
mtu	1006	60 à 4096

Configuration optique série par défaut

Voici les attributs du convertisseur de canal réseau optique série et leurs valeurs par défaut telles qu'elles s'affichent dans le menu SMIT Sélection d'une interface de réseau ou dans **wsm** Web-based System Manager.

Attribut	Valeur par défaut	Valeurs possibles
netaddr		
state	down	up, down, detach
netmask		

Voici l'attribut du gestionnaire d'unité réseau optique et sa valeur par défaut telle qu'elle s'affiche dans le menu SMIT Sélection d'une interface de réseau ou dans **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
mtu	61428	1 à 61428

Configuration ATM par défaut

Voici les attributs de carte réseau ATM et leurs valeurs par défaut telles qu'elles s'affichent dans le menu SMIT Sélection d'une interface de réseau ou dans **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
netaddr		
netmask		
state	up	up, down, detach
Connection Type	svc_s	svc_c, svc_s, pvc
ATM Server Address		
Alternate Device		
idle timer	60	1 à 60
Best Effort Bit Rate (UBR) en kbits/sec	0	1 à 155.000

Voici l'attribut de pilote d'unité réseau ATM et sa valeur par défaut telle qu'elle s'affiche dans le menu SMIT Sélection d'une interface de réseau ou dans **wsm** (Web-based System Manager).

Attribut	Valeur par défaut	Valeurs possibles
mtu	9180	1 à 64K

Remarque : La plus grande prudence est recommandée aux administrateurs réseau s'ils modifient la taille de MTU définie par défaut. La valeur de ce paramètre doit être compatible avec les autres stations du réseau.

Si des PVC sont utilisés sur une interface, les VPI:VCI doivent être définis via la dernière option du menu Sélection d'une interface de réseau, PVCs for IP over ATM Network, qui vous permet de répertorier, d'ajouter, de modifier ou de supprimer des PVC.

Réseaux avec plusieurs interfaces

Si plusieurs interfaces réseau sont connectées à un seul réseau, chaque interface doit avoir une adresse IP unique.

Avant AIX 5.1, si vous configuriez plusieurs interfaces réseau sur le même réseau, seule la première interface configurée avait un routage vers le réseau dans la table de routage IP. La totalité du trafic IP sortant passerait par conséquent uniquement par cette interface, et pas les autres interfaces du réseau. Bien qu'il soit possible d'utiliser cette configuration pour équilibrer le trafic entrant, il est déconseillé de l'utiliser dans les versions antérieures à AIX 5.1.

Dans AIX 5.1 et les versions supérieures, la fonction de Routage multi–chemins permet d'ajouter des routes à la table de routage IP pour les interfaces multi–chemins sur le même sous–réseau. Ceci permet au trafic sortant d'alterner entre les interfaces au lieu d'être envoyées via une seule interface.

Gestion d'interfaces de réseau

Pour gérer des interfaces de réseau, utilisez le gestionnaire système Web, WSM Network, l'application FastPath ou les procédures du tableau suivant.

Gestion des tâches d'interfaces de réseau			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Liste de toutes les unités de réseau	**smit lsinet**	**lsdev –C –c if**	Logiciel —> **Unités** —> **Toutes les unités**.
Configuration d'une unité de réseau	*smit chinet*	Reportez-vous à la commande **ifconfig** et au fichier **rc.net**	Logiciel —> **Réseau** —> **TCPIP (IPv4 et IPv6)** —> **Configuration de protocole** —> **Procédez à la configuration TCP/IP de base**.

Modification des informations d'interface réseau avec **/usr** monté à distance	**smit chdev**[1,2]	**chgif**[1,2]	Logiciel —> **Réseau** —> **TCPIP (IPv4 et IPv6)** —> **Interfaces de réseau** —>. Cliquez avec le bouton droit et sélectionnez **Propriétés** —> **Alias**.
Statistiques sur une interface de réseau		**netstat –v**	Réseau —> **Réseau** —> **TCPIP (IPv4 et IPv6)** —> **Interfaces de réseau** —> **Statistiques sur le réseau**.

Remarques :

1. Les modifications apportées depuis un **/usr** monté à distance n'affectent que l'ODM tant que le réseau n'est pas réinitialisé ou tant que la commande **ifconfig** n'a pas été utilisée pour valider les modifications.
2. Avec **/usr** monté à distance, l'administrateur système doit veiller à ne pas changer l'interface car elle correspond à l'emplacement des bibliothèques, des commandes et du noyau.

Options du réseau spécifiques à l'interface

Les interfaces TCP/IP doivent être spécialement définies pour atteindre une bonne performance réseau à haut débit (au moins 100 Mo). Le fait que plusieurs interfaces de réseau et une combinaison d'interfaces TCP/IP traditionnelles et à haut débit puissent être utilisées sur un seul système complique cet effort. Avant AIX 4.3.3 (4330–08) et AIX 5.1, AIX fournissait un seul ensemble de valeurs au niveau des systèmes pour les paramètres de réglage de réseau d'interface IP principaux, ce qui rendait impossible le réglage d'un système ayant des interfaces de cartes réseau très différentes. Depuis AIX 4.3.3 (4330–08) et AIX 5.1, Interface Specific Network Options (ISNO) permet aux administrateurs système de régler chaque interface TCP/IP pour obtenir la meilleure performance.

Il existe cinq paramètres ISNO par interface prise en charge : **rfc1323**, **tcp_nodelay**, **tcp_sendspace**, **tcp_recvspace** et **tcp_mssdflt**. Lorsqu'elles sont définies, les valeurs de ces paramètres remplacent les paramètres de mêmes noms définis avec la commande **no** au niveau de l'ensemble du système. Lorsque les options ISNO ne sont pas définies pour une interface particulière, les options au niveau de l'ensemble du système sont utilisées. Lorsque des options ont été définies par une application pour un socket donné utilisant la sous–routine **setsockopt**, de telles options remplacent les ISNO.

L'option de réseau **use_isno**, définie avec la commande **no**, doit être égale à 1 pour que les ISNO soient pris en compte. La valeur par défaut de **use_isno** est 1.

Les paramètres ISNO de certaines cartes à haut débit sont définis par défaut dans la base de données d'ODM.

Les interfaces Gigabit Ethernet, lorsqu'elles sont configurées pour utiliser un MTU de 9000, utilisent les valeurs ISNO suivantes par défaut :

Nom	**Valeur pour AIX 4.3.3**	**Valeur pour AIX 4.3.3 (4330–08)**	**Valeur pour AIX 5.1 (ou version supérieure)**
tcp_sendspace	131072	262144	262144
tcp_recvspace	92160	131072	131072
rfc1323	1	1	1

Les interfaces Gigabit Ethernet, lorsqu'elles sont configurées pour utiliser un MTU de 1500, utilisent les valeurs ISNO suivantes par défaut :

Nom	Valeur pour AIX 4.3.3	Valeur pour AIX 4.3.3 (4330–08)	Valeur pour AIX 5.1 (ou version supérieure)
tcp_sendspace	65536	131072	131072
tcp_recvspace	16384	65536	65536
rfc1323	0	non définie	non définie

Les interfaces ATM, lorsqu'elles sont configurées pour utiliser un MTU de 1500, utilisent les valeurs ISNO suivantes par défaut :

Nom	Valeur pour AIX 4.3.3	Valeur pour AIX 4.3.3 (4330–08)	Valeur pour AIX 5.1 (ou version supérieure)
tcp_sendspace	16384	non définie	non définie
tcp_recvspace	16384	non définie	non définie
rfc1323	0	non définie	non définie
tcp_nodelay	0	non définie	non définie
tcp_mssdflt	512	non définie	non définie

Les interfaces ATM, lorsqu'elles sont configurées pour utiliser un MTU de 65527, utilisent les valeurs ISNO suivantes par défaut :

Nom	Valeur pour AIX 4.3.3	Valeur pour AIX 4.3.3 (4330–08)	Valeur pour AIX 5.1 (ou version supérieure)
tcp_sendspace	655360	655360	655360
tcp_recvspace	655360	655360	655360
rfc1323	0	1	1
tcp_nodelay	0	non définie	non définie
tcp_mssdflt	512	non définie	non définie

Les interfaces ATM, lorsqu'elles sont configurées pour utiliser un MTU de 9180, utilisent les valeurs ISNO suivantes par défaut :

Nom	Valeur pour AIX 4.3.3	Valeur pour AIX 4.3.3 (4330–08)	Valeur pour AIX 5.1 (ou version supérieure)
tcp_sendspace	65536	65536	65536
tcp_recvspace	65536	65536	65536
rfc1323	0	non définie	non définie
tcp_nodelay	0	non définie	non définie
tcp_mssdflt	512	non définie	non définie

Les interfaces FDDI, lorsqu'elles sont configurées pour utiliser un MTU de 4352, utilisent les valeurs ISNO suivantes par défaut :

Nom	Valeur
tcp_sendspace	45046
tcp_recvspace	45046

Les paramètres ISNO ne peuvent pas être affichés ou modifiés à l'aide de SMIT. Ils peuvent être définis à l'aide des commandes **chdev** ou **ifconfig**. La commande **ifconfig** ne modifie les valeurs que jusqu'au prochain redémarrage du système. La commande **chdev** modifie les valeurs dans la base de données d'ODM afin qu'elles soient utilisées lors de redémarrages ultérieurs du système. Les commandes **lsattr** ou **ifconfig** peuvent être utilisées pour afficher les valeurs actuelles.

Exemple

Les commandes suivantes peuvent être d'abord utilisées pour vérifier la prise en charge du système et de l'interface, puis pour définir et vérifier les nouvelles valeurs.

1. Vérifiez la prise en charge du système général et de l'interface en utilisant les commandes **no** et **lsattr**.
   - Vérifiez que l'option **use_isno** est activée en utilisant une commande semblable à la suivante :

     ```
     $ no -a | grep isno
         use_isno=1
     ```

   - Vérifiez que l'interface prend en charge les cinq nouveaux ISNO utilisant la commande **lsattr –El**, comme illustré ci–dessous :

     ```
     $ lsattr -E -l en0 -H
         attribute   value  description
         rfc1323                     N/A
         tcp_nodelay                 N/A
         tcp_sendspace               N/A
         tcp_recvspace               N/A
         tcp_mssdflt                 N/A
     ```

2. Définissez les valeurs spécifiques à l'interface en utilisant les commandes **ifconfig** ou **chdev**. La commande **ifconfig** définit temporairement des valeurs, ce qui est recommandé pour effectuer des tests. La commande **chdev** modifie l'ODM, les valeurs personnalisées conservent donc leur validité après un redémarrage du système.
   - Définissez **tcp_recvspace** et **tcp_sendspace** à 64 K et activez **tcp_nodelay** en utilisant l'une des solutions suivantes :

     ```
     $ ifconfig en0 tcp_recvspace 65536 tcp_sendspace 65536 tcp_nodelay 1
     $ chdev -l en0 -a tcp_recvspace=65536 -a tcp_sendspace=65536 -a
      tcp_nodelay=1
     ```

   - En supposant également que la commande **no** donne une valeur globale **rfc1323=1**, l'utilisateur racine peut désactiver **rfc1323** pour toutes les connexions sur en0 avec les commandes suivantes :

     ```
     $ ifconfig en0 rfc1323 0
     $ chdev -l en0 -a rfc1323=0
     ```

3. Vérifiez les paramètres à l'aide des commandes **ifconfig** ou **lsattr**, comme illustré dans l'exemple ci–dessous :

   ```
   $ ifconfig en0 <UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST
   ,GROUPRT,64BIT>
        en0: flags=e080863
         inet 9.19.161.100 netmask 0xffffff00 broadcast 9.19.161.255
         tcp_sendspace 65536 tcp_recvspace 65536 tcp_nodelay 1 rfc1323 0
    $ lsattr -El en0
        rfc1323         0           N/A               True
        tcp_nodelay     1           N/A               True
        tcp_sendspace   65536       N/A               True
        tcp_recvspace   65536       N/A               True
        tcp_mssdflt                 N/A               True
   ```

Adressage TCP/IP

TCP/IP contient un schéma d'adressage Internet qui permet aux utilisateurs et aux applications d'obtenir l'identité d'un réseau ou d'un hôte pour établir une communication. Une adresse Internet fonctionne sur le même principe qu'une adresse postale : elle permet aux données d'être acheminées à destination. TCIP/IP intègre des normes d'adressage de réseaux, sous-réseaux, hôtes, sockets, et des normes d'utilisation des adresses de diffusion et de bouclage.

Une adresse Internet est constituée d'une adresse réseau et d'une adresse d'hôte (locale). Ce format permet de spécifier dans la même adresse le réseau et l'hôte cible. Une adresse officielle unique est attribuée à chaque réseau qui se connecte à d'autres réseaux Internet. Pour les réseaux non connectés à d'autres réseaux Internet, l'adresse peut être déterminée selon la convenance locale.

Le schéma d'adressage Internet propose des adresses IP (Internet Protocol) et deux cas particuliers d'adresse IP : adresses de diffusion et adresses de bouclage.

Adresses Internet

Le protocole IP (Internet Protocol) utilise une zone d'adresse de 32 bits formée de deux parties. Les 32 bits sont répartis en groupes de quatre *octets* comme suit :

01111101 00001101 01001001 00001111

Ces nombres binaires correspondent à :

125 13 73 15

Les deux parties de l'adresse Internet sont respectivement l'adresse réseau et l'adresse hôte. Ainsi, un hôte distant peut expédier des informations en précisant le réseau distant et l'hôte destinataire sur ce réseau. Par convention, le numéro d'hôte 0 (zéro) désigne le réseau lui-même.

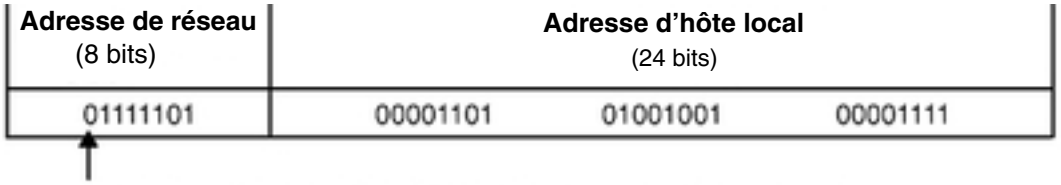
TCP/IP prend en charge trois classes d'adresses Internet : A, B et C, qui se distinguent par l'attribution des 32 bits. L'appartenance à une classe est déterminée par la taille du réseau.

Adresses de classe A

Une adresse de classe A se compose d'une adresse de réseau de 8 bits et d'une adresse hôte local de 24 bits. Le premier bit de l'adresse de réseau sert à désigner la classe du réseau et les 7 autres, l'adresse effective. Le nombre le plus élevé que peuvent représenter ces 7 bits en binaire est 128 ; la classe A offre donc 128 adresses possibles. Deux sont réservées à des cas particuliers : l'adressage de bouclage local pour l'une (code 127) et l'adressage de diffusion pour l'autre (adresse qui couvre la totalité des réseaux).

Il en résulte 126 adresses de réseau de classe A possibles et 16 777 216 adresses d'hôte local. Dans une adresse de classe A, le bit de poids fort est positionné à 0 (voir figure).

Figure 16. Adresse de classe A Cette illustration représente une structure d'adresse de classe A typique. Les 8 premiers bits contiennent l'adresse réseau (commençant toujours par un zéro). Les 24 bits restants contiennent l'adresse hôte locale.



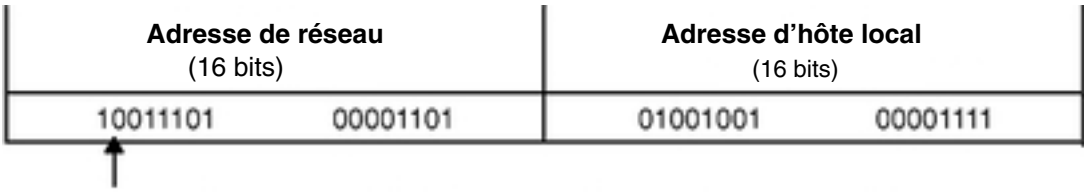
Remarque : Le bit de poids fort (le premier) est toujours positionné à 0 dans une adresse de classe A.

Autrement dit, le premier octet d'une adresse de classe A est compris entre 1 et 126.

Adresse de classe B

Une adresse de classe B se compose d'une adresse de réseau de 16 bits et d'une adresse hôte local de 16 bits. Les 2 premiers bits de l'adresse de réseau désignent la classe de réseau et les 14 autres, l'adresse effective. Par conséquent, il y a 16 384 adresses de réseau possibles et 65 536 adresses hôte local. Dans une adresse de classe B, les bits de poids fort sont positionnés à 1 et 0.

Figure 17. Adresse de classe B Cette illustration représente une structure d'adresse de classe B typique. Les 16 premiers bits contiennent l'adresse réseau. Les deux bits d'ordre supérieur sont toujours un 1 et un zéro. Les 16 bits restants contiennent l'adresse hôte locale.



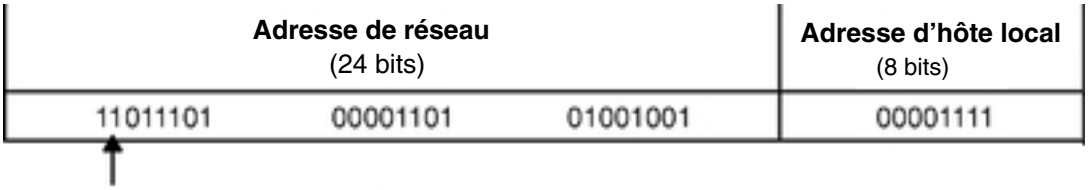
Remarque : Les 2 bits de poids fort (les deux premiers) sont toujours positionnés à 1 et 0 dans une adresse de classe B.

Autrement dit, le premier octet d'une adresse de classe B est compris entre 128 et 191.

Adresse de classe C

Une adresse de classe C se compose d'une adresse de réseau de 24 bits et d'une adresse hôte local de 8 bits. Les 2 premiers bits de l'adresse de réseau désignent la classe de réseau et les 22 autres, l'adresse effective. Par conséquent, il y a 2 097 152 adresses de réseau possibles et 256 adresses hôte local possibles. Dans une adresse de classe C, les bits de poids fort sont positionnés à 1 et 1 (voir figure).

Figure 18. Adresse de classe C Cette illustration représente une structure d'adresse de classe C typique. Les 24 premiers bits contiennent l'adresse réseau (les deux bits d'ordre supérieur sont toujours un 1 et un 1). Les 8 bits restants contiennent l'adresse hôte locale.



Remarque : Les 2 bits de poids fort (les deux premiers) sont toujours positionnés à 1 dans une adresse de classe C.

Autrement dit, le premier octet d'une adresse de classe C est compris entre 192 et 223.

Pour décider de la classe d'adresse, vous devez tenir compte du nombre d'hôtes locaux et de sous-réseaux prévus. Si l'organisation est réduite et que le réseau comporte moins de 256 hôtes, une adresse de classe C est probablement suffisante. Sinon, il faut envisager une adresse de classe A ou B.

Remarque : Les adresses de classe D (1-1-1-0 pour les bits de poids fort), prises en charge par UDP/IP sous ce système, sont utilisées comme adresses de diffusion.

Les machines lisent les adresses en code binaire. Par convention, les adresses hôtes Internet sont exprimées en *notation décimale à points* sur 32 bits répartis en quatre zones de 8 bits. Par exemple, la valeur binaire :

0001010 00000010 00000000 00110100

peut être exprimée comme suit :

010.002.000.052 ou 10.2.0.52

La valeur de chacune de ces zones, séparées par un point, est un nombre décimal.

Remarque : La commande **hostent** reconnaît les adresses suivantes : .08, .008, .09 et .009. Les adresses introduites par des zéros sont interprétées en base octale, laquelle exclut les chiffres 8 et 9.

TCP/IP requiert une adresse Internet unique pour chaque interface (carte) du réseau. Ces adresses, définies par la base de données de configuration, doivent concorder avec celles du fichier **/etc/hosts** ou, si un serveur de noms est utilisé, de la base de données **named**.

Adresses Internet avec zéros

Lorsque la zone d'adresse hôte d'une adresse Internet de classe C a la valeur 0 (par exemple 192.9.200.0), TCP/IP envoie une adresse générique sur le réseau : toutes les machines dotées de l'adresse de classe 192.9.200.*X* (où *X* représente une valeur comprise entre 0 et 254) doivent répondre à la requête. Il en résulte que le réseau est inondé de requêtes adressées à des machines inexistantes.

Le même problème se pose pour une adresse de classe B du type 129.5.0.0 : toutes les machines dotées de l'adresse de classe 129.5.*X.X*. (où *X* représente une valeur comprise entre 0 et 254) doivent répondre à la requête. Mais, dans ce cas, le nombre de requêtes est bien plus important encore que sur un réseau de classe C car les adresses de classe B couvrent des réseaux plus vastes.

Adresses de sous-réseau

Grâce au mécanisme d'adressage de sous-réseau, un système autonome regroupant plusieurs réseaux peut disposer d'une même adresse Internet. Il est également possible de diviser un réseau en plusieurs réseaux logiques (sous-réseaux). Par exemple, une organisation sera dotée d'une adresse Internet unique connue par les utilisateurs extérieurs à l'organisation mais comportera en interne plusieurs sous-réseaux de service. Quel que soit le cas de figure, l'adressage de sous-réseau réduit le nombre d'adresses Internet requises et optimise le routage local.

La zone d'adresse du protocole IP est formée de deux parties : une adresse réseau et une adresse locale. Cette dernière est constituée d'un numéro de sous–réseau et d'un numéro d'hôte, ce qui permet de définir des adresses de sous–réseau. L'identification du sous–réseau est suffisamment précise pour assurer le routage des messages de façon fiable.

Dans l'adresse Internet de classe A (voir figure), qui se compose d'une adresse de réseau de 8 bits et d'une adresse hôte local de 24 bits, l'adresse locale identifie la machine hôte spécifique sur le réseau.

Figure 19. Adresse de classe A Cette illustration représente une structure d'adresse de classe A typique. Les 8 premiers bits contiennent l'adresse réseau (commençant toujours par un zéro). Les 24 bits restants contiennent l'adresse hôte locale.

Adresse de réseau (8 bits)	**Adresse d'hôte local** (24 bits)		
01111101	00001101	01001001	00001111

Pour créer une adresse de sous-réseau pour réseau Internet de classe A, l'adresse locale est composée de deux éléments : le numéro d'identification du réseau physique (ou sous-réseau) et le numéro de l'hôte sur le sous-réseau. Les messages sont renvoyés à l'adresse de réseau indiquée et le système local se charge d'acheminer les messages vers

ses sous-réseaux et hôtes. Le partitionnement de l'adresse locale en adresses sous-réseau et hôte s'effectue en fonction du nombre de sous-réseaux et d'hôtes correspondants.

Le tableau ci–dessous décrit l'adresse locale divisée en une adresse de sous–réseau 12 bits et une adresse hôte 12 bits.

Figure 20. Adresse de classe A avec sous–réseau correspondant Adresse Cette illustration représente une structure d'adresse de classe A typique. Les 8 premiers bits contiennent l'adresse réseau (commençant toujours par un zéro). Les 24 derniers bits contiennent l'adresse hôte locale avec l'adresse de sous–réseau qui occupe les 8 premiers bits et l'adresse hôte qui occupe les 8 derniers bits.

Adresse de réseau (8 bits)	**Adresse d'hôte local** (24 bits)	
Adresse de réseau	**Adresse de sous-réseau**	**Adresse d'hôte**
01111101	00001101 0100	1001 00001111

Remarque : Le bit de poids fort (le premier) est toujours positionné à 0 dans une adresse de classe A.

Vous bénéficiez d'une grande souplesse d'adressage des sous-réseaux et hôtes. Les bits de l'adresse locale peuvent être répartis en fonction de la croissance potentielle de l'organisation et de la structure de réseau. Les règles à respecter sont les suivantes :

- `adresse_réseau` correspond à l'adresse Internet.
- `adresse_sous-réseau` est une zone de longueur constante pour un réseau donné.
- `adresse_hôte` est une zone de 1 bit minimum.

Si la longueur de la zone `adresse de sous-réseau` est 0, le réseau n'est pas organisé en sous-réseaux, et l'adressage du réseau se fait par le biais de l'adresse de réseau Internet.

Il n'est donc pas nécessaire que ces bits soient contigus dans l'adresse, bien que ce soit généralement préférable. De même, il est conseillé de positionner les bits de sous-réseau comme bits de poids fort de l'adresse locale.

Masques de sous-réseau

Lorsqu'un hôte envoie un message, le système doit déterminer si la destination du message se trouve sur le même réseau que la source ou sur un réseau directement accessible par une des interfaces locales. Pour ce faire, il compare l'adresse de destination à l'adresse hôte sur la base d'un *masque de sous-réseau*. Lorsque la destination n'est pas locale, le message transite par une passerelle. La passerelle détermine si la destination est accessible localement en procédant à la même comparaison.

Le masque de sous-réseau fournit au système le schéma de partitionnement du sous-réseau. Ce masque de bits comporte la partie adresse de réseau et la partie adresse de sous-réseau de l'adresse Internet (voir figure). Par exemple, le masque de sous-réseau de l'adresse de classe A répartie comme indiqué précédemment se présente comme suit :

Figure 21. Adresse de classe A avec sous–réseau correspondant Adresse Cette illustration représente une structure d'adresse de classe A typique. Les 8 premiers bits contiennent l'adresse réseau (commençant toujours par un zéro). Les 24 derniers bits contiennent l'adresse hôte locale avec l'adresse de sous–réseau qui occupe les 8 premiers bits et l'adresse hôte qui occupe les 8 derniers bits.

Adresse de réseau (8 bits)	Adresse d'hôte local (24 bits)	
Adresse de réseau	**Adresse de sous-réseau**	**Adresse d'hôte**
01111101	00001101 0100	1001 00001111

Adresse de classe A intégrant une adresse de sous-réseau

Adresse de réseau (8 bits)	**Adresse d'hôte local** (24 bits)	
Adresse de réseau	Adresse de sous-réseau	Adresse d'hôte
Masque de sous-réseau		**Adresse d'hôte**
01111101	00001101 0100	1001 00001111

Adresse de classe A intégrant un masque de sous-réseau

Le masque de sous-réseau est un ensemble de 4 octets, comme l'adresse interréseau. Il comporte des bits de poids fort (les 1) qui correspondent aux emplacements de bits de l'adresse de réseau et de sous-réseau, et des bits de poids faible (les 0) correspondant aux emplacements des bits de l'adresse hôte. Le masque de sous-réseau de l'adresse donnée dans la figure ci-dessus se présente comme suit :

Figure 22. Exemple de masque de sous–réseau Cette illustration représente un exemple d'une structure de masque de sous–réseau. Les 8 premiers bits contiennent l'adresse réseau. Les 24 derniers bits contiennent l'adresse hôte locale avec l'adresse de sous–réseau qui occupe les 8 premiers bits et l'adresse hôte qui occupe les 8 derniers bits.

Adresse de réseau (8 bits)	**Adresse d'hôte local** (24 bits)	
Adresse de réseau	**Adresse de sous-réseau**	**Adresse d'hôte**
11111111	11111111 1111	0000 00000000

Comparaison d'adresses

L'adresse de destination est comparée à l'adresse de réseau local en appliquant l'opérateur logique AND et l'opérateur d'exclusion OR sur le masque de sous-réseau de l'hôte source :

La procédure de comparaison se déroule comme suit :

1. Application de l'opérateur logique AND entre l'adresse de destination et le masque de l'adresse de sous-réseau local.
2. Application de l'opérateur d'exclusion OR entre le résultat de l'opération précédente et l'adresse de réseau local associée à l'interface locale.

   Si le résultat ne fournit que des zéros, la destination est supposée directement accessible via une des interfaces locales.

3. Si un système autonome est équipé de plusieurs interfaces (et donc de plusieurs adresses Internet), la comparaison est effectuée pour chaque interface locale.

Supposons, par exemple, que deux interfaces locales soient définies pour le réseau hôte T125. Leur adresse Internet et la représentation binaire de ces adresses doivent se présenter comme suit :

Adresses d'interface de réseau local

```
CLASS A   73.1.5.2    =  01001001 00000001 00000101 00000010
 CLASS B   145.21.6.3  =  10010001 00010101 00000110 00000011
```

Les masques de sous-réseau correspondants des interfaces de réseau local se présentent comme suit :

Adresses d'interface de réseau local

```
CLASS A   73.1.5.2    =  11111111 11111111 11100000 00000000
 CLASS B   145.21.6.3  =  11111111 11111111 11111111 11000000
```

Si le réseau source T125 est sollicité pour envoyer un message au réseau de destination avec 114.16.23.8 pour adresse hôte (représentée en binaire par 01110010 00010000 00010111 00001000), le système vérifie si la destination est directement accessible via une interface locale.

Remarque: Le mot clé **subnetmask** doit être défini dans la base de données de configuration de chaque hôte appelé à desservir des sous-réseaux. En effet, les sous-réseaux ne sont utilisables que s'ils sont pris en charge par chaque hôte du réseau. Vous devez donc déclarer le masque de sous-réseau comme permanent dans la base de données de configuration, via le menu SMIT Sélection d'une interface de réseau ou via l'application Web-based System Manager Network. Vous pouvez également déclarer le masque de sous–réseau dans le système d'exploitation via la commande **ifconfig**. (si vous utilisez **ifconfig**, la modification n'est pas permanente).

Adresses de diffusion

TCP/IP peut transmettre des données à tous les hôtes du réseau local ou des réseaux directement connectés. Ces transmissions sont appelées *messages de diffusion*. Par exemple, le démon de routage **routed** fait appel à ce type de message pour lancer des requêtes de routage ou y répondre.

Les données à diffuser aux hôtes des réseaux directement connectés sont transmises par les protocoles UDP (User Datagram Protocol) et IP (Internet Protocol), avec, dans l'en-tête IP, tous les bits de l'adresse de destination hôte positionnés à 1. Dans le cas de données à diffuser aux hôtes d'un réseau spécifique, tous les bits de la partie adresse locale de l'adresse IP sont positionnés à 0.

L'adresse de diffusion peut être modifiée temporairement via le paramètre *broadcast* dans la commande **ifconfig**. Modifiez-la de façon permanente avec le raccourci Web-based System Manager **wsm** ou avec le raccourci SMIT **smit chinet**. Ceci peut s'avérer utile pour la compatibilité avec des versions antérieures de logiciels qui utilisent des adresses de diffusion différentes (avec, par exemple, des ID hôte définies à 0).

Adresses de bouclage local

Le protocole IP déclare l'adresse de réseau spéciale 127.0.0.1 comme adresse de bouclage local. Les hôtes utilisent cette adresse pour s'envoyer des messages à eux-mêmes. L'adresse de bouclage local est définie par le gestionnaire de configuration lors du démarrage du système. Le bouclage local est appliqué dans le noyau et peut également être défini avec la commande **ifconfig**. Le bouclage est appelé au lancement du système.

Résolution de noms sous TCP/IP

Bien que les adresses Internet 32–bits fournissent un moyen efficace d'identifier la source et la destination des datagrammes à travers un interréseau, les utilisateurs préfèrent utiliser des noms représentatifs et faciles à mémoriser. TCP/IP propose un système d'attribution de noms applicable à des réseaux hiérarchiques ou plats.

Cette section traite des points suivants :

Système d'appellation

Le système d'appellation des réseaux plats est très simple : les noms attribués aux hôtes sont formés par une chaîne unique de caractères et gérés le plus souvent localement. Chaque machine du réseau plat dispose d'un fichier **/etc/hosts** qui contient, pour chaque hôte du système, l'équivalence entre le nom et l'adresse Internet. Ce fichier s'étoffe avec l'extension du réseau et sa mise à jour représente une tâche de plus en plus lourde. Lorsque des réseaux prennent une grande envergure comme dans le cas d'Internet, leurs systèmes d'appellation sont hiérarchisés. Ces divisions reflètent généralement l'organisation des réseaux. En TCP/IP, le système d'appellation est connu sous le nom de DNS (*domain name system*) et utilise le protocole DOMAIN. Ce protocole DOMAIN est lancé par le démon **named** dans TCP/IP.

Le système d'appellation hiérarchique DNS, comme pour les réseaux plats, attribue aux réseaux et aux hôtes des noms symboliques à la fois représentatifs et faciles à mémoriser. Mais au lieu de tenir un fichier d'équivalence sur chaque machine du réseau, il désigne un ou plusieurs hôtes pour jouer le rôle de *serveurs de noms.* Ces serveurs sont chargés de traduire (résoudre) les noms symboliques des réseaux et des hôtes en adresses Internet interprétables par les machines. Chaque serveur dispose des informations complètes sur la *zone* du domaine dont il a la charge.

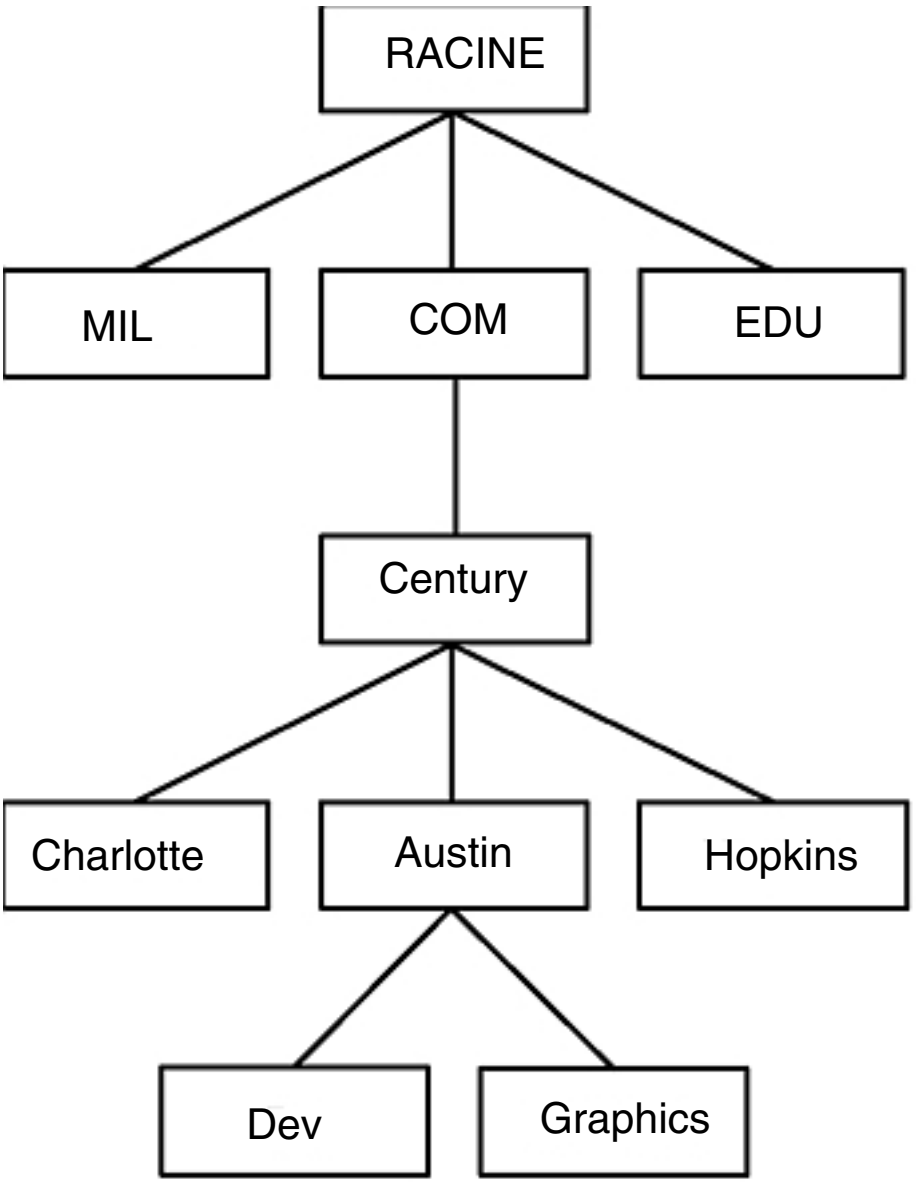
Autorité d'appellation

Dans un réseau plat, tous les hôtes sont gérés par une autorité centrale. Cette forme de réseau implique que tous les hôtes aient un nom unique. Transposé à un réseau étendu, ce système représenterait, pour l'autorité centrale, une charge administrative très lourde.

Dans un réseau hiérarchique (organisé en domaines), les hôtes sont gérés par groupes répartis dans une hiérarchie de domaines et de sous-domaines. Ainsi, l'unicité d'un nom d'hôte n'est exigée que dans son domaine local, et l'autorité centrale n'a en charge que le *domaine racine*. Cette structure, qui permet la gestion des sous-domaines en local, décharge l'autorité centrale. Prenons l'exemple du réseau Internet : son domaine racine est divisé en domaines `com` (secteur commercial), `edu` (secteur éducatif), `gov` (secteur public)

et `mil` (secteur militaire). A ce niveau, seule l'autorité centrale est habilitée à ajouter de nouveaux domaines. Dans chacun de ces domaines, l'appellation de deuxième niveau est déléguée à un agent désigné. Ainsi, l'agent du domaine COM décide de l'appellation de tous les sous-domaines situés sous `com`. De même, l'appellation au troisième niveau (et ainsi de suite) est déléguée aux agents de ce niveau. Dans la figure qui suit, le domaine Century est responsable de l'appellation de ses sous–domaines Austin, Hopkins et Charlotte.

Figure 23. Structure de domaine Internet Cette figure illustre la structure hiérarchique d'Internet. Elle commence par le haut avec la racine et forme de branches jusqu'au niveau suivant contenant les domaines mil, com et edu. Sous le domaine com se trouve un autre niveau contenant Charlotte, Austin et Hopkins. Sous Austin se trouvent Dev et Graphics.



Le sous-domaine Austin pourrait aussi être divisé en zones comme Dev et Graphics. Dans ce cas, la zone `austin.century.com` couvre toutes les données du domaine `austin.century.com`, excepté celles dépendant de Dev et de Graphics. De même, la zone `dev.century.com` contient uniquement les données confiées à Dev et n'a aucune visibilité sur le contenu de la zone Graphics. La zone `austin.century.com` (par opposition au domaine du même nom) ne contient que les données qui n'ont pas été confiées aux autres zones.

Conventions d'appellation

Dans un système d'appellation hiérarchique, les noms sont formés par une suite de noms sans distinction majuscules/minuscules, séparés par un point et dépourvus d'espaces. Selon le protocole DOMAIN, la longueur du nom de domaine local doit être inférieure à 64 caractères et celle du nom d'hôte, à 32 caractères. Le nom de l'hôte vient en premier, suivi d'un point (.), d'une série de noms de domaines locaux et enfin du domaine racine. Au total, le nom complet d'un domaine pour un hôte ne doit pas dépasser 255 caractères (points compris) et se présente sous la forme :

```
hôte.sousdomaine1.[sousdomaine2 . . . sousdomain].domaineracine
```

Les noms d'hôte étant uniques dans un domaine, vous pouvez utiliser des noms abrégés (relatifs) pour envoyer des messages au sein du même domaine. Par exemple, au lieu d'adresser un message à `smith.eng.lsu.edu`, un hôte du domaine `eng` peut indiquer seulement `smith`. Par ailleurs, chaque hôte peut être assorti de plusieurs alias utilisables par les autres hôtes.

Appellation des hôtes de votre réseau

Les noms d'hôte sont conçus pour simplifier la désignation des ordinateurs d'un réseau. Les administrateurs d'Internet ont constaté que, en matière de nom, il existe de bons et de mauvais choix. Il faut donc éviter certains pièges.

Voici quelques conseils pour vous aider à choisir les noms d'hôte de votre réseau :

- Préférez des noms peu usités tels que `sphinx` ou `eclipse`.
- Utilisez aussi des noms thématiques tels que des couleurs, des éléments `helium`, `argon` ou `zinc`), des fleurs, des poissons, etc.
- Pensez encore à utiliser de véritables mots (plutôt que des chaînes de caractères aléatoires).

Puisez dans le vocabulaire existant (n'inventez pas de chaînes de caractères). Inversement, pour limiter les oublis ou les confusions (pour l'utilisateur ou la machine), évitez :

- les termes très courants tels que `up`, `down` ou `crash`,
- les noms contenant uniquement des chiffres,
- les noms contenant des signes de ponctuation,
- les noms différenciés par des majuscules ou des minuscules (par exemple, `Orange` et `orange`),
- le nom ou les initiales de l'utilisateur principal du système,
- les noms de plus de 8 caractères,
- les orthographes inhabituelles ou volontairement incorrectes (par exemple, `czek`, qui peut être confondu avec "check" ou "tech"),
- les noms de domaine ou assimilables, tel que `yale.edu`.

Serveurs de noms

Dans une structure plate, tous les noms doivent être stockés dans le fichier **/etc/hosts** de chaque hôte membre du réseau. Ce système se révèle difficile à gérer lorsque le réseau est très étendu.

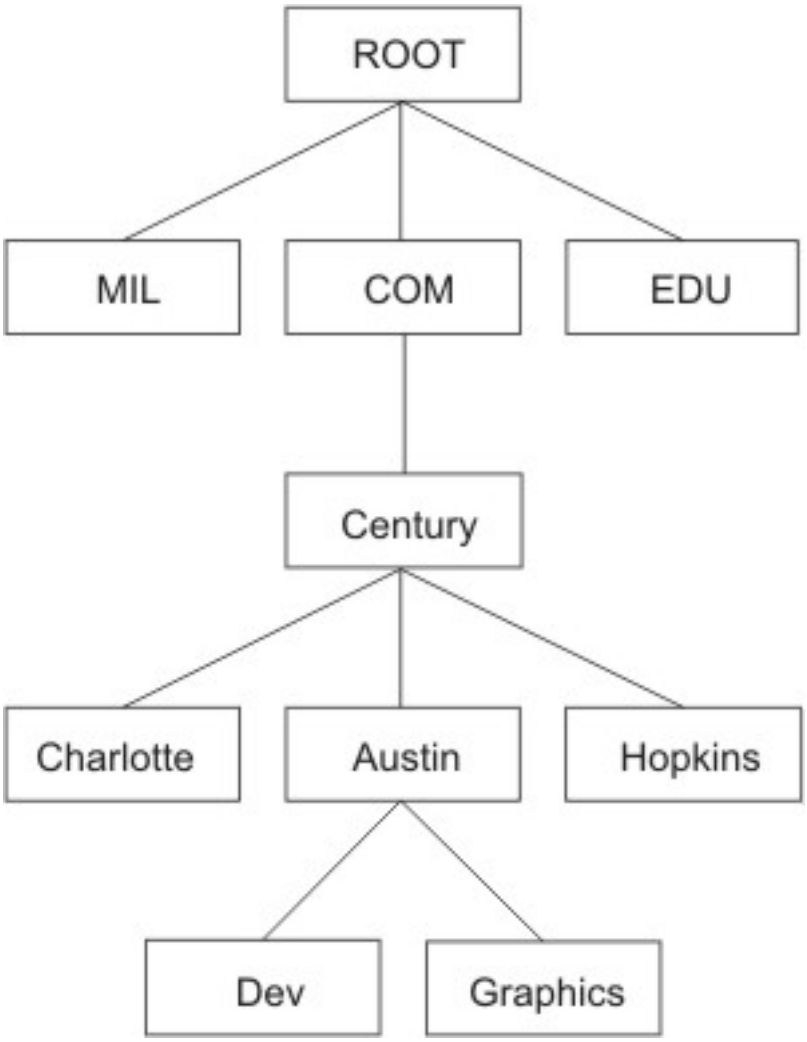
Dans une structure hiérarchique, les hôtes désignés comme *serveurs de noms* se chargent de résoudre le nom de chaque hôte en une adresse Internet. Ce mécanisme présente deux avantages par rapport à la structure plate : les ressources nécessaires à la résolution des noms ne sont pas mobilisées au niveau de chaque hôte et la tâche de l'administrateur système, alors déchargé de la mise à jour de chaque fichier de résolution des noms, s'en trouve allégée. L'ensemble des noms administrés par un serveur de noms est appelé *zone d'autorité* de ce serveur.

Remarque : La machine qui assure la fonction de résolution des noms pour une zone d'autorité est appelée hôte *serveur de noms* mais, en réalité, c'est le process (démon **named**) contrôlant cette fonction qui est le véritable serveur de noms.

Pour optimiser l'activité du réseau, les serveurs de noms stockent en mémoire *cache* (mémoire temporaire) les équivalences noms-adresses. Ainsi, lorsqu'un client demande au serveur de résoudre un nom, ce dernier consulte d'abord la mémoire cache où se trouvent les équivalences des derniers noms résolus. Ces équivalences sont conservées en mémoire pour une durée limitée (définie dans le paramètre TTL "Time-To-Live" de l'article de ressource), les noms de domaine et d'hôtes pouvant être modifiés. Les autorités d'appellation sont donc en mesure de spécifier la durée pendant laquelle la résolution de noms est réputée fiable.

Un système autonome peut comporter plusieurs serveurs de noms. Ces serveurs de noms suivent généralement une structure hiérarchique qui reflète l'organisation du réseau. Comme le montre la figure Structure en domaines d'Internet, chaque domaine peut bénéficier d'un serveur de noms responsable de tous ses sous-domaines. Les serveurs de noms des sous-domaines communiquent avec le serveur de noms du domaine supérieur (ou serveur de noms *père*) et les serveurs des autres sous-domaines.

Figure 24. Structure de domaine Internet Cette figure illustre la structure hiérarchique d'Internet. Elle commence par le haut avec la racine et forme de branches jusqu'au niveau suivant contenant les domaines mil, com et edu. Sous le domaine com se trouve un autre niveau contenant Charlotte, Austin et Hopkins. Sous Austin se trouvent Dev et Graphics.



Dans la figure Structure en domaines d'Internet, Austin, Hopkins et Charlotte sont tous des sous-domaines de Century. Si la hiérarchie du réseau est respectée, le serveur de noms Austin communique avec Charlotte et Hopkins ainsi qu'avec le serveur de noms père Century.

Austin communique également avec les serveurs de noms chargés de ses sous-domaines.

Il existe plusieurs types de serveur de noms :

serveur de noms maître	Il charge ses données à partir d'un fichier ou d'un disque et peut éventuellement déléguer des fonctions à d'autres serveurs de son domaine.
serveur de noms esclave	Au lancement du système, il reçoit du serveur de noms maître les informations sur une zone d'autorité donnée, et interroge périodiquement ce serveur maître pour la mise à jour des informations. Une fois le délai de rafraîchissement écoulé (valeur de l'article SOA sur le serveur de noms esclave), ou à réception d'une notification émise par le serveur maître, le serveur esclave recharge la base de données à partir du serveur maître si la sienne est devenue obsolète (autrement dit, si sa référence est antérieure à celle de la base du serveur maître). S'il devient nécessaire de forcer un transfert de zones, il suffit de supprimer les bases de données en place sur les serveurs esclaves et de régénérer le démon **named** sur le serveur de noms esclave.

Serveur de noms de tronçon (stub)	Bien que la méthode soit similaire à celle utilisée par le serveur de noms esclave, le serveur de noms de tronçon (stub) reproduit uniquement les données de serveurs de noms de la base de données maître, et non l'ensemble de la base.
Serveur d'indices (hint server)	Ce serveur de noms ne fonctionne que d'après les indices accumulés suite aux requêtes antérieures auprès d'autres serveurs. Le serveur d'indices (hint server) répond aux requêtes en demandant les informations souhaitées auprès des autres serveurs "experts" (serveurs ayant autorité) s'il ne dispose pas dans sa mémoire cache de l'équivalence demandée.
Serveur client ou expéditeur	Envoie les requêtes qu'il ne peut satisfaire localement aux serveurs répertoriés dans une liste prédéfinie. Les serveurs exclusivement expéditeurs (simples transmetteurs d'informations, qui ne sont pas à proprement parler des serveurs) ne dialoguent pas avec les serveurs de noms maîtres pour le domaine racine ou les autres domaines. Les requêtes transitent d'un serveur à l'autre de façon récursive : les serveurs expéditeurs sont contactés l'un après l'autre jusqu'à la fin de la liste. Ce type de configuration est généralement utilisé pour éviter que tous les serveurs d'un site dialoguent avec les autres serveurs Internet, ou pour constituer une mémoire cache étendue dans un certain nombre de serveurs de noms.
Serveur distant	Lance tous les programmes réseau qui font appel au serveur de noms, alors que le process serveur de noms n'est pas exécuté sur l'hôte local. Les requêtes sont prises en charge par un serveur de noms exécuté sur une autre machine du réseau.

Un hôte serveur de noms peut exercer diverses fonctions dans des zones d'autorité différentes. Par exemple, il peut faire fonction de serveur de noms maître dans une zone, et de serveur de noms esclave dans une autre.

Résolution de noms

La procédure d'obtention d'une adresse Internet à partir d'un nom d'hôte, appelée "résolution de noms", est exécutée par la sous-routine **gethostbyname**. Inversement, la traduction d'une adresse Internet en nom d'hôte est appelée "résolution inverse de noms", et est exécutée par la sous-routine **gethostbyaddr**. Ces routines permettent essentiellement d'accéder à une bibliothèque de routines de traduction de noms appelées *"routines de résolution"*.

Les routines de résolution sur les hôtes TCP/IP essaient normalement de résoudre les noms en utilisant les sources suivantes :

1. BIND/DNS (nommé),
2. NIS (Network Information Services),
3. Fichier **/etc/hosts** local.

Lors de l'installation de NIS+, les préférences de recherche sont définies dans le fichier **irs.conf**. Pour plus d'informations, reportez–vous à *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide* .

Pour résoudre un nom dans un réseau hiérarchique, la routine de résolution émet tout d'abord une requête auprès de la base de données du serveur de noms de domaine, résidant sur l'hôte local (s'il s'agit d'un hôte serveur de noms de domaine) ou sur un hôte étranger. Les serveurs de noms transforment les noms de domaines en adresses Internet. L'ensemble des noms administrés par un serveur de noms est appelé zone d'autorité de ce

serveur. Si la routine de résolution utilise un serveur de noms distant, elle a recours au protocole DOMAIN (protocole de noms de domaine) pour les requêtes de mappage. Pour résoudre un nom dans un réseau plat, la routine recherche l'entrée correspondante dans le fichier local **/etc/hosts**. Si NIS ou NIS+ est utilisé, le fichier **/etc/hosts** du serveur maître est également vérifié.

Par défaut, les routines de résolution essaient de résoudre les noms à l'aide des ressources mentionnées ci–dessus. Le mécanisme BIND/DNS est lancé en premier. Si le fichier **/etc/resolv.conf** n'existe pas ou si le nom est introuvable, une requête est émise auprès de NIS si ce système est en service. NIS ayant autorité sur le fichier **/etc/hosts** local, la recherche peut s'arrêter là. Si NIS n'est pas en service, la recherche s'effectue sur le fichier local **/etc/hosts**. Si ce nom reste introuvable, les routines de résolution renvoient le message HOST_NOT_FOUND. Si tous les services sont indisponibles, les routines de résolution renvoient le message SERVICE_UNAVAILABLE.

Il est possible de modifier l'ordre de recherche présenté ci–dessus en créant le fichier de configuration **/etc/irs.conf** pour y préciser l'ordre voulu. Ces deux ordres (ordre par défaut et fichier **/etc/irs.conf**) peuvent encore être écrasés par la variable d'environnement **NSORDER**. Si ni le fichier **/etc/irs.conf** ni la variable d'environnement **NSORDER** ne sont définies, au moins une valeur doit être spécifiée avec l'option.

Pour définir l'ordre des hôtes avec le fichier **/etc/irs.conf** :

```
hosts value [ continue ]
```

Pour définir l'ordre, chaque méthode doit être indiquée sur une ligne distincte. La *valeur* correspond à une des méthodes indiquées et le mot clé `continue` indique qu'une autre méthode de résolution figure en ligne suivante.

Dans la variable d'environnement **NSORDER** :

```
 NSORDER= value, value, value
```

L'ordre est spécifié sur une seule ligne avec des valeurs séparées par une virgule. Les espaces sont admis entre les virgules et le signe égal.

Par exemple, si le réseau local est "plat", seul le fichier **/etc/hosts/** est nécessaire. Dans cet exemple, le fichier **/etc/irs.conf** contient la ligne suivante :

```
hosts local
```

Autrement, la variable **NSORDER** peut être renseignée comme suit :

```
NSORDER=local
```

Si le réseau local est "hiérarchique" et fait appel à un serveur de noms pour la résolution des noms et à un fichier **/etc/hosts** pour une copie de secours, les deux services doivent être spécifiés. Dans cet exemple, le fichier **/etc/irs.conf** contiendrait les lignes suivantes :

```
hosts dns continue
hosts local
```

Et la variable **NSORDER** est renseignée comme suit :

```
NSORDER=bind,local
```

Remarque : les valeurs doivent être spécifiées en minuscules.

En suivant un ordre de résolution défini ou l'ordre par défaut, l'algorithme de recherche ne passe d'une routine à la suivante que si :

- le service courant n'est pas accessible (il n'est pas actif),
- le service courant ne trouve pas le nom recherché et n'est pas un serveur "expert".

Si le fichier **/etc/resolv.conf** n'existe pas, le mécanisme BIND/DNS est considéré comme non installé, et par là–même non accessible. En cas d'échec des sous–routines **getdomainname** et **yp_bind**, le service NIS est considéré comme non installé et par là–même non accessible. Si le fichier **/etc/hosts** n'a pas pu être ouvert, il est impossible de procéder à une recherche locale et d'accéder au fichier et au service.

Un service est dit *expert* si, de par les informations qu'il contient, il est jugé mieux à même de répondre aux requêtes que les services cités après lui. Les routines de résolution n'essaient pas les services suivants, puisque ces derniers ne peuvent contenir qu'un sous–ensemble des informations du service expert. La résolution des noms s'arrête à la consultation du service expert même s'il n'est pas parvenu à fournir le nom demandé (message HOST_NOT_FOUND renvoyé). En cas d'indisponibilité d'un service expert, le service suivant spécifié est interrogé.

La source "expert" est déclarée par la chaîne `"=auth"` spécifiée à la suite de son nom. Il est possible de spécifier également tout le mot `"authoritative"`. Par exemple, si la variable **NSORDER** contient :

```
hosts = nis=auth,dns,local
```

Si NIS est actif, la recherche prend fin après consultation de NIS, que le nom ait été trouvé ou non. Si NIS n'est pas actif, elle est étendue à la source suivante (en l'occurrence, DNS).

Les serveurs de noms TCP/IP ont recours à la mémoire cache pour réduire le coût de recherche de noms d'hôte sur réseaux distants. Ainsi, ils consultent d'abord la mémoire cache où se trouvent les équivalences des derniers noms résolus. Ces équivalences sont conservées en mémoire pour une durée limitée (définie dans le paramètre TTL "Time-To-Live" de l'article de ressource), les noms de domaine et d'hôtes pouvant être modifiés. Les serveurs de noms sont donc en mesure de spécifier la durée pendant laquelle leurs réponses sont réputées fiables.

Risques de conflits de noms d'hôte

En environnement DNS, un nom d'hôte défini soit par la commande **hostname** en ligne de commande, soit par le fichier **rc.net**, doit être le nom officiel de l'hôte tel qu'il est renvoyé par le serveur de noms. Ce nom est généralement le nom complet du domaine de l'hôte sous la forme :

```
host.subdomain.subdomain.rootdomain
```

Remarque : pour les routines de résolution, le domaine par défaut doit être défini. S'il n'est pas défini dans **hostname**, il doit l'être dans **/etc/resolv.conf**.

Si le nom de l'hôte n'est pas configuré en nom complet du domaine, et si le système est configuré avec serveur de noms de domaine associé au programme **sendmail**, le fichier de configuration **/etc/sendmail.cf** doit être modifié conformément à ce nom officiel. Pour que le programme **sendmail** fonctionne correctement, il faut de plus que les macros de nom de domaine soient définies dans cette configuration.

Remarque : pour toutes les fonctions de **sendmail**, le domaine spécifié dans le fichier **/etc/sendmail.cf** prime sur celui défini à la commande **hostname**.

Risques de conflits de noms de domaine

Dans le cas d'un hôte membre d'un réseau DOMAIN mais qui n'est pas un serveur de noms, le nom de domaine local et le serveur de noms de domaine sont spécifiés dans le fichier **/etc/resolv.conf**. Or, dans un hôte serveur de noms de domaine, le domaine local et les autres serveurs de noms sont définis dans des fichiers que le démon **named** lit à son lancement.

Protocole RARP

Le protocole RARP (Reverse Address Resolution Protocol) traduit les adresses matérielles uniques en adresses Internet sur la carte LAN Ethernet (protocole Ethernet seulement). Le protocole Ethernet standard est pris en charge dans les limites suivantes :

- Le serveur répond uniquement aux requêtes RARP.
- Le serveur se limite aux entrées permanentes de la table ARP
- Le serveur n'utilise pas les entrées dynamiques de la table ARP.
- Le serveur ne répond pas automatiquement pour lui–même.

L'administrateur système doit créer et tenir à jour manuellement une table des entrées permanentes ARP à l'aide de la commande **arp**. Une entrée de table ARP spécifique doit être ajoutée sur le serveur pour chaque hôte qui sollicite des réponses RARP d'une source "expert".

Résolution locale des noms (/etc/hosts)

Le fichier **/etc/hosts** doit être configuré si vous travaillez sur un réseau limité et plat. Cette configuration peut également être utile sur un réseau hiérarchique pour identifier les hôtes inconnus des serveurs de noms.

Vous pouvez configurer votre système en vue de la résolution locale de noms via Web-based System Manager, SMIT ou les commandes. Si c'est à partir de la ligne de commande, veillez à conserver le format du fichier **/etc/hosts**, comme indiqué à la section "Hosts File Format for TCP/IP" du manuel *AIX 5L Version 5.2 Files Reference*).

Résolution locale des noms			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Afficher la liste des hôtes	**smit lshostent**	affichez **/etc/hosts**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Hosts File** —> **Contents of /etc/hosts file**.
Ajouter un hôte	**smit mkhostent**	éditez **/etc/hosts**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Hosts File**. Dans la boîte de dialogue **Add/Change host entry**, complétez les champs suivants : **IP Addresses**, **Host name**, **Alias(es)** et **Comment**. Cliquez sur **Add/Change Entry** —> **OK**.

Modifier/afficher les caractéristiques d'un hôte	**smit chhostent**	éditez **/etc/hosts**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Hosts File**. Sélectionnez un hôte dans **Contents of /etc/hosts/file**, puis changez les données dans **Add/Change host entry**. Cliquez sur **Add/Change Entry** —> **OK**.
Supprimer un hôte	**smit rmhostent**	éditez **/etc/hosts**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Hosts File**. Sélectionnez un hôte dans **Contents of /etc/hosts/file**, puis cliquez sur **Delete Entry** —> **OK**.

Préparation à la résolution DNS (DOMAIN)

Si vous faites partie d'un interréseau étendu, coordonnez vos serveurs de noms et domaines avec leur autorité centrale.

Les conseils suivants vous aideront à configurer votre système pour la résolution DSN :

- Familiarisez–vous avec TCP/IP, DNS et BIND et les nombreuses fonctionnalités de leur architecture et de leur configuration avant d'arrêter votre choix. Avant d'utiliser un service d'information réseau (NIS), familiarisez–vous également avec NIS. Vous disposez pour cela de nombreux documents. Pour plus d'informations sur les caractéristiques de NIS et de NIS+, reportez–vous au *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide*.
- Planifiez la configuration.

  Rappelez-vous qu'il est *plus* compliqué de changer un nom que de le définir. Avant de configurer vos fichiers, décidez (en accord avec votre organisation) des noms des hôtes, du réseau, de la passerelle et du serveur de noms.
- Définissez des serveurs de noms redondants.

  A défaut, veillez à définir des serveurs de noms esclaves et des serveurs d'indices pour disposer d'un système de secours.
- Pour la sélection des serveurs de noms :
  - choisissez les machines géographiquement les plus proches des systèmes extérieurs ;
  - vos serveurs de noms doivent être aussi indépendants que possible. Si possible, utilisez des alimentations électriques et des câblages distincts.
  - désignez un autre réseau comme réseau de secours pour votre service de résolution des noms ; faites de même pour les autres réseaux.
- testez les serveurs :
  - testez la résolution des noms normale et inverse,
  - testez le transfert de zone du serveur de noms maître au serveur de noms esclave,

– testez chaque serveur de noms, après une panne et un réamorçage du système.

- Faites transiter vos requêtes de résolution de noms par des serveurs expéditeurs avant de les envoyer vers des serveurs de noms extérieurs. Cela permet aux serveurs de noms de partager leur mémoire cache et d'améliorer les performances en allégeant la charge des serveurs de noms maîtres.

```
objectclass: container
        requires
                objectclass,
                cn
objectclass hosts
        requires
                objectclass,
                hname
        allows
                addr
                halias,
                comment
```

Serveur de noms : généralités

Dans un réseau hiérarchisé, certains hôtes sont définis comme *serveurs de noms*. Ces hôtes résolvent les noms en adresses IP pour les autres hôtes. The Le démon **named** contient la fonction du serveur de noms et doit donc être exécuté sur un hôte de serveur de noms.

Avant de procéder à la configuration, déterminez les types de serveur de noms les mieux adaptés à votre réseau. Il existe trois types :

Leurs noms, défini dans le fichier conf, peut être modifié par l'utilisateur. Par convention, ce nom comporte celui du démon (*named*) avec, en extension, le type de fichier et le nom du domaine. *Serveur de noms esclave* ou *serveur de noms de tronçon (stub)* : ceux–ci reçoivent leurs informations d'un serveur maître au démarrage du système pour une zone d'autorité donnée, puis l'interrogent périodiquement pour les mettre à jour. *Serveur d'indices (hint server)* : ce serveur répond aux requêtes de résolution des noms en demandant les informations souhaitées auprès d'autres serveurs experts.

Remarque : les générations antérieures du serveur de noms **named** définissaient le serveur maître comme serveur de noms primaire, le serveur esclave comme serveur de noms secondaire, et le serveur d'indices comme serveur de mémoire cache. Dans cette documentation, toute référence au fichier **named.conf** est spécifique à version 4.3.2 ou versions ultérieures.

Rappelons qu'un serveur de noms peut exercer des fonctions différentes selon les zones d'autorité. Par exemple, un hôte peut faire fonction de serveur de noms maître dans une zone, et de serveur de noms esclave dans une autre. Si NIS ou NIS+ est installé sur votre système, ces services peuvent également vous aider à la résolution des noms. Pour plus d'informations, reportez–vous au *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide*.

Plusieurs fichiers sont impliqués dans la configuration des serveurs de noms.

conf Fichier lu au démarrage du démon **named**. Les articles du fichier **conf** indiquent au démon **named** le type du serveur, ses zones d'autorité (domaines) et l'implantation des données pour la configuration initiale de sa base de données. Son nom par défaut est **/etc/named.conf**. Vous pouvez lui en attribuer un autre en indiquant sur la ligne de commande le nouveau nom complet dès le lancement du démon **named**. Si vous utilisez pour l'amorçage le fichier **/etc/named.conf**, mais que ce dernier n'existe pas, un message est généré dans **syslog** et le démon **named** s'arrête. Toutefois, si un autre fichier **conf** a été prévu et qu'il n'existe pas, il n'y aura pas de message d'erreur et le démon **named** continuera.

cache Fichier contenant les informations sur la mémoire cache locale : nom et adresse des serveurs de noms bénéficiant de la plus haute "autorité". Ce fichier respecte le format des articles de ressource standard (Standard Resource Record Format). Son nom est défini dans le fichier **conf**.

domain data Il existe trois types de fichiers domain data, également nommés fichiers de données **named**. Le fichier **named** *local* contient les informations de résolution d'adresses en bouclage local. Le fichier de *données* **named** contient les données de résolution d'adresses pour toutes les machines de la zone d'autorité du serveur de noms. Le fichier de *données inversées* **named** contient les informations de résolution inversée d'adresses pour toutes les machines de la zone d'autorité du serveur de noms. Ces trois fichiers respectent le format des articles de ressource standard (Standard Resource Record Format). Leurs noms, défini dans le fichier **conf** , peut être modifié par l'utilisateur. Par convention, ce nom comporte celui du démon (`named`) avec, en extension, le type de fichier et le nom du domaine. Par exemple, les fichiers du serveur de noms du domaine `abc` peuvent être :

```
named.abc.data
named.abc.rev
named.abc.local
```

En modifiant les fichiers de données **named**, il est conseillé d'incrémenter le numéro de série donné dans l'article SOA pour les serveurs de noms esclaves afin d'effectuer correctement les modifications de zones.

resolv.conf Ce fichier indique par sa présence que l'hôte doit d'abord faire appel à un serveur de noms pour effectuer une résolution. En l'absence de **resolv.conf**, l'hôte fait ensuite appel au fichier **/etc/hosts**. Obligatoire sur un serveur de noms, le fichier **resolv.conf** peut contenir l'adresse de l'hôte local, l'adresse de bouclage (127.0.0.1), ou être vide.

Remarque :
Pour les routines de résolution, le domaine par défaut doit être défini. S'il n'est pas défini dans **/etc/resolv.conf**, il doit l'être dans **hostname**.

Le paramètre TTL (Time-To-Live) est spécifié dans les articles de ressource. A défaut, le délai appliqué est la plus petite valeur définie dans l'article SOA (Start Of Authority) de la zone d'autorité concernée. TTL est utilisé lorsque les données sont stockées en dehors d'une zone (en mémoire cache) pour s'assurer qu'elles n'y sont pas maintenues indéfiniment.

Configuration des serveurs de noms

Pour savoir comment configurer les serveurs maître, esclave et d'indices, reportez–vous à Configurer les serveurs de noms de domaine.

Configuration d'un serveur de courrier de domaine

En définissant un serveur de courrier de domaine, vous mettez à la disposition des utilisateurs externes une méthode d'adressage simple leur permettant de correspondre avec votre organisation. Sans cela, l'adresse doit obligatoirement préciser un hôte particulier de votre organisation. Par exemple, `sam@orange.widget.com`, `widget.com` étant le nom de domaine de votre organisation, et `orange` l'hôte utilisé par `sam`. Avec le serveur de courrier de domaine, il suffit à l'utilisateur externe d'indiquer le nom de l'utilisateur et le nom du domaine sans le nom de l'hôte, dans notre exemple, `sam@widget.com`.

Vous pouvez configurer un serveur de courrier via le raccourci Web-based System Manager **wsm** ou via l'une des procédures suivantes.

Configuration d'un serveur de courrier de domaine

1. Créez un article MX et un article A pour le serveur de courrier (`black.widget.com`) :

```
widget.com            IN    MX     10 black.widget.com
widget.com            IN     A     192.10.143.9
black.widget.com      IN     A     192.10.143.9
```

2. Editez **sendmail.cf** sur le serveur de courrier (`black.widget.com`) pour ajouter l'alias du domaine (classe **w**) :

```
Cw $w $?D$w.$D$. widget.com
```

3. Les clients de la messagerie doivent savoir où adresser leur courrier non local. Editez **sendmail.cf** sur chaque client pour pointer sur le serveur de courrier (macro **S**) :

```
DRblack.widget.com
```

4. A l'aide de l'option **NameServOpt**, configurez le démon **sendmail** de sorte que chacun puisse utiliser les articles MX définis dans le serveur de noms `brown.widget.com`.
5. Ajoutez l'alias des utilisateurs du domaine qui n'ont pas de compte sur le serveur de courrier, en vous aidant du fichier d'alias.

```
sam:sam@orange.widget.com
david:david@green.widget.com
judy:judy@red.widget.com
```

   Remarque : les articles MB peuvent remplir la même fonction.

6. La base de données ayant été modifiée, il est conseillé d'incrémenter le numéro de série donné dans l'article SOA.
7. Régénérez la base de données du serveur de noms via la commande **refresh –s named**.
8. Sur les clients, exécutez la commande **refresh –s sendmail** pour prendre les modifications en compte.

Il existe d'autres méthodes permettant de configurer un serveur de courrier de domaine. Les procédures qui suivent utilisent les articles MB, MR et MG.

Configuration d'un serveur de courrier de domaine avec des articles MB

1. Définissez un article MB pour chaque utilisateur du domaine. Par exemple :

```
sam IN MB orange.widget.com.
```

   dans le fichier **/usr/named.data** de l'hôte `brown.widget.com` . Cette instruction stipule le serveur de courrier (`black.widget.com`) destinataire pour chaque utilisateur du domaine.

2. Configurez le démon **sendmail** sur le serveur de courrier (`black.widget.com`) pour qu'il utilise les articles MB définis sur le serveur de noms (`brown.widget.com`). Ayez recours à l'option **NameServOpt**.

3. La base de données ayant été modifiée, il est conseillé d'incrémenter le numéro de série donné dans l'article SOA.
4. Régénérez la base de données du serveur de noms via la commande **refresh –s named**.
5. Tapez la commande **refresh –s sendmail** pour prendre les modifications en compte.

Définition d'un article MR (Mail Rename)

1. Editez le fichier **/usr/named.data** sur votre serveur de noms de domaine.
2. Ajoutez un article MR pour chaque alias. Par exemple, l'utilisateur `sam` dont l'alias est `sammy` aura pour article MR :

   ```
   sammy IN MR sam
   ```

   Cet article demande que tous les messages adressés à `sammy` soient livrés à `sam`. Il faut prévoir une ligne par article MR.
3. La base de données ayant été modifiée, il est conseillé d'incrémenter le numéro de série donné dans l'article SOA.
4. Régénérez la base de données du serveur de noms via la commande **refresh –s named**.
5. Tapez la commande **refresh –s sendmail** pour prendre les modifications en compte.

Définition d'un article MG (Mail Group)

1. Editez le fichier **/etc/named.data** sur votre serveur de noms de domaine.
2. Ajoutez des articles MG pour chaque groupe courrier. Ces articles fonctionnent comme le fichier **/usr/aliases**, les alias étant tenus à jour sur le serveur de noms. Par exemple :

   ```
   users IN HINFO users-request widget.com
   users IN MG sam
   users IN MG david
   users IN MG judy
   ```

   Ces articles demandent que tous les messages adressés à `users@widget.com` soient livrés à `sam`, `david` et `judy`. Il faut prévoir une ligne par article MG.

   Remarque : des articles MB doivent avoir été définis pour `sam`, `david` et `judy`.
3. La base de données ayant été modifiée, il est conseillé d'incrémenter le numéro de série donné dans l'article SOA.
4. Régénérez la base de données du serveur de noms via la commande **refresh –s named**.
5. Entrez la commande **sendmail –bz** pour recompiler le fichier **sendmail.cf** sur le serveur de courrier, puis la commande **refresh -s sendmail** pour appliquer les modifications.

Définition d'articles MX (Mail Exchanger)

1. Editez le fichier **/usr/named.data** sur votre serveur de noms de domaine.
2. Ajoutez des articles MX pour chaque machine indirectement connectée à votre réseau et avec laquelle vous souhaitez correspondre. Par exemple, si le courrier adressé aux utilisateurs de `purple.widget.com` doit être transmis à `post.office.widget`, ajoutez un article MX comme suit :

   ```
   purple.widget.com IN MX 0 post.office.widget.
   ```

   Lorsque vous utilisez les articles d'échangeur de courrier (MX), vous devez spécifier le nom de la machine et le nom d'hôte. Il faut prévoir une ligne par article MX. L'utilisation des caractères génériques est admise :

   ```
   *.widget.com IN MX 0 post.office.widget.
   ```

Ces articles demandent que les messages adressés à un hôte inconnu (sans article MX explicite) du domaine `widget.com` soient expédiés à `post.office.widget`.

Remarque : les caractères génériques dans les articles MX sont incompatibles avec l'utilisation d'Internet.

3. La base de données ayant été modifiée, il est conseillé d'incrémenter le numéro de série donné dans l'article SOA.
4. Régénérez la base de données du serveur de noms via la commande **refresh –s named**.
5. Tapez la commande **refresh –s sendmail** pour prendre les modifications en compte.

Configuration d'un serveur expéditeur

Pour configurer un expéditeur, utilisez le raccourci Web-based System Manager **wsm** ou suivez la procédure ci–dessous, qui édite une série de fichiers puis a recours à SMIT ou à la ligne de commande pour démarrer le démon **named**.

1. Editez le fichier **/etc/named.conf**. Si le répertoire **/etc** ne contient pas de fichier **named.conf**, copiez-y le fichier-type **/usr/samples/tcpip/named.conf** et éditez-le. Pour en savoir plus et examiner un exemple de fichier de configuration, reportez-vous à la section "named.boot File Format for TCP/IP" dans le manuel *AIX 5L Version 5.2 Files Reference*.

   – Insérez une ligne "forwarders" dans la strophe d'options du fichier **/etc/named.conf** indiquant toutes les adresses IP des serveurs de noms auxquels des requêtes doivent être expédiées. Par exemple :

   ```
   options {
        ...
        directory "/usr/local/domain";
       forwarders { 192.100.61.1; 129.35.128.222; };
        ...
    };
   ```

   – Spécifiez la zone de bouclage. Par exemple :

   ```
   zone "0.0.127.in-addr.arpa" in {
       type master;
        file "named.abc.local";
    };
   ```

   – Spécifiez la zone d'indices. Par exemple :

   ```
   zone "." IN {
        type hint;
        file "named.ca";
    };
   ```

2. Editez le fichier **/usr/local/domain/named.ca** Pour en savoir plus et disposer d'un exemple de fichier cache, reportez–vous à la section "DOMAIN Cache File Format for TCP/IP" dans le manuel *AIX 5L Version 5.2 Files Reference*.

   Ce fichier contient l'adresse des serveurs de noms "experts" pour le domaine racine (root) du réseau. Par exemple :

   ```
   ; root name servers.
   .          IN    NS    relay.century.com.
   relay.century.com.   3600000    IN    A     129.114.1.2
   ```

   Remarque : toutes les lignes de ce fichier doivent respecter le format des articles de ressource standard (Standard Resource Record Format).

3. Editez le fichier **/usr/local/domain/named.abc.local**. Pour en savoir plus et disposer d'un exemple de fichier de données local, reportez–vous à la section " DOMAIN Local Data File Format for TCP/IP " dans le manuel *AIX 5L Version 5.2 Files Reference*.

a. Spécifiez pour la zone la valeur de SOA (Start Of Authority) et les délais TTL (time–to–live) par défaut. Par exemple :

```
$TTL 3h    ;3 hour

 @ IN SOA venus.abc.aus.century.com. gail.zeus.abc.aus.century.com.  (

                              1       ;serial
                              3600    ;refresh
                             600     ;retry
                              3600000 ;expire
                              86400   ;negative caching TTL
)
```

b. Spécifiez l'article NS (serveur de noms). Par exemple :

```
 <tab>       IN    NS     venus.abc.aus.century.com.
```

c. Spécifiez l'article PTR (pointeur).

```
1       IN    PTR    localhost.
```

Remarque : toutes les lignes de ce fichier doivent respecter le format des articles de ressource standard (Standard Resource Record Format).

4. Créez un fichier **/etc/resolv.conf** via la commande :

```
touch /etc/resolv.conf
```

Ce fichier indique par sa présence que l'hôte doit d'abord faire appel à un serveur de noms pour effectuer une résolution, et non au fichier **/etc/hosts**.

Autrement, le fichier **/etc/resolv.conf** peut contenir l'entrée suivante :

```
nameserver 127.00.0.1
```

`127.0.0.1` est l'adresse de bouclage qui, pour l'accès au serveur de noms, dirige l'hôte vers lui-même. Ce fichier **/etc/resolv.conf** peut également comporter une ligne du type :

```
 domain  NomDomaine
```

Dans cet exemple, *NomDomaine* serait `austin.century.com`.

5. Exécutez l'une des tâches suivantes :
   - Activez le démon **named** en utilisant le raccourci SMIT **smit stnamed**. Cette commande initialise le démon à chaque lancement du système. Indiquez quand vous souhaitez lancer le démon **named** : immédiatement, au prochain lancement du système ou les deux.
   - Editez le fichier **/etc/rc.tcpip**. Activez le démon **named** en retirant la marque de commentaire (#) de la ligne suivante :

```
#start /etc/named "$src_running"
```

Cette commande initialise le démon à chaque lancement du système.

6. Si vous ne souhaitez pas initialiser le démon named via SMIT, lancez-le pour la session en cours par la commande :

```
startsrc -s named
```

Configuration de serveur exclusivement expéditeur

Pour configurer un serveur de noms esclave, utilisez le raccourci Web-based System Manager **wsm** ou suivez la procédure ci–dessous, qui édite une série de fichiers puis a recours à SMIT ou à la ligne de commande pour démarrer le démon **named**.

Remarque : vous pouvez obtenir une configuration similaire sans exécuter de serveur de noms exclusivement expéditeur. Il suffit de créer un fichier **/etc/resolv.conf** en insérant des lignes de serveur de noms qui pointent vers les serveurs expéditeurs souhaités.

1. Editez le fichier **/etc/named.conf**. Si le répertoire **/etc** ne contient pas de fichier **named.conf**, copiez-y le fichier-type **/usr/samples/tcpip/named.conf** et éditez-le. Pour en savoir plus et examiner un exemple de fichier de configuration, reportez-vous à la section ”named.boot File Format for TCP/IP” dans le manuel *AIX 5L Version 5.2 Files Reference*.

   – Insérez les lignes ”forwarders” et ”forward only” dans la strophe d’options du fichier **/etc/named.conf** indiquant toutes les adresses IP des serveurs de noms auxquels des requêtes doivent être expédiées. Par exemple :

   ```
   options {
        ...
        directory ”/usr/local/domain”;
       forwarders { 192.100.61.1; 129.35.128.222; };
       forward only;
        ...
    };
   ```

   – Spécifiez la zone de bouclage. Par exemple :

   ```
   zone ”0.0.127.in-addr.arpa” in {
        type master;
        file ”named.abc.local”;
    };
   ```

   – Spécifiez la zone d’indices. Par exemple :

   ```
   zone ”.” IN {
        type hint;
        file ”named.ca”;
    };
   ```

2. Editez le fichier **/usr/local/domain/named.ca** Pour en savoir plus et disposer d’un exemple de fichier cache, reportez–vous à la section DOMAIN Cache File Format for TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*. Ce fichier contient l’adresse des serveurs de noms ”experts” pour le domaine racine (root) du réseau. Par exemple :

```
; root name servers.
.          IN    NS    relay.century.com.
relay.century.com.   3600000    IN    A     129.114.1.2
```

   Remarque : toutes les lignes de ce fichier doivent respecter le format des articles de ressource standard (Standard Resource Record Format).

3. Editez le fichier **/etc/named.local**. Pour en savoir plus et disposer d’un exemple de fichier de données local, reportez-vous à la section ”DOMAIN Local Data File Format for TCP/IP” dans le manuel *AIX 5L Version 5.2 Files Reference*.

   a. Spécifiez pour la zone la valeur de SOA (Start Of Authority) et les délais TTL (time-to-live) par défaut. Par exemple :

   ```
   $TTL 3h    ;3 hour

    @ IN SOA venus.abc.aus.century.com. gail.zeus.abc.aus.century.com.  (

                               1       ;serial
                               3600    ;refresh
                              600     ;retry
                               3600000 ;expire
                               86400   ;negative caching TTL
    )
   ```

   b. Spécifiez l’article NS (serveur de noms). Par exemple :

   ```
   <tab>      IN    NS     venus.abc.aus.century.com.
   ```

   c. Spécifiez l’article PTR (pointeur).

   Remarque : toutes les lignes de ce fichier doivent respecter le format des articles de ressource standard (Standard Resource Record Format).

4. Créez un fichier **/etc/resolv.conf** via la commande :

   ```
   touch /etc/resolv.conf
   ```

   Ce fichier indique par sa présence que l'hôte doit d'abord faire appel à un serveur de noms pour effectuer une résolution, et non au fichier **/etc/hosts**.

   Autrement, le fichier **/etc/resolv.conf** peut contenir l'entrée suivante :

   ```
   nameserver 127.00.0.1
   ```

   `127.0.0.1` est l'adresse de bouclage qui, pour l'accès au serveur de noms, dirige l'hôte vers lui-même. Ce fichier **/etc/resolv.conf** peut également comporter une ligne du type :

   ```
   domain  NomDomaine
   ```

   Dans cet exemple, *NomDomaine* serait `austin.century.com`.

5. Exécutez l'une des tâches suivantes :
   - Activez le démon **named** en utilisant le raccourci SMIT **smit stnamed**. Cette commande initialise le démon à chaque lancement du système. Indiquez quand vous souhaitez lancer le démon **named** : immédiatement, au prochain lancement du système ou les deux.
   - Editez le fichier **/etc/rc.tcpip**. Activez le démon **named** en retirant la marque de commentaire (#) de la ligne suivante :

   ```
   #start /etc/named "$src_running"
   ```

   Cette commande initialise le démon à chaque lancement du système.

6. Si vous ne souhaitez pas initialiser le démon **named** via SMIT, lancez-le pour la session en cours par la commande :

   ```
   startsrc -s named
   ```

Configuration d'un hôte avec serveur de noms

Vous pouvez configurer un hôte pour un serveur de noms via le raccourci Web-based System Manager, **wsm** ou via l'une des procédures suivantes.

1. Créez un fichier **/etc/resolv.conf** via la commande :

   ```
   touch /etc/resolv.conf
   ```

2. Sur la première ligne du fichier **/etc/resolv.conf**, entrez le mot **domain** puis le nom complet du domaine auquel appartient l'hôte. Par exemple :

   ```
   domain abc.aus.century.com
   ```

3. Sur une ligne vierge après la ligne introduite par `domain`, entrez le mot **nameserver** suivi d'au moins un espace et de l'adresse Internet (en notation décimale à points) du serveur de noms à ajouter (il doit desservir le domaine indiqué dans l'instruction `domain`). Vous pouvez insérer jusqu'à 16 entrées de serveur de noms. Par exemple, votre fichier **/etc/resolv.conf** peut contenir les entrées :

   ```
   nameserver 192.9.201.1
   nameserver 192.9.201.2
   ```

   Le système interroge les serveurs dans l'ordre de leur spécification.

   ```
   search domainname_list
   ```

   Le mot–clé de recherche peut aussi être utilisé pour indiquer l'ordre dans lequel le programme de résolution interrogera la liste des domaines. Dans ce cas, les valeurs de `domainname_list` sont `abc.aus.century.com` et `aus.century.com`. `domainname_list` peut contenir au maximum six noms de domaines, chacun séparés par un espace.

4. En supposant que le serveur de noms est opérationnel, vous pouvez tester sa communication avec l'hôte via la commande suivante :

```
host hostname
```

Indiquez un nom d'hôte que le serveur doit résoudre. Si le processus aboutit, vous obtenez un résultat du type :

```
brown.abc.aus.century.com is 129.35.145.95
```

D'autres tâches de configuration sont présentées dans le tableau suivant.

Configuration d'un hôte avec serveur de noms			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Créer un fichier **/etc/resolv.conf**.	**smit stnamerslv2**	créez et éditez **/etc/resolv.conf**[1]	
Afficher la liste des serveurs utilisés par un hôte	**smit lsnamerslv**	affichez **/etc/resolv.conf**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Hosts File** —> **Contents of /etc/hosts file**.
Ajouter un serveur de noms	**smit mknamerslv**	éditez **/etc/resolv.conf** [2]	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **DNS**. Dans le champ **Name Server IP Address**, tapez l'*Adresse IP*. Cliquez sur **Add** —> **OK**.
Supprimer un serveur de noms	**smit rmnamerslv**	éditez **/etc/resolv.conf** Remarques :	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **DNS**. Sélectionnez un serveur de nom dans **Name server to search**. Cliquez sur **Delete** —> **OK**.
Activer/Réactiver la résolution DNS	**smit stnamerslv**		Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **DNS**. Cochez la case **Enable domain name resolution using Domain Name Service (DNS)**. Cliquez sur **OK**.

Désactiver la résolution DNS	**smit spnamerslv**		Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **DNS**. Décochez la case **Enable domain name resolution using Domain Name Service (DNS)**. Cliquez sur **OK**.
Modifier/Afficher le domaine	**smit mkdomain**	éditez **/etc/resolv.conf** Remarques :	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **DNS**. —> **Domain name to search**. Cliquez sur **Add** —> **OK**.
Supprimer un domaine	**smit rmdomain**	éditez **/etc/resolv.conf** Remarques :	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **DNS**. Sélectionnez un domaine dans la liste **Domain search list**. Cliquez sur **Delete** —> **OK**.

Configuration de zones dynamiques sur le serveur de noms DNS

La commande **named** autorise les mises à jour dynamiques. La base de données nommée et les fichiers de configuration doivent être configurés pour permettre aux machines clientes d'émettre des mises à jour. Une zone peut être dynamique ou statique. La zone par défaut est statique.

Pour rendre une zone dynamique, il faut ajouter le mot clé **allow–update** à la strophe correspondante du fichier **/etc/named.conf** file. Ce mot clé précise la liste de correspondances d'adresses Internet définissant les hôtes autorisés à soumettre des mises à jour. Pour en savoir plus et examiner un exemple de fichier de configuration, reportez-vous à la section "named.boot File Format for TCP/IP" dans le manuel *AIX 5L Version 5.2 Files Reference*. Dans l'exemple ci–dessous, la mise à jour de la zone dynamique est autorisée à tous les hôtes :

```
zone "abc.aus.century.com" IN {
     type master;
     file "named.abc.data";
     allow-update { any; };
 };
```

Sur une zone dynamique, trois modes de sécurité peuvent être définis :

Non sécurisé	N'importe qui peut, à tout moment, mettre à jour les informations de la zone. **Attention :** il est déconseillé d'opter pour ce mode. Des données risquent d'être perdues ou interceptées, et l'utilisateur frustré. Il convient au minimum de limiter la mise à jour d'une zone non sécurisée ("unsecured") à des adresses Internet spécifiques.
Contrôlé	Autorise la création d'informations et le remplacement de données existantes. C'est sans doute le mode le plus adapté à un environnement de transition sécurisé. Ce mode requiert également que les données entrantes soient horodatées et munies d'une signature à clé.
Pré–securisé	Impose que les mises à jour remplacent les informations existantes par des informations similaires. Ne permet pas de créer de nouvelles informations. Ce mode requiert également que les données entrantes soient horodatées et munies d'une signature à clé.

Par défaut, une zone dynamique se trouve en mode non sécurisé. Pour utiliser l'un des autres modes, tapez **controlled** ou **presecured** après le mot de passe **update–security** dans la zone de strophe du fichier **/etc/named.conf** file. Cela indique au serveur **named** le niveau de sécurité à utiliser pour cette zone. Par exemple :

```
zone "abc.aus.century.com" IN {
     type master;
     file "named.abc.data";
     allow-update { any; };
     update-security controlled;
 };
```

Une fois le mode sélectionné, les fichiers de données doivent être amenés au niveau de sécurité choisi. En mode non sécurisé, les fichiers de données sont utilisés tels quels. En mode contrôlé ou pré–sécurisé, vous devez créer un ensemble de paires de clés entre noms de serveur maîtres et hôtes pour chaque nom de la zone. Utilisez pour cela la commande **nsupdate** avec l'option **–g**. Cette commande génère la paire de clés, une privée et une publique. Ces clés sont nécessaires pour authentifier les mises à jour. Après avoir créé toutes les clés pour la liste de noms de zones, il faut les ajouter au fichier de données. Le format de clé (KEY) est le suivant :

Index ttl Class Type KeyFlags Protocol Algorithm KeyData

où :

Index	Nom référençant les données de la zone.
ttl	ttl ("time to live") des données. Ce champ est facultatif.
Classe	Classe des données. Dépend de la zone, mais généralement IN.
Type	Type de l'enregistrement. Dans ce cas, le type est KEY.
IndicClé	Informations sur la clé. En général, l'enregistrement de clé pour un hôte est sous la forme 0x0000. Le code 0x0100 définit l'enregistrement de clé associé au nom de zone.
Protocole	Protocole à utiliser. Pour le moment, il n'y en a qu'un, `0`.

Algorithme Algorithme de la clé. Pour le moment, il n'y en a qu'un, `1`. Cette méthode est celle de l'authentification privé/public.

DonnéesClé Clé exprimée en base 64. La commande **nsupdate** génère les deux clés (publique et privée) en base 64. Dans le fichier de sortie, la clé publique apparaît en dernier.

Exemple

Pour garantir la sécurité d'un nom d'hôte dans une zone dynamique, il faut ajouter au fichier de zone une ligne du type ci–dessous pour la zone contenant ce nom :

```
bears    4660    IN    KEY    0x0000    0    1    AQOtg......
```

Dans cet exemple, `bears` est doté d'un enregistrement KEY défini : toute personne souhaitant mettre à jour `bears` doit signer sa mise à jour avec la clé privée correspondant à la clé publique enregistrée dans la base de données. Pour que la commande **nsupdate** agisse, cette clé privée doit figurer dans un fichier de clé chez le client (fichier **/etc/keyfile** par défaut). Son format est le suivant :

hostname *mastername* *base64* *key*

Une entrée similaire KEY doit se trouver dans la section de définition de la zone. **La clé de zone est obligatoire en mode pré–sécurisé ou contrôlé : sans clé, le mode est considéré non sécurisé.** L'exemple `bears` précédent montre comment procéder, mais l'utilisation de clé privée revient à l'administrateur qui utilise la commande **nsupdate** en mode administrateur.

1. Pour générer une paire de clés avec la commande **nsupdate**, entrez :

   nsupdate -g -h *NomZone* **-p** *NomServeur* **-k** *FichierCléAdmin*

   Une clé est générée pour la zone. Dans cet exemple, **nsupdate** est lié à **nsupdate4**, en tapant ce qui suit :

```
ln -fs /usr/sbin/nsupdate4 /usr/sbin/nsupdate
```

2. Placez la dernière clé de la paire au début de la section relative à la zone, comme suit :

```
IN     KEY     0x0100  0  1   Key
```

   L'entrée du fichier **named.abc.data** est la suivante :

```
$TTL 3h    ;3 hour

 @ IN    SOA     venus.abc.aus.century.com.
gail.zeus.abc.aus.century.com. (
                1       ;serial
                3600    ;refresh
                600     ;retry
                3600000 ;expire
                86400   ;negative caching TTL
 )
         IN      NS      venus.abc.aus.century.com.
         IN      KEY     0x0100  0 1
AQPlwHmIQeZzRk6Q/nQYhs3xwnhfTgF/8YlBVzKSoKxVKPNLINnYW0mB7attTcfhHaZZcZr4u
/vDNikKnhnZwgn/
 venus   IN      A       192.9.201.1
 earth   IN      A       192.9.201.5
 mars    IN      A       192.9.201.3
```

3. La zone est maintenant prête à être chargée en régénérant le serveur de noms. Placez FichierCléAdmin sur le client ou le serveur DHCP qui met la zone à jour. La clé de zone contenue dans FichierCléAdmin peut être utilisée pour appliquer des mises à jour et des opérations de maintenance au serveur de noms.

BIND 9

BIND 9 offre les deux mesures de sécurité suivantes pour **named**:

- Transaction Signatures (TSIG) page 4-85
- Signature (SIG) page 4-87

Le serveur de noms avec BIND 9, par défaut, ne permet pas les mises à jour dynamiques dans les zones d'autorité, comme dans BIND 8.

Transaction Signatures (TSIG)

BIND 9 prend en charge TSIG pour l'administration de serveur à serveur. Ceci comprend les messages de transfert de zones, de notification et d'interrogation récursive. TSIG est également utile pour les mises à jour dynamiques. Un serveur principal pour une zone dynamique doit utiliser le contrôle d'accès pour contrôler les mises à jour, mais le contrôle d'accès IP est insuffisant.

En utilisant un chiffrement de base de clé à la place de la méthode actuelle des listes de contrôle d'accès, TSIG permet de restreindre les utilisateurs autorisés à mettre à jour les zones dynamiques. Contrairement à la méthode ACL (Access Control List) de mises à jour dynamiques, la clé TSIG peut être distribués aux autres auteurs de mises à jour sans qu'il soit nécessaire de modifier les fichiers de configuration sur le serveur de noms, ce qui signifie qu'il n'est pas nécessaire que le serveur de noms relise les fichiers de configuration.

Il est important de noter que BIND 9 ne dispose pas de tous les mots–clés existant dans BIND 8. Dans cet exemple, nous utilisons la configuration maître simple de BIND 8.

Remarque : Pour utiliser named 9, vous devez relier la liaison symbolique au démon **named** à **named9**, et **nsupdate** à **nsupdate9** en exécutant les commandes suivantes :

1. `ln –fs /usr/sbin/named9 /usr/sbin/named`
2. `ln –fs /usr/sbin/nsupdate9 /usr/sbin/nsupdate`

1. Générez la clé à l'aide de la commande **dnssec–keygen** :

   ```
   dnssec-keygen –a HMAC–MD5 –b 128 –n HOST clé
   ```

   – `HMAC–MD5` est l'algorithme de chiffrement

   – `128` est la longueur de la clé à utiliser (ou nombre de bits)

   – `HOST`: `HOST` est le mot–clé TSIG utilisé pour générer une clé hôte pour un chiffrement de clé partagé.

   La commande

   ```
   dnssec-keygen –a HMAC–MD5 –b 128 –n HOST venus–batman.abc.aus.century.com
   ```

   génère deux fichiers de clé, comme suit :

   ```
   Kvenus–batman.abc.aus.century.com.+157+35215.key
    Kvenus–batman.abc.aus.century.com.+157+35215.private
   ```

   – `157` est l'algorithme utilisé (HMAC–MD5)

   – `35215` est l'empreinte, ce qui est utile dans DNNSEC car plusieurs clés par zone sont autorisées

2. Ajoutez l'entrée à named.conf sur le serveur de noms maître :

   ```
   // Clé TSIG
    key venus–batman.abc.aus.century.com. {
            algorithm hmac–md5;
            secret "+UWSvbpxHWFdNwEAdy1Ktw==";
    };
   ```

   En supposant que HMAC–MD5 est utilisé, les deux fichiers de clé contiennent la clé partagée, qui est stockée en tant que dernière entrée des fichiers. Trouvez un moyen

sécurisé de copier la clé secrète partagée sur le client. Vous n'avez pas besoin de copier le fichier de clé, uniquement la clé secrète partagée.

Ce qui suit est l'entrée du fichier **Kvenus–batman.abc.aus.century.com.+157+35215.private**:

```
Format-clé-privée : v1.2
 Algorithme : 157 (HMAC_MD5)
Clé : +UWSvbpxHWFdNwEAdy1Ktw==
```

Vous trouverez ci–après un exemple du fichier **named.conf** du serveur de noms maître. La zone `abc.aus.century.com` permet le transfert de zones et les mises à jour dynamiques uniquement sur les serveurs ayant la clé `venus-batman.abc.aus.century.com`. Procédez de même dans la zone inverse, pour laquelle les auteurs de mises à jour doivent avoir la clé partagée.

```
// Clé TSIG
 key venus-batman.abc.aus.century.com. {
         algorithm hmac-md5;
         secret "+UWSvbpxHWFdNwEAdy1Ktw==";
 };

 options {
         directory "/usr/local/domain";
 };

 zone "abc.aus.century.com" in {
         type master;
         file "named.abc.data";
         allow-transfer { key venus-batman.abc.aus.century.com.;};
         allow-update{ key venus-batman.abc.aus.century.com.; };
 };
```

Comme les transferts de zone sont restreints à ceux qui ont une clé, le fichier **named.conf** du serveur de noms doit aussi être édité. Toutes les demande transmises à 192.9.201.1(venus.abc.aus.century.com) sont signées par une clé. Le nom de la clé (venus–batman.abc.aus.century.com.) doit correspondre à ceux des serveurs qui les utilisent.

Vous trouverez ci–après un exemple du fichier **named.conf** du serveur de noms esclave :

```
// Clé TSIG
 key venus-batman.abc.aus.century.com. {
         algorithm hmac-md5;
         secret "+UWSvbpxHWFdNwEAdy1Ktw==";
 };

 server 192.9.201.1{
         keys { venus-batman.abc.aus.century.com.;};
 };

 options {
         directory "/usr/local/domain";
 };

 zone "abc.aus.century.com" IN {
    type slave;
     file "named.abc.data.bak";
     masters { 192.9.201.1; };
 };
```

Signature (SIG)

BIND 9 prend en partie en charge les signatures de transaction DNSSEC SIG comme indiqué dans RFC 2535. SIG utilise les clés publiques et privées pour authentifier les messages.

Les enregistrements SIG permettent aux administrateurs de signer leurs données de zone afin de les authentifier.

Sécurisation de la zone racine

Supposons que les autres serveurs de noms sur Internet n'utilisent pas BIND 9, et que vous vouliez sécuriser vos données de zone et permettre aux autres serveurs de vérifier vos données de zone. Vous souhaitez indiquer que votre zone (dans notre cas `aus.century.com`) est une racine sécurisée et valide toutes les données de zone sécurisées sous celle–ci.

1. Générez la clé à l'aide de la commande **dnssec–keygen** :

   ```
   dnssec-keygen -a RSA -b 512 -r /usr/sbin/named -n ZONE aus.century.com.
   ```

   Remarque : Le chiffrement RSA peut être utilisé comme l'algorithme de génération de la clé si OpenSSL est installé, mais vous devez d'abord relier la bibliothèque DNS à une bibliothèque DNS sécurisée en exécutant la commande suivante :

   ```
   ln -fs /usr/lib/libdns_secure.a /usr/lib/libdns.a
   ```

   – ZONE: ZONE est le mot–clé DNSSEC utilisé pour générer des clés de zones pour le chiffrement de clé privée/publique.

   – L'indicateur `r` désigne un périphérique aléatoire.

2. Ajoutez l'entrée de clé publique comme dans le fichier **named.conf**. L'entrée utilisée dans l'exemple est indiquée ci–après. Le contenu du fichier de clé **Kaus.century.com.+001+03254.key** est indiqué ci–dessous.

   ```
   abc.aus.century.com. IN KEY 256 3 1
   AQOnfGEAg0xpzSdNRe7KePq3Dl4NqQiq7HkwKl6TygUfaw6vz6ldmauB4UQFcGKOyL68/Zv5Z
   nEvyB1fMTAaDLYz
   ```

   La clé publique est contenue dans le fichier **Kzonename.+algor.+fingerprint.key**, ou dans notre exemple `Kaus.century.com.+001+03254.key`. Vous devez supprimer la classe IN et taper KEY et mettre la clé entre guillemets. Lorsque vous ajoutez cette entrée au fichier **/etc/named.conf** et régénérez le serveur de noms, la zone `aus.century.com` est une racine sécurisée.

```
trusted-keys {
         aus.century.com. 256 3 1
"AQOnfGEAg0xpzSdNRe7KePq3Dl4NqQiq7HkwKl6Tyg
 Ufaw6vz6ldmauB 4UQFcGKOyL68/Zv5ZnEvyB1fMTAaDLYz";
 };
 options {
         directory "/usr/local/domain";
 };

 zone "abc.aus.century.com" in {
         type master;
         file "named.abc.data.signed";
         allow-update{192.9.201.1;};
 };
```

Application de la chaîne de confiance

Une fois votre racine sécurisée, vous pouvez sécuriser le reste de vos zones enfant. Dans ce cas, nous voulons sécuriser la zone `abc.aus.century.com`. Procédez comme suit pour sécuriser vos zones enfants restantes :

1. Générez les paires de clé à l'aide de la commande **dnssec–keygen** :

   ```
   dnssec-keygen -a RSA -b 512 -r /usr/sbin/named -n ZONE
   abc.aus.century.com.
   ```

– L'indicateur `r` désigne un fichier d'entrée aléatoire.

2. Créez un fichier de clé en exécutant la commande **dnssec–makekeyset** :

```
dnssec-makekeyset -t 172800   Kabc.aus.century.com.+001+11515.key
 où  Kabc.aus.century.com.+001+03254.key est votre propre clé publique.
```

   Ceci crée un fichier de clé appelé **keyset–abc.aus.century.com**.

3. Envoyez ce fichier de clé à la zone parente pour le faire signer. Dans cet exemple, notre zone parente est la zone racine sécurisée `aus.century.com`.

4. Le parent doit signer la clé avec sa clé privée.

```
dnssec-signkey keyset-abc.aus.century.com.
Kaus.century.com.+001+03254.private
```

   Ceci génère un fichier appelé **signedkey–abc.aus.century.com**, et le parent doit renvoyer ce fichier à la zone enfant.

5. Sur le serveur de noms enfant de la zone `abc.aus.century.com`, ajoutez `$INCLUDE Kabc.aus.century.com.+001+11515.key` au fichier de zone simple `named.abc.data`. Souvenez–vous de placer le fichier **signedkey–abc.aus.century.com** dans le même emplacement que le fichier de zone **named.abc.data**. Lorsque la zone est signée dans l'étape suivante, le programme sait qu'il doit inclure **signedkey–abc.aus.century.com**, qui a été reçue du parent.

```
$TTL 3h    ;3 hour

 @ IN    SOA     venus.abc.aus.century.com.
gail.zeus.abc.aus.century.com.  (
                1       ;serial
                3600    ;refresh
                600     ;retry
                3600000 ;expire
                86400   ;negative caching TTL
 )
 $INCLUDE Kabc.aus.century.com.+001+03254.key
```

6. Signez la zone à l'aide de la commande **dnssec–signzone** :

```
dnssec-signzone -o abc.aus.century.com. named.abc.data
```

7. Modifiez le fichier **named.conf** dans la zone enfant `abc.aus.century.com` pour utiliser le nouveau fichier de zone signé (`named.abc.data.signed`). Par exemple :

```
options {
         directory "/usr/local/domain";
 };

 zone "abc.aus.century.com" in {
         type master;
         file "named.abc.data.signed";
         allow-update{192.9.201.1;};
 };
```

8. Régénérez le serveur de noms.

Pour plus d'informations sur la résolution des problèmes, reportez–vous à Problèmes de résolution des noms page 4-225.

Planification et configuration pour la résolution de noms LDAP (Schéma de répertoire SecureWay)

LDAP (Lightweight Directory Access Protocol) est un standard du marché qui définit une méthode d'accès et de mise à jour des informations d'un répertoire. Un schéma LDAP définit les règles de classement des données. La classe d'objet **ibm–HostTable**, contenue dans le schéma de répertoire SecureWay Directory, peut être utilisée pour stocker l'équivalence entre le nom et l'adresse Internet pour chaque hôte du système.

La classe d'objet **ibm–HostTable** est définit comme suit :

```
Nom de la classe d'objets :     ibm-HostTable
Description :           Entrée de la table des systèmes hôte regroupant
                      des noms hôte pour des mappages d'adresses IP.
OID :                   TBD
RDN :                   ipAddress
Classe d'objet supérieure : top
Attributs nécessaires :   hôte, ipAddress
Attributs optionnels :    ibm-hostAlias, ipAddressType, description
```

Définitions des attributs :

```
Attribut :        ipAddress
Description :     Adresses IP des noms hôtes de la Table des systèmes hôte
OID :             TBD
Syntaxe :         caseIgnoreString
Longueur :        256
Valeur unique :   Yes
Attribut :        ibm-hostAlias
Description :     Alias de l'hôte dans la table des systèmes hôte
OID :             TBD
Syntaxe :         caseIgnoreString
Longueur :        256
Valeur unique :   Valeur multiple
Attribut :        ipAddressType
Description :     Famille d'adresses d'une adresse IP (1=IPv4, 2=IPv6)
OID :             TBD
Syntaxe :         Entier
Longueur :        11
Valeur unique :   Yes
Attribut :        host
Description :     Nom d'hôte d'un système.
OID :             1.13.18.0.2.4.486
Syntaxe :         caseIgnoreString
Longueur :        256
Valeur unique :   Valeur multiple
Attribut :        description
Description :     Commentaires sur un objet de répertoire.
OID :             2.5.4.13
Syntaxe :         caseIgnoreString
Longueur :        1024
Valeur unique :   Valeur multiple
```

Utilisez la procédure suivante pour configurer le serveur LDAP conformément au schéma SecureWay Directory, de façon à stocker l'équivalence entre les noms et les adresses Internet :

1. Ajoutez un suffixe au serveur LDAP. Le suffixe est le point de départ de la base de données des hôtes. Par exemple, "cn=hosts". Utilisez pour cela l'utilitaire IBM SecureWay Directory Server Administration.
2. Créez un fichier LDIF (Data Interchange Format) LDAP : Vous pouvez le faire manuellement ou à l'aide de la commande **hosts2ldif**, qui crée un fichier LDIF à partir du fichier **/etc/hosts**. Reportez–vous à **hosts2ldif Command** dans le manuel *AIX 5L Version 5.2 Commands Reference* pour plus d'informations. Voici un exemple de fichier LDIF :

```
dn: cn=hosts
 objectclass: top
 objectclass: container
 cn: hosts
 dn: ipAddress=1.1.1.1, cn=hosts
 host: test
 ipAddress: 1.1.1.1
 objectclass: ibm-HostTable
 ipAddressType: 1
 ibm-hostAlias: e-test
 ibm-hostAlias: test.austin.ibm.com
 description: first ethernet interface
 dn: ipAddress=fe80::dead, cn=hosts
 host: test
 ipAddress: fe80::dead
 objectclass: ibm-HostTable
 ipAddressType: 2
 ibm-hostAlias: test-ll
 ibm-hostAlias: test-ll.austin.ibm.com
 description: v6 link level interface
```

3. Importez les données du répertoire d'hôtes à partir du fichier LDIF du serveur LDAP. Pour cela, utilisez la commande **ldif2db** ou l'outil Web SecureWay Directory Server Administration.

Configurez le client pour qu'il accède à la base de données des hôtes sur le serveur LDAP, via le mécanisme LDAP, en procédant comme suit :

1. Créez le fichier **/etc/resolv.ldap** Pour en savoir plus et disposer d'un exemple de fichier **resolv.ldap**, reportez–vous à la section resolv.ldap File Format for TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*.
2. Modifiez le nom de résolution par défaut avec la variable d'environnement **NSORDER**, le fichier **/etc/netsvc.conf** ou le fichier **/etc/irs.conf**. Pour plus de détails, reportez–vous à netsvc.conf File Format for TCP/IP, ou à irs.conf File Format for TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*.

Bien qu'il soit toujours pris en charge, il n'est plus conseillé d'utiliser le mécanisme `ldap` . Le mécanisme `ldap` existant fonctionne avec SecureWay Directory Schema. AIX 5.2 propose le nouveau mécanisme d'attribution des noms, `nis_ldap` (NIS_LDAP), qui fonctionne avec le schéma RFC 2307. Il est conseillé d'utiliser le mécanisme `nis_ldap` à la place du mécanisme `ldap` . Pour plus d'informations sur la résolution de noms `nis_ldap` , reportez–vous à P Planification et configuration pour la résolution de noms NIS_LDAP (Schéma de répertoire RFC 2307) page 4-90 .

Planification et configuration pour la résolution de noms NIS_LDAP (Schéma RFC 2307)

AIX 5.2 proopse un nouveau mécanisme d'attribution de noms appelé NIS_LDAP. La différence entre le mécanisme LDAP existant et ce nouveau mécanisme NIS_LDAP est lié au schéma LDAP (le groupe des attributs et des classes d'objets qui déterminent la façon dont les attributs sont regroupés pour décrire une entité). Le mécanisme LDAP existant fonctionne avec le serveur LDAP compatible avec le schéma SecureWay et prend en charge uniquement le service d'attribution de noms hôte. Le mécanisme NIS_LDAP fonctionne avec le serveur LDAP compatible avec le schéma RFC 2307, et prend en charge tous les services NIS : utilisateurs et groupes, hôtes, services, protocoles, réseaux et groupe réseau. RFC 2307 définit un ensemble d'attributs et de classes d'objets qui peut être utilisé pour décrire les services d'informations réseau, notamment les utilisateurs et les groupes.

Pour configurer le serveur LDAP, vous devez configurer le serveur LDAP et migrer les données requises vers le serveur.

1. Pour configurer un serveur, utilisez la commande **mksecldap**. Le mécanisme `nis_ldap` fonctionne uniquement avec le schéma RFC 2307. Lors de la configuration du serveur

LDAP, la commande **mksecldap** devrait être appelée avec l'option `-S rfc2307` ou `-S rfc2307aix` (pas l'option `-S aix` qui spécifie le schéma SecureWay Directory). Par défaut, la commande **mksecldap** migre les utilisateurs et les groupes définis dans local system sur le serveur LDAP. Si vous souhaitez désactiver cette migration, utilisez l'option `-u NONE` .

```
mksecldap -s -a cn=admin -p adminpwd -S rfc2307aix
```

Le serveur LDAP se voit attribuer `cn=admin` comme DN administrateur et `adminpwd` comme mot de passe. Le suffixe par défaut, `cn=aixdata`, est également ajouté au fichier **/etc/slapd32.conf**, le fichier de configuration du serveur LDAP.

Par défaut, la commande **mksecldap** migre les utilisateurs et les groupes définis dans le système local sur le serveur LDAP. Pour désactiver cette migration, utilisez l'option `-u NONE` , qui empêche la migration des utilisateurs et des groupes locaux sur le serveur LDAP. Vous ne pouvez ainsi ajouter les utilisateurs et les groupes NIS que plus tard.

```
mksecldap -s -a cn=admin -p adminpwd -u NONE
```

Pour plus de détails sur la commande **mksecldap**, reportez–vous à la description de la commande dans le manuel *AIX 5L Version 5.2 Commands Reference*.

2. Migrez les données NIS. Utilisez la commande **nistoldif** du serveur NIS pour migrer les équivalences NIS vers le serveur LDAP. La commande **nistoldif** peut aussi être utilisée pour migrer les données des fichiers plats.

   Exécutez la commande **nistoldif** sur un système contenant des données NIS qui doivent être migrées vers le serveur LDAP.

   ```
   nistoldif -h server1.ibm.com -a cn=admin -p adminpwd -d cn=aixdata
   ```

   Ceci permet de faire migrer les équivalences NIS depuis le local system sur le serveur LDAP, `server1.ibm.com`. Les données NIS sont placées sous le DN`cn=aixdata` . Vous pouvez aussi exécuter la commande **nistoldif** pour migrer les données des fichiers plats de n'importe quel système vers le serveur LDAP. Les fichiers plats seront utilisés pour toutes les équivalences manquantes du serveur NIS.

   Pour plus de détails sur la commande **nistoldif**, reportez–vous à la description de la commande dans le manuel *AIX 5L Version 5.2 Commands Reference*.

   Remarque : Les noms sont représentés par l'attribut `cn` du serveur LDAP. L'attribut `cn` défini par RFC 2307 ne tient pas compte de la casse. Les noms différenciés uniquement par la casse sont fusionnés sur le serveur. Les équivalences exactes ne tiennent pas non plus compte de la casse. Les recherches effectuées sur `TCP`, `tcp`, ou `Tcp` renverraient toutes l'entrée de protocole de TCP.

Pour configurer le client LDAP afin qu'il accède aux noms du serveur LDAP, exécutez la commande **mksecldap** avec les options de configuration du client.

1. La commande **mksecldap** enregistre le nom de serveur LDAP, le port, admindn, le mot de passe et basedn dans le fichier **/etc/security/ldap/ldap.cfg**, qui est lu par le démon **secldapclntd** lors du démarrage. La commande **mksecldap** démarre le démon **secldapclntd** automatiquement, si la configuration réussit.

   Pour plus d'informations sur le fichier **/etc/security/ldap/ldap.cfg**, consultez le manuel *AIX 5L Version 5.2 Files Reference*. Pour plus d'informations sur le démon **secldapclntd**, consultez le manuel *AIX 5L Version 5.2 Commands Reference*.

2. La commande **mksecldap** ajoute le mécanisme `nis_ldap` au fichier **/etc/netsvc.conf** et **/etc/irs.conf** afin de diriger la résolution de noms vers LDAP. Vous pouvez aussi définir la variable d'environnement **NSORDER** en tant que `nis_ldap` pour utiliser la résolution de noms NIS_LDAP.

   ```
   mksecldap -c -a cn=admin -p adminpwd -h server1.ibm.com
   ```

   Ceci configure le local system afin qu'il utilise le serveur LDAP `server1.ibm.com` . Le DN et le mot de passe administrateur du serveur LDAP doivent être fournis au client

pour lui permettre de s'authentifier. Les fichiers **/etc/netsvc.conf** et **/etc/irs.conf** sont mis à jour afin que la résolution d'attribution de noms soit résolue via NIS_LDAP.

Pour plus d'informations, consultez le format de fichier **/etc/netsvc.conf** pour TCP/IP ou **/etc/irs.conf** pour TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*.

3. La résolution des noms pour les utilisateurs et les groupes n'est pas contrôlée par les fichiers **/etc/netsvc.conf** ou **/etc/irs.conf**, mais par le fichier **/etc/security/user**. Pour permettre à un utilisateur LDAP de se connecter à un système AIX, définissez les variables `SYSTEM` et `registry` de l'utilisateur en tant que `LDAP` dans le fichier **/etc/security/user** de ce système client. Vous pouvez effectuer cette opération avec la commande **chuser**.

   ```
   chuser -R LDAP SYSTEM=LDAP registry=LDAP foo
   ```

   Vous pouvez configurer votre système afin d'autoriser tous les utilisateurs LDAP à se connecter à un système. Pour ce faire, éditez le fichier **/etc/security/user**. Ajoutez `registry = files` à la strophe racine. Ajoutez ensuite `SYSTEM = LDAP` et `registry = LDAP` à la strophe par défaut.

   Pour plus d'informations sur l'authentification utilisateur, reportez–vous à la section LDAP Exploitation of the Security Subsystem dans le manuel *AIX 5L Version 5.2 Security Guide*.

Affectation des adresses et paramètres TCP/IP - Protocole DHCP

Le protocole TCP/IP permet la communication entre machines disposant d'adresses configurées. L'affectation des adresses et la distribution des paramètres pour toutes les machines du réseau est une des tâches incombant à l'administrateur de réseau. Généralement, ce processus consiste pour l'administrateur à imposer une configuration à chaque utilisateur, tout en permettant à l'utilisateur de configurer sa propre machine. Toutefois, des erreurs de configuration ou des malentendus peuvent générer des appels de service que l'administrateur doit traiter individuellement. Le protocole DHCP (Dynamic Host Configuration Protocol) offre à l'administrateur une alternative, permettant d'exclure l'utilisateur final des problèmes de configuration et de gérer la configuration du réseau à partir d'un site central.

DHCP est un protocole de couche application qui permet à une machine du réseau, le *client*, d'obtenir du serveur une adresse IP ainsi que d'autres paramètres de configuration. Les informations sont obtenues au moyen d'un échange de paquets réalisé entre un démon sur le client et un autre sur le serveur. La plupart des systèmes d'exploitation proposent à l'heure actuelle un client DHCP dans leur module de base.

Pour obtenir une adresse, le démon du client DHCP (**dhcpcd**) diffuse un message de découverte DHCP, qui est reçu et traité par le serveur. (Il est possible de configurer à cet effet plusieurs serveurs sur le réseau.) S'il existe une adresse disponible pour ce client, un message DHCP de proposition est créé, contenant une adresse IP et d'autres options client. Le client reçoit cette proposition DHCP et la stocke en attendant d'autres propositions. Il choisit ensuite la meilleure et diffuse une demande DHCP indiquant au serveur la proposition retenue.

Tous les serveurs DHCP configurés reçoivent la demande. Chacun d'eux vérifie qu'il n'est pas le serveur demandé. Si ce n'est pas le cas, le serveur libère l'adresse qu'il a affecté au client. En revanche, le serveur demandé marque que l'adresse est affectée et renvoie un accusé de réception DHCP, qui finalise la transaction et attribue au client une adresse pour une durée (délai) définie par le serveur.

A échéance de la moitié de ce délai, le client tente de renouveler la réservation de son adresse en envoyant au serveur un paquet de *renouvellement*. Si le serveur accepte la demande, il envoie un accusé de réception DHCP. Si le client ne parvient pas à obtenir une réponse de son serveur attitré, il diffuse un paquet de nouvelle liaison DHCP afin de tenter de joindre le serveur (celui–ci a pu, par exemple, être déplacé d'un réseau à un autre). Si, à l'expiration de la totalité du délai, le client n'a pas renouvelé son adresse, l'interface est arrêtée et le processus recommence à zéro. Ce cycle permet d'éviter que plusieurs clients d'un réseau ne se voient affecter la même adresse.

Le serveur DHCP procède à l'attribution des adresses en fonction de *clés*. Les quatre clés les plus courantes sont le réseau, la classe, le fournisseur et l'ID de client. Le serveur se sert de ces clés pour obtenir une adresse et un jeu d'options de configuration qu'il envoie au client.

réseau
: Identifie le segment de réseau d'où est issu le paquet. La clé réseau permet au serveur de vérifier sa base de données d'adresses et d'attribuer une adresse correspondant au segment de réseau.

classe
: Elle est entièrement configurable par le client. Elle peut comprendre une adresse et des options. Cette clé peut être utilisée pour préciser la fonction d'une machine du réseau ou décrire le mode de regroupement des machines adopté à des fins administratives. Ainsi, l'administrateur du réseau peut créer une classe `netbios` contenant les options destinées aux clients NetBIOS ou une classe `comptabilité` représentant les machines du service Comptabilité qui ont besoin d'accéder à une imprimante spécifique.

fournisseur — Facilite l'identification du client à l'aide de sa plate–forme matérielle/logicielle (par exemple, un client Windows 95 ou un client OS/2 Warp).

ID client — Identifie le client, soit par le nom d'hôte de sa machine soit par son adresse de couche MAC (medium access control). L'ID client figure dans le fichier de configuration du démon **dhcpcd**. Par ailleurs, il peut être utilisé par le serveur pour transmettre des options à un client ou pour empêcher un client de recevoir des paramètres.

Ces clés peuvent figurer dans le fichier de configuration soit seules, soit en combinaison. Si un client fournit plusieurs clés et que plusieurs adresses peuvent être allouées, le choix porte sur une clé et le jeu d'options découle de la clé choisie en premier. Pour plus d'informations sur la sélection des clés et des adresses, reportez–vous à la section Configuration de DHCP, page 4-97.

Un agent relais est requis pour que les diffusions initiales du client puissent quitter le réseau local. Cet agent est appelé agent relais BOOTP. Ces agents assurent le relais des paquets DHCP et BOOTP.

Le serveur DHCP

A partir de AIX Version 4.3.1, le serveur DHCP a été divisé en trois grandes parties : une base de données, un moteur de protocole et un ensemble de routines de service, chaque partie disposant de ses propres informations de configuration.

La base de données DHCP

La base de données **db_file.dhcpo** permet d'effectuer le suivi des clients et des adresses et de contrôler les accès (par exemple, pour autoriser certains clients exclusivement à accéder à certains réseaux ou pour désactiver les clients BOOTP sur un réseau particulier). Les options sont également enregistrées dans la base de données d'où elles peuvent être extraites et distribuées aux clients. La base de données est implémentée sous la forme d'un objet pouvant être chargé de façon dynamique, ce qui facilite les mises à niveau et la maintenance du serveur.

A partir des informations du fichier de configuration, la base de données est amorcée et sa cohérence est vérifiée. Un ensemble de fichiers de points de contrôle met à jour la base de données et réduit le volume d'écritures vers le fichier de stockage principal. La base de données contient également des pools d'adresses et d'options, mais ceux–ci sont statiques et sont étudiés dans la section Configuration de DHCP, page 4-97.

Le fichier de stockage principal et sa copie de sauvegarde sont de simples fichiers ASCII qui peuvent, si nécessaire, être modifiés. Leur format est le suivant :

```
DF01
 "   ID CLIENT  " "   0.0.0.0  "   Etat      LeaseTimeStart
LeaseTimeDuration       LeaseTimeEnd
   "    Adresse IP serveur " " ID classe " "ID fournisseur" "Hôte " "Nom de
domaine "
 "   ID CLIENT  " "   0.0.0.0  "   Etat      LeaseTimeStart
LeaseTimeDuration       LeaseTimeEnd
   "   Adresse IP serveur  " "   ID classe " "   ID fournisseur  " "   Hôte
 " "   Nom de domaine  "
...
```

La première ligne indique la version du fichier : `DF01c`. Les lignes qui suivent définissent des enregistrements client. Le serveur procède à la lecture de la seconde ligne jusqu'à la fin du fichier. (Les paramètres entre guillemets doivent être indiqués entre guillemets.)

"CLIENT ID" — ID utilisé par le client pour se présenter au serveur.

"0.0.0.0" — est l'adresse IP actuellement attribuée au serveur DHCP. Si aucune adresse n'a été attribuée, "0.0.0.0" sera adopté par défaut.

State — Etat actuel du client. Le moteur de protocole DHCP contient le jeu de valeurs attribuables et les états sont gérés dans la base de données DHCP.

Le nombre en regard de *State* représente sa valeur. Les différents états possibles sont :

(1) FREE Représente les adresses qui sont disponibles. En général, les clients n'ont pas cet état, à moins qu'aucune adresse ne leur ait encore été attribuée. **dadmin** et la sortie de **lssrc** signalent pour cet état "Free".

(2) BOUND Indique que le client et l'adresse sont liés et que l'adresse a été attribuée au client il y a déjà un certain temps. **dadmin** et la sortie de **lssrc** indiquent pour cet état "Leased".

(3) EXPIRED Indique que le client et l'adresse sont liés, à titre d'information uniquement, de la même manière que l'état released. Cet état signale toutefois que le client a laissé son bail arriver à expiration. Une adresse arrivée à expiration est disponible et est réaffectée lorsque toutes les adresses libres sont indisponibles et avant que les adresses libérées ne soient réattribuées. **dadmin** et la sortie de **lssrc** indiquent pour cet état "Expired".

(4) RELEASED Indique que le client et l'adresse sont liés, à titre d'information uniquement. Le protocole DHCP conseille aux serveurs DHCP de gérer les informations concernant leurs clients précédents à des fins de référence ultérieure (principalement pour essayer de redonner à un client une adresse qu'il a déjà utilisée dans le passé). Cet état signale que le client a libéré l'adresse. Cette adresse peut donc être utilisée par d'autres clients si aucune autre adresse n'est disponible. **dadmin** et la sortie de **lssrc** indiquent pour cet état "Released".

(5) RESERVED
Indique qu'une liaison lâche existe entre le client et l'adresse. Le client a envoyé un message de découverte DHCP, auquel le serveur DHCP a répondu, et le client n'a pas encore répondu par une requête DHCP demandant cette adresse. **dadmin** et la sortie de **lssrc** indiquent pour cet état "Reserved".

(6) BAD Représente une adresse utilisée sur le réseau mais qui n'a pas été distribuée par le serveur DHCP. Cet état qualifie également les adresses qui ont été rejetées par les clients. Cet état ne s'applique pas à des clients. **dadmin** et la sortie de **lssrc** indiquent que cet état est "Utilisé" (Used) et "Mauvais" (Bad), respectivement.

LeaseTimeStart Début du bail actuel (en nombre de secondes écoulées depuis le 1er janvier 1970).

LeaseTimeDuration
Durée du bail (en secondes).

LeaseTimeEnd Utilise le même format que *LeaseTimeStart*, pour indiquer la fin du bail. Certaines options de configuration utilisent des valeurs différentes pour le début et la fin d'un bail et il est possible de substituer à ces valeurs des options du fichier de configuration. Reportez–vous à Syntaxe du fichier de serveur DHCP pour la base de données db_file, page 4-117.

"*Server IP Address*"
Adresse IP du serveur DHCP détenteur de cet enregistrement.

"*Class ID*"
"*Vendor ID*"*Host Name*"
"*Domain Name*"
Valeurs utilisées par le serveur pour déterminer les options qui sont envoyées au serveur (stockées sous la forme de chaînes entre guillemets). Ces paramètres permettent d'améliorer les performances, puisque les listes d'options peuvent être générées à l'avance pour ces clients au démarrage du serveur DHCP.

Fichiers de points de contrôle

La syntaxe des fichiers de points de contrôle n'est pas spécifiée. En cas de panne du serveur, ou si vous devez l'arrêter sans avoir pu fermer normalement la base de données, le serveur peut utiliser les fichiers de points de contrôle et les fichiers de sauvegarde pour reconstruire une base de données correcte. La pire situation serait de perdre un client (si le client était en cours d'écriture dans le fichier de point de contrôle au moment de la panne). Les fichiers par défaut sont :

/etc/db_file.cr fonctionnement normal de la base de données

/etc/db_file.crbk
sauvegardes de la base de données

/etc/db_file.chkpt et **/etc/db_file.chkpt2**
fichiers de point de contrôle en alternance

Le serveur DHCP pour AIX Version 4.3.1 et ultérieures est du type enchaîné. Pour garantir un débit élevé, les opérations sur la base de données (y compris les opérations de sauvegarde) sont optimisées pour le type enchaîné. Lorsqu'une sauvegarde est demandée, le fichier de points de contrôle existant est remplacé par le fichier de points de contrôle suivant, le fichier de base de données existant est copié dans le fichier de secours et un nouveau fichier de sauvegarde est créé. Chaque enregistrement client est consigné et un bit est modifié afin d'indiquer que le client doit utiliser le nouveau fichier de points de contrôle pour la journalisation. Lorsque tous les enregistrements client sont pris en compte, la sauvegarde est fermée et les anciens fichiers de secours et de points de contrôle sont supprimés. De cette manière, les clients peuvent toujours être traités et, si l'enregistrement du client a été sauvegardé, les modifications s'inscrivent dans un nouveau fichier de sauvegarde ou un nouveau fichier de points de contrôle.

Le moteur de protocole DHCP

Pour AIX Version 4.3.1 et ultérieures, le moteur de protocole DHCP a été mis au niveau de la norme RFC 2131, mais reste compatible avec RFC 1541. (Le serveur peut également traiter des options définies dans RFC 2132.) Le moteur de protocole utilise la base de donnés pour déterminer quelles informations doivent être retournées au client.

La configuration des pools d'adresses fait intervenir certaines options qui affectent l'état de la machine. Par exemple, le serveur DHCP interroge (ping) les adresses avant de les attribuer. La durée d'attente de la réponse par le serveur peut désormais être configurée pour chaque pool d'adresses.

Opérations DHCP enchaînées

Le dernier élément du serveur DHCP est en fait un ensemble d'opérations qui permettent d'assurer la continuité des opérations. Comme le serveur DHCP est du type enchaîné, ces opérations sont en fait définies sous la forme de routines qui interviennent occasionnellement pour s'assurer du bon déroulement des opérations.

La première routine, ou routine **principale**, gère les requêtes SRC (par exemple **startsrc**, **stopsrc**, **lssrc**, **traceson** et **refresh**). Cette routine coordonne également toutes les opérations qui affectent toutes les routines et gère les signaux. Par exemple :

- A SIGHUP (–1) provoque un rafraîchissement de toutes les bases de données du fichier de configuration.
- A SIGTERM (–15) entraîne l'arrêt en douceur du serveur.

La routine suivante, **dadmin**, interface avec le programme client **dadmin** et le serveur DHCP. L'outil dadmin peut être utilisé pour obtenir des informations sur l'état de la base de données et la modifier, et évite de modifier manuellement les différents fichiers de la base de données. Les versions antérieures du serveur DHCP empêchaient l'attribution d'adresses aux clients lorsqu'une requête d'état était en cours. Grâce aux routines **dadmin** et **src**, le serveur est désormais en mesure de gérer les requêtes de services tout en continuant à traiter les requêtes des clients.

La routine suivante est **garbage** qui, à intervalles réguliers, nettoie la base de données, la sauvegarde, purge les clients ne possédant pas d'adresse et supprime les adresses réservées qui le sont depuis trop longtemps. Les intervalles peuvent être configurés (reportez–vous à la section Configuration de DHCP, page 4-97). Les autres routines correspondent à des processeurs de paquet. Leur nombre peut être configuré et il est de 10 par défaut. Chaque routine peut traiter une requête émise par un client DHCP. Le nombre de processeurs de paquets requis est fonction de la charge et de la machine. Si la machine assure d'autres services que DHCP, il n'est peut être pas très sage de lancer 500 routines.

Préparation de DHCP

Pour exploiter ce protocole, l'administrateur réseau doit configurer un serveur DHCP ainsi que les agents relais BOOTP sur les liaisons dépourvues de serveur DHCP. Une planification anticipée peut permettre de réduire la charge de DHCP sur le réseau. Par exemple, si vous configurez un seul serveur pour gérer tous les clients, tous les paquets doivent transiter par ce serveur. Si vous ne disposez que d'un routeur entre deux grands réseaux, il est plus sage de prévoir deux serveurs, un sur chaque liaison.

Un autre aspect à considérer est le fait que DHCP implique une trame de trafic. Par exemple, si vous définissez un délai par défaut inférieur à 2 jours et que vous arrêtez les machines pendant le week-end, le trafic DHCP connaîtra une pointe le lundi matin. Bien que le trafic DHCP ne constitue pas une charge supplémentaire considérable, il doit néanmoins être pris en compte au moment de décider du nombre et de l'emplacement des serveurs DHCP sur le réseau.

L'objectif de DHCP est de libérer le client de toute saisie une fois DHCP activé pour intégrer le client au réseau. Le client DHCP, `dhcpcd`, lit un fichier de configuration, **dhcpcd.ini**, qui contient des informations sur la journalisation ainsi que les paramètres requis pour démarrer. L'installation terminée, il vous faut sélectionner la méthode de configuration de TCP/IP : configuration minimale ou DHCP. Si vous optez pour DHCP, vous devez choisir une interface et vous pouvez spécifier des paramètres facultatifs. Pour l'interface, vous pouvez sélectionner le mot–clé **any**, qui indique à `dhcpcd` d'utiliser la première interface en état de fonctionnement qu'il rencontre. Cette méthode minimise la quantité d'entrées côté client.

Configuration de DHCP

Par défaut, la configuration du serveur DHCP est effectuée par la lecture du fichier **/etc/dhcpsd.cnf**, qui spécifie la base de données initiale d'adresses et d'options du serveur. Le serveur est lancé dans le fichier **/etc/rc.tcpip**, à partir de Web-based System Manager, de SMIT, ou à l'aide de commandes SRC. Vous pouvez configurer un client DHCP via Web-based System Manager, SMIT ou en éditant un fichier ASCII plat.

La configuration de DHCP constitue la tâche la plus délicate dans le cadre de l'utilisation de DHCP dans votre réseau. Vous devez d'abord déterminer le nombre de réseaux qui devront accueillir des clients DHCP. Chaque sous–réseau du réseau principal représente un pool d'adresses que le serveur DHCP doit ajouter à sa base de données. Par exemple :

```
database db_file
 {
   subnet 9.3.149.0 255.255.255.0
   {  option 3 9.3.149.1 # Passerelle par défaut que les clients de
ce réseau doivent utiliser
    option 6 9.3.149.2 # Serveur de noms que les clients de ce résea
u doivent utiliser
    }
   ... options ou autres conteneurs ajoutés ultérieurement
 }
```

L'exemple ci–dessus représente un sous–réseau, `9.3.149.0`, avec un masque de sous–réseau `255.255.255.0`. Toutes les adresses de ce sous–réseau, de 9.3.149.1 à 9.3.149.254, sont contenues dans le pool. Eventuellement, il est possible de spécifier un

intervalle à la fin de la ligne, ou d'inclure un intervalle ou une instruction d'exclusion dans le conteneur de sous–réseau. Pour plus d'informations sur les définitions et méthodes de configuration classiques, reportez–vous à Options connues du fichier de serveur DHCP, page 4-106.

La clause de base de données mentionnant `db_file` indique la méthode à utiliser pour le traitement de cette portion du fichier de configuration. Les commentaires sont introduits par le symbole #. Le texte placé entre le # initial et la fin de la ligne est ignoré par le serveur DHCP. Chaque ligne `option` est utilisée par le serveur pour indiquer au client ce qu'il doit faire. La section Options connues du fichier de serveur DHCP, page 4-106 décrit les options reconnues et prises en charge à l'heure actuelle. Pour savoir comment définir des options inconnues du serveur, reportez–vous à la section Syntaxe du fichier de serveur DHCP pour le fonctionnement général du serveur, page 4-113.

Si le serveur ne comprend pas comment analyser une option, il utilise des méthodes par défaut pour transmettre l'option au client. Ceci permet au serveur DHCP d'envoyer des options spécifiques à certains sites, qui ne sont pas définies dans les normes RFC, mais sont utilisables par certains clients ou certaines configurations de client.

Le fichier de configuration

Le fichier de configuration comprend une section d'adresses et une section de définition d'options, basées sur le concept des conteneurs, qui renferment les options, les modificateurs et, le cas échéant, d'autres conteneurs.

Un *conteneur* (qui est finalement une méthode de regroupement des options) fait appel à un identificateur pour classer les clients en plusieurs groupes. Les types de conteneur sont le sous–réseau, la classe, le fournisseur et le client. A l'heure actuelle, il n'existe pas de conteneur générique définissable par l'utilisateur. L'identificateur définit le client de manière unique, de sorte qu'il soit possible de suivre sa trace même s'il est déplacé vers un autre sous–réseau. Il est possible d'utiliser plusieurs types de conteneur pour définir les droits d'accès du client.

Les *options* sont les identificateurs qui sont retournés au client, par exemple la passerelle par défaut et l'adresse de DNS.

Les *modificateurs* sont des instructions isolées qui modifient l'aspect d'un conteneur, par exemple la valeur par défaut de la durée du bail.

Conteneurs

Lorsque le serveur DHCP reçoit une requête, le paquet est analysé et les clés d'identification permettent de déterminer les conteneurs, les options et les adresses à extraire.

L'exemple précédent présente un conteneur de sous–réseau. La clé d'identification est la position du client au sein du réseau. Si le client fait partie de ce réseau, alors il est intégré à ce conteneur.

Chaque type de conteneur utilise une option différente pour identifier les clients :

- Le conteneur sous–réseau utilise le champ giaddr ou l'adresse de l'interface réceptrice pour déterminer le sous–réseau d'origine du client.
- Le conteneur classe utilise la valeur de l'option 77 (User Site Class Identifier – identificateur de la classe du site utilisateur).
- Le conteneur fournisseur utilise la valeur de l'option 60 (Vendor Class Identifier – identificateur de la classe du fournisseur).
- Le conteneur client utilise la valeur de l'option 61 (Client Identifier – identificateur du client) pour les clients DHCP et le champ chaddr du paquet BOOTP pour les clients BOOTP.

Sauf pour les sous–réseaux, chaque conteneur accepte la spécification de la valeur de correspondance à l'aide d'expressions régulières.

A ces conteneurs, il faut ajouter un conteneur implicite, le conteneur *global*. Sauf spécification contraire ou refus explicite, les options et modificateurs sont placés dans le conteneur global. La plupart des conteneurs peuvent être inclus dans d'autres conteneurs, ce qui implique une certaine visibilité. Les conteneurs peuvent ou non être associés à des plages d'adresses. Tel est le cas, par nature, des sous–réseaux.

Les règles de base s'appliquant aux conteneurs et sous–conteneurs sont les suivantes :

- Tous les conteneurs sont valides au niveau général.
- Les sous–réseaux ne doivent jamais être inclus dans d'autres conteneurs.
- Des conteneurs restreints ne peuvent englober des conteneurs réguliers du même type. (Par exemple, un conteneur doté d'une option autorisant uniquement la classe `Comptabilité` ne peut receler un conteneur doté d'une option autorisant toutes les classes commençant par la lettre "c". Ceci n'est pas autorisé.)
- Les conteneurs client restreints ne peuvent englober de sous–conteneurs.

En tenant compte des règles ci–dessus, vous pouvez générer une hiérarchie de conteneurs qui répartissent les options en différents groupes pour des clients ou des ensembles de clients spécifiques.

Comment sont gérées les options et adresses lorsqu'un client correspond à plusieurs conteneurs ? Le serveur DHCP reçoit les messages, il transmet la requête à la base de données (fichier **db_file** en l'occurrence) et une liste de conteneurs est générée. La liste est organisée par ordre de profondeur et de priorité. La priorité se définit comme une hiérarchie implicite au sein des conteneurs. Les conteneurs stricts ont une priorité supérieure à celle des conteneurs réguliers. Les clients, les classes, les fournisseurs et enfin, les sous–réseaux sont triés, dans cet ordre, et à l'intérieur de chaque conteneur en fonction de leur profondeur. Ceci aboutit à une liste allant du plus spécifique au moins spécifique. Par exemple :

```
Sous-réseau 1
--Classe 1
--Client 1
Sous-réseau 2
--Classe 1
----Fournisseur 1
----Client 1
--Client 1
```

Cet exemple présente deux sous–réseaux, `Sous-réseau 1` et `Sous-réseau 2`. Il y a un nom de classe, `Classe 1`, un nom de fournisseur, `Fournisseur 1` et un nom de client, `Client 1`. `Classe 1` et `Client 1` sont définis en plusieurs endroits. Comme ils résident dans des conteneurs différents, leurs noms peuvent être identique mais leurs valeurs, différentes. Si `Client 1` envoie un message au serveur DHCP depuis `Sous-réseau 1` avec `Classe 1` spécifiée dans sa liste d'options, le serveur DHCP va générer le chemin de conteneur suivant :

```
Sous-réseau 1, Classe 1, Client 1
```

Le conteneur le plus spécifique apparaît en dernier. Pour obtenir une adresse, la liste est étudiée dans l'ordre inverse de la hiérarchie et la première adresse disponible est retenue. Ensuite, l'étude de la liste de poursuit en remontant dans la hiérarchie afin d'obtenir les options. Les options peuvent remplacer des valeurs précédentes, sauf si une option *deny* a été incluse dans le conteneur. Par ailleurs, puisque `Classe 1` et `Client 1` figurent dans `Sous-réseau 1`, ils sont ordonnés en fonction de la priorité de leur conteneur. Si le même client se trouve dans `Sous-réseau 2` et envoie le même message, la liste de conteneur générée sera :

`Sous-réseau 2, Classe 1, Client 1` (au niveau de `Sous-réseau 2`), `Client 1` (au niveau de `Classe 1`)

`Sous-réseau 2` apparaît en premier, suivi de `Classe 1` , puis de `Client 1` au niveau de `Sous-réseau 2` (car cette instruction client ne se trouve qu'à un niveau en dessous

dans la hiérarchie). Cette hiérarchie implique qu'un client correspondant à la première instruction client est moins spécifique que le client correspondant à `Client 1` de `Classe 1` au sein de `Sous-réseau 2`.

La priorité sélectionnée en fonction de la profondeur dans la hiérarchie prend le pas sur la priorité des conteneurs eux–mêmes. Par exemple, si le même client émet le même message, en précisant cette fois un identificateur de fournisseur, la liste de conteneur devient :

`Sous-réseau 2, Classe 1, Fournisseur 1, Client 1` (au niveau de `Sous-réseau 2`), `Client 1` (au niveau de `Classe 1`)

La priorité au niveau des conteneurs améliore les performances en matière de recherche car elle correspond à un concept général selon lequel les conteneurs client constituent le moyen le plus spécifique de définir un ou plusieurs clients. Le conteneur client contient des adresses plus spécifiques qu'un conteneur classe, lui–même plus spécifique qu'un conteneur fournisseur, le conteneur sous–réseau étant le moins spécifique de tous.

Adresses et plages d'adresses

Les plages d'adresses, obligatoires pour les conteneurs sous–réseau, peuvent être associées à tout type de conteneur. Chaque plage définie pour un conteneur doit être un sous–ensemble de la plage du conteneur parent et ne doit pas présenter de chevauchement avec la plage d'un autre conteneur. Par exemple, si une classe définie dans un sous–réseau est associée à une plage d'adresses, cette plage doit constituer un sous–ensemble des adresses de la plage du sous–réseau. En outre, le conteneur de la classe ne doit pas recouvrir, même partiellement, d'autres plages d'adresses au même niveau.

Les plages peuvent être définies sur la ligne du conteneur et modifiées au moyen d'instructions de plages et d'exclusion afin que des jeux d'adresse non contigus puissent être associés à un conteneur. Ainsi, si les dix premières adresses d'un sous–réseau sont disponibles, ainsi que les dix suivantes, le sous–réseau peut spécifier ces adresses par plage dans la clause de sous–réseau afin de réduire l'utilisation de la mémoire et les risques de collision d'adresses avec d'autres clients ne se trouvant pas dans les plages spécifiées.

Dès qu'une adresse est sélectionnée, tout conteneur suivant dans la liste contenant les plages d'adresses est retiré de la liste, avec ses enfants. Les options spécifiques au réseau dans les conteneurs supprimés ne sont pas valides si l'adresse n'est pas utilisée à partir de ce conteneur.

Options

Une fois la liste ponctionnée pour déterminer les adresses, un ensemble d'options est généré pour le client. Lors de ce processus de sélection, les nouvelles options remplacent les options précédemment sélectionnées, sauf si une clause *deny* est rencontrée, auquel cas l'option refusée est retirée de la liste envoyée au client. Cette méthode autorise les héritages à partir des conteneurs parents afin de réduire la quantité de données à spécifier.

Modificateurs

Les modificateurs sont des éléments qui modifient l'aspect de certains conteneurs, par exemple le type d'accès ou la durée du bail. Après avoir défini les pools d'options et d'adresses, réfléchissez aux modificateurs à ajouter aux conteneurs. Les plus courants sont **leasetimedefault**, **supportBootp** et **supportUnlistedclients**.

leasetimedefault
: Définit la durée pendant laquelle une adresse est louée à un client.

supportBootp
: Détermine si le serveur doit répondre aux clients BOOTP.

supportUnlistedclients
: Indique si un client doit être explicitement défini par une instruction de client pour recevoir une adresse. La valeur de supportUnlistedClients peut être

none (aucun) , **dhcp**, **bootp** ou **both (les deux)**. Vous pouvez ainsi restreindre l'accès des clients bootp et autoriser tous les clients DHCP à obtenir des adresses.

Pour connaître les autres modificateurs, reportez–vous à Syntaxe du fichier de serveur DHCP pour la base de données db_file.

Journalisation

Une fois les modificateurs sélectionnés, configurez la fonction de journalisation. Les paramètres de journalisation sont précisés dans un conteneur tel que la base de données, mais le mot de passe du conteneur est : **logging_info**. Au démarrage, il est conseillé d'activer le niveau de journalisation le plus élevé. En outre, il est préférable de configurer cette fonction préalablement à toute autre afin que les erreurs de configuration puissent être consignées après initialisation du sous–système de journalisation. Le mot–clé **logitem** active le niveau de journalisation ; si vous supprimez **logitem,** le niveau de journalisation sera désactivé. Les autres mots–clé concernant la journalisation permettent d'indiquer le nom du fichier journal, sa taille et le nombre de journaux utilisés en alternance.

Options spécifiques au serveur

Le dernier groupe de paramètres concerne les options spécifiques au serveur, et permet à l'utilisateur de contrôler le nombre de processeurs de paquets, la fréquence d'exécution des routines de nettoyage, etc.

Voici deux exemples d'options spécifiques au serveur :

reservedTime Indique pendant combien de temps une adresse doit rester à l'état réservé après l'envoi d'une OFFRE au client DHCP

reservedTimeInterval
Indique à quelle fréquence le serveur DHCP analyse les adresses pour vérifier si certaines ne sont pas à l'état réservé depuis une durée supérieure à celle définie par **reservedTime**.

Ces options sont pratiques si vous avez plusieurs clients qui diffusent des messages DISCOVER, mais qui n'envoient pas de message REQUEST ou que leur message REQUEST se perd sur le réseau. Ces paramètres permettent d'éviter la réservation indéfinie des adresses pour un client non conforme.

Une autre option particulièrement importante, **SaveInterval**, permet de définir la fréquence de sauvegarde. Toutes les options spécifiques au serveur sont abordées dans la section Syntaxe du fichier de serveur DHCP pour le fonctionnement général du serveur, avec les mots–clés de journalisation.

Considérations de performance

Vous n'êtes pas sans savoir que certains mots–clé de configuration ainsi que la structure du fichier de configuration ont une incidence sur l'utilisation de la mémoire et les performances du serveur DHCP.

Premièrement, il est possible d'éviter toute sollicitation excessive de la mémoire en appréhendant le modèle d'héritage des options des conteneurs parents vers les conteneurs enfants. Dans un environnement qui ne prend pas en charge les clients non répertoriés, l'administrateur doit expressément lister chaque client du fichier. Lorsque des options sont répertoriées pour chaque client en particulier, le serveur sollicite plus de capacité mémoire pour stocker cette structure de configuration arborescente que lorsque des options sont héritées d'un conteneur parent (conteneurs de sous–réseau, de réseau ou conteneurs globaux, par exemple). Par conséquent, l'administrateur doit vérifier la répétition ou non des options relatives au client au sein du fichier de configuration. Si tel est le cas, il doit décider si ces options peuvent ou non être définies dans le conteneur parent et partagées par l'ensemble des clients.

Deuxièmement, l'utilisation des entrées **logItem** INFO et TRACE entraîne la consignation de nombreux messages au cours du traitement de chaque message du client DHCP. L'ajout

d'une ligne au journal peut s'avérer une opération onéreuse. C'est pourquoi, la limitation du volume de journalisation améliore les performances du serveur DHCP. En cas de présomption d'erreur sur le serveur DHCP, la journalisation peut être dynamiquement réactivée à l'aide des commandes SRC traceson ou dadmin.

Troisièmement, la sélection d'une valeur **numprocessors** doit dépendre de la taille du réseau DHCP, du paramètre de configuration **pingTime db_file** et du délai de propagation type sur le réseau. Etant donné que chaque routine de processeur de paquet émet une requête d'écho ICMP pour vérifier l'état de l'adresse serveur avant de l'attribuer à un client, le délai de réponse affecte directement la durée de traitement d'un message DISCOVER. La routine de processeur de paquet se borne essentiellement à attendre une réponse ou le **pingTime**. Par conséquent, la réduction de la valeur **numprocessors** améliore le temps de réponse du serveur, et réduit par là–même le nombre de retransmissions par clients, sans pour autant sacrifier les avantages que présentent le ping inhérent à la conception du serveur.

Pour optimiser les performances, sélectionnez une valeur **pingTime** basée sur le délai de propagation des réseaux distants pris en charge par le serveur DHCP. Sélectionnez également **numprocessors** en fonction de la valeur **pingTime** et de la taille du réseau. La sélection d'une valeur trop basse peut entraîner l'arrêt de toutes les routines de traitement de paquet dans l'attente des réponses d'écho tandis que les messages client DHCP entrants sont mis en attente sur le port du serveur. Celui–ci traite alors les messages client par lots au lieu de les traiter en continu.

Une valeur sélectionnée trop petite peut causer l'arrêt du traitement de tous les paquets dans l'attente de Echo Responses, ce qui provoquerait le.

Afin d'éviter ce cas de figure, la valeur de **numprocessors** doit être supérieure au nombre prévu de messages DISCOVER pouvant être reçus dans un intervalle **pingTime** au cours d'une période de forte activité client sur le DHCP. Toutefois, ne définissez pas une valeur trop élevée pour **numprocessors** car la gestion de routines risquerait d'encombrer le noyau.

A titre d'exemple, les valeurs **numprocessors 5** et **pingTime 300** offrent de faibles performances dans un environnement pouvoir recevoir 10 messages DISCOVER par seconde. En effet, en cas de forte sollicitation, 5 messages seulement peuvent être traités toutes les 3 secondes. Cet environnement doit être configuré avec des valeurs se rapprochant de **numprocessors 20** et de **pingTime 80**.

Personnalisation d'un fichier de configuration

De nombreux administrateurs réseau ont à gérer des réseaux comprenant plusieurs types de clients : ainsi, on peut trouver dans le même réseau des ordinateurs exécutant différents systèmes d'exploitation, tels que Windows, OS/2, Java OS et UNIX. Chaque type de machine requiert des identificateurs de fournisseurs uniques (c'est ce champ qui permet d'indiquer le type de machine au serveur DHCP). Les clients Java et les machines Thin Client peuvent exiger des paramètres qui leur sont propres, par exemple bootfiles, et il est possible que vous deviez adapter les options de configuration en conséquence. En revanche, les ordinateurs Windows 95 ne vont pas gérer correctement les options spécifiques à Java. Il est donc possible d'encapsuler les options spécifiques à chaque machine au sein de son conteneur fournisseur.

Pour reprendre notre exemple, imaginez une tâche principale dédiée à certaines machines en fonction de leurs utilisateurs. Par exemple, le personnel de développement peut travailler sur des clients de ce système d'exploitation pour effectuer des travaux de programmation, le personnel du service marketing peut utiliser des clients OS/2, les membres du service des ventes peuvent préférer les clients Java et les machines Thin Client, tandis que la comptabilité a adopté des machines Windows 95. Chacune de ces familles d'utilisateurs peut avoir besoin d'options de configuration différentes (imprimantes, serveurs de noms ou serveurs Web par défaut, etc.). Dans un tel cas, il est possible d'inclure ces options dans le conteneur fournisseur, puisque chaque groupe utilise un type de machine différent. Si le même type de machine était utilisé par plusieurs groupes, il serait possible de placer les

options au sein d'un identificateur de classe subordonné, ce qui permettrait, par exemple, aux directeurs du marketing d'utiliser un groupe d'imprimantes non accessible au reste du personnel.

Remarque : L'exemple suivant représente une portion d'un fichier de configuration. Les commentaires sont précédés d'un symbole # et indiquent l'effet de chaque ligne sur l'installation.

```
vendor "AIX_CLIENT"
 {
 # Pas d'option spécifique, les différents éléments sont traités
 en fonction de leur classe
 }
 vendor "OS/2 Client"
 {
 # Pas d'option spécifique, les différents éléments sont traités
 en fonction de leur classe
 }
      vendor "Windows 95"
 { option 44 9.3.150.3          # Serveur de noms NetBIOS par
                                  défaut
 }
 vendor "Java OS"
 { bootstrapserver 9.3.150.4    # Serveur TFTP par défaut pour
 les boîtes Java
  option 67 "javaos.bin"       # Fichier de démarrage de la boîte
                               Java OS
 }
 vendor "IBM Thin Client"
 { bootstrapserver 9.3.150.5    # Serveur TFTP par défaut pour
                                  les boîtes Thin Client
 option 67 "thinos.bin"         # Fichier de démarrage par défaut
                                pour les boîtes Thin Client
 }
 subnet 9.3.149.0 255.255.255.0
 { option 3 9.3.149.1           # Passerelle par défaut pour le
                                 sous-réseau
  option 6 9.3.150.2           # Serveur de noms pour le
sous-réseau
  class accounting 9.3.149.5-9.3.149.20
  {       # La classe de facturation est limitée à la plage
 d'adresses  9.3.149.5-9.3.149.20
         # L'imprimante destinée à ce groupe fait également
 partie de cette plage, elle est donc exclue.
     exclude 9.3.149.15
     option 9 9.3.149.15       # Serveur LPR (serveur
 d'impression)
     vendor "Windows 95"
     {
     option 9 deny              # Cette installation de Windows
                                 95 ne prend pas en charge
                                # cette imprimante, l'option est
                                 donc refusée.
     }
  }
 . . .
}
```

DHCP et DDNS (Dynamic Domain Name System – Système de noms de domaine dynamique)

Le serveur DHCP fournit des options permettant le fonctionnement en environnement DDNS. Pour utiliser DHCP dans l'environnement DDNS, vous devez définir et utiliser une zone dynamique sur un serveur DNS.

Une fois le serveur DDNS configuré, vous devez décider si le serveur DHCP doit effectuer des mises à jour d'enregistrement A, des mises à jour d'enregistrement PTR, des mises à jour pour les deux types d'enregistrement ou aucune mise à jour. Cette décision dépendra de la part de travail que peut prendre en charge la machine client.

- Si le client peut assumer une partie de la mise à jour, vous pouvez confier les mises à jour d'enregistrement PTR au serveur et les mises à jour d'enregistrement A au client.
- Si le client peut tout assumer, configurez le serveur de sorte qu'il n'effectue aucune mise à jour.
- Si le client ne peut se charger de rien, configurez le serveur de sorte qu'il effectue les deux types de mise à jour.

Le serveur DHCP dispose d'un jeu de mots–clés de configuration qui vous permettent de déclencher l'exécution d'une commande lorsqu'une mise à jour est requise. Ce sont les suivants :

updatedns (déconseillé) Représente la commande à exécuter pour effectuer n'importe quel type de mise à jour. Elle sera appelée pour les enregistrements A et les enregistrements PTR.

updatednsA Spécifie la commande de mise à jour de l'enregistrement A.

updatednsP Spécifie la commande de mise à jour de l'enregistrement PTR.

Ces mots–clés définissent des chaînes exécutables que le serveur DHCP exécute lorsqu'une mise à jour est nécessaire. Les chaînes de mot–clé doivent contenir quatre `%s` (symbole de pourcentage, lettre s). Le premier `%s` correspond au nom d'hôte ; le second, au nom de domaine ; le troisième, à l'adresse IP et le quatrième, à la durée du bail. Ils sont utilisés comme quatre premiers paramètres de la commande **dhcpaction**. Les deux autres paramètres de **dhcpaction** indiquent l'enregistrement à mettre à jour (A, PTR, NONE ou BOTH) et si NIM doit être actualisé (NIM or NONIM). Pour plus d'informations sur les interactions NIM et DHCP, reportez–vous à DHCP et gestion NIM (Network Installation Management), page 4-129. Par exemple :

```
updatednsA "/usr/sbin/dhcpaction '%s' '%s' '%s' '%s' A NONIM"
          # Ceci applique la commande dhcpaction uniquement à
l'enregistrement A
          updatednsP "/usr/sbin/dhcpaction '%s' '%s' '%s' '%s'
          PTR NONIM"
         # Ceci applique la commande uniquement à
l'enregistrement PTR
          updatedns "/usr/sbin/dhcpaction '%s' '%s' '%s' '%s'
          BOTH NIM"
         # Ceci applique la commande aux deux enregistrements
         et actualise NIM
```

Le serveur DHCP dispose également d'un jeu de mots–clés pour supprimer les entrées DNS lorsqu'un bail est libéré ou arrive à expiration. Ce sont les suivants :

releasednsA Supprime l'enregistrement A.

releasednsP Supprime l'enregistrement PTR.

removedns Supprime les deux types d'enregistrement.

Ces mots–clés définissent des chaînes exécutables que le serveur DHCP exécute lorsqu'une adresse est libérée ou périmée. La commande **dhcpremove** fonctionne de la même manière que **dhcpaction**, mais n'accepte que trois paramètres :

1. L'adresse IP, spécifiée sous la forme d'un `%s` dans la chaîne de commande.
2. L'enregistrement à supprimer (A, PTR, NONE ou BOTH).
3. L'actualisation éventuelle de NIM (NIM ou NONIM).

```
releasednsA "/usr/sbin/dhcpremove '%s' A NONIM"
          # Ceci applique la commande dhcpremove uniquement à
l'enregistrement A
releasednsP "/usr/sbin/dhcpremove '%s' PTR NONIM"
         # Ceci applique la commande uniquement à
l'enregistrement PTR
removedns "/usr/sbin/dhcpremove '%s' BOTH NIM"
         # Ceci applique la commande aux deux enregistrements
         et actualise NIM
```

Les scripts **dhcpaction** et **dhcpremove** effectuent quelques vérifications sur les paramètres, puis définissent un appel vers **nsupdate**, qui a été adapté pour fonctionner avec les serveurs de ce système d'exploitation et les serveurs DDNS OS/2. Pour plus d'informations, reportez–vous à la description de la commande **nsupdate**.

Si l'interaction NIM n'est **PAS** requise par la mise à jour des noms, le serveur DHCP peut être configuré afin d'utiliser un transfert de sockets entre le démon DHCP et la commande **nsupdate** afin d'améliorer les performances et de permettre la reprise des mises à jour DNS à la suite d'une défaillance. Pour configurer cette option, le premier mot cité dans le mot–clé **updateDNSA**, **updateDNSP**, **releaseDNSA** ou **releaseDNSP** doit être "nsupdate_daemon". Les paramètres et les indicateurs de mise à jour sont identiques à ceux qui sont acceptés par la commande **nsupdate**. De plus, les noms de variables suivants peuvent être utilisés en remplacement :

$hostname	Remplacé par le nom d'hôte du client lors de la mise à jour DNS ou par le nom d'hôte préalablement associé au client pour le retrait DNS.
$domain	Remplacé par le domaine DNS relatif à la mise à jour ou par le domaine préalablement utilisé pour le nom d'hôte du client dans le cas de retrait DNS.
$ipadress	Remplacé par l'adresse IP associée ou dissociée du nom du client DHCP.
$leasetime	Remplacé par la durée du bail (en secondes).
$clientid	Remplacé par la représentation en chaîne de l'identificateur du client DHCP ou par l'association du type de matériel et de l'adresse matérielle pour les clients BOOTP.

Par exemple :

```
 updateDNSA "nsupdate_daemon -p 9.3.149.2 -h $hostname -d $domain

 updateDNSP "nsupdate_daemon -p 9.3.149.2 -r $ipaddress

releaseDNSA "nsupdate_daemon -p 9.3.149.2 -h $hostname -d $domain -
s"d;a;*;s;1;3110400""

releaseDNSP "nsupdate_daemon -p 9.3.149.2 -r $ipaddress -s"d;ptr;*;
s;1;3110400""
```

Pour plus d'informations, reportez–vous à la description de la commande **nsupdate**.

Des dispositifs définis par l'administrateur ont également été ajoutés pour les échanges de noms d'hôte entre le serveur et les clients. Par défaut, le nom d'hôte retourné au client et utilisé pour une mise à jour DDNS correspond à l'option 12 (définie dans le fichier de configuration du serveur). Toutefois, ce nom d'hôte par défaut peut également être un nom d'hôte suggéré par le client, par le biais de l'option 81 (option DHCPDDNS) ou de l'option 12 (option HOSTNAME). L'administrateur a la possibilité de remplacer ce nom d'hôte par défaut en utilisant les mots–clés de configuration **hostnamepolicy**, **proxyarec** et **appenddomain**. Ces options et leurs paramètres sont définies dans Syntaxe du fichier de serveur DHCP pour la base de données db_file.

Compatibilité DHCP avec les versions antérieures

Le serveur DHCP pour AIX Version 4.3.1 et ultérieures reconnaît les fichiers de configuration et de base de données des versions antérieures, **dhcps.ar** et **dhcps.cr**. Il analyse les anciens fichiers de configuration et génère de nouveaux fichiers de base de données aux anciens emplacements. Les anciennes bases de données sont automatiquement converties au nouveau format. Le fichier de configuration lui–même n'est pas converti.

Le module de base de données du serveur DHCP, **db_file**, est capable de lire l'ancien format. Le serveur DHCP est en mesure de détecter si un conteneur de base de données est absent du fichier de configuration et considère dans ce cas que le fichier contient tous les paramètres de serveur, les paramètres de journalisation et les paramètres de base de données **db_file**.

1. Une partie de la syntaxe de l'ancien fichier de configuration est déconseillée mais toujours prise en charge. Les autres éléments obsolètes sont les suivants :
2. Le conteneur réseau est totalement obsolète. Pour obtenir une spécification correcte, convertissez la clause réseau en une plage au sein d'un conteneur de sous–réseau correct mentionnant une adresse de sous–réseau, un masque de sous–réseau et la plage d'adresses. Si le conteneur réseau renferme des conteneurs de sous–réseau, supprimez le mot–clé du conteneur réseau et ses accolades, puis placez le masque de sous–réseau à l'endroit approprié sur la ligne. Pour démarrer à l'aide du conteneur base de données, regroupez tous les éléments ayant trait au réseau et aux accès client dans un seul conteneur de base de données de type **db_file**.
3. Les mots–clés **updatedns** et **removedns** sont obsolètes et seront remplacés de préférence par la spécification des actions à appliquer individuellement aux enregistrements A et PTR.
4. Les mots–clés **clientrecorddb** et **addressrecorddb** ont été supplantés respectivement par **clientrecorddb** et **backupfile**.
5. Les mots–clés **option sa** et **option ga** ont été remplacés respectivement par **bootstrapserver** et **giaddrfield**. Pour plus d'informations, reportez–vous à la section Syntaxe du fichier de serveur DHCP pour le fonctionnement général du serveur, page 4-113 et à Syntaxe du fichier de serveur DHCP pour la base de données db_file, page 4-117.

Options connues du fichier de serveur DHCP

Remarque : Les options du tableau suivant qui sont marquées non autorisées peuvent être spécifiées (Non dans la colonne Autorisée ?) dans le fichier de configuration mais seront remplacées par la valeur réelle. Pour une définition plus complète de chaque option, reportez–vous à la norme RFC 2132.

Numéro de l'option	Type de données par défaut	Autorisée ?	Description/Emploi
0	Aucune	Non	Complète le champ d'option, si nécessaire. Le serveur ajoute des caractères de remplissage le cas échéant.
1	Dotted quad (quatre numéros séparés par points)	Non	Masque du sous–réseau d'où est tiré l'adresse.
2	Entier 32 bits	Oui	Indique le décalage du sous–réseau du client, en secondes du système UTC (Coordinated Universal Time).
3	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP de la passerelle par défaut.
4	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des serveurs horaires.
5	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des serveurs de noms.
6	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des DNS.
7	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des serveurs de journaux.
8	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des serveurs de "cookies".
9	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des serveurs LPR.
10	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des serveurs Impress.
11	Un ou plusieurs "dotted quad"	Oui	Liste des adresses IP des serveurs de localisation des ressources.
12	Chaîne ASCII	Oui	Nom d'hôte du client à utiliser.
13	Entier 16 bits non signé	Oui	Taille du fichier de démarrage.
14	Chaîne ASCII	Oui	Chemin d'accès du fichier Merit Dump.
15	Chaîne ASCII	Oui	Nom de domaine DNS par défaut.
16	Adresse IP	Oui	Adresse du serveur Swap.
17	Chaîne ASCII	Oui	Chemin d'accès racine par défaut.
18	Chaîne ASCII	Oui	Chemin d'accès aux extensions pour le client.
19	Yes, No, True, False, 1, 0	Oui	Indique si le réacheminement IP doit être activé ou non.
20	Yes, No, True, False, 1, 0	Oui	Indique si le routage source non local doit être utilisé.
21	Une ou plusieurs paires de "dotted quad", sous la forme DottedQuad:DottedQuad	Oui	Dispositifs de filtre pour les adresses IP.
22	Entier 16 bits non signé	Oui	Taille maximale autorisée pour les fragments de datagrammes.
23	Entier 8 bits non signé	Oui	TTL (time–to–live) IP.

24	Entier 32 bits non signé	Oui	Nombre de secondes à utiliser dans le délai de vieillissement du MTU d'accès.
25	Liste d'un ou plusieurs entiers 16 bits non signés	Oui	Table des valeurs MTU d'accès. Spécifie un ensemble de valeurs représentant les tailles MTU à utiliser lors de la recherche de MTU d'accès.
26	Entier 16 bits non signé	Oui	Taille MTU pour l'interface réceptrice.
27	Yes, No, True, False, 1, 0	Oui	Indique si tous les sous–réseaux sont locaux.
28	Une adresse IP (”dotted quad”)	Oui	Diffuse une adresse pour l'interface.
29	Yes, No, True, False, 1, 0	Oui	Indique si la recherche de masque de réseau ICMP doit être utilisée.
30	Yes, No, True, False, 1, 0	Oui	Indique si le client doit devenir un fournisseur de masque de réseau ICMP.
31	Yes, No, True, False, 1, 0	Oui	Indique si les messages de recherche de routeur ICMP doivent être utilisés.
32	Une adresse IP (”dotted quad”)	Oui	Adresse à utiliser pour la sollicitation du routeur.
33	Une ou plusieurs paires d'adresses IP, sous la forme DottedQuad:DottedQuad	Oui	Chaque paire d'adresses représente une route statique.
34	Yes/No, True/False, 1/0	Oui	Indique si l'encapsulation de fin doit être utilisée.
35	Entier 32 bits non signé	Oui	Valeur du délai de cache ARP.
36	Yes/No, True/False, 1/0	Oui	Indique si l'encapsulation Ethernet doit être utilisée.
37	Entier 8 bits non signé	Oui	TTL (time–to–live) TCP.
38	Entier 32 bits non signé	Oui	Intervalle de garde en vie (keep alive) TCP.
39	Yes/No, True/False, 1/0	Oui	Indique si la garde en vie (keep alive) TCP doit être utilisée.
40	Chaîne ASCII	Oui	Domaine NIS par défaut.
41	Un ou plusieurs ”dotted quad”	Oui	Adresses IP des serveurs NIS.
42	Un ou plusieurs ”dotted quad”	Oui	Adresses IP des serveurs NTP.
43	Chaînes hexadécimales de chiffres, sous la forme hex ”*digits*”, hex ”*digits*” ou 0x*digits*	Oui, mais spécifiée en fait uniquement pour le conteneur fournisseur	Conteneur en option encapsulé pour le conteneur fournisseur.
44	Un ou plusieurs ”dotted quad”	Oui	Adresses IP des serveurs de noms NetBIOS.

45	Un ou plusieurs ”dotted quad”	Oui	Adresses IP des serveurs de distribution de datagramme NetBIOS.
46	Entier 8 bits non signé	Oui	Type de nœud NetBIOS.
47	Chaînes hexadécimales de chiffres, sous la forme hex ”*digits*”, hex ”*digits*” ou 0x*digits*	Oui	Portée NetBIOS.
48	Un ou plusieurs ”dotted quad”	Oui	Adresses IP des serveurs de polices X Windows.
49	Un ou plusieurs ”dotted quad”	Oui	Gestionnaire d’affichage X Windows.
50	Aucune	Non	Adresse IP demandée, utilisée par le client pour indiquer l’adresse souhaitée.
51	Entier 32 bits non signé	Oui	Durée du bail pour l’adresse retournée. Par défaut, le serveur DHCP utilise le mot–clé **leasetimedefault**, mais la spécification directe de l’option `51` prend le pas sur la valeur.
52	Aucune	Non	Options éventuelles. Le client utilise ce paramètre pour indiquer que les champs **sname** et **file** du paquet BOOTP peuvent avoir des options.
53	Aucune	Non	Le serveur ou le client DHCP utilise cette option pour indiquer le type de message DHCP.
54	Aucune	Non	Le serveur ou le client DHCP utilise cette option pour indiquer l’adresse du serveur ou le serveur auquel le message est envoyé.
55	Aucune	Non	Le client DHCP utilise ce paramètre pour indiquer les options souhaitées.
56	Chaîne ASCII	Oui	Chaîne que le serveur DHCP envoie au client. En général, elle peut être utilisée par le client et le serveur DHCP pour signaler des problèmes.
57	Non	Non	Le client DHCP utilise cette option pour indiquer au serveur DHCP la taille de paquet DHCP maximale que le client peut recevoir.
58	Entier 32 bits non signé	Oui	Nombre de secondes pendant lesquelles le client doit attendre avant d’envoyer un paquet de renouvellement.
59	Entier 32 bits non signé	Oui	Nombre de secondes pendant lesquelles le client doit attendre avant d’envoyer un paquet de nouvelle liaison.

60	Aucune	Non	Le client DHCP utilise cette option pour indiquer son type de fournisseur. Le client DHCP utilise ce champ pour la correspondance avec les conteneurs fournisseur.
61	Aucune	Non	Le client DHCP utilise ce paramètre pour s'identifier de manière unique. Le serveur DHCP utilise ce champ pour la correspondance avec les conteneurs client.
64	Chaîne ASCII	Oui	Spécifie le domaine NIS+.
65	Un ou plusieurs "dotted quad"	Oui	Adresses IP des serveurs NIS+.
66	Chaîne ASCII	Oui	Spécifie le nom du serveur TFTP. Ce nom d'hôte est utilisé à la place du champ **siaddr** si le client comprend cette option.
67	Chaîne ASCII	Oui	Spécifie le nom du fichier de démarrage. Ce paramètre peut être utilisé à la place du mot–clé **bootfile**, qui insère le nom du fichier dans le champ **nom de fichier** du paquet.
68	Un ou plusieurs "dotted quad" ou NONE	Oui	Adresses des agents personnels.
69	Un ou plusieurs "dotted quad"	Oui	Serveurs SMTP par défaut à utiliser.
70	Un ou plusieurs "dotted quad"	Oui	Serveurs POP3 par défaut à utiliser.
71	Un ou plusieurs "dotted quad"	Oui	Serveurs NNTP par défaut à utiliser.
72	Un ou plusieurs "dotted quad"	Oui	Serveurs WWW par défaut à utiliser.
73	Un ou plusieurs "dotted quad"	Oui	Serveurs Finger par défaut à utiliser.
74	Un ou plusieurs "dotted quad"	Oui	Serveurs IRC par défaut à utiliser.
75	Un ou plusieurs "dotted quad"	Oui	Serveurs Street Talk par défaut à utiliser.
76	Un ou plusieurs "dotted quad"	Oui	Serveurs de renseignements Street Talk par défaut à utiliser.
77	Chaîne ASCII	Oui	Identificateur de la classe du site utilisateur. Le serveur DHCP utilise ce champ pour la correspondance avec les conteneurs classe.
78	Octet obligatoire, un ou plusieurs « dotted quads »	Oui	L'option SLP directory Agent indique la liste des adresses IP des agents de répertoire
79	Octet obligatoire et chaîne ASCII	Oui	La chaîne ASCII est une liste de portées, c'est–à–dire une liste délimitée par des virgules qui indique les portées qu'un agent SLP est configuré pour utiliser.

81	Chaîne ASCII plus d'autres éléments	Non	Le client DHCP utilise cette option pour définir la politique que doit suivre le serveur DHCP vis à vis de DDNS.
85	Un ou plusieurs "dotted quad"	Oui	L'option de serveur NDS indique un ou plusieurs serveurs que le client doit contacter pour accéder à la base de données DNS. Les serveurs doivent être répertoriés dans l'ordre de préférence.
86	Chaîne ASCII	Oui	L'option d'arborescence NDS indique le nom de l'arborescence NDS que le client va contacter.
87	Chaîne ASCII	Oui	L'option de contexte NDS indique le contexte NDS initial que le client doit utiliser.
93	Aucune	Non	Le client DHCP utilise cette option pour définir l'architecture du système client.
94	Aucune	Non	Le client DHCP utilise cette option pour définir l'identifiant de l'interface du réseau client.

117	Liste d'un ou plusieurs entiers 16 bits non signés	Oui	L'option Name Service Search Option indique l'ordre de préférence du code d'option des entiers pour les services de noms. Par exemple :
			`Services de nom valeur`
			`Option de serveur de noms de domaine 6`
			`Option NIS 41`
			`Option NIS+ 65`

118	Un "dotted quad"	Non	Subnet Selection Option est une option envoyée par le client demandant au serveur dhcp d'allouer l'adresse IP à partir du sous–réseau indiqué.
255	Aucune	Non	Le serveur et le client DHCP utilisent cette option pour signaler la fin d'une liste d'options.

Sous–option de conteneur fournisseur de l'environnement PXE (Preboot Execution Environment)

Dans le cadre de la prise en charge d'un environnement PXE client, le serveur DHCP transmet l'option suivante au serveur BINLD, qui l'utilise pour sa configuration :

Opt Num	Type de données par défaut	Autorisée ?	Description
7	Un "dotted quad"	Oui	Adresse IP de multi–diffusion. Adresse IP de multi–diffusion de découverte du serveur de démarrage.

L'exemple ci–dessous montre comment cette option peut être utilisée :

```
pxeservertype    proxy_on_dhcp_server

Vendor pxeserver
{
  option   7    9.3.4.68
}
```

Dans cet exemple, le serveur DHCP informe le client que le serveur proxy est exécuté sur la même machine mais est à l'écoute des requêtes sur le port 4011. Le conteneur fournisseur est requis ici car le serveur BINLD diffuse un message INFORM/REQUEST sur le port 67, l'option 60 étant définie sur "PXEServer". En retour, le serveur DHCP envoie l'adresse IP de multi–diffusion sur laquelle le serveur BINLD doit écouter les requêtes du client PXE.

Exemple de fichier de configuration prenant en charge les clients PXE

Dans l'exemple ci–dessous, le serveur **dhcpsd** donne le nom du fichier de démarrage au PXEClient ou dirige celui–ci sur le serveur BINLD en envoyant des sous–options. Le mot–clé **pxeboofile** est utilisé pour créer une liste de fichiers de démarrage pour l'architecture d'un client donné et des versions majeures et mineures du système client.

```
pxeservertype     dhcp_pxe_binld
subnet default
{
  vendor pxe
  {
    option 6 2    # Désactiver la multidiffusion
    option 8 5 4 10.10.10.1 12.1.1.15 12.5.5.5 12.6.6.6\
        2 2 10.1.1.10 9.3.4.5 1 1 10.5.5.9\
        1 1 9.3.149.15\
        4 0
    option 9 5 "WorkSpace On Demand" 2 "Intel"\
        1 "Microsoft WindowsNT" 4 "NEC ESMPRO"
    option 10 2 "Press F8 to View Menu"
  }
  vendor pxeserver
  {
    option 7 239.0.0.239
  }

}

subnet  9.3.149.0  255.255.255.0
{
  option 3   9.3.149.1
  option 6   9.3.149.15

  vendor pxe
  {

  option    6    4    # bootfile est présent dans le paquet propos
é
   pxebootfile   1   2   1   os2.one
   pxebootfile   2   2   1   aix.one
  }
}
```

Chaque ligne option du conteneur pxe est utilisée par le serveur pour indiquer au client ce qu'il doit faire. La section Sous–options du conteneur fournisseur PXE décrit les sous–options PXE connues et prises en charge.

Syntaxe du fichier de serveur DHCP pour le fonctionnement général du serveur

Remarque : Les unités de temps (*time_units*) indiquées dans le tableau suivant sont facultatives et correspondent à un modificateur du temps réel. L'unité de temps par défaut est exprimée en minutes. Les valeurs autorisées sont les secondes (1), les minutes (60), les heures (3600), les jours (86400), les semaines (604800), les mois (2392000) et les années (31536000). Le nombre entre parenthèses est un multiplicateur appliqué à la valeur *n* spécifiée pour exprimer cette valeur en secondes.

Mot–clé	Forme	Sous–conteneurs ?	Valeur par défaut	Signification
database	database db type	Oui	Aucune	Conteneur principal renfermant les définitions des pools d'adresses, options et instructions d'accès client. db type est le nom du module chargé pour traiter cette portion du fichier. La seule valeur actuellement disponible est **db_file**.
logging_info	logging_info	Oui	Aucune	Conteneur de journalisation principal définissant les paramètres de journalisation.
logitem	logitem NONE	Non	Non activé pour tous par défaut.	Active le niveau de journalisation. Plusieurs lignes sont autorisées.
	logitem SYSERR			
	logitem OBJERR			
	logitem PROTOCOL			
	logitem PROTERR			
	logitem WARN			
	logitem WARNING			
	logitem CONFIG			
	logitem EVENT			
	logitem PARSEERR			
	logitem ACTION			
	logitem ACNTING			
	logitem STAT			
	logitem TRACE			
	logitem RTRACE			
	logitem START			
numLogFiles	numLogFiles *n*	Non	0	Indique le nombre de fichiers journaux à créer. Les journaux alternent lorsque le premier journal est rempli. *n* est le nombre de journaux à créer.

logFileSize	logFileSize *n*	Non	0	Indique la taille de chaque fichier journal, exprimée en unités de 1024 octets.
logFileName	logFileName *path*	Non	Aucune	Indique le chemin d'accès au premier fichier journal. Le nom d'origine du fichier journal est *nomfichier* ou *nomfichier.extension*. *nomfichier* est limité à huit caractères. Lorsque la permutation des fichiers est effectuée, le premier fichier est renommé en conservant le base du nom, nomfichier, et en lui ajoutant un numéro, ou en remplaçant l'extension par un numéro. Par exemple, si le nom d'origine du fichier est `file`, le nom du fichier après permutation devient `file01`. Si le nom du fichier d'origine est `file.log`, il devient `file.01`.
CharFlag	charflag yes charflag true charflag false charflag no	Non	true	Non applicable au serveur DHCP de ce système d'exploitation, mais utilisé par le serveur DHCP OS/2 pour générer des fenêtres de débogage.
StatisticSnapShot	StatisticSnapShot n	Non	–1, jamais	Indique, en secondes, à quelle fréquence les statistiques sont écrites dans le fichier journal.
UsedIpAddress ExpireInterval	UsedIpAddressExpireInterval n time_units	Non	–1, jamais	Indique à quelle fréquence les adresses présentant l'état BAD sont recoupées et testées afin de vérifier leur validité.
leaseExpireInterval	leaseExpireInterval n time_units	Non	900 secondes	Indique à quelle fréquence les adresses à l'état BOUND sont vérifiées pour voir si elles sont arrivées à expiration. Si l'adresse est arrivée à expiration, l'état devient EXPIRED.
reservedTime	reservedTime n time_units	Non	–1, jamais	Indique pendant combien de temps les adresses peuvent rester à l'état RESERVED avant de reprendre l'état FREE.
reservedTime Interval	reservedTimeInterval n time_units	Non	900 secondes	Indique à quelle fréquence les adresses à l'état RESERVE sont vérifiées pour voir si elles peuvent reprendre l'état FREE.
saveInterval	saveInterval n time_units	Non	3600 secondes	Indique à quelle fréquence le serveur DHCP doit déclencher une sauvegarde des bases de données ouvertes. Pour les serveurs très chargés, cette valeur doit tourner autour de 60 ou 120 secondes.

clientpruneintv	clientpruneintv n time_units	Non	3600 secondes	Indique à quelle fréquence le serveur DHCP supprime des bases de données les clients non associés à une adresse (état UNKNOWN). Ceci permet d'économiser la mémoire du serveur DHCP.
num processors	numprocessors n	Non	10	Indique le nombre de processeurs de paquets à créer. Le minimum est de un.
userObject	userObject *obj_name*	Oui	Aucune	Indique que le serveur doit charger un objet partagé défini par l'utilisateur et appeler des routines au sein de cet objet par le biais de chaque interaction avec les clients DHCP. L'objet à charger est situé dans le répertoire `/usr/sbin` sous le nom de *`obj_name`*`.dhcpo`. Pour plus d'informations, reportez–vous au DHCP Server User–Defined Extension API.

pxeservertype	pxeservertype *server_type*	No	dhcp_only	Indique le type du serveur **dhcpd**. *server_type* peut avoir l'une des valeurs suivantes : dhcp_pxe_binld DHCP exécute les fonctions **dhcpsd**, **pxed** et **bindl**. proxy_on_dhcp_server DHCP renvoie le client PXE au port du serveur proxy sur la même machine. La valeur par défaut est `dhcp_only`, ce qui signifie que **dhcpsd** ne prend pas en charge les clients PXE en mode par défaut.
supportsubnetselection	supportsubnetselection **global** supportsubnetselection **subnetlevel** supportsubnetselection **no**	No	Aucune	Indique si le serveur dhcp prend en charge l'option 118 (option de sélection du sous–réseau) dans le paquet DISCOVER ou REQUEST des clients. **global**: tous les sous–réseaux du fichier de configuration prennent en charge l'option 118. **subnetlevel**: les sous–réseaux configurés pour prendre en charge cette option via le mot–clé supportoption118 acceptent cette option. **no**: ne prend pas en charge l'option 118.

Remarques sur la syntaxe du fichier de serveur DHCP pour la base de données db_file :

1. Les unités de temps (*time_units*) indiquées dans le tableau suivant sont facultatives et correspondent à un modificateur du temps réel. L'unité de temps par défaut est exprimée en minutes. Les valeurs autorisées sont les secondes (1), les minutes (60), les heures (3600), les jours (86400), les semaines (604800), les mois (2392000) et les années (31536000). Le nombre entre parenthèses est un multiplicateur appliqué à la valeur *n* spécifiée pour exprimer cette valeur en secondes.
2. Par ailleurs, les éléments spécifiés dans un conteneur peuvent être remplacés par ceux d'un sous–conteneur. Vous pouvez par exemple définir les clients BOOTP de manière globale, et, au sein d'un sous–réseau particulier, autoriser les clients BOOTP en indiquant le mot–clé supportBootp dans les deux conteneurs.
3. Les conteneurs client, classe et fournisseur acceptent les expressions régulières. Pour la classe et le vendeur, une chaîne entre guillemets dont le premier caractère à l'intérieur des guillemets est un point d'exclamation (!) indique que le reste de la chaîne doit être considéré comme une expression régulière. Le conteneur client accepte les expressions régulières dans les champs hwtype et hwaddr. Une chaîne unique est utilisée pour représenter les deux champs, selon la syntaxe suivante :

   `nombre_décimal-données`

   Si nombre_décimal est égal à zéro, les données constituent une chaîne ASCII. Pour tout autre nombre, les données sont des chiffres hexadécimaux.

Mot–clé	Forme	Sous–conteneurs ?	Valeur par défaut	Signification
subnet	subnet default	Oui	Aucune	Spécifie un sous–réseau sans plage associée. Ce sous–réseau est utilisé par le serveur uniquement pour répondre à un paquet INFORM/REQUEST du client et si aucun conteneur de sous–réseau ne correspond à l'adresse de ce dernier.
subnet	subnet subnet id netmask subnet *subnet id netmask range* subnet *subnet id netmask label*:*priority* subnet *subnet id netmask range label*:*priority*	Oui	Aucune	Spécifie un sous–réseau et un pool d'adresses. Toutes les adresses sont supposées faire partie du pool, sauf si une plage est spécifiée sur la ligne ou si les adresses sont modifiées ultérieurement dans le conteneur par une instruction de plage ou d'exclusion. La plage facultative est une paire d'adresses IP en format de "dotted quad" séparées par un tiret. Il est possible de préciser un label et une priorité. Ceux–ci sont utilisés dans les sous–réseaux virtuels pour identifier et classer les sous–réseaux du sous–réseau virtuel. Le label et la priorité sont séparés par un signe deux–points. Ces conteneurs ne sont autorisés qu'au niveau global ou au niveau du conteneur base de données.

subnet	subnet *subnet id range*	Oui	Aucune	Spécifie un sous–réseau qui s'inscrit dans un conteneur réseau. Il définit une plage d'adresses formant la totalité du sous–réseau, sauf si la plage facultative est indiquée. Le masque de réseau associé au sous–réseau est issu du conteneur réseau environnant. **Remarque :** Cette méthode est déconseillée au profit des autres formes de sous–réseaux.
option	option *number data* ...	Non	Aucune	Spécifie une option à envoyer à un client ou, dans le cas d'un refus (deny), une option qui ne doit pas être envoyée à un client. La clause option `* deny` signifie que toutes les options non spécifiées dans le conteneur en cours ne doivent pas être retournées au client. L'option *number*deny ne refuse que l'option spécifiée. *number* est un entier 8 bits non signé. *data* est spécifique à l'option (voir ci–dessus) ou peut être définie sous la forme d'une chaîne entre guillemets (texte ASCII) ou 0x*hexdigits* ou hex"*hexdigits*" ou encore hex "*hexdigits*". Si l'option correspond à un conteneur fournisseur, elle sera encapsulée avec les autres options dans une option 43.
	option *number*deny			
	option * deny			
exclude	exclude *an IP address*	Non	Aucune	Modifie la plage sur le conteneur qui comporte l'instruction exclude. L'instruction exclude n'est pas valide au niveau des conteneurs de base de données ou au niveau général. L'instruction exclude supprime l'adresse ou la plage spécifiée de la plage actuelle sur le conteneur. Elle permet de créer des plages non contiguës pour sous sous–réseaux ou d'autres conteneurs.
	exclude *dotted_quad–dotted_quad*			

range	range *IP_address* range *dotted_quad–dotted_quad*	Non	Aucune	Modifie la plage sur le conteneur qui comporte l'instruction range. L'instruction range n'est pas valide au niveau des conteneurs de base de données ou au niveau général. S'il s'agit de la première plage du conteneur qui ne spécifie pas une plage sur la ligne de définition du conteneur, la plage du conteneur devient alors la plage spécifiée par l'instruction range. Toute instruction range suivante, ou toutes les instructions range dans le cas d'un conteneur spécifiant des plages dans sa définition sont ajoutées à la page actuelle. Avec l'instruction range, il est possible d'ajouter à la plage existante une adresse unique ou un jeu d'adresses. La plage doit être incorporée dans la définition du conteneur de sous–réseau.
client	client *hwtype hwaddr* NONE client *hwtype hwaddr* ANY client *hwtype hwaddr dotted_quad* client *hwtype hwaddr range*	Oui	Aucune	Spécifie un conteneur client qui empêche le client indiqué par *hwaddr* et *hwtype* d'obtenir une adresse. Si *hwtype* est 0, alors *hwaddr* est une chaîne ASCII. Sinon, *hwtype* correspond au type de matériel du client et *hwaddr* à l'adresse du matériel du client. Si *hwaddr* est une chaîne, des guillemets peuvent encadrer la chaîne. Si *hwaddr* est une chaîne hexadécimale, l'adresse peut être spécifiée sous la forme 0x*hexdigits* ou hex *digits*. *range* permet au client spécifié par *hwaddr* et *hwtype* d'obtenir une adresse faisant partie de cette *plage*. Pour faire référence à plusieurs clients, il faut utiliser une expression régulière.
class	class *string* class *string range*	Oui	Aucune	Spécifie un conteneur classe portant le nom *string*. La chaîne peut ou non être placée entre guillemets. Si oui, les guillemets sont supprimés avant la comparaison. Les guillemets sont obligatoires si la chaîne contient des espaces ou des tabulations. Ce conteneur est autorisé à tous les niveaux. Il est possible d'indiquer une plage pour spécifier le jeu d'adresses à proposer au client avec cette classe. La plage est soit une adresse IP en format de "dotted quad", soit deux adresses IP en format de "dotted quad" séparées par un tiret.

réseau	network *network id netmask* network *network id* network *network id range*	Oui	Aucune	Spécifie un ID de réseau à l'aide des informations de classe (par exemple 9.3.149.0 avec un masque de réseau de 255.255.255.0 correspond au réseau 9.0.0.0 255.255.255.0). Cette version du conteneur de réseau est utilisée pour englober les sous–réseaux partageant le même masque et le même ID de réseau. Lorsqu'une plage est fournie, toutes les adresses de la plage font partie du pool. La plage doit être comprise dans le réseau de l'ID de réseau. Elle fait appel à l'adresse intégrale de la classe. Elle n'est valide qu'au niveau général ou au niveau du conteneur de base de données. **Remarque :** Le mot–clé network est déconseillé au profit du conteneur de sous–réseau.
vendor	vendor *vendor_id* vendor *vendor_id* hex"" vendor *vendor_id* hex "" vendor *vendor_id* 0xdata vendor *vendor_id* "" vendor *vendor_id range* vendor *vendor_id range* hex"" vendor *vendor_id range* hex "" vendor *vendor_id range* 0xdata vendor *vendor_id range* ""	Oui	Aucune	Spécifie un conteneur de fournisseur. Les conteneurs fournisseur sont utilisés pour retourner l'option 43 au client. L'id de fournisseur peut être spécifié sous la forme d'une chaîne entre guillemets ou d'un chaîne binaire du type 0x*hexdigits* ou hex"*digits*". Il est possible d'ajouter à l'id de fournisseur une plage facultative, en utilisant deux "dotted quad" séparés par un tiret. A la suite de la plage facultative, une chaîne hexadécimale ou ASCII également facultative peut être indiquée comme première partie de l'option 43. Si des options figurent dans le conteneur, elles sont annexées aux données de l'option 43. Une fois toutes les options traitées, une option End Of Option List (fin de la liste d'options) est ajoutée aux données. Pour retourner les options en dehors d'une option 43, utilisez une expression régulière correspondant à tous les clients pour spécifier les options normales à renvoyer en fonction de l'ID fournisseur.

inoption	inoption *number option_data* inoption *number option_data range*	Oui	Aucune	Indique un conteneur à rapprocher d'une option entrante arbitraire définie par le client. *number* indique le numéro de l'option. *option_data* définit la clé correspondant au conteneur à sélectionner lors du choix de l'adresse et de l'option pour ce client. La clé *option_data* se présente sous forme de chaîne entre guillemets, d'adresse IP ou de nombre entier pour les options connues mais peut également se présenter sous forme de chaîne hexadécimale d'octets si elle est précédée des caractères `0x`. Pour les options que le serveur connaît mal, il est possible de définir une chaîne hexadécimale d'octets sur le même schéma. En outre, la valeur *option_data* peut faire référence à une expression régulière à rapprocher de la représentation en chaîne des données d'option du client. Ces expressions régulières se présentent sous la forme d'une chaîne entre guillemets (dont le premier caractère est un point d'exclamation `"!`). Les options peu connues du serveur se présentent sous forme de chaîne hexadécimale d'octets NON précédée des caractères `0x`.
virtual	virtual fill *id id* ... virtual sfill *id id* ... virtual rotate *id id* ... virtual srotate *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique. `fill` signifie utiliser toutes les adresses de ce conteneur avant de passer au suivant. `rotate` signifie sélectionner une adresse du pool suivant de la liste sur chaque requête. `sfill` et `srotate` sont identiques à `fill` et `rotate`, mais une recherche est effectuée pour savoir si le client correspond aux conteneurs, aux fournisseurs ou aux classes du sous–réseau. Si une correspondance permet d'obtenir une adresse, cette adresse est adoptée à partir du conteneur au lieu de suivre la politique indiquée. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.

inorder:	inorder: *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique de remplissage, ce qui signifie utiliser toutes les adresses de ce conteneur avant de passer au conteneur suivant. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.
balance:	balance: *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique de rotation, ce qui signifie utiliser l'adresse suivante du conteneur suivant. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.
support Bootp	supportBootp true supportBootp 1 supportBootp yes supportBootp false supportBootp 0 supportBootp no	Non	Oui	Indique si le conteneur en cours et tous ceux qui en découlent (jusqu'à mention contraire) doivent accepter les clients BOOTP.
supportUnlisted clients	supportUnlistedclients BOTH supportUnlistedclients DHCP supportUnlistedclients BOOTP supportUnlistedclients NONE supportUnlistedclients true supportUnlistedclients yes supportUnlistedclients 1 supportUnlistedclients false supportUnlistedclients no supportUnlistedclients 0	Non	Both	Indique si le conteneur en cours et tous ceux qui en découlent (jusqu'à mention contraire) doivent accepter les clients non répertoriés. La valeur indique si tous les clients bénéficient d'un accès sans instructions client particulières, si seuls les clients DHCP ont un accès, si seuls les clients BOOTP sont autorisés ou aucun des deux. **Remarque :** Les valeurs true et false ont été conservées par souci de compatibilité avec les versions antérieures mais sont déconseillées. La valeur true est équivalente à BOTH et la valeur false à NONE.

address recorddb	addressrecrddb *path*	Non	Aucune	Lorsqu'elle est spécifiée, cette option fonctionne comme le mot–clé **backupfile**. L'option n'est valide qu'au niveau général ou au niveau du conteneur de base de données. **Remarque :** Cette méthode est déconseillée.
backup file	backupfile *path*	Non	/etc/db_file.crbk	Indique le fichier à utiliser pour les sauvegardes de la base de données. L'option n'est valide qu'au niveau général ou au niveau du conteneur de base de données.
check pointfile	checkpointfile *path*	Non	/etc/db_file.crbk	Indique le fichier de points de contrôle de la base de données. Le premier fichier de points de contrôle correspond à *path*. Le second est path, avec le dernier caractère remplacé par un 2. Le nom du fichier de contrôle ne doit donc pas se terminer à l'origine par un 2. Cette option n'est valable qu'au niveau général ou au niveau du conteneur de base de données.
client recorddb	clientrecorddb *path*	Non	/etc/db_file.crbk	Indique le fichier de sauvegarde de la base de données. Le fichier contient tous les enregistrements client que le serveur DHCP a traités. L'option n'est valide qu'au niveau général ou au niveau du conteneur de base de données.
bootstrap server	bootstrapserver *IP address*	Non	Aucune	Indique le serveur que les clients doivent utiliser comme point de départ vers les fichiers TFTP à l'issue de la réception de paquets BOOTP ou DHCP. Cette valeur complète le champ **siaddr** du paquet. Cette option est valide à tous les niveaux de conteneur.
giaddr field	giaddrfield *IP address*	Non	Aucune	Définit le champ giaddrfield pour les paquets de réponse. **Remarque :** Cette spécification n'est pas autorisée pour les protocoles BOOTP et DHCP, mais certains clients exigent le champ **giaddr** comme passerelle par défaut pour le réseau. En raison de ce risque de conflit, il est conseillé de n'utiliser giaddrfield qu'au sein d'un conteneur client, bien que l'option fonctionne à tous les niveaux.

pingTime	pingTime *n time_unit*	Non	3 seconds	Indique la durée pendant laquelle la réponse ping doit être attendue avant qu'une adresse ne soit suspendue. L'unité de temps par défaut est de l'ordre des centièmes de seconde. La valeur de l'unité de temps est définie dans la remarque qui précède ce tableau. Cette option est valide à tous les niveaux de conteneur. Le paramètre time_unit est facultatif.
bootptime	bootptime *n time_unit*	Non	–1, illimité	Indique la durée pendant laquelle louer une adresse à un client BOOTP. La valeur par défaut est –1, ce qui signifie durée illimitée. Les valeurs classiques d'unités de temps sont acceptées. Le paramètre time unit est facultatif. Cette option est valide à tous les niveaux de conteneur.
AllRoutesBroadcast	allroutesbroadcast no allroutesbroadcast false allroutesbroadcast 0 allroutesbroadcast yes allroutesbroadcast true allroutesbroadcast 1	Non	0	Si un réponse de diffusion est requise, indique si cette réponse doit être diffusée sur toutes les routes. Cette option est valide à tous les niveaux de conteneur. Elle est ignorée par les serveurs DHCP du système d'exploitation car l'adresse MAC réelle du client, y compris les RIF, est stockée pour le paquet en retour. Cette option est valide à tous les niveaux de conteneur.
addressassigned	addressassigned *"string"*	Non	Aucune	Indique une chaîne entre guillemets à exécuter lorsqu'une adresse est attribuée à un client. La chaîne doit comporter deux %s. Le premier %s correspond à l'ID client, sous la forme type–string. Le second %s est une adresse IP en format de "dotted quad". Cette option est valide à tous les niveaux de conteneur.
addressreleased	addressreleased *"string"*	Non	Aucune	Indique une chaîne entre guillemets à exécuter lorsqu'une adresse est libérée par un client. La chaîne ne doit comporter qu'un %s, correspondant à l'adresse IP libérée en format de "dotted quad". Cette option est valide à tous les niveaux de conteneur.

appenddomain	appenddomain 0 appenddomain no appenddomain false appenddomain 1 appenddomain yes appenddomain true	Non	Non	Indique s'il convient d'ajouter le nom de domaine défini par l'option 15 au nom d'hôte suggéré par le client lorsque ce dernier ne propose pas de nom de domaine. Cette option est valide à tous les niveaux de conteneur.
canonical	canonical 0 canonical no canonical false canonical 1 canonical yes canonical true	Non	0	Indique que l'ID du client est en format canonique. Cette option n'est valide qu'au niveau du conteneur client.
leaseTimeDefault	leaseTimeDefault *n time_unit*	Non	86400 seconds	Indique la durée du bail par défaut pour les clients. Cette option est valide à tous les niveaux de conteneur. Le paramètre time_unit est facultatif.
proxyarec	proxyarec never proxyarec usedhcpddns proxyarec usedhcpddnsplus proxyarec always proxyarec usedhcpddnsprotected proxyarec usedhcpddnsplus protected proxyarec alwaysprotected proxyarec standard proxyarec protected	Non	usedhcpddnsplus	Indique les options et méthodes qui doivent être utilisées pour la mise à jour des enregistrements A dans DNS. `never` signifie que l'enregistrement A ne doit jamais être actualisé. `usedhcpddns` signifie utiliser l'option 81 si le client l'a définie. `usedhcpddnsplus` signifie utiliser l'option 81, ou les options 12 et 15, si spécifié. `always` signifie que l'enregistrement A doit être actualisé pour tous les clients. `XXXXprotected` modifie la commande **nsupdate** pour s'assurer que le client est autorisé. `standard` est synonyme de `always`. `protected` est synonyme de `alwaysprotected`. Cette option est valide à tous les niveaux de conteneur.
releasednsA	releasednsA *"string"*	Non	Aucune	Indique la chaîne d'exécution à utiliser lors de la libération d'une adresse. La chaîne est utilisée pour supprimer l'enregistrement A associé à l'adresse libérée. Cette option est valide à tous les niveaux de conteneur.

releasednsP	releasednsP *"string"*	Non	Aucune	Indique la chaîne d'exécution à utiliser lors de la libération d'une adresse. La chaîne est utilisée pour supprimer l'enregistrement PTR associé à l'adresse libérée. Cette option est valide à tous les niveaux de conteneur.
removedns	removedns *"string"*	Non	Aucune	Indique la chaîne d'exécution à utiliser lors de la libération d'une adresse. La chaîne est utilisée pour supprimer les enregistrements A et PTR associés à l'adresse libérée. Cette option est valide à tous les niveaux de conteneur. **Remarque :** Cette option est déconseillée au profit des mots–clés **releasednsA** et **releasednsP**.
updatedns	updatedns *"string"*	Non	Aucune	Indique la chaîne d'exécution à utiliser lors de la liaison d'une adresse. La chaîne est utilisée pour mettre à jour les enregistrements A et PTR associés à l'adresse. Cette option est valide à tous les niveaux de conteneur. **Remarque :** Cette option est déconseillée au profit des mots–clés **updatednsA** et **updatednsP**.
updatednsA	updatednsA *"string"*	Non	Aucune	Indique la chaîne d'exécution à utiliser lors de la liaison d'une adresse. La chaîne est utilisée pour mettre à jour l'enregistrement A associé à l'adresse. Cette option est valide à tous les niveaux de conteneur.
updatednsP	updatednsP *"string"*	Non	Aucune	Indique la chaîne d'exécution à utiliser lors de la liaison d'une adresse. La chaîne est utilisée pour mettre à jour l'enregistrement PTR associé à l'adresse. Cette option est valide à tous les niveaux de conteneur.

hostnamepolicy	hostnamepolicy suggested hostnamepolicy resolved hostnamepolicy always_resolved hostnamepolicy defined hostnamepolicy always_defined hostnamepolicy default	Non	par défaut	Spécifie le nom d'hôte à retourner au client. La politique par défaut préfère le nom d'hôte et le nom de domaine explicitement définis par rapport aux noms suggérés. Les autres politiques respectent strictement les consignes (par exemple : `defined` retourne le nom défini ou rien si aucun nom n'est défini dans la configuration). En outre, les politiques utilisant le modificateur `always` demandent au serveur de toujours retourner l'option nom d'hôte même si le client ne l'a pas demandé au moyen de l'option liste des paramètres. A noter que suggérer un nom d'hôte implique également de le demander, et que les noms d'hôte peuvent être suggérés à l'aide de l'option 81 ou des options 12 et 15. Ce mot–clé est valide à tous les niveaux de conteneur.
bootfilepolicy	bootfilepolicy suggested bootfilepolicy merge bootfilepolicy defined bootfilepolicy always	Non	suggested	Définit une préférence pour retourner le nom du fichier de démarrage à un client. `suggested` préfère le nom du fichier de démarrage suggéré par le client à n'importe quel autre nom configuré par le serveur. `merge` ajoute le nom suggéré par le client au répertoire personnel configuré par le serveur. `defined` préfère le nom défini à n'importe quel autre nom suggéré. `always` retourne le nom défini même si le client ne l'a pas demandé à l'aide de l'option liste des paramètres.
stealfromchildren	stealfromchildren true stealfromchildren 1 stealfromchildren yes stealfromchildren false stealfromchildren 0 stealfromchildren no	Non	Non	Indique si le conteneur parent est autorisé à "voler" des adresses dans ses conteneurs enfants lorsqu'il est à court d'adresses. Cela signifie que si vous avez un sous–réseau avec une classe définie à l'aide d'une plage d'adresses, ces adresses sont réservées aux clients qui mentionnent cette classe. Si `stealfromchildren` a la valeur true, les adresses seront récupérées chez un enfant afin de tenter de satisfaire la requête. La valeur par défaut n'autorise pas les vols d'adresses.

homedirectory	homedirectory *path*	Non	Aucune	Indique le répertoire personnel à utiliser dans la section fichier du paquet de réponse. Cette option peut être définie à tous les niveaux de conteneur. La politique bootfile définit comment les éléments spécifiés dans la section fichier du paquet entrant se conjuguent avec les instructions du fichier de démarrage et du répertoire personnel.
bootfile	bootfile *path*	Non	Aucune	Indique le fichier de démarrage à utiliser dans la section fichier du paquet de réponse. Cette option peut être définie à tous les niveaux de conteneur. La politique bootfile définit comment les éléments spécifiés dans la section fichier du paquet entrant se conjuguent avec les instructions du fichier de démarrage et du répertoire personnel.
pxebootfile	pxebootfile *system_architecture major_version minor_version boofilename*	Non	Aucune	Indique le fichier de démarrage à donner pour un client. Il n'est utilisé que lorsque **dhcpsd** prend en charge les clients PXE (**pxeservertype** a la valeur **dhcp_pxe_binld**). L'analyseur du fichier de configuration génère une erreur si le nombre de paramètres après `pxebootfile` est inférieur à quatre, et il ignore les paramètres supplémentaires. `pxebootfile` ne peut être utilisé que dans un conteneur.
supportoption118	supportoption118 *no* / *yes*	Non. Peut être défini uniquement dans le conteneur du sous–réseau.	Aucune	Ce mot–clé indique si ce conteneur prend en charge l'option 118. Yes signifie qu'elle est prise en charge, No qu'elle ne l'est pas. Pour que cette option soit prise en compte, vous devez aussi utiliser le mot–clé **supportsubnetselection**.

DHCP et gestion NIM (Network Installation Management)

Le concept d'affectation dynamique d'adresses IP est relativement nouveau. Voici quelques suggestions relatives à l'interaction entre DHCP et NIM.

1. Lorsque vous configurez des objets dans l'environnement NIM, utilisez des noms d'hôte chaque fois que possible : vous pouvez ainsi exploiter un serveur de noms dynamique qui met à jour les adresses IP lorsque le nom d'hôte est converti en adresse IP dans l'environnement NIM.
2. Placez le maître NIM et le serveur DHCP sur le même système. Le serveur DHCP est doté, dans la chaîne DNS de mise à jour, d'une option qui, affectée de la valeur `NIM`, tente de conserver les objets NIM hors des états qui requièrent des adresses IP statiques quand ces adresses changent.
3. Pour les clients NIM, vous devez définir un délai dédié double du temps requis pour installer un client. Cela permet à une adresse IP dédiée de rester valide pendant l'installation. Celle-ci terminée, le client réamorce et DHCP est lancé ou doit être configuré, selon le type de l'installation.
4. Le serveur `dhcpsd` doit être responsable des enregistrements système noms de domaine PTR et A. Lorsque NIM réinstalle la machine, le fichier contenant le RSA est supprimé et le client ne peut mettre ses enregistrements à jour. C'est pourquoi le serveur doit mettre à jour les enregistrements système. Pour ce faire, modifiez la ligne `updatedns` du fichier **/etc/dhcpcd.ini** :

   ```
   updatedns "/usr/sbin/dhcpaction '%s' '%s' '%s' '%s' NONE NONIM"
   ```

   Dans le fichier **/etc/dhcpsd.cnf**, changez la ligne `updatedns` en :

   ```
   updatedns "/usr/sbin/dhcpaction '%s' '%s' '%s' '%s' BOTH NIM"
   ```

   Remarque : Lorsqu'un objet NIM est placé en état d'attente de l'installation BOS, le serveur `dhcpsd` peut passer des arguments différents de ceux prévus à l'origine. Pour éviter cette situation, réduisez au minimum le délai pendant lequel le client se trouve en état d'attente.

Suivez ces suggestions : les clients dynamiques pourront exploiter l'environnement NIM.

Pour en savoir plus sur l'environnement NIM, reportez-vous au manuel *AIX 5L Version 5.2Network Installation Management Guide and Reference*.

Démon DHCP avec structure PXED (Preboot Execution Environment Proxy)

Le serveur DHCP proxy PXE se comporte comme un serveur DHCP ; il écoute le trafic client DHCP ordinaire et répond à certaines requêtes. Toutefois, contrairement au serveur DHCP, le serveur DHCP proxy ne gère pas les adresses réseau, et il ne répond qu'aux clients qui s'identifient en tant que clients PXE. Les réponses données par le serveur DHCP proxy PXE contiennent le mécanisme selon lequel le client localise les serveurs de démarrage ou les adresses réseau et les descriptions des serveurs de démarrage compatibles pris en charge.

L'utilisation d'un serveur DHCP proxy PXE avec un serveur DHCP fournit trois fonctionnalités clés. Tout d'abord, vous pouvez séparer l'administration des adresses réseau de l'administration des images de démarrage. En utilisant deux processus différents sur le même système, vous pouvez configurer les informations de démarrage gérées par le serveur DHCP proxy PXE sans intervenir sur la configuration du serveur DHCP ou avoir besoin d'y accéder. Ensuite, vous pouvez définir plusieurs serveurs de démarrage et laisser le client PXE en sélectionner un lors du démarrage. Chaque serveur de démarrage peut, par exemple, offrir un type différent de système d'exploitation ou de configuration système. Enfin, l'utilisation du serveur proxy offre la possibilité de configurer le client PXE de telle sorte qu'il utilise l'adressage IP multi–diffusion pour trouver la localisation des serveurs de démarrage compatibles.

Le serveur DHCP proxy PXE peut être configuré pour s'exécuter sur le même système que celui exécutant le serveur DHCP ou sur un système différent. En outre, il peut être configuré pour s'exécuter sur le système qui exécute également le démon du serveur de démarrage ou sur un système différent.

Le serveur DHCP proxy PXE

Le serveur PXED est divisé en trois grandes parties : une base de données, un moteur de protocole et un ensemble de routines de service, chaque partie disposant de ses propres informations de configuration.

La base de données PXED

La base de données **db_file.dhcpo** est utilisée pour générer les options à transmettre au client lorsqu'il envoie un paquet REQUEST. Les options renvoyées par la base de données dépendent du type de serveur choisi. Celui–ci est défini à l'aide du mot–clé **pxeservertype** dans le fichier **pxed.cnf**.

A partir des informations du fichier de configuration, la base de données est amorcée et sa cohérence est vérifiée.

Le moteur de protocole PXED

Pour AIX Version 4.3.1 et ultérieures, le moteur de protocole PXED est basé sur Preboot Execution Environment (PXE) Specification Version 2.1 d'Intel, mais il reste compatible avec PXE Specification Version 1.1. Le moteur de protocole utilise la base de donnés pour déterminer quelles informations doivent être retournées au client.

Opérations PXED enchaînées

Le dernier élément du serveur PXED est en fait un ensemble d'opérations qui permettent d'assurer la continuité de l'exécution. Comme le serveur PXED est du type enchaîné, ces opérations sont définies sous la forme de routines qui interviennent occasionnellement pour s'assurer du bon déroulement de l'exécution.

La première routine, ou routine *principale*, gère les requêtes SRC (par exemple **startsrc**, **stopsrc**, **lssrc**, **traceson** et **refresh**). Cette routine coordonne également toutes les opérations qui affectent toutes les routines et gère les signaux. Par exemple :

- A SIGHUP (–1) provoque un rafraîchissement de toutes les bases de données du fichier de configuration.
- A SIGTERM (–15) entraîne l'arrêt en douceur du serveur.

L'autre routine traite les paquets. Selon le type du serveur, une ou deux routines sont utilisées. L'une d'entre elles écoute le port 67 et la deuxième le port 4011. Chacune peut traiter une requête d'un client.

Configuration du serveur PXED

Par défaut, la configuration du serveur PXED est effectuée par la lecture du fichier **/etc/pxed.cnf**, qui spécifie la base de données initiale d'adresses et d'options du serveur. Le serveur est démarré à partir de Web-based System Manager, de SMIT ou via les commandes SRC.

La configuration de PXED constitue la tâche la plus délicate dans le cadre de l'utilisation de PXED sur votre réseau. Vous devez d'abord déterminer le nombre de réseaux qui devront accueillir des clients PXE. L'exemple suivant configure le démon **pxed** de telle sorte qu'il s'exécute sur la même machine que le serveur DHCP :

```
pxeservertype       proxy_on_dhcp_server

subnet default
{
    vendor pxe
    {
      option  6    2       # Désactiver la découverte du serveur de
démarrage multi-diffusion
      option  8    1   2     9.3.4.5    9.3.4.6   2   1    9.3.149.29
                           # L'option ci-dessus fournit la liste des

serveurs de démarrage
      option  9    0       "PXE bootstrap server" \
                   1       "Microsoft Windows NT Boot Server"  \
                   2       "DOS/UNDI Boot Server"
      option  10  20    "secondes avant la sélection automatique de
la première option du menu de démarrage"
     }
}
```

Les sous–options du conteneur fournisseur ne sont envoyées aux clients PXE que si l'adresse IP du client figure dans la plage d'adresses IP du sous–réseau (de 9.3.149.0 à 9.3.149.255 par exemple).

L'exemple suivant configure le démon **pxed** de telle sorte qu'il s'exécute sur une autre machine que le serveur DHCP :

```
subnet default
{
    vendor pxe
    {
        option    6      10      # Le nom du fichier de démarrage est
présent dans la proposition de paquet
                                 # pxed du client.
        option    8      1   2     9.3.4.5    9.3.4.6   2    1
9.3.149.29
                           # L'option ci-dessus fournit la liste des
serveurs de démarrage
        option    9      0       "PXE bootstrap server" \
                         1       "Microsoft Windows NT Boot Server"
\
                         2       "DOS/UNDI Boot Server"
      option  10  20    "secondes avant la sélection automatique de
la première option du menu de démarrage"
        bootstrapserver          9.3.148.65
        pxebootfile    1    2    1   window.one
        pxebootfile    2    2    1   linux.one
        pxebootfile    1    2    1   hello.one
        client 6 10005a8ad14d any
        {
           pxebootfile    1    2    1   aix.one
           pxebootfile    2    2    1   window.one
        }
    }
Vendor pxeserver
{
    option     7     224.234.202.202
}
```

Le mot–clé **pxeservertype** n'est pas défini dans le fichier de configuration. La valeur par défaut **pdhcp_only** est donc utilisée, ce qui signifie que le serveur PXED est exécuté sur une machine différente que le serveur DHCP. Dans cette configuration, le serveur PXED est à l'écoute des paquets BINLD REQUEST/INFORM des clients sur deux ports (67 et 4011). L'option 7 est envoyée au serveur BINLD lorsque le serveur PXED reçoit un paquet REQUEST/INFORM en provenance de BINLD sur le port 67 et si l'option 60 est définie sur le serveur PXED.

La clause de base de données `db_file` indique la méthode à utiliser pour le traitement de cette portion du fichier de configuration. Les commentaires sont introduits par le symbole #. Tout le texte placé entre le # et la fin de la ligne est ignoré par le serveur PXED. Chaque ligne `option` est utilisée par le serveur pour indiquer au client ce qu'il doit faire. La section Sous–options du conteneur fournisseur PXE décrit les sous–options reconnues et prises en charge à l'heure actuelle. Pour savoir comment définir des options inconnues du serveur, reportez–vous à la section Syntaxe du fichier de serveur PXED pour le fonctionnement général du serveur.

Le fichier de configuration

Le fichier de configuration comprend une section d'adresses et une section de définition d'options, basées sur le concept des conteneurs, qui renferment les options, les modificateurs et, le cas échéant, d'autres conteneurs.

Un *conteneur* (qui est finalement une méthode de regroupement des options) fait appel à un identificateur pour classer les clients en plusieurs groupes. Les types de conteneur sont le sous–réseau, la classe, le fournisseur et le client. A l'heure actuelle, il n'existe pas de conteneur générique définissable par l'utilisateur. L'Identificateur définit le client de manière unique, de sorte qu'il soit possible de suivre sa trace même s'il est déplacé vers un autre sous–réseau. Il est possible d'utiliser plusieurs types de conteneur pour définir les droits d'accès du client.

Les *options* sont les identificateurs qui sont retournés au client, par exemple la passerelle par défaut et l'adresse de DNS.

Conteneurs

Lorsque le serveur DHCP reçoit une requête, le paquet est analysé et les clés d'identification permettent de déterminer les conteneurs, les options et les adresses à extraire.

L'exemple précédent présente un conteneur de sous–réseau. La clé d'identification est la position du client au sein du réseau. Si le client fait partie de ce réseau, alors il est intégré à ce conteneur.

Chaque type de conteneur utilise une option différente pour identifier les clients :

- Le conteneur sous–réseau utilise le champ `giaddr` ou l'adresse de l'interface réceptrice pour déterminer le sous–réseau d'origine du client.
- Le conteneur classe utilise la valeur de l'option 77 (User Site Class Identifier – identificateur de la classe du site utilisateur).
- Le conteneur fournisseur utilise la valeur de l'option 60 (Vendor Class Identifier – identificateur de la classe du fournisseur).
- Le conteneur client utilise la valeur de l'option 61 (Client Identifier – identificateur du client) pour les clients PXE et le champ `chaddr` du paquet BOOTP pour les clients BOOTP.

  Sauf pour les sous–réseaux, chaque conteneur accepte la spécification de la valeur de correspondance à l'aide d'expressions régulières.

A ces conteneurs, il faut ajouter un conteneur implicite, le conteneur *global*. Sauf spécification contraire ou refus explicite, les options et modificateurs placés dans le conteneur global s'appliquent à tous les conteneurs. La plupart des conteneurs peuvent être inclus dans d'autres conteneurs, ce qui implique une certaine visibilité. Les conteneurs peuvent ou non être associés à des plages d'adresses. Tel est le cas, par nature, des sous–réseaux.

Les règles de base s'appliquant aux conteneurs et sous–conteneurs sont les suivantes :

- Tous les conteneurs sont valides au niveau général.
- Les sous–réseaux ne doivent jamais être inclus dans d'autres conteneurs.
- Des conteneurs restreints ne peuvent englober des conteneurs réguliers du même type. (Par exemple, un conteneur doté d'une option autorisant uniquement la classe `Comptabilité` ne peut receler un conteneur doté d'une option autorisant toutes les classes commençant par la lettre "c". Ceci n'est pas autorisé.)
- Les conteneurs client restreints ne peuvent englober de sous–conteneurs.

En tenant compte des règles ci–dessus, vous pouvez générer une hiérarchie de conteneurs qui répartissent les options en différents groupes pour des clients ou des ensembles de clients spécifiques.

Comment sont gérées les options et adresses lorsqu'un client correspond à plusieurs conteneurs ? Le serveur DHCP reçoit les messages, il transmet la requête à la base de données (fichier **db_file** en l'occurrence) et une liste de conteneurs est générée. La liste est organisée par ordre de profondeur et de priorité. La priorité se définit comme une hiérarchie implicite au sein des conteneurs. Les conteneurs stricts ont une priorité supérieure à celle des conteneurs réguliers. Les clients, les classes, les fournisseurs et enfin, les sous–réseaux sont triés, dans cet ordre, et à l'intérieur de chaque conteneur en fonction de leur profondeur. Ceci aboutit à une liste allant du plus spécifique au moins spécifique. Par exemple :

```
Sous-réseau 1
--Classe 1
--Client 1
Sous-réseau 2
--Classe 1
----Fournisseur 1
----Client 1
--Client 1
```

L'exemple ci–dessus présente deux sous–réseaux, `Sous-réseau 1` et `Sous-réseau 2`. Il y a un nom de classe, `Classe 1`, un nom de fournisseur, `Fournisseur 1` et un nom de client, `Client 1`. `Classe 1` et `Client 1` sont définis en plusieurs endroits. Comme ils résident dans des conteneurs différents, leurs noms peuvent être identiques mais leurs valeurs, différentes. Si `Client 1` envoie un message au serveur DHCP depuis `Sous-réseau 1` avec `Classe 1` spécifiée dans sa liste d'options, le serveur DHCP va générer le chemin de conteneur suivant :

```
Sous-réseau 1, Classe 1, Client 1
```

Le conteneur le plus spécifique apparaît en dernier. Pour obtenir une adresse, la liste est étudiée dans l'ordre inverse de la hiérarchie et la première adresse disponible est retenue. Ensuite, l'étude de la liste de poursuit en remontant dans la hiérarchie afin d'obtenir les options. Les options peuvent remplacer des valeurs précédentes, sauf si une option *deny* a été incluse dans le conteneur. Par ailleurs, puisque `Classe 1` et `Client 1` figurent dans `Sous-réseau 1`, ils sont ordonnés en fonction de la priorité de leur conteneur. Si le même client se trouve dans `Sous-réseau 2` et envoie le même message, la liste de conteneur générée sera :

`Sous-réseau 2, Classe 1, Client 1` (au niveau de `Sous-réseau 2`), `Client 1` (au niveau de `Classe 1`)

`Sous-réseau 2` apparaît en premier, suivi de `Classe 1` , puis de `Client 1` au niveau de `Sous-réseau 2` (car cette instruction client ne se trouve qu'à un niveau en dessous dans la hiérarchie). Cette hiérarchie implique qu'un client correspondant à la première instruction client est moins spécifique que le client correspondant à `Client 1` de `Classe 1` au sein de `Sous-réseau 2`.

La priorité sélectionnée en fonction de la profondeur dans la hiérarchie prend le pas sur la priorité des conteneurs eux–mêmes. Par exemple, si le même client émet le même message, en précisant cette fois un identificateur de fournisseur, la liste de conteneur devient :

`Sous-réseau 2, Classe 1, Fournisseur 1, Client 1` (au niveau de `Sous-réseau 2`), `Client 1` (au niveau de `Classe 1`)

La priorité au niveau des conteneurs améliore les performances en matière de recherche car elle correspond à un concept général selon lequel les conteneurs client constituent le moyen le plus spécifique de définir un ou plusieurs clients. Le conteneur client contient des adresses plus spécifiques qu'un conteneur classe, lui–même plus spécifique qu'un conteneur fournisseur, le conteneur sous–réseau étant le moins spécifique de tous.

Adresses et plages d'adresses

Les plages d'adresses, obligatoires pour les conteneurs sous–réseau, peuvent être associées à tout type de conteneur. Chaque plage définie pour un conteneur doit être un sous–ensemble de la plage du conteneur parent et ne doit pas présenter de chevauchement avec la plage d'un autre conteneur. Par exemple, si une classe définie dans un sous–réseau est associée à une plage d'adresses, cette plage doit constituer un sous–ensemble des adresses de la plage du sous–réseau. En outre, le conteneur de la classe ne doit pas recouvrir, même partiellement, d'autres plages d'adresses au même niveau.

Les plages peuvent être définies sur la ligne du conteneur et modifiées au moyen d'instructions de plages et d'exclusion afin que des jeux d'adresse non contigus puissent être associés à un conteneur. Ainsi, si les dix premières adresses d'un sous–réseau sont

disponibles, ainsi que les dix suivantes, le sous–réseau peut spécifier ces adresses par plage dans la clause de sous–réseau afin de réduire l'utilisation de la mémoire et les risques de collision d'adresses avec d'autres clients ne se trouvant pas dans les plages spécifiées.

Dès qu'une adresse est sélectionnée, tout conteneur suivant dans la liste contenant les plages d'adresses est retiré de la liste, avec ses enfants. La raison en est que les options spécifiques au réseau dans les conteneurs supprimés ne sont pas valides si l'adresse n'est pas utilisée à partir de ce conteneur.

Options

Une fois la liste ponctionnée pour déterminer les adresses, un ensemble d'options est généré pour le client. Lors de ce processus de sélection, les nouvelles options remplacent les options précédemment sélectionnées, sauf si une clause *deny* est rencontrée, auquel cas l'option refusée est retirée de la liste envoyée au client. Cette méthode autorise les héritages à partir des conteneurs parents afin de réduire la quantité de données à spécifier.

Journalisation

Les paramètres de journalisation sont précisés dans un conteneur tel que la base de données, mais le mot de passe du conteneur est : **logging_info**. Au démarrage, il est conseillé d'activer le niveau de journalisation le plus élevé. En outre, il est préférable de configurer cette fonction préalablement à toute autre afin que les erreurs de configuration puissent être consignées après initialisation du sous–système de journalisation. Le mot–clé **logitem** active le niveau de journalisation ; si vous supprimez **logitem,** le niveau de journalisation sera désactivé. Les autres mots–clé concernant la journalisation permettent d'indiquer le nom du fichier journal, sa taille et le nombre de journaux utilisés en alternance.

Considérations de performance

Vous n'êtes pas sans savoir que certains mots–clé de configuration ainsi que la structure du fichier de configuration ont une incidence sur l'utilisation de la mémoire et les performances du serveur PXED.

Premièrement, il est possible d'éviter toute sollicitation excessive de la mémoire en appréhendant le modèle d'héritage des options des conteneurs parents vers les conteneurs enfants. Dans un environnement qui ne prend pas en charge les clients non répertoriés, l'administrateur doit expressément lister chaque client du fichier. Lorsque des options sont répertoriées pour chaque client en particulier, le serveur sollicite plus de capacité mémoire pour stocker cette structure de configuration arborescente que lorsque des options sont héritées d'un conteneur parent (conteneurs de sous–réseau, de réseau ou conteneurs globaux, par exemple). Par conséquent, l'administrateur doit vérifier la répétition ou non des options relatives au client au sein du fichier de configuration. Si tel est le cas, il doit décider si ces options peuvent ou non être définies dans le conteneur parent et partagées par l'ensemble des clients.

Deuxièmement, l'utilisation des entrées **logItem** INFO et TRACE entraîne la consignation de nombreux messages au cours du traitement de chaque message du client PXE. L'ajout d'une ligne au journal peut s'avérer une opération onéreuse. C'est pourquoi la limitation du volume de journalisation améliore les performances du serveur PXED. En cas de présomption d'erreur sur le serveur PXED, la journalisation peut être dynamiquement réactivée à l'aide de la commande SRC traceson.

Sous–options du conteneur fournisseur PXE

Dans le cadre de la prise en charge d'un client PXE, le serveur DHCP transmet l'option suivante au serveur BINLD, qui l'utilise pour sa configuration :

Opt Num	Type de données par défaut	Autorisée ?	Description
6	Nombre décimal	Oui	PXE_DISCOVERY_CONTROL. Limite 0–16. Ceci est un champ de bit. Bit 0 est le bit le moins significatif. **bit 0** S'il est défini, il désactive la découverte de diffusion. **bit 1** S'il est défini, il désactive la découverte de multidiffusion. **bit 2** S'il est défini, seuls les serveurs de PXE_BOOT_ SERVERS sont utilisés/acceptés. **bit 3** S'il est défini, et si un nom de fichier de démarrage est présent dans le paquet PXED initial, le fichier de démarrage est téléchargé (sans invite préalable). **bit 4–7** Doit être défini sur 0. Si cette option n'est pas fournie, le client suppose que tous les bits sont égaux à 0.
7	Un "dotted quad"	Oui	Adresse IP de multi–diffusion. Adresse IP de multi–diffusion de découverte du serveur de démarrage. Les serveurs dotés de cette fonctionnalité doivent écouter cette adresse de multi–diffusion. Cette option est requise si le bit de désactivation de la découverte multi–diffusion (bit 1) de l'option PXE_DISCOVERY_ CONTROL n'est pas défini.

8	*Boot server type(0–65535)*	Oui	PXE_BOOT_SERVERS *IP address count (0–256)* **Type 0** Microsoft Windows *IP address...IP address* NT Boot Server *Boot server type IP address* **Type 1** Intel LCM Boot Server *count IP address ...* **Type 3** DOS/UNDI Boot Server *IP address* **Type 4** NEC ESMPRO Boot Server **Type 5** WSoD Boot Server **Type 6** LCCM Boot Server **Type 7** CA Unicenter TNG Boot Server. **Type 8** HP OpenView Boot Server. **Types 9 à 32767** Réservés **Types 32768 à 65534** A l'usage du fournisseur **Type 65535** PXE API Test Server. Si *IP address count* a la valeur zéro pour un type de serveur, le client peut accepter des offres de n'importe quel serveur de démarrage de ce type. Les serveurs de démarrage ne répondent pas aux requêtes de découverte des types qu'ils ne prennent pas en charge.
9	*Boot server type (0–65535)*	Oui	PXE_BOOT_MENU *"description"* "order" du serveur de démarrage implicite dans le type. *"description"...menu order.*
10	*Délai d'attente en secondes (0–255)*	Oui	PXE_MENU_PROMPT *"prompt"* Le délai d'attente correspond au nombre de secondes avant la sélection automatique de la première option du menu de démarrage. Sur le système client, l'invite est affichée suivie du nombre de secondes restant avant cette sélection. Si l'utilisateur appuie sur la touche F8 sur le système client, un menu est affiché. Si cette option est fournie au client, le menu est affiché sans invite ni délai d'attente. Si le délai d'attente est égal à 0, la première option du menu est automatiquement sélectionnée. Si le délai d'attente est égal à 255, le menu et l'invite sont affichés sans sélection automatique ni délai d'attente.

Syntaxe du fichier de serveur PXED pour le fonctionnement général du serveur

Remarque : Les unités de temps (*time_units*) indiquées dans le tableau suivant sont facultatives et correspondent à un modificateur du temps réel. L'unité de temps par défaut est exprimée en minutes. Les valeurs autorisées sont les secondes (1), les minutes (60), les heures (3600), les jours (86400), les semaines (604800), les mois (2392000) et les années (31536000). Le nombre entre parenthèses est un multiplicateur appliqué à la valeur *n* spécifiée pour exprimer cette valeur en secondes.

Mot–clé	Forme	Sous–conteneurs ?	Valeur par défaut	Signification
database	database *db type*	Oui	Aucune	Conteneur principal renfermant les définitions des pools d'adresses, options et instructions d'accès client. *db type* est le nom du module chargé pour traiter cette portion du fichier. La seule valeur actuellement disponible est **db_file**.
logging_ info	logging_info	Oui	Aucune	Conteneur de journalisation principal définissant les paramètres de journalisation.
logitem	logitem NONE	Non	Non activé pour tous par défaut.	Active le niveau de journalisation. Plusieurs lignes sont autorisées.
	logitem SYSERR			
	logitem OBJERR			
	logitem PROTOCOL			
	logitem PROTERR			
	logitem WARN			
	logitem WARNING			
	logitem CONFIG			
	logitem EVENT			
	logitem PARSEERR			
	logitem ACTION			
	logitem ACNTING			
	logitem STAT			
	logitem TRACE			
	logitem RTRACE			
	logitem START			
numLog Files	numLogFiles *n*	Non	0	Indique le nombre de fichiers journaux à créer. Les journaux alternent lorsque le premier journal est rempli. *n* est le nombre de journaux à créer.
logFile Size	logFileSize *n*	Non	0	Indique la taille de chaque fichier journal, exprimée en unités de 1024 octets.

logFile Name	logFileName *path*	Non	Aucune	Indique le chemin d'accès au premier fichier journal. Le nom d'origine du fichier journal est *nomfichier* ou *nomfichier.extension*. *nomfichier* est limité à huit caractères. Lorsque la permutation des fichiers est effectuée, le premier fichier est renommé en conservant la base du nom, *nomfichier*, et en lui ajoutant un numéro, ou en remplaçant l'extension par un numéro. Par exemple, si le nom original du fichier est `file`, le nom du fichier après permutation devient `file01`. Si le nom du fichier d'origine est `file.log`, il devient `file.01`.
pxeserver type	pxeservertype *servertype*	Non	dhcp_only	Indique le type du serveur **dhcpsd**. *servertype* peut avoir la valeur **proxy_on_dhcp_server**, ce qui signifie que PXED est exécuté sur la même machine que le serveur DHCP et est à l'écoute des requêtes client PXE sur le port 4011 uniquement, ou la valeur par défaut **pdhcp_only**, ce qui signifie que PXED est exécuté sur une machine à part et doit écouter les paquets client sur les ports 67 et 4011.

Remarques sur la syntaxe du fichier de serveur PXED pour la base de données db_file :

Remarques :

1. Les unités de temps (*time_units*) indiquées dans le tableau suivant sont facultatives et correspondent à un modificateur du temps réel. L'unité de temps par défaut est exprimée en minutes. Les valeurs autorisées sont les secondes (1), les minutes (60), les heures (3600), les jours (86400), les semaines (604800), les mois (2392000) et les années (31536000). Le nombre entre parenthèses est un multiplicateur appliqué à la valeur *n* spécifiée pour exprimer cette valeur en secondes.
2. Les éléments spécifiés dans un conteneur peuvent être remplacés par ceux d'un sous–conteneur. Vous pouvez par exemple définir les clients BOOTP de manière globale, et, au sein d'un sous–réseau particulier, autoriser les clients BOOTP en indiquant le mot–clé supportBootp dans les deux conteneurs.
3. Les conteneurs client, classe et fournisseur acceptent les expressions régulières. Pour la classe et le vendeur, une chaîne entre guillemets dont le premier caractère à l'intérieur des guillemets est un point d'exclamation (!) indique que le reste de la chaîne doit être considéré comme une expression régulière. Le conteneur client accepte les expressions régulières dans les champs hwtype et hwaddr. Une chaîne unique est utilisée pour représenter les deux champs, selon la syntaxe suivante :

   `nombre_décimal-données`

   Si nombre_décimal est égal à zéro, les données constituent une chaîne ASCII. Pour tout autre nombre, les données sont des chiffres hexadécimaux.

Mot–clé	Forme	Sous–conteneurs ?	Valeur par défaut	Signification
subnet	subnet default	Oui	Aucune	Spécifie un sous–réseau sans plage associée. Il n'est utilisé par le serveur que lorsqu'il répond à un paquet INFORM émanant du client.
subnet	subnet *subnet id netmask* subnet *subnet id netmask range* subnet *subnet id netmask label*:*priority* subnet *subnet id netmask range label*:*priority*	Oui	Aucune	Spécifie un sous–réseau et un pool d'adresses. Toutes les adresses sont supposées faire partie du pool, sauf si une plage est spécifiée sur la ligne ou si les adresses sont modifiées ultérieurement dans le conteneur par une instruction de plage ou d'exclusion. La plage facultative est une paire d'adresses IP en format de "dotted quad" séparées par un tiret. Il est possible de préciser un label et une priorité. Ceux–ci sont utilisés dans les sous–réseaux virtuels pour identifier et classer les sous–réseaux du sous–réseau virtuel. Le label et la priorité sont séparés par un signe deux–points. Ces conteneurs ne sont autorisés qu'au niveau global ou au niveau du conteneur base de données.
subnet	subnet *subnet id range*	Oui	Aucune	Spécifie un sous–réseau qui s'inscrit dans un conteneur réseau. Il définit une plage d'adresses formant la totalité du sous–réseau, sauf si la plage facultative est indiquée. Le masque de réseau associé au sous–réseau est issu du conteneur réseau environnant. **Remarque:** Cette méthode est déconseillée au profit des autres formes de sous–réseaux.

option	option *number data* ... option *number*deny option * deny	Non	Aucune	Spécifie une option à envoyer à un client ou, dans le cas d'un refus (deny), une option qui ne doit pas être envoyée à un client. La clause optionnelle `* deny` signifie que toutes les options non spécifiées dans le conteneur en cours ne doivent pas être retournées au client. L'option *number*deny ne refuse que l'option spécifiée. *number* est un entier 8 bits non signé. *data* est spécifique à l'option (voir ci–dessus) ou peut être définie sous la forme d'une chaîne entre guillemets (texte ASCII) ou 0x*hexdigits* ou hex"*hexdigits*" ou encore hex "*hexdigits*". Si l'option correspond à un conteneur fournisseur, elle sera encapsulée avec les autres options dans une option 43.
exclude	exclude *an IP address* exclude *dotted_quad –dotted_ quad*	Non	Aucune	Modifie la plage sur le conteneur qui comporte l'instruction exclude. L'instruction exclude n'est pas valide au niveau des conteneurs de base de données ou au niveau général. L'instruction exclude supprime l'adresse ou la plage spécifiée de la plage actuelle sur le conteneur. Elle permet de créer des plages non contiguës pour sous sous–réseaux ou d'autres conteneurs.
range	range *IP_address* range *dotted_quad –dotted_ quad*	Non	Aucune	Modifie la plage sur le conteneur qui comporte l'instruction range. L'instruction range n'est pas valide au niveau des conteneurs de base de données ou au niveau général. S'il s'agit de la première plage du conteneur qui ne spécifie pas une plage sur la ligne de définition du conteneur, la plage du conteneur devient alors la plage spécifiée par l'instruction range. Toute instruction range suivante, ou toutes les instructions range dans le cas d'un conteneur spécifiant des plages dans sa définition sont ajoutées à la page actuelle. Avec l'instruction range, il est possible d'ajouter à la plage existante une adresse unique ou un jeu d'adresses. La plage doit être incorporée dans la définition du conteneur de sous–réseau.

client	client *hwtype hwaddr* NONE client *hwtype hwaddr* ANY client *hwtype hwaddr dotted_quad* client *hwtype hwaddr range*	Oui	Aucune	Spécifie un conteneur client qui empêche le client indiqué par *hwaddr* et *hwtype* d'obtenir une adresse. Si *hwtype* est 0, alors *hwaddr* est une chaîne ASCII. Sinon, *hwtype* correspond au type de matériel du client et *hwaddr* à l'adresse du matériel du client. Si *hwaddr* est une chaîne, des guillemets peuvent encadrer la chaîne. Si *hwaddr* est une chaîne hexadécimale, l'adresse peut être spécifiée sous la forme 0x*hexdigits* ou hex *digits*. *range* permet au client spécifié par *hwaddr* et *hwtype* d'obtenir une adresse faisant partie de cette *plage*. Pour faire référence à plusieurs clients, il faut utiliser une expression régulière.
class	class *string* class *string range*	Oui	Aucune	Spécifie un conteneur classe portant le nom *string*. La chaîne peut ou non être placée entre guillemets. Si oui, les guillemets sont supprimés avant la comparaison. Les guillemets sont obligatoires si la chaîne contient des espaces ou des tabulations. Ce conteneur est autorisé à tous les niveaux. Il est possible d'indiquer une plage pour spécifier le jeu d'adresses à proposer au client avec cette classe. La plage est soit une adresse IP en format de "dotted quad", soit deux adresses IP en format de "dotted quad" séparées par un tiret.

réseau	network *network id netmask* network *network id* network *network id range*	Oui	Aucune	Spécifie un ID de réseau à l'aide des informations de classe (par exemple 9.3.149.0 avec un masque de réseau de 255.255.255.0 correspond au réseau 9.0.0.0 255.255.255.0). Cette version du conteneur de réseau est utilisée pour englober les sous–réseaux partageant le même masque et le même ID de réseau. Lorsqu'une plage est fournie, toutes les adresses de la plage font partie du pool. La plage doit être comprise dans le réseau de l'ID de réseau. Elle fait appel à l'adresse intégrale de la classe. Elle n'est valide qu'au niveau général ou au niveau du conteneur de base de données. **Remarque:** Le mot–clé network est déconseillé au profit du conteneur de sous–réseau.
vendor	vendor *vendor_id* vendor *vendor_id* hex"" vendor *vendor_id* hex "" vendor *vendor_id* 0xdata vendor *vendor_id* "" vendor *vendor_id range* vendor *vendor_id range* hex"" vendor *vendor_id range* hex "" vendor *vendor_id range* 0xdata vendor *vendor_id range* ""	Oui	Aucune	Spécifie un conteneur de fournisseur. Les conteneurs fournisseur sont utilisés pour retourner l'option 43 au client. L'id de fournisseur peut être spécifié sous la forme d'une chaîne entre guillemets ou d'une chaîne binaire du type 0x*hexdigits* ou hex"*digits*". Il est possible d'ajouter à l'id de fournisseur une plage facultative, en utilisant deux "dotted quad" séparés par un tiret. A la suite de la plage facultative, une chaîne hexadécimale ou ASCII également facultative peut être indiquée comme première partie de l'option 43. Si des options figurent dans le conteneur, elles sont annexées aux données de l'option 43. Une fois toutes les options traitées, une option End Of Option List (fin de la liste d'options) est ajoutée aux données. Pour retourner les options en dehors d'une option 43, utilisez une expression régulière correspondant à tous les clients pour spécifier les options normales à renvoyer en fonction de l'ID fournisseur.

inoption	inoption *number option_data* inoption *number option_data range*	Oui	Aucune	Indique un conteneur à rapprocher d'une option entrante arbitraire définie par le client. *number* indique le numéro de l'option. *option_data* définit la clé correspondant au conteneur à sélectionner lors du choix de l'adresse et de l'option pour ce client. La clé *option_data* se présente sous forme de chaîne entre guillemets, d'adresse IP ou de nombre entier pour les options connues mais peut également se présenter sous forme de chaîne hexadécimale d'octets si elle est précédée des caractères `0x`. Pour les options que le serveur connaît mal, il est possible de définir une chaîne hexadécimale d'octets sur le même schéma. En outre, la valeur *option_data* peut faire référence à une expression régulière à rapprocher de la représentation en chaîne des données d'option du client. Ces expressions régulières se présentent sous la forme d'une chaîne entre guillemets (dont le premier caractère est un point d'exclamation `"!`). Les options peu connues du serveur se présentent sous forme de chaîne hexadécimale d'octets NON précédée des caractères `0x`.
virtual	virtual fill *id id* ... virtual sfill *id id* ... virtual rotate *id id* ... virtual srotate *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique. `fill` signifie utiliser toutes les adresses de ce conteneur avant de passer au suivant. `rotate` signifie sélectionner une adresse du pool suivant de la liste sur chaque requête. `sfill` et `srotate` sont identiques à `fill` et `rotate`, mais une recherche est effectuée pour savoir si le client correspond aux conteneurs, aux fournisseurs ou aux classes du sous–réseau. Si une correspondance permet d'obtenir une adresse, cette adresse est adoptée à partir du conteneur au lieu de suivre la politique indiquée. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.

inorder:	inorder: *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique de remplissage, ce qui signifie utiliser toutes les adresses de ce conteneur avant de passer au conteneur suivant. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.
balance:	balance: *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique de rotation, ce qui signifie utiliser l'adresse suivante du conteneur suivant. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.
bootstrap server	bootstrapserver *IP address*	Non	Aucune	Indique le serveur que les clients doivent utiliser comme point de départ vers les fichiers TFTP à l'issue de la réception de paquets BOOTP ou DHCP. Cette valeur complète le champ **siaddr** du paquet. Cette option est valide à tous les niveaux de conteneur.
giaddr field	giaddrfield *IP address*	Non	Aucune	Définit le champ giaddrfield pour les paquets de réponse. **Remarque :** Cette spécification n'est pas autorisée pour les protocoles BOOTP et DHCP, mais certains clients exigent le champ **giaddr** comme passerelle par défaut pour le réseau. En raison de ce risque de conflit, il est conseillé de n'utiliser giaddrfield qu'au sein d'un conteneur client, bien que l'option fonctionne à tous les niveaux.

bootfile	bootfile *path*	Non	Aucune	Indique le fichier de démarrage à utiliser dans la section fichier du paquet de réponse. Cette option peut être définie à tous les niveaux de conteneur. La politique bootfile définit comment les éléments spécifiés dans la section fichier du paquet entrant se conjuguent avec les instructions du fichier de démarrage et du répertoire personnel.
pxeboot file	pxebootfile *System Arch MajorVer MinorVer Bootfilename*	Non	Aucune	Indique le fichier de démarrage à donner à un client. L'analyseur du fichier de configuration génère une erreur si le nombre de paramètres après le mot–clé est inférieur à quatre, et il ignore les paramètres supplémentaires. Ce mot–clé ne peut être utilisé que dans un conteneur.

Pour plus d'informations sur les autres options, reportez–vous à la section Options connues du fichier de serveur DHCP.

Démon BINLD (Boot Image Negotiation Layer Daemon)

Le serveur BINLD constitue le troisième contact des clients PXE. Après avoir communiqué avec le serveur DHCP pour obtenir une adresse IP, et avec le serveur DHCP proxy PHE pour connaître la localisation du serveur de démarrage, ce dernier est contacté afin d'obtenir le chemin d'accès à partir duquel télécharger l'image de démarrage. Le client PXE peut revenir communiquer plusieurs fois avec le serveur de démarrage au cours de l'initialisation s'il a besoin de plusieurs fichiers pour son processus de démarrage.

La dernière étape du démarrage du réseau PXE est le téléchargement de l'image de démarrage fournie par le serveur de démarrage. La localisation du serveur TFTP et le nom du fichier qui doit être téléchargé sont donnés par le serveur de démarrage au client PXE.

Le serveur BINLD

A partir de la mise à jour version 4.3.3, le serveur BINLD est segmenté en trois composants principaux : une base de données, un moteur de protocole et un ensemble de routines de service, chaque élément disposant de ses propres informations de configuration.

La base de données BINLD

La base de données **db_file.dhcpo** est utilisée pour générer les options qui répondent à un paquet REQUEST d'un client. Les options renvoyées par la base de données dépendent du type de serveur choisi. Les options sont définies à l'aide du mot–clé **pxeservertype** dans le fichier **binld.cnf**.

A partir des informations du fichier de configuration, la base de données est amorcée et sa cohérence est vérifiée.

Le moteur de protocole BINLD

Le moteur de protocole PXED est basé sur Preboot Execution Environment (PXE) Specification Version 2.1 d'Intel, mais il reste compatible avec PXE Specification Version 1.1. Le moteur de protocole utilise la base de donnés pour déterminer quelles informations doivent être retournées au client.

Opérations BINLD enchaînées

Le dernier élément du serveur BINLD est en fait un ensemble d'opérations qui permettent d'assurer la continuité de l'exécution. Comme le serveur BINLD est du type enchaîné, ces opérations sont définies sous la forme de routines qui interviennent occasionnellement pour s'assurer du bon déroulement de l'exécution.

La première routine, ou routine **principale**, gère les requêtes SRC (par exemple **startsrc**, **stopsrc**, **lssrc**, **traceson** et **refresh**). Cette routine coordonne également toutes les opérations qui affectent toutes les routines et gère les signaux. Par exemple :

- A SIGHUP (–1) provoque un rafraîchissement de toutes les bases de données du fichier de configuration.
- A SIGTERM (–15) entraîne l'arrêt en douceur du serveur.

L'autre routine traite les paquets. Selon le type du serveur, une ou deux routines sont utilisées. L'une d'entre elles écoute le port 67 et la deuxième le port 4011. Chacune peut traiter une requête d'un client.

Configuration de BINLD

Par défaut, la configuration du serveur BINLD est effectuée par la lecture du fichier **/etc/binld.cnf**, qui spécifie la base de données initiale d'adresses et d'options du serveur. Le serveur est démarré à partir de Web-based System Manager, de SMIT ou via les commandes SRC.

La configuration de BINLD constitue la tâche la plus délicate dans le cadre de l'utilisation de BINLD sur votre réseau. Vous devez d'abord déterminer le nombre de réseaux qui devront accueillir des clients PXE. L'exemple suivant configure un serveur BINLD exécuté sur la même machine que le serveur DHCP :

```
pxeservertype       binld_on_dhcp_server

 subnet default
 {
     vendor pxe
     {
                 bootstrapserver  9.3.149.6      #TFTP server IP address
                 pxebootfile   1   2   1   window.one   1   0
                 pxebootfile   2   2   1   linux.one    2   3
                 pxebootfile   1   2   1   hello.one    3   4
                 client 6 10005a8ad14d any
                 {
                    pxebootfile   1   2   1   aix.one     5   6
                    pxebootfile   2   2   1   window.one  6   7
         }
      }
 }
```

Dans la configuration ci–dessus, le serveur BINLD écoute les paquets uni–diffusés d'un client sur le port 4011 et les paquets multi–diffusés sur ce même port si BINLD obtient l'adresse de multi–diffusion de dhcpsd/pxed. Le serveur BINLD répond aux paquets REQUEST/INFORM du client avec le nom du fichier de démarrage et l'adresse IP du serveur TFTP. Si BINLD ne trouve pas le fichier de démarrage avec une couche correspondante spécifiée par le client, il tente ensuite de trouver un fichier de démarrage pour la couche suivante. BINLD ne répond pas lorsqu'aucun fichier de démarrage ne correspond aux requêtes du client (*Type*, *SystemArch*, *MajorVers*, *MinorVers* et *Layer*).

L'exemple ci–dessous configure BINLD pour une exécution sur une machine à part (DHCP et PXED ne sont pas exécutés sur la même machine).

```
  subnet 9.3.149.0 255.255.255.0
  {

       vendor pxe
       {

       bootstrapserver         9.3.149.6      # Adresse IP du serveur TFTP.
       pxebootfile    1    2    1   window.one    1   0
       pxebootfile    2    2    1   linux.one     2   3
       pxebootfile    1    2    1   hello.one     3   4
       client 6 10005a8ad14d any
         {
         pxebootfile    1    2    1   aix.one      5   6
         pxebootfile    2    2    1   window.one   6   7
         }
       }
  }
```

Dans l'exemple ci–dessus, *pxeservertype* n'est pas défini, le type de serveur par défaut est donc **binld_only**. Le serveur BINLD écoute les paquets uni–diffusés d'un client sur le port 4011, les paquets diffusés et uni–diffusés sur le port 67 et les paquets multi–diffusés sur le port 4011 si BINLD obtient l'adresse de multi–diffusion de dhcpsd/pxed. Le nom du fichier de démarrage et l'adresse IP du serveur TFTP ne sont envoyés à un client PXE que si son adresse IP figure dans la plage d'adresses IP du sous–réseau (de 9.3.149.0 à 9.3.149.255).

L'exemple suivant configure BINLD pour une exécution sur la même machine que le serveur PXED :

```
pxeservertype       binld_on_proxy_server
subnet default
{
     vendor
     {
     bootstrapserver        9.3.149.6      # Adresse IP du serveur TFTP.
     pxebootfile    1    2    1   window.one    1    0
     pxebootfile    2    2    1   linux.one     2    3
     pxebootfile    1    2    1   hello.one     3    4
     client 6 10005a8ad14d any
       {
       pxebootfile    1    2    1   aix.one     5   6
       pxebootfile    2    2    1   window.one  6   7
       }
     }
}
```

Dans cette configuration, le serveur BINLD n'écoute les paquets multi–diffusés sur le port 4011 que si BINLD obtient une adresse de multi–diffusion de dhcpsd/pxed. S'il ne reçoit pas d'adresse de multidiffusion, BINLD est fermé et un message d'erreur est enregistré dans le fichier journal.

La clause de base de données `db_file` indique la méthode à utiliser pour le traitement de cette portion du fichier de configuration. Les commentaires sont introduits par le symbole #. Tout le texte placé entre le # et la fin de la ligne est ignoré par le serveur PXED. Chaque ligne `option` est utilisée par le serveur pour indiquer au client ce qu'il doit faire. La section Sous–options du conteneur fournisseur PXE décrit les sous–options reconnues et prises en charge à l'heure actuelle. Pour savoir comment définir des options inconnues du serveur, reportez–vous à la section Syntaxe du fichier de serveur BINLD pour le fonctionnement général du serveur.

Le fichier de configuration

Le fichier de configuration comprend une section d'adresses et une section de définition d'options, basées sur le concept des conteneurs, qui renferment les options, les modificateurs et, le cas échéant, d'autres conteneurs.

Un *conteneur* (qui est finalement une méthode de regroupement des options) fait appel à un identificateur pour classer les clients en plusieurs groupes. Les types de conteneur sont le sous–réseau, la classe, le fournisseur et le client. A l'heure actuelle, il n'existe pas de conteneur générique définissable par l'utilisateur. L'Identificateur définit le client de manière unique, de sorte qu'il soit possible de suivre sa trace même s'il est déplacé vers un autre sous–réseau. Il est possible d'utiliser plusieurs types de conteneur pour définir les droits d'accès du client.

Les *options* sont les identificateurs qui sont retournés au client, par exemple la passerelle par défaut et l'adresse de DNS.

Conteneurs

Lorsque le serveur DHCP reçoit une requête, le paquet est analysé et les clés d'identification permettent de déterminer les conteneurs, les options et les adresses à extraire.

L'exemple précédent présente un conteneur de sous–réseau. La clé d'identification est la position du client au sein du réseau. Si le client fait partie de ce réseau, alors il est intégré à ce conteneur.

Chaque type de conteneur utilise une option différente pour identifier les clients :

- Le conteneur sous–réseau utilise le champ giaddr ou l'adresse de l'interface réceptrice pour déterminer le sous–réseau d'origine du client.
- Le conteneur classe utilise la valeur de l'option 77 (User Site Class Identifier – identificateur de la classe du site utilisateur).
- Le conteneur fournisseur utilise la valeur de l'option 60 (Vendor Class Identifier – identificateur de la classe du fournisseur).
- Le conteneur client utilise la valeur de l'option 61 (Client Identifier – identificateur du client) pour les clients PXED et le champ chaddr du paquet BOOTP pour les clients BOOTP.

Sauf pour les sous–réseaux, chaque conteneur accepte la spécification de la valeur de correspondance à l'aide d'expressions régulières.

A ces conteneurs, il faut ajouter un conteneur implicite, le conteneur *global*. Sauf spécification contraire ou refus explicite, les options et modificateurs placés dans le conteneur global s'appliquent à tous les conteneurs. La plupart des conteneurs peuvent être inclus dans d'autres conteneurs, ce qui implique une certaine visibilité. Les conteneurs peuvent ou non être associés à des plages d'adresses. Tel est le cas, par nature, des sous–réseaux.

Les règles de base s'appliquant aux conteneurs et sous–conteneurs sont les suivantes :

- Tous les conteneurs sont valides au niveau général.
- Les sous–réseaux ne doivent jamais être inclus dans d'autres conteneurs.
- Des conteneurs restreints ne peuvent englober des conteneurs réguliers du même type. (Par exemple, un conteneur doté d'une option autorisant uniquement la classe `Comptabilité` ne peut receler un conteneur doté d'une option autorisant toutes les classes commençant par la lettre "c". Ceci n'est pas autorisé.)
- Les conteneurs client restreints ne peuvent englober de sous–conteneurs. En tenant compte des règles ci–dessus, vous pouvez générer une hiérarchie de conteneurs qui répartissent les options en différents groupes pour des clients ou des ensembles de clients spécifiques.

Comment sont gérées les options et adresses lorsqu'un client correspond à plusieurs conteneurs ? Le serveur DHCP reçoit les messages, il transmet la requête à la base de données (fichier **db_file** en l'occurrence) et une liste de conteneurs est générée. La liste est organisée par ordre de profondeur et de priorité. La priorité se définit comme une hiérarchie implicite au sein des conteneurs. Les conteneurs stricts ont une priorité supérieure à celle des conteneurs réguliers. Les clients, les classes, les fournisseurs et enfin, les sous–réseaux sont triés, dans cet ordre, et à l'intérieur de chaque conteneur en fonction de leur profondeur. Ceci aboutit à une liste allant du plus spécifique au moins spécifique. Par exemple :

```
Sous-réseau 1
--Classe 1
--Client 1
Sous-réseau 2
--Classe 1
----Fournisseur 1
----Client 1
--Client 1
```

Cet exemple présente deux sous–réseaux, `Sous-réseau 1` et `Sous-réseau 2`. Il y a un nom de classe, `Classe 1`, un nom de fournisseur, `Fournisseur 1` et un nom de client, `Client 1`. `Classe 1` et `Client 1` sont définis en plusieurs endroits. Comme ils résident dans des conteneurs différents, leurs noms peuvent être identique mais leurs valeurs, différentes. Si `Client 1` envoie un message au serveur DHCP depuis `Sous-réseau 1`

avec `Classe 1` spécifiée dans sa liste d'options, le serveur DHCP va générer le chemin de conteneur suivant :

```
Sous-réseau 1, Classe 1, Client 1
```

Le conteneur le plus spécifique apparaît en dernier. Pour obtenir une adresse, la liste est étudiée dans l'ordre inverse de la hiérarchie et la première adresse disponible est retenue. Ensuite, l'étude de la liste de poursuit en remontant dans la hiérarchie afin d'obtenir les options. Les options peuvent remplacer des valeurs précédentes, sauf si une option *deny* a été incluse dans le conteneur. Par ailleurs, puisque `Classe 1` et `Client 1` figurent dans `Sous-réseau 1`, ils sont ordonnés en fonction de la priorité de leur conteneur. Si le même client se trouve dans `Sous-réseau 2` et envoie le même message, la liste de conteneur générée sera :

`Sous-réseau 2, Classe 1, Client 1` (au niveau de `Sous-réseau 2`), `Client 1` (au niveau de `Classe 1`)

`Sous-réseau 2` apparaît en premier, suivi de `Classe 1`, puis de `Client 1` au niveau de `Sous-réseau 2` (car cette instruction client ne se trouve qu'à un niveau en dessous dans la hiérarchie). Cette hiérarchie implique qu'un client correspondant à la première instruction client est moins spécifique que le client correspondant à `Client 1` de `Classe 1` au sein de `Sous-réseau 2`.

La priorité sélectionnée en fonction de la profondeur dans la hiérarchie prend le pas sur la priorité des conteneurs eux–mêmes. Par exemple, si le même client émet le même message, en précisant cette fois un identificateur de fournisseur, la liste de conteneur devient :

`Sous-réseau 2, Classe 1, Fournisseur 1, Client 1` (au niveau de `Sous-réseau 2`), `Client 1` (au niveau de `Classe 1`)

La priorité au niveau des conteneurs améliore les performances en matière de recherche car elle correspond à un concept général selon lequel les conteneurs client constituent le moyen le plus spécifique de définir un ou plusieurs clients. Le conteneur client contient des adresses plus spécifiques qu'un conteneur classe, lui–même plus spécifique qu'un conteneur fournisseur, le conteneur sous–réseau étant le moins spécifique de tous.

Adresses et plages d'adresses

Les plages d'adresses, obligatoires pour les conteneurs sous–réseau, peuvent être associées à tout type de conteneur. Chaque plage définie pour un conteneur doit être un sous–ensemble de la plage du conteneur parent et ne doit pas présenter de chevauchement avec la plage d'un autre conteneur. Par exemple, si une classe définie dans un sous–réseau est associée à une plage d'adresses, cette plage doit constituer un sous–ensemble des adresses de la plage du sous–réseau. En outre, le conteneur de la classe ne doit pas recouvrir, même partiellement, d'autres plages d'adresses au même niveau.

Les plages peuvent être définies sur la ligne du conteneur et modifiées au moyen d'instructions de plages et d'exclusion afin que des jeux d'adresse non contigus puissent être associés à un conteneur. Ainsi, si les dix premières adresses d'un sous–réseau sont disponibles, ainsi que les dix suivantes, le sous–réseau peut spécifier ces adresses par plage dans la clause de sous–réseau afin de réduire l'utilisation de la mémoire et les risques de collision d'adresses avec d'autres clients ne se trouvant pas dans les plages spécifiées.

Dès qu'une adresse est sélectionnée, tout conteneur suivant dans la liste contenant les plages d'adresses est retiré de la liste, avec ses enfants. La raison en est que les options spécifiques au réseau dans les conteneurs supprimés ne sont pas valides si l'adresse n'est pas utilisée à partir de ce conteneur.

Options

Une fois la liste ponctionnée pour déterminer les adresses, un ensemble d'options est généré pour le client. Lors de ce processus de sélection, les nouvelles options remplacent

les options précédemment sélectionnées, sauf si une clause *deny* est rencontrée, auquel cas l'option refusée est retirée de la liste envoyée au client. Cette méthode autorise les héritages à partir des conteneurs parents afin de réduire la quantité de données à spécifier.

Journalisation

Les paramètres de journalisation sont précisés dans un conteneur tel que la base de données, mais le mot de passe du conteneur est : **logging_info**. Au démarrage, il est conseillé d'activer le niveau de journalisation le plus élevé. En outre, il est préférable de configurer cette fonction préalablement à toute autre afin que les erreurs de configuration puissent être consignées après initialisation du sous–système de journalisation. Le mot–clé **logitem** active le niveau de journalisation ; si vous supprimez **logitem,** le niveau de journalisation sera désactivé. Les autres mots–clé concernant la journalisation permettent d'indiquer le nom du fichier journal, sa taille et le nombre de journaux utilisés en alternance.

Considérations de performance

Vous n'êtes pas sans savoir que certains mots–clés de configuration ainsi que la structure du fichier de configuration ont une incidence sur l'utilisation de la mémoire et les performances du serveur PXED.

Premièrement, il est possible d'éviter toute sollicitation excessive de la mémoire en appréhendant le modèle d'héritage des options des conteneurs parents vers les conteneurs enfants. Dans un environnement qui ne prend pas en charge les clients non répertoriés, l'administrateur doit expressément lister chaque client du fichier. Lorsque des options sont répertoriées pour chaque client en particulier, le serveur sollicite plus de capacité mémoire pour stocker cette structure de configuration arborescente que lorsque des options sont héritées d'un conteneur parent (conteneurs de sous–réseau, de réseau ou conteneurs globaux, par exemple). Par conséquent, l'administrateur doit vérifier la répétition ou non des options relatives au client au sein du fichier de configuration. Si tel est le cas, il doit décider si ces options peuvent ou non être définies dans le conteneur parent et partagées par l'ensemble des clients.

Deuxièmement, l'utilisation des entrées **logItem** INFO et TRACE entraîne la consignation de nombreux messages au cours du traitement de chaque message du client PXE. L'ajout d'une ligne au journal peut s'avérer une opération onéreuse. C'est pourquoi la limitation du volume de journalisation améliore les performances du serveur PXED. En cas de présomption d'erreur sur le serveur PXED, la journalisation peut être dynamiquement réactivée à l'aide de la commande SRC traceson.

Syntaxe du fichier de serveur BINLD pour le fonctionnement général du serveur

Remarque : Les unités de temps (*time_units*) indiquées dans le tableau suivant sont facultatives et correspondent à un modificateur du temps réel. L'unité de temps par défaut est exprimée en minutes. Les valeurs autorisées sont les secondes (1), les minutes (60), les heures (3600), les jours (86400), les semaines (604800), les mois (2392000) et les années (31536000). Le nombre entre parenthèses est un multiplicateur appliqué à la valeur *n* spécifiée pour exprimer cette valeur en secondes.

Mot–clé	Forme	Sous–conte–neurs ?	Valeur par défaut	Signification
database	database *db type*	Oui	Aucune	Conteneur principal renfermant les définitions des pools d'adresses, options et instructions d'accès client. *db type* est le nom du module chargé pour traiter cette portion du fichier. La seule valeur actuellement disponible est **db_file**.

logging_info	logging_info	Oui	Aucune	Conteneur de journalisation principal définissant les paramètres de journalisation.
logitem	logitem NONE	Non	Non activé pour tous par défaut.	Active le niveau de journalisation. Plusieurs lignes sont autorisées.
	logitem SYSERR			
	logitem OBJERR			
	logitem PROTOCOL			
	logitem PROTERR			
	logitem WARN			
	logitem WARNING			
	logitem CONFIG			
	logitem EVENT			
	logitem PARSEERR			
	logitem ACTION			
	logitem ACNTING			
	logitem STAT			
	logitem TRACE			
	logitem RTRACE			
	logitem START			
numLogFiles	numLogFiles *n*	Non	0	Indique le nombre de fichiers journaux à créer. Les journaux alternent lorsque le premier journal est rempli. *n* est le nombre de journaux à créer.
logFileSize	logFileSize *n*	Non	0	Indique la taille de chaque fichier journal, exprimée en unités de 1024 octets.
logFileName	logFileName *path*	Non	Aucune	Indique le chemin d'accès au premier fichier journal. Le nom d'origine du fichier journal est *nomfichier* ou *nomfichier.extension*. *nomfichier* est limité à huit caractères. Lorsque la permutation des fichiers est effectuée, le premier fichier est renommé en conservant la base du nom, *nomfichier*, et en lui ajoutant un numéro, ou en remplaçant l'extension par un numéro. Par exemple, si le nom original du fichier est `file`, le nom du fichier après permutation devient `file01`. Si le nom du fichier d'origine est `file.log`, il devient `file.01`.

pxeservertype	pxeservertype *servertype*	Non	dhcp_only	Indique le type de serveur dhcpsd. *servertype* peut avoir l'une des valeurs suivantes : **binld_on_dhcp_server** Cela signifie que **BINLD** est exécuté sur la même machine que le serveur DHCP, qu'il écoute les requêtes client PXE sur le port 4011 et l'adresse de multi–diffusion si celle–ci est reçue du serveur DHCP / PXED.**binld_on_proxy_server** Cela signifie que **BINLD**est exécuté sur la même machine que le serveur PXED et écoute les requêtes client PXE sur l'adresse de multi–diffusion si celle–ci est reçue du serveur DHCP / PXED. La valeur par défaut est **binld_only** : le serveur BINLD est exécuté sur une machine à part et doit écouter les paquets du client sur les ports 67 et 4011 et sur l'adresse de multidiffusion si celle–ci est reçue du serveur DHCP / PXED.
dhcp_or_proxy _address	dhcp_or_proxy_address *IP address*	Non	Aucune	Ce mot–clé fournit l'adresse IP du serveur dhcp ou pxed auquel le serveur BINLD peut envoyer un paquet uni–diffusé de type REQUEST/INFORM pour recevoir l'adresse de multi–diffusion. Ce mot–clé n'est défini que lorsque le serveur dhcp ou pxed est exécuté sur un sous–réseau différent de BINLD.

Syntaxe du fichier de serveur BINLD pour le fonctionnement général du serveur

Remarques :

1. Les unités de temps (*time_units*) indiquées dans le tableau suivant sont facultatives et correspondent à un modificateur du temps réel. L'unité de temps par défaut est exprimée en minutes. Les valeurs autorisées sont les secondes (1), les minutes (60), les heures (3600), les jours (86400), les semaines (604800), les mois (2392000) et les années (31536000). Le nombre entre parenthèses est un multiplicateur appliqué à la valeur *n* spécifiée pour exprimer cette valeur en secondes.
2. Les éléments spécifiés dans un conteneur peuvent être remplacés par ceux d'un sous–conteneur. Vous pouvez par exemple définir les clients BOOTP de manière globale, et, au sein d'un sous–réseau donné, autoriser les clients BOOTP en indiquant le mot–clé supportBootp dans les deux conteneurs.
3. Les conteneurs client, classe et fournisseur acceptent les expressions régulières. Pour la classe et le vendeur, une chaîne entre guillemets dont le premier caractère à l'intérieur des guillemets est un point d'exclamation (!) indique que le reste de la chaîne doit être considéré comme une expression régulière. Le conteneur client accepte les expressions régulières dans les champs hwtype et hwaddr. Une chaîne unique est utilisée pour représenter les deux champs, selon la syntaxe suivante :

   ```
   nombre_décimal-données
   ```

   Si nombre_décimal est égal à zéro, les données constituent une chaîne ASCII. Pour tout autre nombre, les données sont des chiffres hexadécimaux.

Mot–clé	Forme	Sous–conteneurs ?	Valeur par défaut	Signification
subnet	subnet default	Oui	Aucune	Spécifie un sous–réseau sans plage associée. Ce sous–réseau est utilisé par un serveur uniquement pour répondre à un paquet INFORM d'un client et si aucun conteneur de sous–réseau ne correspond à l'adresse de ce dernier.
subnet	subnet *subnet id netmask* subnet *subnet id netmask range* subnet *subnet id netmask label*:*priority* subnet *subnet id netmask range label*:*priority*	Oui	Aucune	Spécifie un sous–réseau et un pool d'adresses. Toutes les adresses sont supposées faire partie du pool, sauf si une plage est spécifiée sur la ligne ou si les adresses sont modifiées ultérieurement dans le conteneur par une instruction de plage ou d'exclusion. La plage facultative est une paire d'adresses IP en format de "dotted quad" séparées par un tiret. Il est possible de préciser un label et une priorité. Ceux–ci sont utilisés dans les sous–réseaux virtuels pour identifier et classer les sous–réseaux du sous–réseau virtuel. Le label et la priorité sont séparés par un signe deux–points. Ces conteneurs ne sont autorisés qu'au niveau global ou au niveau du conteneur base de données.

subnet	subnet *subnet id range*	Oui	Aucune	Spécifie un sous–réseau qui s'inscrit dans un conteneur réseau. Il définit une plage d'adresses formant la totalité du sous–réseau, sauf si la plage facultative est indiquée. Le masque de réseau associé au sous–réseau est issu du conteneur réseau environnant. **Remarque :** Cette méthode est déconseillée au profit des autres formes de sous–réseaux.
option	option *number data* ...	Non	Aucune	Spécifie une option à envoyer à un client ou, dans le cas d'un refus (deny), une option qui ne doit pas être envoyée à un client. La clause option `* deny` signifie que toutes les options non spécifiées dans le conteneur en cours ne doivent pas être retournées au client. L'option *number*deny ne refuse que l'option spécifiée. *number* est un entier 8 bits non signé. *data* est spécifique à l'option (voir ci–dessus) ou peut être définie sous la forme d'une chaîne entre guillemets (texte ASCII) ou 0x*hexdigits* ou hex"*hexdigits*" ou encore hex "*hexdigits*". Si l'option correspond à un conteneur fournisseur, elle sera encapsulée avec les autres options dans une option 43.
	option *number*deny			
	option * deny			
exclude	exclude *an IP address*	Non	Aucune	Modifie la plage sur le conteneur qui comporte l'instruction exclude. L'instruction exclude n'est pas valide au niveau des conteneurs de base de données ou au niveau général. L'instruction exclude supprime l'adresse ou la plage spécifiée de la plage actuelle sur le conteneur. Elle permet de créer des plages non contiguës pour sous sous–réseaux ou d'autres conteneurs.
	exclude *dotted_quad–dotted_quad*			
range	range *IP_address*	Non	Aucune	Modifie la plage sur le conteneur qui comporte l'instruction range. L'instruction range n'est pas valide au niveau des conteneurs de base de données ou au niveau général. S'il s'agit de la première plage du conteneur qui ne spécifie pas une plage sur la ligne de définition du conteneur, la plage du conteneur devient alors la plage spécifiée par l'instruction range. Toute instruction range suivante, ou toutes les instructions range dans le cas d'un conteneur spécifiant des plages dans sa définition sont ajoutées à la page actuelle. Avec l'instruction range, il est possible d'ajouter à la plage existante une adresse unique ou un jeu d'adresses. La plage doit être incorporée dans la définition du conteneur de sous–réseau.
	range *dotted_quad–dotted_quad*			

client	client *hwtype hwaddr* NONE client *hwtype hwaddr* ANY client *hwtype hwaddr dotted_quad* client *hwtype hwaddr range*	Oui	Aucune	Spécifie un conteneur client qui empêche le client indiqué par *hwaddr* et *hwtype* d'obtenir une adresse. Si *hwtype* est 0, alors *hwaddr* est une chaîne ASCII. Sinon, *hwtype* correspond au type de matériel du client et *hwaddr* à l'adresse du matériel du client. Si *hwaddr* est une chaîne, des guillemets peuvent encadrer la chaîne. Si *hwaddr* est une chaîne hexadécimale, l'adresse peut être spécifiée sous la forme 0x*hexdigits* ou hex *digits*. *range* permet au client spécifié par *hwaddr* et *hwtype* d'obtenir une adresse faisant partie de cette *plage*. Pour faire référence à plusieurs clients, il faut utiliser une expression régulière.
class	class *string* class *string range*	Oui	Aucune	Spécifie un conteneur classe portant le nom *string*. La chaîne peut ou non être placée entre guillemets. Si oui, les guillemets sont supprimés avant la comparaison. Les guillemets sont obligatoires si la chaîne contient des espaces ou des tabulations. Ce conteneur est autorisé à tous les niveaux. Il est possible d'indiquer une plage pour spécifier le jeu d'adresses à proposer au client avec cette classe. La plage est soit une adresse IP en format de "dotted quad", soit deux adresses IP en format de "dotted quad" séparées par un tiret.
réseau	network *network id netmask* network *network id* network *network id range*	Oui	Aucune	Spécifie un ID de réseau à l'aide des informations de classe (par exemple 9.3.149.0 avec un masque de réseau de 255.255.255.0 correspond au réseau 9.0.0.0 255.255.255.0). Cette version du conteneur de réseau est utilisée pour englober les sous–réseaux partageant le même masque et le même ID de réseau. Lorsqu'une plage est fournie, toutes les adresses de la plage font partie du pool. La plage doit être comprise dans le réseau de l'ID de réseau. Elle fait appel à l'adresse intégrale de la classe. Elle n'est valide qu'au niveau général ou au niveau du conteneur de base de données. **Remarque :** Le mot–clé network est déconseillé au profit du conteneur de sous–réseau.

vendor	vendor *vendor_id* vendor *vendor_id* hex”” vendor *vendor_id* hex ”” vendor *vendor_id* 0xdata vendor *vendor_id* ”” vendor *vendor_id range* vendor *vendor_id range* hex”” vendor *vendor_id range* hex ”” vendor *vendor_id range* 0xdata vendor *vendor_id range* ”” vendor pxe vendor pxeserver	Oui	Aucune	Spécifie un conteneur de fournisseur. Les conteneurs fournisseur sont utilisés pour retourner l'option 43 au client. L'id de fournisseur peut être spécifié sous la forme d'une chaîne entre guillemets ou d'une chaîne binaire du type 0x*hexdigits* ou hex”*digits*”. Il est possible d'ajouter à l'id de fournisseur une plage facultative, en utilisant deux ”dotted quad” séparés par un tiret. A la suite de la plage facultative, une chaîne hexadécimale ou ASCII également facultative peut être indiquée comme première partie de l'option 43. Si les options figurent dans le conteneur, elles sont annexées aux données de l'option 43. Une fois toutes les options traitées, une option End Of Option List (fin de la liste d'options) est ajoutée aux données. Pour retourner les options en dehors d'une option 43, utilisez une expression régulière correspondant à tous les clients pour spécifier les options normales à renvoyer en fonction de l'ID fournisseur. **pxe** après le mot–clé **vendor** crée un conteneur fournisseur pour PXEClient. **pxeserver** après le mot–clé **vendor** crée un conteneur fournisseur pour PXEServer.

inoption	inoption *number option_data* inoption *number option_data range*	Oui	Aucune	Indique un conteneur à rapprocher d'une option entrante arbitraire définie par le client. *number* indique le numéro de l'option. *option_data* définit la clé correspondant au conteneur à sélectionner lors du choix de l'adresse et de l'option pour ce client. La clé *option_data* se présente sous forme de chaîne entre guillemets, d'adresse IP ou de nombre entier pour les options connues mais peut également se présenter sous forme de chaîne hexadécimale d'octets si elle est précédée des caractères `0x`. Pour les options que le serveur connaît mal, il est possible de définir une chaîne hexadécimale d'octets sur le même schéma. En outre, la valeur *option_data* peut faire référence à une expression régulière à rapprocher de la représentation en chaîne des données d'option du client. Ces expressions régulières se présentent sous la forme d'une chaîne entre guillemets (dont le premier caractère est un point d'exclamation `"!`). Les options peu connues du serveur se présentent sous forme de chaîne hexadécimale d'octets NON précédée des caractères `0x`.
virtual	virtual fill *id id* ... virtual sfill *id id* ... virtual rotate *id id* ... virtual srotate *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique. `fill` signifie utiliser toutes les adresses de ce conteneur avant de passer au suivant. `rotate` signifie sélectionner une adresse du pool suivant de la liste sur chaque requête. `sfill` et `srotate` sont identiques à `fill` et `rotate`, mais une recherche est effectuée pour savoir si le client correspond aux conteneurs, aux fournisseurs ou aux classes du sous–réseau. Si une correspondance permet d'obtenir une adresse, cette adresse est adoptée à partir du conteneur au lieu de suivre la politique indiquée. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.
inorder:	inorder: *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique de remplissage, ce qui signifie utiliser toutes les adresses de ce conteneur avant de passer au conteneur suivant. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.

balance:	balance: *id id* ...	Non	Aucune	Spécifie un sous–réseau virtuel avec une politique de rotation, ce qui signifie utiliser l'adresse suivante du conteneur suivant. Il peut y avoir autant d'ID que nécessaire. *id* est soit l'ID de sous–réseau de la définition de sous–réseau, soit le label de cette même définition. Le label est nécessaire si plusieurs sous–réseaux partagent le même ID de sous–réseau.
boots trapserver	bootstrap server *IP address*	Non	Aucune	Indique le serveur que les clients doivent utiliser comme point de départ vers les fichiers TFTP à l'issue de la réception de paquets BOOTP ou DHCP. Cette valeur complète le champ **siaddr** du paquet. Cette option est valide à tous les niveaux de conteneur.
giaddrfield	giaddrfield *IP address*	Non	Aucune	Définit le champ giaddrfield pour les paquets de réponse. **Remarque :** Cette spécification n'est pas autorisée pour les protocoles BOOTP et DHCP, mais certains clients exigent le champ **giaddr** comme passerelle par défaut pour le réseau. En raison de ce risque de conflit, il est conseillé de n'utiliser giaddrfield qu'au sein d'un conteneur client, bien que l'option fonctionne à tous les niveaux.
bootfile	bootfile *path*	Non	Aucune	Indique le fichier de démarrage à utiliser dans la section fichier du paquet de réponse. Cette option peut être définie à tous les niveaux de conteneur. La politique bootfile définit comment les éléments spécifiés dans la section fichier du paquet entrant se conjuguent avec les instructions du fichier de démarrage et du répertoire personnel.
pxebootfile	pxebootfile *SystemArch MajorVer MinorVer Bootfilename Type Layer*	Non	Aucune	Indique le fichier de démarrage à donner à un PXEClient. L'analyseur du fichier de configuration génère une erreur si le nombre de paramètres après le mot–clé est inférieur à 4 et les ignore s'il est supérieur à 7. Si 4 paramètres sont indiqués, il suppose les valeurs Type = 0 et Layer = 0. Ce mot–clé ne peut être utilisé que dans un conteneur.

Pour plus d'informations sur les autres options, reportez–vous aux sections Options connues du fichier de serveur DHCP, page 4-106 et Sous–options du conteneur fournisseur PXE, page 4-136.

Démons TCP/IP

Les démons (ou *serveurs*) sont des process qui fonctionnent en continu, en arrière-plan, pour exécuter des fonctions requises par d'autres process. TCP/IP fournit des démons pour implémenter certaines fonctions sur le système. Leur exécution en arrière-plan n'interrompt pas les autres processus (à moins qu'ils en soient chargés).

Les démons sont appelés par des commandes au niveau de la gestion système, par d'autres démons ou scripts shell. Vous pouvez également les contrôler à l'aide du démon **inetd**, du script shell **rc.tcpip** et du contrôleur SRC (System Resource Controller).

Sous-systèmes et sous-serveurs

Un *sous-système* est un démon ou serveur contrôlé par SRC. Un *sous-serveur* est un démon contrôlé par un sous-système. (Les commandes et noms de démon sont généralement suffixés par un **d**.) Sous–système et sous–serveurs sont deux catégories opposées et incompatibles : un démon ne peut relever des deux catégories à la fois. Le seul sous-système TCP/IP qui contrôle d'autres démons est **inetd**. Ainsi, tout sous-serveur TCP/IP est également un sous-serveur **inetd**.

Les démons TCP/IP contrôlés par SRC sont :

Sous-systèmes

gated	Fournit des fonctions de routage de passerelle et prend en charge les protocoles RIP (Routing Information Protocol), RIPng (Routing Information Protocol Next Generation), EGP (Exterior Gateway Protocol), BGP (Border Gateway Protocol) et BGP4+, HELLO, OSPF (Open Shortest Path First), IS–IS (Intermediate System to Intermediate System), ICMP et ICMPv6 (Internet Control Message Protocol /Router Discovery). Le démon **gated** prend également le protocole SNMP (Simple Network Monitoring Protocol) en charge. Le démon **gated** est l'un des deux démons de routage dédiés aux adresses de réseau. Le démon **gated** est préféré au démon **routed** car il admet davantage de protocoles de passerelle.
inetd	Appelle et planifie l'exécution d'autres démons à la réception des demandes de services de démons. Ce démon peut aussi en lancer d'autres. **inetd** est aussi appelé " super démon ".
iptrace	Suivi des paquets au niveau interface pour les protocoles Internet.
named	Fournit la fonction d'appellation au protocole de serveur de noms DOMAIN.
routed	Gère les tables de routage de réseau et prend en charge le protocole RIP (Routing Information Protocol). Le démon **gated** est préféré au démon **routed** car il admet davantage de protocoles de passerelle.
rwhod	Diffuse des messages à l'ensemble des hôtes, toutes les trois minutes, et stocke l'information relative aux utilisateurs connectés et à l'état du réseau. Utilisez **rwhod** avec précaution car il monopolise une part importante des ressources machine.
timed	Fournit la fonction serveur horaire.

Remarque : Les démons **routed** et **gated** relèvent de la catégorie des sous-systèmes TCP/IP. N'exécutez pas la commande **startsrc –g tcpip**, qui lance ces deux démons de routage avec tous les autres sous-systèmes TCP/IP. Ces deux démons lancés ensemble produiraient des résultats imprévisibles.

Les démons TCP/IP contrôlés par le sous-système **inetd** sont :

Sous-serveurs inetd

comsat	Avertit les utilisateurs de l'arrivée d'un courrier.
fingerd	Dresse un compte rendu concernant l'état de tous les utilisateurs connectés et l'état du réseau sur l'hôte distant spécifié. Ce démon utilise le protocole FINGER.
ftpd	Assure le transfert des fichiers pour un processus client en appliquant le protocole FTP (File Transfer Protocol).
rexecd	Assure la fonction de serveur hôte étranger, pour la commande **rexec**.
rlogind	Effectue la connexion à distance pour la commande **rlogin**.
rshd	Effectue la fonction serveur d'exécution des commandes à distance pour les commandes **rcp** et **rsh**.
talkd	Apport de la fonction conversation à la commande **talk.**
syslogd	Lecture et consignation des messages système.Ce démon appartient au groupe de sous-systèmes **Remote Access Service** (RAS).
telnetd	Apport de la fonction serveur au protocole TELNET.
tftpd	Assure la fonction serveur pour le protocole TFTP (Trivial File Transfer Protocol).
uucpd	Gère les communications entre BNU et TCP/IP.

Fonction SRC

Le contrôleur de ressources système (SRC) permet, entre autres, de lancer les démons, les arrêter et suivre leurs activités. De plus, SRC permet de grouper des démons en sous-systèmes et sous-serveurs.

Cet outil a été conçu pour aider l'administrateur système à contrôler les démons. Ce contrôle s'effectue au-delà des indicateurs et paramètres disponibles pour chaque commande de démon.

Pour en savoir plus sur SRC, reportez-vous à la section Contrôleur SRC dans *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*.

Commandes SRC

Les commandes SRC sont applicables à un seul démon, à un groupe de démons ou à un démon et à ceux qu'il contrôle (sous-système avec sous-serveurs). Par ailleurs, certains démons TCP/IP ne répondent pas à toutes les commandes SRC. Voici la liste des commandes SRC disponibles pour contrôler des démons TCP/IP et leurs exceptions.

startsrc	Démarre tous les sous-systèmes TCP/IP et sous-serveurs **inetd**, sans exception. La commande **startsrc** fonctionne pour tous les sous-systèmes TCP/IP et sous-serveurs **inetd**.
stopsrc	Arrête tous les sous-systèmes TCP/IP et sous-serveurs **inetd**, sans exception. Cette commande s'appelle également **stop normal**. La commande **stop normal** permet aux sous-systèmes de traiter tout le travail en cours et d'y mettre fin en douceur. Pour les sous–serveurs **inetd**, toutes les connexions en attente sont lancées et celles en exécution, terminées. La commande **stop normal** fonctionne pour tous les sous-systèmes TCP/IP et sous-serveurs **inetd**.

stopsrc –f Arrête tous les sous-systèmes TCP/IP et sous-serveurs **inetd**, sans exception. Cette commande s'appelle également **stop normal**. La commande **stop force** arrête immédiatement tous les sous–systèmes. Pour les sous-serveurs **inetd**, toutes les connexions en cours ou en attente sont immédiatement terminées.

refresh Rafraîchit les sous–systèmes et sous–serveurs suivants : sous–systèmes **inetd**, **syslogd**, **named**, **dhcpsd** et **gated**.

lssrc Fournit un bref compte rendu de l'état du sous–système spécifié (actif ou non) et des sous–serveurs **inetd**. Fournit un bref compte rendu d'état de **inetd** accompagné du nom, de l'état et de la description du sous–serveur, du nom de la commande et des arguments qui ont permis de le lancer.

lssrc –l Fournit un bref compte rendu d'état accompagné d'informations supplémentaires (état détaillé) sur les sous systèmes.

gated Etat de la mise au point ou du suivi, protocoles de routage activés, tables de routage, signaux acceptés avec leur fonctions.

inetd Etat de la mise au point, liste des sous-serveurs actifs avec état succinct, signaux acceptés avec leurs fonctions.

named Etat de la mise au point, informations sur le fichier **named.conf**.

dhcpsd Etat de la mise au point, toutes les adresses IP contrôlées et leur état actuel.

routed Etat de la mise au point et du suivi, état des informations de routage source, tables de routage.

syslogd Données de configuration de **syslogd**.

La commande **lssrc –l** indique également l'état détaillé des sous–serveurs **inetd**. L'état détaillé comprend un compte rendu et des informations sur la connexion active. Certains sous–serveurs fournissent des informations supplémentaires. Il s'agit de :

ftpd Etat de la mise au point et de la journalisation.

telnetd Type d'émulation de terminal.

rlogind Etat de la mise au point.

fingerd Etat de la mise au point et de la journalisation.

Les sous-serveurs **rwhod** et **timed** ne fournissent pas d'état détaillé.

traceson	Active la mise au point au niveau socket. Utilisez la commande **trpt** pour mettre la sortie en forme. Cette commande n'est pas prise en charge par les sous-systèmes **timed** et **iptraced**.
tracesoff	Désactive la mise au point au niveau socket. Utilisez la commande **trpt** pour mettre la sortie en forme. Cette commande n'est pas prise en charge par les sous-systèmes **timed** et **iptraced**.

Pour des exemples d'utilisation, reportez-vous à la description de la commande qui vous intéresse. Pour en savoir plus sur SRC, reportez-vous à la section Contrôleur SRC dans *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices* .

Configuration du démon inetd

Pour configurer le démon **inetd** :

1. Définissez les sous-serveurs que le démon doit appeler en ajoutant un sous-serveur **inetd**.
2. Définissez ses caractéristiques de relance, en modifiant les caractéristiques de relance du démon **inetd**.

Configuration des tâches du démon inetd

Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Démarrage du démon **inetd**	**smit mkinetd**	**startsrc –s inetd**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems**. Cliquez avec le bouton droit de la souris sur un sous–système inactif, puis sélectionnez **Activate**.
Modification des caractéristiques de relance du démon **inetd**	**smit chinetd** ou **smit lsinetd**		Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems** —> **Selected** —> **Properties**.
Arrêt du démon **inetd**	**smit rminetd**	**stopsrc –s inetd**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems**. Cliquez avec le bouton droit de la souris sur un sous–système actif, puis sélectionnez —> **Deactivate**.
Liste des sous–serveurs **inetd**	**smit inetdconf**		Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems**.
Ajout d'un sous-serveur **inetd**[1]	**smit mkinetdconf**	modifiez **/etc/inetd.conf** puis exécutez **refresh –s inetd** ou **tuez –1** *inetdPID*[2]	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems** —> **Subsystems** (menu déroulant) —> **New inetd Subserver**.

Modification/Affichage des caractéristiques d'un sous–serveur **inetd**	**smit inetdconf**	modifiez **/etc/inetd.conf** puis exécutez **refresh –s inetd** ou **tuez –1** *inetdPID*[2]	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems** —> **Selected** —> **Properties**.
Suppression d'un sous-serveur **inetd**	**smit rminetd**	modifiez **/etc/inetd.conf** puis exécutez **refresh –s inetd** ou **tuez –1** *inetdPID*[2]	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems** —> **Selected** —> **Deactivate**.

Remarques :

1. Ajouter un sous-serveur **inetd** revient à configurer le démon **inetd** pour qu'il puisse appeler le sous-serveur lorsque nécessaire.
2. La commande **refresh** ou **kill** signale au démon **inetd** les modifications apportées à son fichier de configuration.

Services réseau client

Les services réseau client (accessibles via le raccourci Web-based System Manager **wsm** ou via le raccourci **smit clientnet**) sont les protocoles TCP/IP applicables sous ce système d'exploitation. Chaque protocole ou service est identifié par le numéro de port qu'il utilise sur le réseau, d'où l'expression **port connu**. Par commodité, ces numéros de port peuvent être associés à des noms ou numéros. Par exemple, le protocole de messagerie TCP/IP qui utilise le port 25 est connu sous le nom **smtp**. Si un protocole est déclaré (pas de marque de commentaire) dans le fichier **/etc/services**, il peut être utilisé par un hôte.

Par défaut, tous les protocoles TCP/IP sont définis dans ce fichier **/etc/services**. Vous n'avez donc pas besoin de configurer ce fichier. Cependant, si vous avez écrit vos propres programmes client/serveur, vous pouvez être amené à les déclarer dans le fichier **/etc/services** et à leur réserver un nom et un numéro de port. Si vous décidez d'ajouter un service à **/etc/services**, notez que les ports 0 à 1024 sont réservés au système.

Tâches des services réseau client			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Liste des services disponibles	**smit lsservices**	Affichez **/etc/services**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Services**.
Ajout d'un service	**smit mkservices**	Editez **/etc/services**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Services** —> **New Service**.

Modification/ affichage des caractéristiques d'un service	**smit chservices**	Editez **/etc/services**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Services**. Sélectionnez un service, puis cliquez sur **Selected** —> **Properties**.
Suppression d'un service	**smit rmservices**	Editez **/etc/services**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Services**. Sélectionnez un service, puis cliquez sur **Selected** —> **Delete**.

Services réseau serveur

Les services réseau serveur se composent du contrôle de l'accès distant, du démarrage ou de l'arrêt de TCP/IP, et de la gestion du pilote d'unité **pty**, comme indiqué dans le tableau suivant.

Le pilote d'unité **pty** est installé automatiquement avec le système. Par défaut, ce pilote, configuré pour des liaisons symboliques 16 BSD, est disponible dès l'amorçage.

Tâches des services réseau serveur			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Contrôle d'accès à distance	Reportez–vous à "Exécution de commandes à distance et "Restrictions d'accès FTP".		Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Access Control**. Cliquez avec le bouton droit de la souris sur **Remote Access**, puis sélectionnez **Properties**.
Démarrage, redémarrage ou arrêt des sous-systèmes TCP/IP	**smit otherserv**	Reportez–vous à System Resource Control (SRC) page 4-162.	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **Subsystems**. Cliquez avec le bouton droit de la souris sur un sous–système, puis sélectionnez **Properties**.
Modification/affichage des caractéristiques du pilote d'unité pty	**smit chgpty**	**chdev –l pty0 –P –a num=** *X* *X* étant une valeur comprise entre 0 et 64	

Désactivation du pilote d'unité pty	**smit pty** puis sélectionnez **Retrait du PTY ; conserver définition**		
Activation du pilote d'unité pty	**smit pty** puis sélectionnez **Configuration du PTY défini**		
Génération d'un compte rendu d'erreur	**smit errpt**		
Suivi de pty	**smit trace**		

Routage TCP/IP

Cette section traite des points suivants :

Une *route* indique l'itinéraire des paquets à travers le réseau Internet. Elle ne définit pas le parcours complet, mais seulement le segment entre un hôte et une passerelle vers la destination (ou une autre passerelle). Il existe trois types de route :

route hôte	Passerelle capable d'envoyer les paquets vers un hôte ou une passerelle d'un autre réseau.
route réseau	Passerelle capable d'envoyer les paquets vers n'importe quel hôte d'un réseau spécifique.
route par défaut	Passerelle utilisable lorsqu'aucune route hôte ou réseau n'est définie.

Les routes sont définies dans la *table de routage* du noyau. Chaque définition donne des informations sur les réseaux accessibles à partir de l'hôte local et sur les passerelles disponibles pour atteindre les réseaux distants. A réception d'un datagramme, la passerelle recherche dans la table de routage l'étape suivante du parcours.

A partir de Aix 5.1, vous pouvez ajouter plusieurs routes dans la table de routage du noyau pour indiquer la même destination. La recherche d'un routage évalue toutes les routes qui correspondent à la demande, puis choisit la route ayant la distance métrique la plus courte. Si la recherche trouve plusieurs routes de même distance, elle choisit la route plus adéquate. Si les deux critères sont équivalents pour plusieurs routes, la recherche se concentre sur les critères alternatifs des routes correspondantes.

Routage statique ou dynamique

TCP/IP propose deux types de routage : *statique* ou *dynamique*. Avec le routage statique, la table de routage est gérée manuellement à l'aide d'une commande de **routage**. Le routage statique est conseillé lorsqu'un réseau communique avec un ou plusieurs réseaux. Toutefois, si ce type de routage est pratique lorsque la communication se limite à deux ou trois réseaux, il devient fastidieux sur une plus grande échelle, avec la multiplication du nombre de passerelles.

Avec le routage dynamique, ce sont les démons qui mettent à jour la table de routage automatiquement. Les démons de routage reçoivent en permanence les informations émises par d'autres démons de routage, et mettent systématiquement à jour la table de routage en conséquence.

TCP/IP propose deux démons de routage dynamique : **routed** et **gated**. Le démon **gated** gère les protocoles RIP (Routing Information Protocol), RIPng (Routing Information Protocol Next Generation), EGP (Exterior Gateway Protocol), BGP (Border Gateway Protocol) et BGP4+, HELLO (Defense Communications Network Local–Network Protocol), OSPF

(Open Shortest Path First), IS–IS (Intermediate System to Intermediate System), ainsi que ICMP et ICMPv6 (Internet Control Message Protocol) / Router Discovery simultanément. Le démon **gated** prend également le protocole SNMP (Simple Network Monitoring Protocol) en charge. Le démon **routed** n'admet que le protocole RIP.

Les démons de routage peuvent fonctionner en mode *passif* ou *actif* (selon l'option définie à leur lancement). En mode actif, ils diffusent périodiquement des informations de routage sur leur réseau local aux passerelles et aux hôtes, et en reçoivent d'eux. En mode passif, ils se limitent à recevoir les informations et ne tiennent pas à jour les passerelles distantes.

Ces deux types de routage sont applicables aux passerelles mais aussi à d'autres hôtes d'un réseau. Les travaux de routage statique fonctionnent de la même façon pour les passerelles que pour les autres hôtes. Les démons de routage dynamique, toutefois, doivent être exécutés en mode passif (quiet) sur les hôtes qui ne sont pas des passerelles.

Passerelles

Les passerelles sont des types de routeur. Les *routeurs* interconnectent des réseaux et assurent la fonction de routage. Certains routeurs opèrent le routage au niveau de l'interface de réseau ou de la couche physique.

Les *passerelles*, quant à elles, assurent le routage au niveau de la couche réseau. Elles reçoivent les datagrammes IP des autres passerelles ou hôtes, les transmettent aux hôtes du réseau local et acheminent les datagrammes IP d'un réseau à l'autre. Par exemple, une passerelle reliant deux réseaux en anneau à jeton est équipée de deux cartes de réseau en anneau à jeton dotée chacune de sa propre interface de réseau en anneau à jeton. Pour la transmission des informations, la passerelle reçoit les datagrammes via une interface de réseau et les envoie par l'autre. Les passerelles contrôlent périodiquement leurs connexions réseau à partir de messages d'état sur les interfaces.

Pour l'aiguillage des paquets, les passerelles se fondent sur le réseau de destination et non sur l'hôte de destination. Ainsi, elles n'ont pas à garder trace des diverses destinations hôte possibles d'un paquet. Au lieu de cela, elles acheminent les paquets en fonction du réseau de l'hôte de destination. C'est le réseau de destination qui se charge ensuite d'envoyer les paquets à l'hôte de destination. Généralement, une passerelle ne requiert qu'une capacité limitée de stockage disque (éventuellement) et de mémoire centrale.

La distance à parcourir entre l'hôte émetteur et l'hôte destinataire dépend du n à traverser (*sauts de passerelles* à traverser). 0 si la passerelle est rattachée directement au réseau, 1 si le réseau est accessible via une passerelle, etc. La distance d'un message s'exprime généralement en nombre de passerelles, ou *nombre de bonds* (ou *distance métrique*).

Passerelles intérieures et extérieures

Les passerelles intérieures font partie du même système autonome. Elles communiquent entre elles à l'aide des protocoles RIP (Routing Information Protocol), RIPng (Routing Information Protocol Next Generation), Intermediate System to Intermediate System, OSPF (Open Shortest Path First protocol) ou du protocole HELLO. Les passerelles extérieures appartiennent à des systèmes autonomes distincts. Elles utilisent les protocoles EGP (Exterior Gateway Protocol), BGP (Border Gateway Protocol) ou BGP4+.

Prenons l'exemple de deux systèmes autonomes. Le premier correspond à tous les réseaux administrés par la société Widget. Le second correspond à tous les réseaux administrés par la société Gadget. La société Widget possède une machine pomme, qui est la passerelle de Widget pour Internet. La société Gadget possède une machine orange, qui est la passerelle de Gadget pour Internet. Les deux sociétés possèdent plusieurs réseaux distincts en interne. Les passerelles reliant les réseaux internes sont des passerelles intérieures. Mais les passerelles pomme et orange sont extérieures.

Chaque passerelle extérieure ne communique pas avec toutes les autres passerelles extérieures. En fait, la passerelle extérieure acquiert un ensemble de passerelles limitrophes (les autres passerelles extérieures) avec lesquelles elle communique. Ces passerelles limitrophes ne sont pas définies par une proximité géographique, mais plutôt par les communications qui s'établissent entre elles. Les passerelles limitrophes, à leur tour,

possèdent d'autres passerelles limitrophes extérieures. Ainsi, les tables de routage des passerelles extérieures sont mises à jour et les informations de routage sont diffusées vers l'ensemble des passerelles extérieures.

Les informations de routage sont expédiées avec les coordonnées (R,D), R étant le réseau cible et D la distance à parcourir (et donc le coût correspondant) pour l'atteindre. Chaque passerelle indique les réseaux qui lui sont accessibles et le coût de leur accès. La passerelle réceptrice détermine les chemins les plus courts et les indique aux passerelles limitrophes. Ainsi, chaque passerelle extérieure reçoit en continu des informations (et met alors à jour ses tables de routage) qu'elle retransmet aux passerelles limitrophes.

Protocoles de passerelle

Toute passerelle, interne ou externe, communique avec les autres via des protocoles. Voici une présentation succincte des protocoles de passerelle TCP/IP courants :

Protocole HELLO

Le protocole HELLO est utilisé par les passerelles intérieures pour communiquer entre elles. HELLO est chargé de calculer le chemin d'accès le plus court (en durée) aux autres réseaux.

RIP (Routing Information Protocol)

Le protocole RIP est également utilisé par les passerelles intérieures pour communiquer entre elles. Comme le protocole HELLO, RIP calcule le chemin d'accès le plus court aux autres réseaux. A la différence de HELLO cependant, RIP calcule la distance en nombre de sauts, et non en durée. Comme le démon **gated** enregistre toutes les distances métriques en interne en tant que durée, il convertit les nombres de sauts calculés par RIP en durée.

Routing Information Protocol Next Generation

RIPng est le protocole RIP étendu qui permet de gérer IPv6.

OSPF (Open Shortest Path First)

Le protocole OPSF est utilisé par les passerelles intérieures pour communiquer entre elles. Ce protocole de communication est plus approprié que RIP pour les réseaux complexes comprenant plusieurs routeurs. Il fournit un routage multi–itinéraire au même coût.

EGP (Exterior Gateway Protocol)

Les passerelles extérieures utilisent ce protocole pour communiquer entre elles. Le protocole EGP ne calcule par le plus court chemin vers les autres réseaux. Il indique simplement si un réseau particulier est accessible ou non.

Border Gateway Protocol (BGP)

Les passerelles extérieures utilisent ce protocole pour communiquer entre elles. Ce protocole permet l'échange d'informations d'accessibilité entre des systèmes autonomes, mais il fournit davantage de fonctions que le protocole EGP. BGP utilise les attributs de chemin pour fournir des informations supplémentaires sur chaque route afin de sélectionner la plus appropriée.

Border Gateway Protocol 4+

BGP4+ est le protocole BGP version 4, qui gère IPv6 et propose d'autres fonctions étendues par rapport aux versions précédentes.

IS–IS (Intermediate System to Intermediate System)

Les passerelles intérieures utilisent le protocole IS–IS pour communiquer entre elles. Ce protocole de communication permet de router des paquets IP et ISO/CLNP et, comme OSPF, il utilise un algorithme de détection du chemin le plus court pour déterminer les routes les plus rapides.

Planification des passerelles

Avant de configurer les passerelles de votre réseau, vous devez :

1. Evaluez le nombre de passerelles à utiliser
   (reportez–vous à Evaluation du nombre de passerelles à utiliser).
2. Déterminez le type de routage à utiliser
   (reportez–vous à Détermination du type de routage à utiliser).

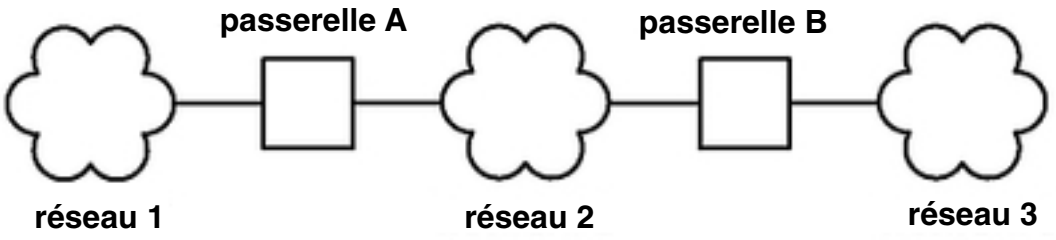
Nombre de passerelles

Le nombre de passerelles nécessaires dépend :

- du nombre de réseaux à connecter,
- du type de connexion des réseaux,
- du niveau d'activité des réseaux connectés.

Par exemple, si les utilisateurs des réseau 1, réseau 2 et réseau 3 doivent tous communiquer ensembles (comme le montre la figure Exemple de configuration de passerelle).

Figure 25. Configuration de passerelle simple Cette illustration contient trois nuages réseau un, deux et trois. Les réseaux un et deux sont connectés avec la passerelle A. Les réseaux deux et trois sont connectés avec la passerelle B.



Pour relier le réseau 1 directement au réseau 2, vous devez utiliser une première passerelle (passerelle A). Pour relier le réseau 2 directement au réseau 3, vous devez utiliser une autre passerelle (passerelle B). Supposons maintenant que les routes appropriées sont déterminées et que tous les utilisateurs des trois réseaux parviennent à communiquer.

Cependant, si le réseau 2 est très occupé, les communications entre le réseau 1 et le réseau 3 peuvent s'en trouver ralenties. De plus, si la communication entre ces deux réseaux est la plus importante, il peut être utile de les connecter directement. Pour ce faire, vous devez ajouter deux passerelles supplémentaires, une sur le réseau 1 (passerelle C), l'autre sur le réseau 3 (passerelle D), reliées par une connexion directe. Cette solution n'est peut–être pas suffisante, une passerelle pouvant raccorder plus de deux réseaux.

Un moyen plus efficace peut consister à connecter directement la passerelle A à la passerelle B et au réseau 2, ce qui suppose d'équiper A et B d'une seconde carte réseau. En règle générale, le nombre de connexions réseau assuré par une passerelle est limité au nombre de cartes réseau qu'elle peut prendre en charge.

Type de routage

Si votre réseau est limité et sa configuration relativement fixe, le routage statique est une solution satisfaisante. En revanche, si votre réseau est étendu et sa configuration très variable, il est préférable d'opter pour un routage dynamique. Une solution intermédiaire peut également être envisagée en panachant les routages statique et dynamique. Par exemple, il est possible de définir statiquement certaines routes et d'autoriser la mise à jour d'autres routes par les démons. Les routes statiques créées ne sont ni notifiées aux autres passerelles ni mises à jour par les démons de routage.

Routage dynamique

Déterminez le démon de routage à utiliser en fonction du type de passerelle nécessaire et des protocoles qu'elle peut prendre en charge. S'il s'agit d'une passerelle intérieure et qu'elle ne requiert que le protocole RIP, optez pour le démon **routed**. Sinon, utilisez **gated**.

Remarque : Vous risquez d'obtenir des résultats imprévisibles si les démons **gated** et **routed** sont exécutés sur le même hôte simultanément.

Configuration d'une passerelle

Pour définir une machine comme passerelle, procédez comme suit. Dans un souci de clarté, on suppose que la passerelle doit être connectée à deux réseaux et qu'elle a déjà fait l'objet d'une configuration minimale sur un des deux réseaux.

1. Installez et configurez la deuxième carte de réseau, si ce n'est déjà fait. (Reportez–vous à Installation d'une carte réseau, page 4-36 et à Configuration et gestion des cartes, page 4-37.)
2. Choisissez une adresse IP pour la seconde interface de réseau et configurez l'interface comme indiqué à Configuration d'une interface de réseau, page 4-53.
3. Ajoutez une route d'accès au second réseau.
4. Pour utiliser une machine comme routeur interréseau sur les réseaux TCP/IP, entrez :

   ```
   no -o ipforwarding=1
   ```

5. La passerelle peut désormais accéder aux deux réseaux directement raccordés.
   a. Pour que le routage statique serve à communiquer avec des hôtes et réseaux en dehors de ces deux réseaux, ajoutez les routes nécessaires.
   b. Pour le routage dynamique, procédez comme indiqué dans Configuration du démon routed, page 4-175 ou dans 1Configuration du démon gated, page 4-175. Si votre interréseau doit rejoindre le réseau Internet, suivez les instructions de la section Obtention d'un numéro de système autonome, page 4-179.

Configuration d'une passerelle			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Affichage du tableau de routage	**smit lsroute**	**netstat –rn**[1]	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Static Routes** —> **Statistics**.

Ajout d'une route statique	**smit mkroute**	**route ajoutée** *destination passerelle*[2]	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Static Routes**. Complétez les informations dans **Add/Change a static route** (ajout/modification d'une route statique) : **Destination Type** (type de destination), **Gateway address** (adresse de la passerelle), **Network interface name** (menu déroulant de l'interface de réseau), **Subnet mask** (masque de sous–réseau), **Metric (coût de la distance métrique)** et **Enable active dead gateway detection** (activer la détection des passerelles non opérationnelles). Cliquez sur **Add/Change Route** (ajout/modification d'une route).
Suppression d'une route statique	**smit rmroute**	**route supprimée** *destination passerelle*[2]	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Static Routes**. Sélectionnez une route, puis cliquez sur **Delete Route** (supprimer la route).
Vidage de la table de routage	**smit fshrttbl**	**vidage de la route**	Software —> **Network** —> **TCPIP (IPv4 and IPv6)** —> **TCPIP Protocol Configuration** —> **TCP/IP** —> **Configure TCP/IP** —> **Advanced Methods** —> **Static Routes** —> **Delete All**.

Remarques :

1. La table est divisée en colonnes, où sont répertoriés l'adresse de destination, l'adresse de passerelle, les indicateurs, le nombre de sauts et l'interface de réseau. Pour la description de ces colonnes, reportez-vous à la commande **netstat** dans le manuel *AIX 5L Version 5.2 Commands Reference*. En cas d'échec de livraison de trames, si les tables de routage sont correctes, un ou plusieurs des événements ci-dessous sont probablement en cause :
   - réseau défaillant,
   - passerelle ou hôte distant défaillant,
   - passerelle ou hôte distant en panne, ou non disponible pour réceptionner des trames,
   - hôte distant ne disposant pas de route retour au réseau source.
2. *destination* représente l'adresse ou le nom symbolique de l'hôte ou réseau de destination et *passerelle*, l'adresse ou le nom symbolique de la passerelle. (Une route implicite a 0 comme valeur de destination.)

Sécurité des routes

Les routes peuvent être sécurisées en limitant leur accès à certains utilisateurs. Les restrictions d'accès sont basées sur les ID de groupe primaire des utilisateurs. Avec la commande **route**, vous pouvez établir une liste de 32 ID groupe maximum et les autoriser ou non à utiliser une route. Si la liste contient des groupes autorisés, n'importe quel utilisateur de n'importe quel groupe a accès à la route. Si au contraire, la liste est formée de groupes non autorisés, seuls les utilisateurs n'appartenant pas aux groupes de la liste ont accès à la route. L'utilisateur racine a accès à toutes les routes.

En outre, les groupes peuvent être associés à une interface via la commande **ifconfig**. Dans ce cas, tout paquet à expédier peut utiliser n'importe quelle route dont l'accès est autorisé aux groupes associés à son interface en entrée.

Si plusieurs routes ont la même destination, les réceptions de réacheminement ICMP pour cette destination sont ignorés et la recherche de MTU d'accès n'est pas effectuée sur ces routes.

Détection des passerelles non opérationnelles

A partir d'AIX 5.1 et des versions ultérieures, vous pouvez configurer un hôte pour qu'il détecte si une passerelle est désactivée et pour qu'il modifie la table de routage en fonction des situations. Lorsque l'option de réseau **–passive_dgd** est définie sur `1`, la détection passive des passerelles non opérationnelles est activée pour l'ensemble du système. Si les demandes ARP consécutives pour **dgd_packets_lost** n'obtiennent pas de réponse, cette passerelle est considérée non opérationnelle et le système attribue aux distances métriques (également nommées *nombre de bonds* ou *coût*) de toutes les routes empruntant cette passerelle les valeurs les plus élevées possibles. Après un dépassement de temps (minutes) associé à **dgd_retry_time**, les coûts des routes sont restaurés sur la base des valeurs définies par l'utilisateur. L'hôte est également impliqué dans l'échec des connexions TCP. Si les paquets TCP consécutifs **dgd_packets_lost** sont perdus, l'entrée ARP de la passerelle est supprimée et la connexion TCP tente la route suivante la plus appropriée. Si la passerelle est effectivement non opérationnelle, à l'utilisation suivante de la passerelle, les actions citées ci–dessus se reproduisent. Les paramètres **passive_dgd**, **dgd_packets_lost**, and **dgd_retry_time** peuvent tous être configurés à l'aide de la commande **no**.

Vous pouvez également configurer les hôtes pour qu'ils appliquent la détection des passerelles non opérationnelles selon la route à l'aide de l'indicateur **–active_dgd** de la commande **route**. La détection des passerelles non opérationnelles exécute un ping sur toutes les passerelles utilisées par les routes associées à la détection selon l'intervalle en seconde lié au paramètre **dgd_ping_time**. Si la passerelle ne donne pas de réponse, l'exécution du ping est répétée plus rapidement en fonction de la valeur associée à **dgd_packets_lost**. Si la passerelle ne donne toujours pas de réponse, le système augmente tous les coûts des routes qui utilisent cette passerelle. La passerelle poursuit l'exécution du ping et lorsque la réponse arrive, les coûts des routes sont restaurés sur la base des valeurs définies par l'utilisateur. Le paramètre **dgd_ping_time** peut être configuré à l'aide de la commande **no**.

Normalement, la détection des passerelles non opérationnelles est plus utile pour les hôtes avec un routage statique qu'avec un routage dynamique. La détection des passerelles non opérationnelles comporte une charge réduite et elle est conseillée pour les réseaux ayant des passerelles redondantes. Toutefois, la détection des passerelles non opérationnelles est appliquée uniquement dans l'optique d'offrir le meilleur service possible. Certains protocoles, tels que UDP, ne fournissent aucun retour d'informations à l'hôte en cas d'échec de la transmission des données ; dans de telles circonstances, la détection des passerelles non opérationnelles ne peut prendre aucune mesure.

La détection des passerelles non opérationnelles est plus utile lorsqu'un hôte doit découvrir immédiatement l'arrêt d'une passerelle. Toutefois, du fait des requêtes répétées vers les passerelles définies, elle implique une certaine charge sur le réseau. La détection des passerelles non opérationnelles est recommandée uniquement pour les hôtes qui fournissent des services vitaux et les réseaux ayant un nombre limité d'hôtes.

Remarque : La détection des passerelles non opérationnelles et les protocoles de routage utilisés par les démons **gated** et **routed** exécutent une fonction similaire en détectant des modifications de la configuration du réseau et en ajustant la table de routage. Toutefois, ils utilisent des mécanismes différents et s'ils sont exécutés en même temps, ils peuvent entrer en conflit l'un avec l'autre. Pour cette raison, la détection des passerelles non opérationnelles ne doit pas être utilisée sur les systèmes exécutant les démons **gated** ou **routed**.

Clonage de route

Le clonage de route permet de créer une route hôte pour tous les hôtes avec lesquels un système communique. Lorsque vous êtes sur le point d'envoyer du trafic sur le réseau, une recherche est effectuée dans la table de routage pour trouver une route jusqu'à l'hôte. Si une route spécifique est trouvée jusqu'à l'hôte, elle est utilisée. Dans le cas contraire, une route réseau ou la route par défaut peut être trouvée. Si l'indicateur de clonage 'c' est défini pour la route, une route hôte pour la destination est créée à l'aide de la passerelle de la route clonée. Les recherches suivantes effectuées sur la table de routage trouvent la route hôte clonée. Les routes clonées sont associées à l'indicateur 'W'. Ces routes expirent et sont supprimées de la table de routage si elles restent inutilisées pendant **route_expire** minutes. Vous pouvez modifier **route_expire** à l'aide de la commande `no` .

La fonction de clonage de route est utilisée principalement par le protocole de détection MTU de chemin dans AIX afin de lui permettre de suivre les informations MTU de chemin pour chaque destination avec laquelle il communique. Si les options de réseau **tcp_pmtu_discover** ou **udp_pmtu_discover** (qui peuvent être définies avec la commande `no`) sont égales à 1, l'indicateur de clonage est activé pour toutes les routes réseau du système. Dans AIX 4.3.3 et les versions ultérieures, la détection MTU de chemin est activée par défaut.

Suppression manuelle de routes dynamiques

Si le démon **routed** est actif, *aucune route supprimée manuellement n'est* remplacée par les informations RIP entrantes (du fait des contrôles d'E/S). Si vous utilisez le démon **gated** sans l'indicateur –n, la route supprimée manuellement *est* remplacée par celle fournie par les informations RIP entrantes.

Configuration du démon routed

Pour configurer le démon **routed** :

1. Supprimez le symbole de commentaire (#) et modifiez la clause associée au démon routed dans le script du shell **/etc/rc.tcpip**. Cette commande initialise automatiquement le démon **routed** à chaque lancement du système.
   - Indiquez le mode d'exécution souhaité : actif (indicateur **–s**) ou passif (indicateur**–q**).
   - Activez éventuellement le suivi des paquets (indicateur **–t**). Vous pouvez également le faire pendant l'exécution du démon **routed**, via la commande **kill**. Cette commande communique au démon un signal **SIGUSR1**. Ce signal peut également servir à incrémenter le niveau de suivi sur quatre niveaux. Vous pouvez également désactiver le suivi de paquet pendant l'exécution du démon **routed,** via la commande **kill** : cette commande communique au démon un signal **SIGUSR2**. Pour en savoir plus, reportez-vous au démon **routed** et à la commande **kill**.

– Activez éventuellement la mise au point (indicateur **–d**). Précisez alors également le fichier journal dans lequel consigner les informations de mise au point (ou indiquez que vous souhaitez les diriger vers l'écran de la console).

– Indiquez si vous exécutez le démon **routed** sur une passerelle (indicateur **–g**).

Remarque : Un hôte non passerelle peut exécuter le démon **routed**, mais en mode passif uniquement.

2. Identifiez tous les réseaux connus en les répertoriant dans le fichier **/etc/network.** Pour en savoir plus, reportez-vous à la section Networks File Format for TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*. Un exemple du fichier **networks** est proposé dans le répertoire **/usr/samples/tcpip**.

3. Définissez dans le fichier **/etc/gateways** les routes d'accès à toutes les passerelles connues qui ne sont pas directement connectées à votre réseau. Pour des exemples détaillés d'entrées de fichier **/etc/gateways.**, reportez-vous à la section Gateways File Format for TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*. Un exemple du fichier **gateways** est proposé dans le répertoire **/usr/samples/tcpip**.

Attention : N'exécutez pas le démon **routed** et le démon **gated** sur la même machine. Des résultats imprévisibles pourraient survenir.

Configuration du démon gated

Pour configurer le démon **gated**, procédez comme suit :

1. Déterminez les protocoles de passerelle appropriés pour votre système. Vous pouvez utiliser les protocoles de routage EGP, BGP, RIP, RIPng, HELLO, OSPF, ICMP/Router Discovery ou IS–IS. Vous pouvez également prévoir le protocole SNMP, qui permet d'afficher et de modifier à partir d'un hôte distant les informations de gestion d'un élément de réseau.

Remarque : Utilisez les protocoles EGP, BGP ou BGP4+ pour notifier les adresses des réseaux d'un système autonome auprès des passerelles des autres systèmes autonomes. Si vous faites partie du réseau Internet, EGP, BGP, or BGP4+ doivent être appliqués pour notifier le système de passerelles noyau de l'accessibilité du réseau. Utilisez les protocoles de routage interne pour communiquer les informations d'accessibilité à l'intérieur d'un système autonome.

2. Identifiez tous les réseaux connus en les répertoriant dans le fichier **/etc/network.** Pour en savoir plus, reportez-vous à la section Networks File Format for TCP/IP dans le manuel *AIX 5L Version 5.2 Files Reference*. Un exemple du fichier **networks** est proposé dans le répertoire **/usr/samples/tcpip**.

3. Modifiez le fichier **/etc/gated.conf** pour intégrer la configuration souhaitée pour le démon **gated**.

Remarque : Le démon **gated** fourni avec AIX Version 4.3.2 et versions ultérieures correspond à la version 3.5.9. La syntaxe du fichier **/etc/gated.conf** a été modifiée. Les exemples ci–dessous concernent la version 3.5.9 du démon **gated**. Le fichier **/etc/gated.conf** fournit également la syntaxe pour sa configuration si vous devez l'utiliser avec les versions antérieures à AIX Version 4.3.2.

a. Indiquez le niveau de suivi souhaité. S'il doit débuter avant l'analyse du fichier **gated.conf**, spécifiez l'indicateur **–t** pour activer le suivi au lancement du démon. Pour en savoir plus, reportez-vous à la section gated Daemon dans le manuel *AIX 5L Version 5.2 Commands Reference*.

b. Indiquez les protocoles de routage souhaités. Vous devez spécifier une instruction par protocole. Supprimez la marque de commentaire (#) et modifiez les instructions correspondant aux protocoles à utiliser.

   . Avec EGP :

     – Insérez la clause `autonomoussystem`. Demandez un numéro de système autonome à Internet si vous êtes sur Internet, sinon, attribuez vous-même ce numéro en fonction des numéros sur votre réseau.

     – Positionnez la clause EGP sur `yes`.

     – Insérez une clause `group` pour chaque système autonome.

     – Insérez une clause `neighbor` pour chaque passerelle limitrophe dans ce système autonome. Par exemple :

```
autonomoussystem 283 ;

 egp yes {
    group maxup 1 {
      neighbor nogendefault 192.9.201.1 ;
      neighbor nogendefault 192.9.201.2 ;
    } ;
    group {
      neighbor 192.10.201.1 ;
      neighbor 192.10.201.2 ;
    } ;
 } ;
```

   . Avec RIP ou HELLO :

     – Positionnez l'instruction RIP ou HELLO sur `yes`.

     – Dans l'instruction RIP ou HELLO, spécifiez `nobroadcast` pour que la passerelle se contente de recevoir des informations de routage, mais n'en diffuse pas. Sinon, spécifiez `broadcast` pour qu'elle puisse recevoir et diffuser ces informations.

     – Pour que la passerelle envoie les informations directement aux passerelles source, utilisez l'instruction `sourcegateways`. Spécifiez le nom ou l'adresse Internet d'une passerelle en notation décimale à points dans la clause `sourcegateways`. Par exemple :

```
# Notification à des  passerelles spécifiques

 rip/hello yes {
     sourcegateways
       101.25.32.1
       101.25.32.2 ;
 } ;
```

L'exemple suivant illustre la syntaxe de RIP/HELLO du fichier **gated.conf** d'une machine qui n'envoie aucun paquet RIP, ni n'en reçoit sur son interface tr0.

```
rip/hello nobroadcast {
     interface tr0 noripin ;
 } ;
```

   . Avec BGP :

     – Insérez la clause `autonomoussystem`. Demandez un numéro de système autonome à Internet si vous êtes sur Internet, sinon, attribuez vous-même ce numéro en fonction des numéros sur votre réseau.

     – Positionnez la clause BGP sur `yes`.

– Insérez une clause `peer` pour chaque passerelle limitrophe dans ce système autonome. Par exemple :

```
# Exécuter toutes les opérations BGP

 bgp yes {
    peer 192.9.201.1 ;
 } ;
```

. Avec SNMP :

– Positionnez la clause SNMP sur `yes`.

```
snmp yes ;
```

Configuration du démon gated pour l'exécution de IPv6

Pour configurer le démon **gated** pour l'exécution avec IPv6 (Internet Protocol version 6), vérifiez d'abord que votre système est configuré pour IPv6 et le routage IPv6 :

1. Exécutez **autoconf6** pour configurer automatiquement vos interfaces pour IPv6.
2. Configurez les adresses locales de chaque interface IPv6 sur laquelle vous voulez utiliser le routage IPv6, via la commande :

   `ifconfig` *`interface`* `inet6 fec0:`*`n`*`::`*`address`*`/64 alias`

   où

   interface est le nom de l'interface, comme `tr0` ou `en0`.

   n est un nombre décimal quelconque, par exemple `11`

   address est la portion de l'interface IPv6 qui suit les deux colonnes, par exemple, avec l'adresse IPv6 `fe80::204:acff:fe86:298d`, l'entrée *address* serait `204:acff:fe86:298d` .

   Remarque : La commande **netstat –i** permet d'afficher votre adresse IPv6 pour chaque interface configurée.

   Ainsi, si l'anneau à jeton `tr0` est associé à l'adresse IPv6 `fe80::204:acff:fe86:298d` , entrez la commande :

   ```
   ifconfig tr0 inet6 fec0:13::204:acff:fe86:298d/64 alias
   ```

3. Pour activer le réacheminement IPv6, utilisez la commande :

   ```
   no –o ip6forwarding=1
   ```

4. Pour lancer **ndpd–router**, utilisez la commande :

   ```
   ndpd-router –g
   ```

   Affichez **ndpd–router** pour déterminer les indicateurs à utiliser dans votre configuration réseau.

   Si vous lancez **ndpd–router**, votre système pourra être utilisé comme routeur pour le protocole Neighbor Discovery Protocol. Les routeurs Neighbor Discovery Protocol communiquent les informations de routage aux hôtes Neighbor Discovery afin qu'ils acheminent les paquets IPv6.

Tout hôte du réseau devant appartenir au réseau IPv6 doit exécuter **ndpd–host**. Les hôtes du réseau qui exécutent **ndpd–host** se reconnaîtront comme appartenant à un réseau IPv6 et utiliseront par conséquent le protocole Neighbor Discovery Protocol. Ce protocole leur permet de déterminer et de contrôler les adresses de communication, non seulement pour autoriser le routage limitrophe, mais aussi pour rechercher les routeurs limitrophes afin de réacheminer les paquets.

Pour plus d'informations, reportez–vous aux sections **ndpd–router**, **ndpd–host**, ou consultez RFC 1970, *Neighbor Discovery*.

Ensuite, configurez le démon gated :

1. Déterminez les protocoles de passerelle IPv6 appropriés pour votre système. Vous pouvez utiliser les protocoles de routage IPv6 BGP4+ (Border Gateway Protocol étendu pour IPv6) et RIPng (Routing Information Protocol Next Generation).
2. Modifiez le fichier **/etc/gated.conf** pour intégrer la configuration souhaitée pour le démon **gated**.

   Remarque : AIX Version 4.3.2 et ultérieures exécutent **gated** version 3.5.9. La syntaxe du fichier **gated.conf** est légèrement modifiée par rapport aux versions précédentes. Pour connaître la syntaxe appropriée, reportez–vous à la documentation **gated.conf** ou utilisez le fichier exemple disponible dans le répertoire **/usr/sample/tcpip**.

   Pour configurer BGP4+ ou RIPng, utilisez les adresses IPv6 dont la syntaxe spécifie une adresse IP.

   Remarque : Par défaut, le protocole RIPng envoie des paquets à plusieurs destinataires.

Dès que le fichier **/etc/gated.conf** a été modifié, le démon **gated** peut être lancé.

Obtention d'un numéro de système autonome

Si vous utilisez EGP ou BGP, il est recommandé de solliciter auprès du NIC un *numéro de système autonome* officiel pour votre passerelle. Pour ce faire, contactez NIC à l'adresse INFO@INTERNIC.NET.

IPv6 Mobile

Mobile IPv6 apporte la mobilité à IPv6. Il vous permet de conserver la même adresse Internet dans le monde entier et aux applications utilisant cette adresse de mettre à jour les connexions de transport et de couche supérieure lorsque vous changez de lieu. Il permet aussi la mobilité dans des milieux homogènes et hétérogènes. Par exemple, Mobile IPv6 facilite le déplacement de nœud d'un segment Ethernet vers une cellule LAN sans fil, tandis que l'adresse IP du nœud mobile reste inchangée.

Dans Mobile IPv6, chaque nœud mobile est identifié par deux adresses IP : l'adresse d'origine et l'adresse provisoire. L'adresse d'origine est une adresse IP permanente qui identifie le nœud mobile quel que soit son emplacement. L'adresse provisoire change à chaque nouveau point de connexion et fournit des informations sur la situation actuelle du nœud mobile. Lorsqu'un nœud mobile arrive sur un réseau visité, il doit se procurer une adresse provisoire qu'il utilise pendant tout le temps qu'il occupera cet emplacement dans le réseau visité. Il peut utiliser les méthodes d'IPv6 Neighborhood Discovery pour obtenir l'adresse provisoire (reportez–vous à Neighbor Discovery/autoconfiguration d'une adresse sans état page 4-11). Une autoconfiguration sans état ou avec état est possible. L'adresse provisoire peut aussi être configurée manuellement. Le mode d'acquisition de l'adresse provisoire n'a pas d'importance pour Mobile IPv6.

Un agent d'origine au moins doit être configuré sur le réseau d'origine et le nœud mobile doit être configuré pour connaître l'adresse IP de cet agent. Le nœud mobile envoie un paquet contenant une option de destination de mise à jour de lien à l'agent d'origine. L'agent d'origine obtient le paquet et établit une association entre l'adresse d'origine et le nœud mobile et l'adresse provisoire qu'il a reçue. L'agent d'origine répond en envoyant un paquet contenant une option de destination d'accusé de réception de lien.

L'agent d'origine dispose d'une cache de lien contenant les associations entre les adresses d'origine et les adresses provisoires des nœuds mobiles qu'il dessert. L'agent d'origine intercepte les paquets destinés à l'adresse d'origine et les transmet aux nœuds mobiles. Un nœud mobile envoie alors une mise à jour de lien au nœud correspondant, en l'informant de son adresse provisoire. Le nœud correspondant crée une entrée de cache de lien afin de pouvoir envoyer du trafic futur directement au nœud mobile de son adresse provisoire.

La prise en charge de la mobilité par AIX offre les fonctions suivantes :

En tant qu'agent **Agent d'origine** :

- Tenue à jour d'une entrée dans sa cache de liens pour chaque nœud mobile servi.
- Interception des paquets adressés à un nœud mobile qu'il dessert en tant qu'agent d'origine, sur le lien d'origine de ce nœud mobile, lorsque le nœud mobile est en déplacement.
- Encapsulation des paquets interceptés afin de les tunnéliser vers l'adresse provisoire principale du nœud mobile indiqué dans son lien, dans la cache de liens de l'agent d'origine.
- Renvoi d'une option d'accusé de réception de liens en réponse à une option de mise à jour de liens reçue avec le groupe de bits de l'accusé de réception.

En tant que nœud **Correspondant stationnaire** :

- Traitement d'une option d'adresse d'origine reçue dans un paquet IPv6.
- Traitement d'une option de mise à jour de liens reçue dans un paquet et renvoi d'une option d'accusé de réception de liens si le bit d'accusé de réception (A) est défini dans la mise à jour de liens reçue.
- Tenue à jour d'une cache de liens contenant les lien reçus dans les mises à jour de liens acceptées.

- Envoi de paquets utilisant un en–tête de routage lorsqu'il existe une entrée de cache de liens pour un nœud mobile contenant l'adresse provisoire en cours du nœud mobile.

En tant que nœud **Routeur** dans un réseau visité par le nœud mobile.

- Envoi d'une option d'intervalle d'annonce dans ses annonces de routeur afin de faciliter la détection de mouvements par les nœuds mobiles. Il est configurable par le paramètre `-m` dans le routeur **ndpd–router**.
- Prise en charge de l'envoi d'annonces de routeur multidiffusion non sollicitées à la vitesse plus élevée décrite dans RFC 2461. Configurable avec le paramètre `-m` dans le démon **ndpd–router**

sécurité IP

- Les tunnels doivent être définis de façon statique à l'aide de l'adresse d'origine entre l'agent d'origine et le nœud mobile ou entre le correspondant et le nœud mobile.
- Seul AH–ESP en mode de transport est pris en charge.
- En cas de filtrage sur le protocole 60, seuls les paquets ayant les options de destination BU, BA et BR sont sécurisés.
- En cas de filtrage sur la totalité du trafic, tous les paquets de mobilité (BU, BA signalisation et autres paquets contenant des données) sont sécurisés.
- Le filtrage de sécurité IP sur le protocole 60 doit toujours être employé lorsque la mobilité est utilisée. Certains nœuds mobiles peuvent ne pas accepter de paquets BA et BU sauf si la sécurité IP est utilisée. L'acceptation de ces paquets en cas non–utilisation de la sécurité IP peut poser un grave problème de sécurité.

Sécurité IP avec IKE

- IKE envoie une réponse à l'agent d'origine ou au correspondant.
- Seul le mode agressif est pris en charge.

Configuration de Mobile IPv6

Lancement de Mobile IPv6 avec la sécurité IP

Agent d'origine

1. Définition de tunnels IKE (phases 1 et 2) et de réponse et d'un protocole AH dans la base de données entre l'adresse IP de l'agent d'origine et de l'adresse d'origine de chaque nœud mobile avec lequel l'agent d'origine est susceptible de communiquer. Pour plus de détails, reportez–vous à IP Security dans le manuel AIX Security Guide.
2. Définition de l'association de sécurité AH IP entre l'adresse IP de l'agent d'origine et chaque adresse d'origine mobile avec laquelle le correspondant est susceptible de communiquer.
3. Exécutez la commande suivante :

```
/etc/rc.mobip6 start -H -S
```

Correspondant

1. Définition de tunnels IKE (phases 1 et 2) et de réponse et d'un protocole AH dans la base de données entre l'adresse IP de l'agent d'origine et de l'adresse d'origine de chaque nœud mobile avec lequel l'agent d'origine est susceptible de communiquer. Pour plus de détails, reportez–vous à IP Security dans le manuel AIX Security Guide.
2. Définition de l'association de sécurité AH IP entre l'adresse IP de l'agent d'origine et chaque adresse d'origine mobile avec laquelle le correspondant est susceptible de communiquer.
3. Exécutez la commande suivante :

```
/etc/rc.mobip6 start -S
```

Routeur

Pour faciliter la détection de mouvement, exécutez ce qui suit :

```
ndpd-router -m
```

Lancement de Mobile IPv6 sans la sécurité IP

Mobile IPv6 peut être démarré sans la sécurité IP, mais ceci est déconseillé. La sécurité IP protège les paquets de liens (le filtrage sur le protocole 60). L'utilisation de Mobile IPv6 sans la sécurité IP crée une lacune dans la sécurité.

Agent d'origine

Exécutez la commande suivante :

```
/etc/rc.mobip6 start -H
```

Correspondant

Exécutez la commande suivante :

```
/etc/rc.mobip6 start
```

Routeur

Pour faciliter la détection de mouvement, exécutez ce qui suit :

```
ndpd-router -m
```

Arrêt de Mobile IPv6

Pour arrêter Mobile IPv6 et faire fonctionner le système comme une passerelle IPv6, exécutez la commande suivante :

```
/etc/rc.mobip6 stop
```

Pour arrêter Mobile IPv6 et désactiver la fonctionnalité de passerelle IPv6, exécutez la commande suivante :

```
/etc/rc.mobip6 stop -N -F
```

Identification des incidents Mobile IPv6

- Obtenez les état des liens en exécutant la commande suivante :

```
mobip6ctrl -b
```

- Reportez–vous à la Détermination des problèmes TCP/IP pour savoir comment employer les utilitaires de dépannage TCP/IP.

Adresse IP virtuelle (VIPA)

Une adresse IP virtuelle évite à l'hôte de dépendre d'interface réseau précises. Les paquets entrants sont envoyés à l'adresse VIPA du système mais tous les paquets passent par le réseau réel.

Auparavant, en cas de défaillance d'une interface, les connexions à cette interface étaient perdues. Avec l'activation de VIPA sur le système et le réacheminement automatique assuré par les protocoles de routage dans le réseau, la reprise après incident se déroule sans interruption des connexions utilisateur existantes passant par l'interface virtuelle, à condition que les paquets puissent arriver via une autre interface physique. Les systèmes exécutant VIPA sont plus disponibles car les pannes de carte n'affectent plus les connexions actives. Comme de multiples cartes physiques transmettent le trafic IP du système, la charge globale n'est pas concentrée sur une seule carte et son sous–réseau associé.

La fonction AIX VIPA est transparente pour l'équipement réseau. Aucun équipement réseau spécial ou d'autre matériel n'est requis. Pour implémenter VIPA, vous devez disposer de la configuration suivante :

- deux interfaces IP existantes de type physique indifférent sur des sous–réseaux différents qui se connectent au réseau d'entreprise
- des protocoles de routage IP s'exécutant dans le réseau de l'entreprise

Configuration de VIPA

VIPA doit être configuré dans SMIT, comme toutes les interfaces réseau IP. Vous pouvez aussi définir un groupe d'interfaces tout en configurant VIPA. Lorsqu'elle est configurée de cette façon, pour toutes les connexions initialisées par l'hôte VIPA via ces interfaces, qui sont conçues pour utiliser un VIPA, l'adresse virtuelle devient l'adresse source placée dans l'en–tête du paquet TCP/IP des paquets en sortie.

1. Pour VIPA IPv4, tapez `smit mkinetvi` sur la ligne de commande.
   Pour VIPA IPv6, tapez `smit mkinetvi6` sur la ligne de commande.
2. Remplissez tous les champs requis et appuyez sur Entrée.

Gestion de VIPA

Cette section traite des points suivants :

- Ajout d'une carte à un VIPA page 4-183
- Retrait d'une carte d'un VIPA page 4-184
- Exemple d'environnement VIPA dans AIX 5.2 page 4-184
- Autres informations techniques page 4-185

Ajout d'une carte à un VIPA

Pour ajouter une carte à votre interface VIPA, procédez comme suit :

1. Tapez `smit chvi` sur la ligne de commande.
2. Sélectionnez le VIPA auquel ajouter une carte et appuyez sur Entrée.
3. Indiquez la carte à ajouter dans le champ `Interface Name(s)`.
4. Entrez `ADD` dans le champ `ADD/REMOVE interface(s)` et appuyez sur Entrée.

Retrait d'une carte d'un VIPA

Pour supprimer une carte d'un VIPA, procédez comme suit :

1. Tapez `smit chvi` sur la ligne de commande.
2. Sélectionnez le VIPA duquel retirer une carte et appuyez sur Entrée.
3. Indiquez la carte à retirer dans le champ `Interface Name(s).`
4. Entrez `REMOVE` dans le champ `ADD/REMOVE interface(s)` et appuyez sur Entrée.

Exemple d'environnement VIPA dans AIX 5.2

Un système a une adresse IP virtuelle, `vi0`, de `10.68.6.1` et deux connexions physiques, `en1` avec l'adresse IP `10.68.1.1` et `en5`, avec l'adresse IP `10.68.5.1`. Dans cet exemple, les deux connexions physiques sont Ethernet, mais toute combinaison d'interfaces IP, par exemple en anneau à jeton ou FDDI, sera prise en charge à partir du moment où les sous–réseaux ont été rattachés au réseau principal d'entreprise et sont connus des routeurs d'entreprise.

L'exécution de la commande `lsattr -El vi0` génère les résultats suivants :

```
netaddr          10.68.6.1      N/A
True
 state            up             Standard Ethernet Network Interface
 True
 netmask          255.255.255.0  Maximum IP Packet Size for This Device
 True
 netaddr6                        Maximum IP Packet Size for REMOTE Networks
True
 alias6                          Internet Address
 True
 prefixlen                       Current Interface Status
 True
 alias4                          TRAILER Link-Level Encapsulation
 True
 interface_names  en1,en5        Interfaces using the Virtual Address
 True
```

L'exécution de la commande `ifconfig vi0` génère les résultats suivants :

```
vi0: flags=84000041<UP,RUNNING,64BIT>
        inet 10.68.6.1 netmask 0xffffff00
        iflist : en1 en5
```

L'exécution de la commande `netstat -rn` génère les résultats suivants :

```
Tables de routage
 Destination       Gateway       Flags     Refs    Use    If     PMTU Exp
Groups

 Route Tree for Protocol Family 2 (Internet):
 default           10.68.1.2      UG        3      1055   en1      -   -
 10.68.1/24        10.68.1.1      U         0      665    en1      -   -
 10.68.5/24        10.68.5.1      U         0      1216   en5      -   -
 127/8             127.0.0.1      U         4      236    lo0      -   -
 10.68.6.1         127.0.0.1      UH        0      0      lo0      -   -
```

L'adresse source des paquets en sortie pour lesquels une adresse source n'est pas définie et qui sont routés via les interfaces `en1` et `en5` est définie comme l'adresse virtuelle (`10.68.6.1`). Les paquets entrants sont routés vers l'adresse VIPA (`10.68.6.1`) annoncée sur le réseau. Comme `vi0` est virtuel, c'est–à–dire qu'il n'est associé à aucune unité, il ne doit pas exister d'entrées lui correspondant dans la table de routage à l'échelle de tout le système affichée par la commande `netstat -rn`. Ceci signifie qu'aucune route d'interface n'est ajoutée lorsque l'interface est configurée dans SMIT.

Si l'une des interfaces physiques, une connexion réseau ou un chemin réseau échoue, les protocoles réseau effectuent l'acheminement vers l'autre interface physique du même système. Si un système éloigné envoie une commande telnet à l'adresse `vi0`, les paquets destinés à `vi0` peuvent arriver via `en1` ou `en5`. Si `en1` est en panne, par exemple, les

paquets peuvent toujours arriver sur `en5`. Les protocoles de routage peuvent prendre un certain temps pour propager les routes.

Lorsque vous utilisez le VIPA, les systèmes finals et les routeurs intermédiaires doivent pouvoir acheminer les paquets destinés à VIPA (`vi0`) vers l'une des interfaces physiques (`en1` ou `en5`).

Autres informations techniques

Comparaison VIPA/alias

Le concept VIPA est semblable aux alias IP, sauf que les adresses ne sont pas associées à une interface matérielle. VIPA offre plusieurs avantages que les alias IP ne possèdent pas :

- VIPA propose une unité virtuelle qui peut être activée et arrêtée de façon indépendante, sans impact sur les interfaces physiques.
- Les adresses VIPA peuvent être modifiées, tandis que les alias peuvent seulement être ajoutés ou supprimés.

Accès via l'adresse IP des cartes réelles

Les interfaces individuelles sont toujours accessibles aux autres systèmes une fois VIPA implémenté. Toutefois, l'utilisation des adresses IP réelles pour les sessions ping et telnet renforce l'avantage VIPA qui communique indépendamment des cartes physiques. VIPA cache les défaillances de carte physique aux clients externes. L'utilisation des adresses réelles réintroduit la dépendance par rapport aux cartes physiques.

Si le système éloigné contacte le système VIPA à l'aide de l'adresse VIPA ou si une application sur le système VIPA initialise la communication avec un autre système, l'adresse VIPA est utilisée comme adresse IP source dans le paquet. Toutefois, si le système éloigné initialise la session à l'aide de l'adresse IP de l'interface réelle, cette adresse IP réelle est l'adresse IP source dans les paquets de réponse. Il y a cependant une exception. Pour les applications établissant une liaison à une interface IP particulière, les paquets sortants transfèrent l'adresse source de l'interface à laquelle ils sont liés.

VIPA et les protocoles de routage

Le démon gated a été modifié pour VIPA afin de ne pas ajouter de route d'interface ou d'envoyer d'annonces via des interfaces virtuelles. Le protocole OSPF, pris en charge par gated, annonce l'interface virtuelle aux routeurs de voisinage. Les autres hôtes du réseau peuvent parler à l'hôte VIPA via le routeur du premier tronçon.

Adresses VIPA multiples

Il est possible de configurer plusieurs interfaces virtuelles.

Il est utile d'avoir plusieurs interfaces VIPA, par exemple si des routeurs réseau peuvent offrir un traitement préférentiel aux paquets envoyés de ou vers certaines adresses VIPA. Vous pouvez aussi utiliser plusieurs interfaces VIPA si elles liaient des applications à une interface VIPA spécifique. Par exemple, pour exécuter plusieurs serveurs Web pour plusieurs entreprises sur une seule machine, vous pouvez configurer ce qui suit :

- vi0 200.1.1.1 www.companyA.com
- vi1 200.1.1.2 www.companyB.com
- vi2 200.1.1.3 www.companyC.com

VIPA sous AIX 5.1

Il n'était pas possible de définir un groupe d'interfaces utilisant un VIPA particulier dans AIX 5.1. Le premier VIPA de la liste d'adresses sera choisi comme adresse source par défaut lorsque l'application n'établit pas de lien explicite à une adresse.

EtherChannel et IEEE 802.3ad Link Aggregation

EtherChannel et IEEE 802.3ad Link Aggregation désignent des technologies d'agrégation de port réseau permettant à plusieurs cartes Ethernet d'être rassemblées pour former une seule pseudo–unité Ethernet. Par exemple, `ent0` et `ent1` peuvent être réunies dans l'interface `ent3`. L'interface `en3` serait alors configurée avec une adresse IP. Le système considère cet agrégat de cartes comme une carte unique. Par conséquent, IP est configuré via ces cartes comme s'il s'agissait d'une carte Ethernet normale. En outre, toutes les cartes d'EtherChannel ou de Link Aggregation reçoivent la même adresse matérielle (MAC) et sont donc traitées par les systèmes distants comme s'il s'agissait d'une seule carte.

L'avantage principal d'EtherChannel et de IEEE 802.3ad Link Aggregation est qu'ils disposent de toute la bande passante de toutes leurs cartes tout en étant uniques dans le réseau. En cas de panne d'une carte, les paquets sont automatiquement envoyés à la carte disponible suivante sans interruption des connexions utilisateur existantes. La carte est remise automatiquement en service sur EtherChannel ou Link Aggregation lorsque son fonctionnement est rétabli.

Il existe des différences entre EtherChannel et IEEE 802.3ad Link Aggregation. Consultez les différences indiquées dans le tableau suivant pour déterminer la solution vous convenant le mieux.

EtherChannel	**IEEE 802.3ad**
Nécessite la configuration d'un commutateur	Peu ou pas de configuration de commutateur pour former l'agrégat. Une configuration initiale du commutateur peut être nécessaire.
Fonctionnement possible dans de nombreux modes	Fonctionne uniquement en mode standard

Pour plus d'informations sur la configuration et l'utilisation d'EtherChannel, reportez–vous à EtherChannel page 4-186. Pour plus d'informations sur la configuration et l'utilisation de IEEE 802.3ad Link Aggregation, reportez–vous à IEEE 802.3ad Link Aggregation page 4-194. Pour plus d'informations sur les différentes combinaisons de configuration d'AIX et de commutateur, et sur les résultats obtenus, reportez–vous à Scénarios d'interopérabilité page 4-197.

EtherChannel

Les cartes appartenant à un EtherChannel sont câblés sur le même commutateur réseau EtherChannel, qui doit être configuré manuellement pour identifier les ports appartenant à l'EtherChannel.

Le trafic est distribué entre les cartes soit de façon standard (la carte via laquelle les paquets sont envoyés est choisie en fonction de l'adresse de destination), soit de façon circulaire (les paquets sont répartis équitablement entre toutes les cartes). Le trafic entrant est distribué en fonction de la configuration du commutateur et n'est pas contrôlé par le mode d'exploitation d'EtherChannel.

Dans AIX, les utilisateurs peuvent configurer plusieurs EtherChannels par système, mais la norme exige que toutes les liaisons d'un EtherChannel soient rattachées à un seul commutateur. Comme EtherChannel ne peut pas être réparti entre plusieurs commutateurs, la totalité d'EtherChannel est perdue en cas de coupure du courant ou de panne du commutateur. Pour résoudre ce problème, une nouvelle option disponible dans AIX 5.2 et les versions supérieures maintient le service actif en cas de défaillance de l'EtherChannel principal. Les cartes de secours et EtherChannel doivent être rattachées à différents commutateurs réseau. En cas de panne de toutes les cartes de l'EtherChannel, les adresses IP et MAC sont automatiquement transférées vers la carte de secours. Lors de la restauration d'une liaison d'EtherChannel, le service est retransféré vers l'EtherChannel.

Network Interface Backup, un mode d'exploitation disponible pour EtherChannel dans AIX 4.3.3 et AIX 5.1, offre une protection contre un point de défaillance Ethernet unique. Aucun matériel spécial n'est requis pour utiliser Network Interface Backup, mais la carte de secours doit être connectée à un commutateur distinct. Dans le mode Network Interface Backup, une seule carte à la fois est utilisée activement pour le trafic réseau. L'EtherChannel teste la carte active et facultativement, le chemin réseau vers un nœud spécifié par l'utilisateur. Lorsqu'une panne est détectée, les adresses MAC et IP sont transférées vers la carte suivante, qui sera utilisée jusqu'à sa défaillance. Network Interface Backup fournit des fonctions de détection et de prise de relais sans interruption des connexions utilisateur. Network Interface Backup a à l'origine été mis en œuvre en tant que mode dans le menu EtherChannel SMIT. Dans AIX 5.2 et les versions supérieures, la carte de secours fournit une fonction équivalente et ce mode a donc été supprimé du menu SMIT. Pour utiliser Network Interface Backup dans AIX 5.2 et les versions supérieures, reportez–vous à Configuration de Network Interface Backup.

Configuration d'EtherChannel

Procédez comme suit pour configurer un EtherChannel.

Remarques

- Vous pouvez utiliser jusqu'à huit cartes Ethernet par EtherChannel.
- Vous pouvez configurer plusieurs EtherChannels sur un seul système, mais chaque EtherChannel constitue une interface Ethernet supplémentaire. Par conséquent, vous devez augmenter la valeur de l'option **ifsize** de la commande **no** pour inclure les interfaces Ethernet pour chaque carte, mais également toutes les unités logiques VLAN configurées. La valeur par défaut de **ifsize** est huit.
- Vous pouvez utiliser toutes les cartes Ethernet prises en charge dans EtherChannel. Toutefois, les cartes Ethernet doivent être connectées à un commutateur prenant en charge EtherChannel. Reportez–vous à la documentation fournie avec le commutateur pour déterminer s'il prend en charge EtherChannel.
- Toutes les cartes d'EtherChannel doivent être configurées pour la même vitesse (100 Mbps, par exemple) et doivent utiliser le mode duplex intégral.
- Les cartes que vous prévoyez d'utiliser pour votre EtherChannel ne doivent pas avoir d'adresse IP configurée avant de commencer cette procédure. Utilisez la commande **ifconfig** pour déconfigurer les cartes. Par exemple, `ifconfig en5 detach` déconfigure la carte `en5`.
- Les cartes utilisées dans l'EtherChannel ne sont pas accessibles par le système une fois l'EtherChannel configuré. Leurs attributs doivent être configurés avant de créer l'EtherChannel.
- Les cartes à ajouter à l'EtherChannel ne peuvent pas avoir d'interfaces configurées dans l'état `up` dans ODM (comme ce serait le cas si leurs adresses IP ont été configurées avec SMIT). Ceci peut causer des problèmes pour activer EtherChannel lors du redémarrage de la machine, car l'interface sous–jacente est configurée avant l'EtherChannel avec les informations trouvées dans ODM. Par conséquent, lorsque l'EtherChannel est configuré, il détecte que l'une de ses cartes est déjà utilisée. Pour modifier ceci, avant de créer l'EtherChannel, tapez `smit chinet`, sélectionnez chacune des interfaces des cartes à inclure dans l'EtherChannel, et remplacez sa valeur **state** par `detach`. Lors du redémarrage de la machine, l'EtherChannel peut être configuré sans erreurs.
- Si vous utilisez des cartes 10/100 Ethernet dans l'EtherChannel, vous devez activer le sondage de liaison sur ces cartes avant de les ajouter à l'EtherChannel. Tapez `smitty chgenet` sur la ligne de commande. Remplacez la valeur de **Enable Link Polling** par `yes`, puis appuyez sur Entrée. Procédez de cette façon pour chaque carte 10/100 Ethernet que vous ajoutez à votre EtherChannel. Si vous ne le faites pas, l'EtherChannel fonctionnera mais la détection de défaillance de liaison et la prise de relais ne fonctionneront pas.

- Si vous avez l'intention d'utiliser des trames jumbo, vous devez activer cette fonctionnalité sur chaque carte avant de créer l'EtherChannel et l'activer aussi dans l'EtherChannel lui–même. Tapez `smitty chgenet` sur la ligne de commande. Remplacez la valeur de **Enable Jumbo Frames** par `yes`, puis appuyez sur Entrée. Procédez de cette façon pour toutes les cartes pour lesquelles vous voulez activer les trames jumbo. Vous activerez ultérieurement les trames jumbo dans l'EtherChannel lui–même.

Configuration d'un EtherChannel

1. Tapez `smit etherchannel` sur la ligne de commande.
2. Sélectionnez **Add an EtherChannel / Link Aggregation** dans la liste et appuyez sur Entrée.
3. Sélectionnez les cartes Ethernet principales de l'EtherChannel et appuyez sur Entrée. Si vous prévoyez d'utiliser la sauvegarde de secours EtherChannel, ne sélectionnez pas la carte de secours à ce stade. L'option de secours EtherChannel est disponible dans AIX 5.2 et les versions supérieures.

   Remarque : **Cartes réseau disponibles** affiche toutes les cartes Ethernet. Si vous sélectionnez une carte Ethernet déjà utilisée (dont l'interface est définie), vous obtenez un message d'erreur. Vous devez d'abord détacher cette interface si vous souhaitez l'utiliser.
4. Entrez les informations dans les champs en respectant les consignes suivantes :
   - **Cartes EtherChannel / Link Aggregation :** Vous devez voir s'afficher toutes les cartes principales utilisées dans votre EtherChannel. Vous avez sélectionné ces cartes à l'étape précédente.
   - **Enable Alternate Address:** Ce champ est facultatif. Le choix de la valeur `yes` vous permet de préciser une adresse MAC que l'EtherChannel doit utiliser. Si vous définissez la valeur `no` pour cette option, l'EtherChannel utilisera l'adresse MAC de la première carte indiquée.
   - **Alternate Address:** Si vous attribuez à **Enable Alternate Address** la valeur `yes`, indiquez l'adresse MAC à utiliser ici. L'adresse indiquée doit commencer par `0x` et être une valeur hexadécimale à 12 chiffres.
   - **Enable Gigabit Ethernet Jumbo Frames:** Ce champ est facultatif. Pour l'utiliser, le commutateur doit prendre en charge les trames jumbo. Ceci fonctionne uniquement avec une interface Ethernet Standard (en) mais pas avec une interface IEEE 802.3. Affectez la valeur `yes` si vous voulez l'activer.
   - **Mode:** Vous pouvez choisir entre les modes suivants :
     - **standard**: Dans ce mode, l'EtherChannel utilise l'adresse IP de destination pour choisir la carte à laquelle il va envoyer les paquets sortants. L'EtherChannel divise le dernier octet de l'adresse IP de destination du paquet par le nombre de cartes dans l'EtherChannel et utilise le reste (à l'aide de l'opérateur modulus) pour identifier la liaison sortante. Par exemple, si l'adresse IP de destination est 10.10.10.1, et qu'il y a 2 cartes dans l'EtherChannel, (1 / 2) = 0 avec comme reste 1, la deuxième carte est utilisée (les cartes sont numérotées à partir de 0). Les cartes sont numérotées dans l'ordre indiqué dans le menu SMIT. Pour le trafic non–IP (par exemple ARP), le dernier octet de l'adresse MAC de destination est utilisé pour effectuer le calcul. Ce mode garantit que les paquets sont envoyés via l'EtherChannel dans l'ordre dans lequel ils ont été reçus, mais il peut ne pas utiliser toute la bande passante. C'est le mode de fonctionnement par défaut.
     - **round_robin**: Dans ce mode, l'EtherChannel utilise tour à tour les cartes, en envoyant un seul paquet à chaque carte avant de recommencer. les paquets peuvent être envoyés dans un ordre légèrement différent que celui de leur envoi à l'EtherChannel, mais l'utilisation de la bande passante est optimisée.

. **netif_backup:** Cette option est disponible dans AIX 5.1 et AIX 4.3.3. Dans ce mode, l'EtherChannel active une seule carte à la fois. Le but est de connecter les cartes à différents commutateurs Ethernet, chacun pouvant accéder à n'importe quelle autre machine du sous–réseau ou du réseau. Lorsqu'un problème lié à la connexion directe est détecté (ou facultativement parce qu'il est impossible d'envoyer une commande ping à une machine), l'EtherChannel désactive la carte en cours et active une carte de secours. Ce mode est le seul qui utilise les champs **Internet Address to Ping**, **Number of Retries**, et **Retry Timeout**.

  Le mode Network Interface Backup Mode n'existe pas en tant que mode explicite dans AIX 5.2 et les versions supérieures. Pour activer le mode Network Interface Backup dans AIX 5.2 et les versions supérieures, vous devez configurer une seule carte dans l'EtherChannel principale et une carte de secours. Pour plus d'informations, consultez la section Configuration de Network Interface Backup page 4-189.

– **Backup Adapter:** Ce champ est facultatif. Indiquez la carte à utiliser comme carte de secours EtherChannel. L'option de secours EtherChannel est disponible dans AIX 5.2 et les versions supérieures.

– **Internet Address to Ping:** Ce champ est facultatif et disponible uniquement si vous exécutez le mode **Network Interface Backup**. L'EtherChannel lance une commande ping sur l'adresse IP indiquée ici. Si l'EtherChannel ne parvient pas à lancer une commande ping au terme du nombre de tentatives précisé dans **Number of Retries** dans les intervalles **Retry Timeout**, l'EtherChannel effectue un basculement des cartes.

– **Number of Retries:** Entrez le nombre d'échecs de réponses ping autorisés avant que l'EtherChannel ne change de cartes. La valeur par défaut est 3. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.

– **Retry Timeout:** Entrez le nombre de secondes entre les envois de commande ping de l'EtherChannel à **Internet Address to Ping**. La valeur par défaut est une seconde. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.

5. Appuyez sur Entrée après avoir modifié les champs voulus pour créer l'EtherChannel.
6. Configurez IP via la nouvelle unité EtherChannel en tapant `smit chinet` sur la ligne de commande.
7. Sélectionnez votre nouvelle interface EtherChannel dans la liste.
8. Remplissez tous les champs requis et appuyez sur Entrée.

Configuration de Network Interface Backup

Network Interface Backup offre une protection contre un point de défaillance réseau unique en permettant la détection des défaillances et la prise de relais, sans interruption des connexions utilisateur. Dans ce mode, une seule carte est active à un moment donné. En cas de défaillance de la carte active, la carte suivante de l'EtherChannel est utilisée pour tout le trafic. En mode Network Interface Backup, il n'est pas nécessaire d'établir de connexion aux commutateurs EtherChannel.

La configuration de Network Interface Backup est plus efficace lorsque les cartes sont connectées à différents commutateurs réseau, car ceci permet de bénéficier d'une meilleure redondance que la connexion de toutes les cartes à un seul commutateur. Lors de la connexion à différents commutateurs, assurez–vous qu'il existe une connexion entre les commutateurs. Ceci fournit des fonctions de prise de relais d'une carte à l'autre en veillant à ce qu'il existe toujours un accès à la carte active.

Dans les éditions antérieures à AIX 5.2, le mode Network Interface Backup a été implémenté en tant que mode explicite de fonctionnement dans le menu EtherChannel SMIT. Dans AIX 5.2 et les versions supérieures, la carte de secours fournit une fonction équivalente et ce mode a donc été supprimé du menu SMIT.

En outre, AIX 5.2 et les versions supérieures fournissent la priorité, c'est–à–dire que la carte configurée dans l'EtherChannel principal sera utilisé en priorité par rapport à la carte de secours. Tant que la carte principale est fonctionnelle, elle sera utilisée. Ceci s'oppose au comportement du mode Network Interface Backup, dans lequel la carte de secours était utilisée jusqu'à sa défaillance, que la carte principale soit ou non déjà rétablie.

Tout en fonction en mode Network Interface Backup, il est aussi possible de configurer l'EtherChannel pour détecter la défaillance de la liaison et l'inaccessibilité du réseau. Pour ce faire, indiquez l'adresse IP d'un hôte éloigné auquel la connexion doit toujours être établie. L'EtherChannel lance régulièrement une commande ping sur cet hôte pour déterminer s'il existe toujours un chemin réseau permettant d'y accéder. Si un certain nombre de tentatives de commande ping restent sans réponse, l'EtherChannel bascule vers la carte de secours suivante dans l'espoir qu'il existe un chemin d'accès réseau à l'hôte éloigné via la carte suivante. Dans cette configuration, non seulement chaque carte doit être connectée à un commutateur différent, mais chaque commutateur doit avoir un chemin d'accès différent à l'hôte recevant la commande ping.

La commande ping est disponible uniquement en mode Network Interface Backup. Cependant, dans AIX 5.2 et les versions supérieures, si la commande ping est activée et qu'une prise de relais a eu lieu, l'EtherChannel ne rebascule pas vers la carte principale. La carte de secours restera le canal actif tant qu'elle sera opérationnelle, car il n'existe aucun moyen de savoir à quel moment le chemin d'accès à l'hôte recevant la commande ping sera accessible à partir de la carte principale. Si une défaillance est détectée lorsque la carte de secours est active (c'est–à–dire si les tentatives de commande ping échouent à partir de la carte de secours ou si la carte de secours elle–même a une défaillance), l'EtherChannel bascule alors vers la carte principale. Si la prise de relais s'est produite parce que la carte principale a échoué, l'EtherChannel revient alors à la carte principale dès qu'elle est rétablie.

Pour configurer Network Interface Backup dans AIX 5.2, reportez–vous à Configuration de Network Interface Backup dans AIX 5.2 et dans les versions supérieures. Pour configurer Network Interface Backup dans les versions précédentes d'AIX, reportez–vous à l'Annexe B. Configuration de Network Interface Backup dans les versions précédentes d'AIX.

Configuration de Network Interface Backup dans AIX 5.2 et versions supérieures

1. Avec les droits root, tapez `smit etherchannel` sur la ligne de commande.
2. Sélectionnez **Add an EtherChannel / Link Aggregation** dans la liste et appuyez sur Entrée.
3. Sélectionnez la carte Ethernet principale et appuyez sur Entrée. C'est la carte qui est utilisée tant qu'elle n'a pas de défaillance.

   Remarque : **Cartes réseau disponibles** affiche toutes les cartes Ethernet. Si vous sélectionnez une carte Ethernet déjà utilisée, vous obtenez un message d'erreur et devrez détacher l'interface avant de l'utiliser. Reportez–vous à la commande **ifconfig** pour savoir comment détacher une interface.

4. Entrez les informations dans les champs en respectant les consignes suivantes :
   - **EtherChannel / Link Aggregation Adapters**: Vous devez voir s'afficher la carte principale sélectionnée à l'étape précédente.
   - **Enable Alternate Address**: Ce champ est facultatif. Le choix de la valeur `yes` vous permet de préciser une adresse MAC que l'EtherChannel doit utiliser. Si vous définissez la valeur `no` pour cette option, l'EtherChannel utilisera l'adresse MAC de la carte principale.
   - **Alternate Address**: Si vous attribuez à **Enable Alternate Address** la valeur `yes`, indiquez l'adresse MAC à utiliser ici. L'adresse indiquée doit commencer par `0x` et être une valeur hexadécimale à 12 chiffres.
   - **Enable Gigabit Ethernet Jumbo Frames**: Ce champ est facultatif. Pour l'utiliser, le commutateur doit prendre en charge les trames jumbo. Ceci fonctionne uniquement

avec une interface Ethernet Standard (en) mais pas avec une interface IEEE 802.3. Affectez la valeur `yes` si vous voulez l'activer.

- **Mode**: Le mode de fonctionnement que vous sélectionnez n'a pas d'importance car il n'existe qu'une seule carte dans l'EtherChannel principal. Tous les paquets sont envoyés via cette carte jusqu'à ce qu'elle ait une défaillance. Il n'existe pas de mode `netif_backup` car ce mode peut être émulé via une carte de secours.
- **Backup Adapter**: Indiquez la carte à utiliser comme carte de secours. A la suite d'une prise de relais, cette carte est utilisée jusqu'à ce que la carte principale soit rétablie. Il est conseillé d'utiliser la carte de prédilection comme carte principale.
- **Internet Address to Ping**: Ce champ est facultatif. L'EtherChannel lance une commande ping sur l'adresse IP indiquée ici. Si l'EtherChannel ne parvient pas à lancer une commande ping au terme du nombre de tentatives précisé dans **Number of Retries** dasn les intervalles **Retry Timeout**, l'EtherChannel effectue un basculement des cartes.
- **Number of Retries**: Entrez le nombre d'échecs de réponses ping autorisés avant que l'EtherChannel ne change de cartes. La valeur par défaut est 3. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.
- **Retry Timeout**: Entrez le nombre de secondes entre les envois de commande ping de l'EtherChannel à **Internet Address to Ping**. La valeur par défaut est une seconde. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.

5. Appuyez sur Entrée après avoir modifié les champs voulus pour créer l'EtherChannel.
6. Configurez IP via la nouvelle interface en tapant `smit chinet` sur la ligne de commande.
7. Sélectionnez votre nouvelle interface EtherChannel dans la liste.
8. Remplissez tous les champs requis et appuyez sur Entrée.

Pour savoir quelles sont les autres tâches à exécuter une fois l'EtherChannel configuré, reportez–vous à Gestion d'EtherChannel et de IEEE 802.3ad Link Aggregation.

Gestion d'EtherChannel et de IEEE 802.3ad Link Aggregation

Cette section vous indique comment exécuter les tâches suivantes :

Affichage de la liste des EtherChannels ou des Link Aggregations

1. Sur la ligne de commande, tapez `smit etherchannel`.
2. Sélectionnez **List All EtherChannels / Link Aggregations** et appuyez sur Entrée.

Modification de l'adresse de remplacement

Ceci vous permet d'indiquer une adresse MAC pour votre EtherChannel ou Link Aggregation.

1. Tapez `ifconfig` *`interface`* `detach`, où *`interface`* désigne votre interface EtherChannel ou Link Aggregation.
2. Sur la ligne de commande, tapez `smit etherchannel`.
3. Sélectionnez **Change / Show Characteristics of an EtherChannel** et appuyez sur Entrée.
4. Si vous avez plusieurs EtherChannels, sélectionnez celui pour lequel vous voulez créer une adresse de remplacement.
5. Remplacez la valeur de **Enable Alternate EtherChannel Address** par `yes`.
6. Entrez l'adresse de remplacement dans le champ **Alternate EtherChannel Address**. L'adresse doit commencer par `0x` et être une valeur hexadécimale à 12 chiffres.
7. Appuyez sur Entrée pour exécuter la procédure.

Ajout, suppression ou changement de cartes dans un EtherChannel ou Link Aggregation

1. Tapez `ifconfig` *`interface`* `detach`, où *`interface`* désigne votre interface EtherChannel.
2. Sur la ligne de commande, tapez `smit etherchannel`.
3. Sélectionnez **Change / Show Characteristics of an EtherChannel / Link Aggregation** et appuyez sur Entrée.
4. Sélectionnez l'EtherChannel ou le Link Aggregation à modifier.
5. Sélectionnez les cartes Ethernet principales de l'EtherChannel ou du Link Aggregation et appuyez sur Entrée. Si vous utilisez une carte de secours, ne sélectionnez pas cette carte ici.
6. Remplissez tous les champs requis et appuyez sur Entrée.

Suppression d'un EtherChannel ou d'un Link Aggregation

1. Tapez `ifconfig` *`interface`* `detach`, où *`interface`* désigne votre interface EtherChannel.
2. Sur la ligne de commande, tapez `smit etherchannel`.
3. Sélectionnez **Remove an Etherchannel** et appuyez sur Entrée.
4. Sélectionnez l' EtherChannel que vous souhaitez supprimer et appuyez sur Entrée.

Configuration ou suppression d'une carte de secours sur un EtherChannel ou un Link Aggregation existant

La procédure suivante configure ou supprime une carte de secours sur un EtherChannel ou Link Aggregation. L'option est disponible uniquement dans AIX 5.2 et les versions supérieures.

1. Tapez `ifconfig` *`interface`* `detach`, où *`interface`* désigne votre interface EtherChannel ou Link Aggregation.
2. Sur la ligne de commande, tapez `smit etherchannel`.
3. Sélectionnez **Change / Show Characteristics of an EtherChannel / Link Aggregation**.
4. Sélectionnez l'EtherChannel ou Link Aggregation sur lequel vous ajoutez ou modifiez la carte de secours.
5. Entrez la carte à utiliser comme carte de secours dans le champ **Backup Adapter** ou sélectionnez **NONE** si vous voulez cesser d'utiliser la carte de secours.

Identification des incidents d'EtherChannel

En cas de problème avec l'EtherChannel, envisagez les points suivants :

Suivi d'EtherChannel

Utilisez **tcpdump** et **iptrace** pour identifier et résoudre les incidents liés à l'EtherChannel. L'ID d'ancrage de trace pour les paquets de transmission est 2FA et 2FB pour les autres événements. Vous ne pouvez pas effectuer le suivi de paquets sur l'EtherChannel tout entier, mais pouvez suivre les points d'ancrage de suivi de réception de chaque carte.

Affichage des statistiques d' EtherChannel

Utilisez la commande **entstat** pour obtenir l'agrégat des statistiques de toutes les cartes de l'EtherChannel. Par exemple, `entstat ent7` affichera l'agrégat de statistique de ent7. L'ajout de l'indicateur `-d` affiche également les statistiques de chaque carte. Par exemple, la commande `entstat -d ent7` affiche l'agrégat des statistiques de l'EtherChannel ainsi que les statistiques de chaque carte de l'EtherChannel.

Remarque : Dans la section `Statistiques générales` , le chiffre indiqué dans `Adapter Reset Count` est celui des prises de relais. Dans l'option de secours d'EtherChannel, le retour à l'EtherChannel principal à partir de la carte de secours n'est pas comptabilisé comme une prise de relais. Seule une prise de relais à partir du canal principal au secours est comptabilisé.

Dans le champ `Number of Adapters` , la carte de secours est comptabilisée dans le chiffre affiché.

Amélioration de la prise de relais lente

Si la prise de relais lorsque vous utilisez le mode Network Interface Backup ou l'option de secours EtherChannel est lente, vérifiez que le commutateur n'exécute pas le protocole STP (Spanning Tree Protocol). Lorsque le commutateur détecte une modification de son équivalence port de commutateur/adresse MAC, il exécute l'algorithme STP pour voir s'il y a des boucles dans le réseau. Network Interface Backup et l'option de secours EtherChannel peuvent provoquer une modification de l'équivalence port/adresse MAC.

Les ports de commutation ont un compteur de retard de transmission qui détermine au bout de combien de temps après l'initialisation chaque port doit commencer à retransmettre ou envoyer des paquets. Pour cette raison, lorsque le canal principal est réactivé, il se produit un retard avant le rétablissement de la connexion, tandis que la prise de relais par la carte de secours est plus rapide. Vérifiez le compteur de retard de retransmission du commutateur et indiquez une valeur aussi basse que possible afin de pouvoir revenir aussi vite que possible au canal principal.

Pour que l'option de secours EtherChannel fonctionne correctement, le compteur de retard de retransmission ne doit pas excéder 10 secondes ou le retour à l'EtherChannel principal risque de ne pas se dérouler normalement. Il est conseillé d'affecter la valeur la plus basse possible autorisée par le commutateur au compteur de retard de retransmission.

Les cartes ne prennent pas le relais

Si les défaillances des cartes ne déclenchent pas des prises de relais, regardez si vos cartes ont besoin d'activer le sondage de liaison pour détecter la défaillance de liaison. Certaines cartes ne peuvent pas détecter automatiquement l'état de leur liaison. Pour détecter cet état, ces cartes doivent activer un mécanisme de sondage de liaison qui démarre un compteur qui vérifie régulièrement l'état de la liaison. Le sondage de liaison est désactivé par défaut. Toutefois, pour qu'EtherChannel fonctionne correctement avec ces cartes, le mécanisme de sondage de liaison doit être activé sur chaque carte avant que l'EtherChannel soit créé.

Les cartes possédant un mécanisme de sondage de liaison ont un attribut ODM appelé **poll_link**, qui doit avoir la valeur `yes` pour que le sondage de liaison soit activé. Avant de créer l'EtherChannel, lancez la commande suivante sur chaque carte à inclure :

```
smit chgenet
```

Remplacez la valeur de **Enable Link Polling** par `yes` puis appuyez sur Entrée.

Utilisation de trames jumbo

Pour que les trames jumbo fonctionnent correctement, vous devez activer l'attribut **use_jumbo_frame** sur l'EtherChannel, mais aussi les trames jumbo sur chaque carte avant de créer l'EtherChannel à l'aide de la commande suivante :

```
smitty chgenet
```

Remplacez la valeur de **Enable Jumbo Frames** par `yes`, puis appuyez sur Entrée.

Vidage à distance

Le vidage à distance n'est pas pris en charge via un EtherChannel.

IEEE 802.3ad Link Aggregation

IEEE 802.3ad est une méthode standard permettant de créer un agrégat de liaisons. Sur le plan conceptuel, il fonctionne de la même façon qu'EtherChannel : plusieurs cartes Ethernet sont regroupées en une seule carte virtuelle, fournissant une bande passante plus élevée et la protection contre les échecs. Comme EtherChannel, IEEE 802.3ad nécessite d'être pris en charge par le commutateur.

Dans IEEE 802.3ad, le protocole indique automatiquement au commutateur les ports qui doivent être regroupés en agrégat. Lorsqu'un agrégat IEEE 802.3ad est configuré, des unités Link Aggregation Control Protocol Data Units (LACPDU) sont échangées entre le serveur et le commutateur. Ce protocole LACP (Link Aggregation Control Protocol) prévient le commutateur que les cartes configurées dans l'agrégat doivent être considérées comme une seule carte sur le commutateur sans autre intervention de l'utilisateur.

Bien que le spécification IEEE 802.3ad ne permette pas à l'utilisateur de choisir les cartes qui doivent être regroupées en un agrégat, AIX permet à l'utilisateur de choisir les cartes.

Pour pouvoir être regroupées en un agrégat (c'est–à–dire que le commutateur leur permettra d'appartenir au même agrégat), les cartes doivent avoir la même vitesse de ligne (par exemple 100 Mbps ou 1 Gbps) et utiliser toutes le mode de duplex intégral. Si vous

tentez de placer des cartes de vitesses différentes ou utilisant des modes duplex différents, vous parviendrez à créer un agrégat sur le système, mais le commutateur risque de ne pas créer l'agrégat des cartes. Si c'est le cas, vous remarquerez une baisse des performances du réseau. Pour savoir si la création d'un agrégat a réussi, reportez–vous à Identification des incidents IEEE 802.3ad page 4-197.

Selon la spécification IEEE 802.3ad, les paquets envoyés à la même adresse IP sont tous envoyés via la même carte. Ainsi, en mode `8023ad` , les paquets seront toujours distribués de façon standard, jamais circulaire (round–robin).

La carte de secours est disponible pour IEEE 802.3ad Link Aggregations. La carte de secours n'a pas besoin d'être connectée à un commutateur IEEE 802.3ad, mais si elle l'est, la carte de secours continuera d'utiliser IEEE 802.3ad LACP.

Vous pouvez aussi configurer un IEEE 802.3ad Link Aggregation si le commutateur prend en charge EtherChannel mais pas IEEE 802.3ad. Dans ce cas, vous devrez configurer manuellement les ports en tant qu'un EtherChannel sur le commutateur (tout comme si un EtherChannel normal avait été créé). En définissant le mode à `8023ad`, l'agrégat fonctionne avec EtherChannel ainsi qu'avec les commutateurs IEEE 802.3ad–enabled. Pour plus d'informations sur l'interopérabilité, reportez–vous à la section Scénarios d'interopérabilité page 4-197 .

Remarque : Les étapes d'activation de l'utilisation de IEEE 802.3ad varient d'un commutateur à l'autre. Consultez la documentation du commutateur pour déterminer les étapes initiales, le cas échéant, qui doivent être effectuées pour activer LACP sur le commutateur.

Pour plus d'informations sur la configuration et l'utilisation de IEEE 802.3ad Link Aggregation, reportez–vous à Configuration de IEEE 802.3ad Link Aggregation page 4-195.

Remarques

Prenez en compte les éléments suivants avant de configurer IEEE 802.3ad Link Aggregation :

- Bien que la mise en œuvre de IEEE 802.3ad sous AIX permette à Link Aggregation de contenir des cartes de différentes vitesses de ligne, vous devez uniquement créer des agrégats de cartes ayant la même vitesse et utilisant le mode duplex intégral. Ceci évite les problèmes potentiels de configuration de Link Aggregation sur le commutateur. Reportez–vous à la documentation du commutateur pour plus d'informations sur les types d'agrégats autorisés par le commutateur.
- Si vous utilisez des cartes 10/100 Ethernet dans le Link Aggregation, vous devez activer le sondage de liaison sur ces cartes avant de les ajouter à l'agrégat. Tapez `smitty chgenet` sur la ligne de commande. Remplacez la valeur de **Enable Link Polling** par `yes`, puis appuyez sur Entrée. Procédez de cette façon pour chaque carte 10/100 Ethernet que vous ajoutez au Link Aggregation.

Configuration de IEEE 802.3ad Link Aggregation

Procédez comme suit pour configurer un IEEE 802.3ad Link Aggregation :

1. Tapez `smit etherchannel` sur la ligne de commande.
2. Sélectionnez **Add an EtherChannel / Link Aggregation** dans la liste et appuyez sur Entrée.
3. Sélectionnez les cartes Ethernet principales de Link Aggregation et appuyez sur Entrée. Si vous prévoyez d'utiliser une carte de secours, ne sélectionnez pas la carte de secours à ce stade. L'option de la carte de secours est disponible dans AIX 5.2 et les versions supérieures.

Remarque : **Cartes réseau disponibles** affiche toutes les cartes Ethernet. Si vous sélectionnez une carte Ethernet déjà utilisée (dont l'interface est définie), vous obtenez un message d'erreur. Vous devez d'abord détacher ces interfaces si vous souhaitez les utiliser.

4. Entrez les informations dans les champs en respectant les consignes suivantes :
   - **Cartes EtherChannel / Link Aggregation :** Vous devez voir s'afficher toutes les cartes principales utilisées dans le Link Aggregation. Vous avez sélectionné ces cartes à l'étape précédente.
   - **Enable Alternate Address:** Ce champ est facultatif. Le choix de la valeur `yes` vous permet de préciser une adresse MAC que le Link Aggregation doit utiliser. Si vous définissez la valeur `no` pour cette option, le Link Aggregation utilisera l'adresse MAC de la première carte indiquée.
   - **Alternate Address:** Si vous attribuez à **Enable Alternate Address** la valeur `yes`, indiquez l'adresse MAC à utiliser ici. L'adresse indiquée doit commencer par `0x` et être une valeur hexadécimale à 12 chiffres.
   - **Enable Gigabit Ethernet Jumbo Frames:** Ce champ est facultatif. Pour l'utiliser, le commutateur doit prendre en charge les trames jumbo. Ceci fonctionne uniquement avec une interface Ethernet Standard (en) mais pas avec une interface IEEE 802.3. Affectez la valeur `yes` si vous voulez l'activer.
   - **Mode:** Entrez `8023ad`.
   - **Backup Adapter:** Ce champ est facultatif. Indiquez la carte à utiliser comme carte de secours. L'option de la carte de secours est disponible dans AIX 5.2 et les versions supérieures.
   - **Internet Address to Ping:** Ce champ est facultatif et n'est disponible que si vous avez une seule carte dans l'agrégat et une carte de secours. Link Aggregation lance une commande ping sur l'adresse IP indiquée ici. Si Link Aggregation ne parvient pas à lancer une commande ping au terme du nombre de tentatives précisé dans **Number of Retries** dans les intervalles **Retry Timeout**, Link Aggregation effectue un basculement des cartes.
   - **Number of Retries:** Entrez le nombre d'échecs de réponses ping autorisés avant que Link Aggregation ne change de cartes. La valeur par défaut est 3. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.
   - **Retry Timeout:** Entrez le nombre de secondes entre les envois de commande ping de Link Aggregation à **Internet Address to Ping**. La valeur par défaut est une seconde. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.
5. Appuyez sur Entrée après avoir modifié les champs voulus pour créer le Link Aggregation.
6. Configurez IP via la nouvelle unité Link Aggregation en tapant `smit chinet` sur la ligne de commande.
7. Sélectionnez votre nouvelle interface Link Aggregation dans la liste.
8. Remplissez tous les champs requis et appuyez sur Entrée.

Gestion de IEEE 802.3ad

Pour les tâches de gestion pouvant être exécutées sur un IEEE 802.3ad Link Aggregation après la configuration, reportez–vous à Gestion d'EtherChannel et de IEEE 802.3ad Link Aggregation.

Identification des incidents de IEEE 802.3ad

Si des problèmes liés au IEEE 802.3ad Link Aggregation se produisent, lancez la commande suivante pour vérifier le mode de fonctionnement de Link Aggregation :

```
entstat -d   device
```

où *device* est l'unité Link Aggregation.

Ceci permet également de déterminer le meilleur effort de l'état de la progression du LACP basé sur les unités LACPDU reçues du commutateur. Les valeurs d'état possibles sont les suivantes :

- `Inactive`: LACP n'a pas été initialisé. Dans cet état, un Link Aggregation n'a pas encore été configuré, soit parce qu'il n'a pas encore reçu d'adresse IP soit parce que son interface a été détachée.
- `Negotiating`: LACP est en cours, mais le commutateur n'a pas encore regroupé les cartes en agrégat. Si le Link Aggregation conserve cet état plus d'une minute, vérifiez que le commutateur est correctement configuré. Vérifiez par exemple que LACP est activé sur les ports.
- `Aggregated`: LACP a réussi et le commutateur a regroupé les cartes dans un agrégat.
- `Failed`: LACP a échoué. Certaines des causes possibles sont que les cartes de l'agrégat ont des vitesses de ligne ou des modes duplex différents ou qu'elles sont connectées à des commutateurs différents. Vérifiez les configurations des cartes.

  En outre, certains commutateurs permettent uniquement aux ports contigus d'être regroupés et imposent parfois une limite au nombre de cartes pouvant être regroupées. Consultez la documentation du commutateur pour connaître les limites éventuelles du commutateur, puis vérifiez sa configuration.

Remarque : L'état du Link Aggregation est une valeur de diagnostic et n'a pas d'incidence sur le côté AIX de la configuration. Cette valeur d'état a été calculée via une tentative de meilleur effort. Pour résoudre les autres problèmes d'agrégat, il est conseillé de vérifier la configuration du commutateur.

Scénarios d'interopérabilité

Le tableau suivant représente plusieurs scénarios d'interopérabilité. Prenez en compte ces scénarios lorsque vous configurez un EtherChannel ou IEEE 802.3ad Link Aggregation. Vous trouverez une explication supplémentaire de chaque scénario à la suite du tableau.

Mode de configuration AIX	Configuration de commutateur	Résultat
`8023ad`	IEEE 802.3ad LACP	OK – AIX lance des unités LACPDU qui déclenchent un IEEE 802.3ad Link Aggregation sur le commutateur.
`standard` ou `round_robin`	EtherChannel	OK – Résultats dans le comportement EtherChannel traditionnel.

`8023ad`	EtherChannel	OK – Résultats dans le comportement EtherChannel traditionnel. AIX initialise des unités LACPDU mais le commutateur n'en tient pas compte.
`standard` ou `round_robin`	IEEE 802.3ad LACP	Non souhaitable – Le commutateur ne peut pas former d'agrégat. Il en résulte des performances médiocres car AIX transfère l'adresse MAC entre les ports du commutateur.

- `8023ad` avec IEEE 802.3ad LACP :

  Il s'agit de la configuration IEEE 802.3ad la plus courante. Ce commutateur peut être un LACP passif ou actif.

- `standard` ou `round_robin` avec EtherChannel :

  Il s'agit de la configuration EtherChannel la plus courante.

- `8023ad` avec EtherChannel :

  Dans ce cas, AIX enverra les unités LACPDU mais elles n'obtiendront pas de réponse car le commutateur opère comme un EtherChannel. Le fonctionnement sera cependant correct car le commutateur continuera de traiter ces ports comme une liaison unique.

  Remarque : Dans ce cas, la commande `entstat -d` signalera toujours que l'agrégat a l'état `Negotiating`.

- `standard` ou `round_robin` avec IEEE 802.3ad LACP :

  Cette configuration est incorrecte. Si le commutateur utilise LACP pour créer un agrégat, celui n'est jamais formé car AIX ne répond jamais aux unités LACPDU. Pour obtenir un fonctionnement correct, définissez le mode `8023ad` dans AIX.

Recherche de MTU d'accès

Pour deux hôtes communiquant via un chemin d'accès à des réseaux multiples, les paquets transmis sont fragmentés si leur taille dépasse celle de la plus petite MTU d'un réseau quelconque du chemin d'accès. La fragmentation étant susceptible de réduire les performances du réseau, il suffit, pour l'éviter, de transmettre des paquets de taille inférieure ou égale à celle de la plus petite MTU du chemin d'accès du réseau : vous faites alors appel à la MTU d'accès.

Un algorithme de recherche de MTU d'accès est pris en charge par ce système tel que défini dans le RFC 1191. Pour l'activer pour les applications TCP et UDP, modifiez les options **tcp_pmtu_discover** et **udp_pmtu_discover** de la commande **no**. Quand elle est activée pour TCP, la recherche de MTU d'accès impose automatiquement aux paquets transmis par les applications TCP une taille ne dépassant pas la MTU d'accès. Les applications UDP déterminent elles-mêmes la taille de leurs paquets transmis : aussi doivent-elles être configurées pour utiliser l'information de MTU d'accès via l'option socket **IP_FINDPMTU**, même si l'option **udp_pmtu_discover no** est activée. Les options **tcp_pmtu_discover** et **udp_pmtu_discover** sont désactivées par défaut de la version AIX Version 4.2.1 à AIX Version 4.3.1, et activées sur la version AIX Version 4.3.2 et les versions ultérieures.

Une fois la MTU d'accès trouvée pour une route de réseau, une route hôte distincte est "clonée" pour le chemin d'accès. Vous pouvez afficher les routes hôte "clonées" et la valeur MTU d'accès pour la route avec la commande **netstat –r**. L'accumulation des routes "clonées" peut être évitée en permettant l'expiration et la suppression des routes inutilisées. L'option **route_expire** de la commande **no** contrôle l'expiration des routes ; elle est désactivée par défaut. L'option de contrôle de l'expiration des routes est désactivée par défaut de la version AIX Version 4.2.1 à AIX Version 4.3.1, et définie sur 1 minute dans les versions AIX Version 4.3.2 et ultérieures.

Les routes pouvant être modifiées dynamiquement, les valeurs MTU d'accès peuvent également changer dans le temps. La diminution de ces valeurs étant susceptible de provoquer la fragmentation de paquets, ces valeurs sont analysées régulièrement (toutes les 10 minutes par défaut). Vous pouvez modifier la fréquence d'analyse avec l'option **pmtu_default_age** de la commande **no**.

L'augmentation des valeurs MTU d'accès peut accroître les performances du réseau. Les valeurs trouvées sont donc analysées régulièrement pour y rechercher une augmentation (toutes les 30 minutes par défaut). Vous pouvez modifier la fréquence d'analyse avec l'option **pmtu_rediscover_interval** de la commande **no**.

Si tous les routeurs du chemin d'accès au réseau n'admettent pas le RFC 1191, déterminer la valeur MTU d'accès exacte peut s'avérer impossible. Dans ce cas, la commande **mmtu** permet d'entériner ou non les valeurs testées.

Remarques :

1. Sur les routes en double et sur celles définies avec group routing, la recherche de MTU d'accès n'est pas possible.
2. Avec la recherche de MTU d'accès activée, l'option **arpqsize** de la commande **no** a sa valeur minimale définie à 5. Si, par la suite, la recherche de MTU d'accès est désactivée, cette valeur n'est pas diminuée.

protocole SLIP

Configuration de SLIP pour modem

Pour configurer un protocole SLIP entre deux systèmes communiquant via un modem, vous pouvez utiliser le raccourci Web-based System Manager **wsm** ou la procédure suivante, qui fait appel à SMIT et à la ligne de commande. Les deux hôtes sont appelés bronze et gold.

1. Connectez physiquement les modems à bronze et gold.
2. Pour créer un tty sur bronze via SMIT :
   a. Entrez :

   ```
   smit maktty
   ```

   b. Sélectionnez **rs232** comme type de tty.
   c. Sélectionnez un port série disponible, par exemple **sa0** (port série système 1).
   d. Sélectionnez dans la liste un numéro de port pour le tty.
   e. Définissez le débit (en bauds) de votre modem.
   f. Désactivez l'option Activation de la connexion.
   g. Quittez SMIT.
3. Créez un tty sur gold.

   Suivez la même procédure que pour bronze (étape 2), excepté pour l'option Activation de la connexion, qui doit être activée (**enable**).

   Dans la suite de la procédure, le numéro tty de bronze et de gold est supposé être tty1.
4. Testez la connexion physique avec ATE.
   a. Côté bronze, entrez :

   ```
   ate
   ```

   b. Dans le MENU PRINCIPAL (ETAT NON CONNECTE), sélectionnez **Alter**. Indiquez, en bauds, le débit de votre modem (Rate), et tty1 comme unité (Device).
   c. Dans le MENU PRINCIPAL (ETAT NON CONNECTE), sélectionnez **Connect**. Lorsque vous y êtes invité par ATE, composez le numéro d'appel de gold et appuyez sur Entrée.
   d. A ce stade, vous devez recevoir une invite de connexion pour gold. Login.
   e. Enfin, à partir de l'écran connecté, déconnectez-vous de gold, appuyez sur **Ctrl–v** (pour appeler le MENU PRINCIPAL (ETAT CONNECTE), entrez **t** pour mettre fin à la connexion, puis **q** pour quitter ATE.

   Remarque : Si vous ne recevez pas d'invite de connexion, revenez à l'étape 1 et vérifiez la configuration. Ne poursuivez qu'une fois la connexion établie avec gold. La configuration de tty pour ATE est légèrement différente de celle pour SLIP. Pour cette raison, vous devez apporter les modifications suivantes :

   a. Côté bronze, entrez :

   ```
   smit chgtty
   ```

   b. Côté gold, entrez :

   ```
   smit chgtty-pdisable tty1
   ```

   Sélectionnez **tty1**, puis **Modification/affichage d'un programme TTY**. Désactivez l'option Activation de la connexion puis quittez SMIT.

5. Insérez la ligne suivante dans le fichier **/usr/lib/uucp/Devices** de bronze et de gold :

   ```
   Direct tty1 - 9600 direct
   ```

   ou remplacez `9600` par tout autre débit de modem.

6. Créez une interface de réseau SLIP sur bronze.

   a. Entrez :

   ```
   smit mkinet1sl
   ```

   b. Pour le port TTY de l'interface de réseau SLIP, sélectionnez **tty1**.

   c. Spécifiez une adresse Internet, par exemple 130.130.130.1.

   d. Spécifiez une adresse de destination (de gold), par exemple 130.130.130.2.

   e. Spécifiez le débit (en bauds) de votre modem.

   f. Spécifiez la chaîne de numérotation, par exemple :

      - "" AT OK ATDT555-1234 CONNECT ""
      - Cette commande signifie : Utilisez `tty1` à `9600` bauds. Envoyez AT au modem. Le modem doit répondre OK. Composez le numéro d'appel `555-1234`. Le modem doit répondre CONNECTE. Les espaces avant et après les doubles guillemets sont obligatoires.

   g. Quittez SMIT.

7. Créez une interface de réseau SLIP sur gold.

   Suivez la même procédure que pour bronze (étape 5), mais en inversant les adresses Internet et de destination.

8. Ajoutez les deux lignes ci-dessous dans le fichier **/etc/hosts** de bronze et de gold :

   ```
   130.130.130.1   bronze
    130.130.130.2   gold
   ```

   Le nom attribué doit être unique. Autrement dit, si le nom `bronze` est déjà attribué à l'interface de réseau en anneau à jeton de l'hôte bronze, choisissez-en un autre pour l'interface SLIP, tel que `bronze_slip`.

   Remarque : Le script `/usr/sbin/slipcall.` fournit une interface simplifiée pour la commande **slattach**.

9. Testez la connexion SLIP.

   a. Côté bronze, entrez :

   ```
   ping gold
   ```

   b. Côté gold, entrez :

   ```
   ping bronze
   ```

   Si les deux tests aboutissent, la connexion SLIP peut être utilisée. Sinon, revenez à l'étape 5 et vérifiez la configuration sur bronze et sur gold.

Configuration de SLIP pour câble de modem nul

Pour configurer un protocole SLIP entre deux systèmes communiquant via un câble de modem nul, vous pouvez utiliser le raccourci Web-based System Manager **wsm** ou la procédure suivante, qui fait appel à SMIT et à la ligne de commande. Les deux hôtes sont appelés bronze et gold.

1. Reliez physiquement bronze et gold par un câble de modem nul. Les câbles suivants vous sont nécessaires. (Ils sont répertoriés dans l'ordre de leur connexion, du bronze au gold.)

   a. Câble B (référence 00G0943). Câble de raccordement port série : livrés avec chaque système (sauf pour les modèles 220, 340 et 350 qui ne le requièrent pas).

b. Câble D (référence 6323741, code 2936). Câble asynchrone EIA-232/V.24.

c. Câble E (référence 59F2861, code 2937). Interposeur imprimante/terminal EIA-232 (câble de modem nul).

d. Carte échangeur (prises des deux côtés).

2. Créez un tty sur bronze.

   a. Entrez :

   ```
   smit maktty
   ```

   b. Sélectionnez **rs232** comme type de tty.

   c. Sélectionnez un port série disponible, par exemple **sa0** (port série système 1).

   d. Sélectionnez dans la liste un numéro de port pour le tty.

   e. Fixez le débit, en bauds, à19200 (vous le passerez ultérieurement à 38400).

   f. Désactivez l'option Activation de la connexion puis quittez SMIT.

3. Créez un tty sur gold.

   Suivez la même procédure que pour bronze (étape 2), excepté pour l'option Activation de la connexion, qui doit être activée (**enable**).

   Remarque : Dans la suite de la procédure, le numéro tty de bronze et de gold est supposé être tty1.

4. Testez la connexion physique avec ATE.

   a. Côté bronze, entrez :

   ```
   ate
   ```

   b. Dans le MENU PRINCIPAL (ETAT NON CONNECTE), sélectionnez **Alter**. Indiquez 19200 comme débit (Rate) et tty1 comme unité (Device).

   c. Dans le MENU PRINCIPAL (ETAT NON CONNECTE), sélectionnez **Connect**. Lorsque vous y êtes invité par ATE, à composer un numéro de téléphone, appuyez sur Entrée. Le message suivant doit s'afficher :

   ```
   ate: 0828-010 La Commande Connect a établi une connexion via
   le port tty1
   ```

   d. Appuyez sur Entrée. Vous devez recevoir une invite de connexion pour gold. Connectez-vous.

   e. Enfin, à partir de l'écran connecté, déconnectez-vous de gold, appuyez sur **Ctrl–v** (pour appeler le MENU PRINCIPAL (ETAT CONNECTE), entrez **t** pour mettre fin à la connexion, puis **q** pour quitter ATE.

   Remarque : Si vous ne recevez pas d'invite de connexion, revenez à l'étape 1 et vérifiez la configuration. Ne poursuivez qu'une fois la connexion établie avec gold. La configuration de tty pour ATE est légèrement différente de celle pour SLIP. Pour cette raison, vous devez apporter les modifications suivantes :

   a. Côté bronze, entrez :

   ```
   smit chgtty
   ```
   Sélectionnez **tty1**. Fixez le débit en bauds à 38400 puis quittez SMIT.

   b. Côté gold, entrez :

   ```
   pdisable tty1
   ```

   c. Côté gold, entrez :

   ```
   smit chgtty
   ```

Sélectionnez **tty1**. Désactivez l'option Activation de la connexion, fixez le débit (en bauds) à 38400, puis quittez SMIT.

5. Insérez la ligne suivante dans le fichier **/usr/lib/uucp/Devices** de bronze et de gold :

   ```
   Direct tty1 - 38400 direct
   ```

6. Créez une interface de réseau SLIP sur **bronze**.

   a. Entrez :

   ```
   smit mkinet1sl
   ```

   b. Pour le port TTY de l'interface de réseau SLIP, sélectionnez **tty1**.

   c. Spécifiez l'adresse Internet, par exemple 130.130.130.1.

   d. Spécifiez l'adresse de destination (de gold), par exemple 130.130.130.2, puis appuyez sur Entrée.

7. Créez une interface de réseau SLIP sur gold.

   Suivez la même procédure que pour bronze (étape 5), mais en inversant les adresses Internet et de destination.

8. Ajoutez les deux lignes ci-dessous dans le fichier **/etc/hosts** de bronze et de gold :

   ```
   130.130.130.1   bronze
    130.130.130.2   gold
   ```

   Le nom attribué doit être unique. Autrement dit, si le nom `bronze` est déjà attribué à l'interface de réseau en anneau à jeton de l'hôte bronze, choisissez-en un autre pour l'interface SLIP, tel que `bronze_slip`.

9. Lancez SLIP sur bronze et gold.

   Entrez :

   ```
   slattach tty1
   ```

10. Testez la connexion SLIP.

    a. Côté bronze, entrez :

    ```
    ping gold
    ```

    b. Côté gold, entrez :

    ```
    ping bronze
    ```

    Si les deux tests aboutissent, la connexion SLIP peut être utilisée. Sinon, revenez à l'étape 5 et vérifiez la configuration sur bronze et sur gold.

Désactivation d'une connexion SLIP

Pour désactiver une connexion SLIP :

1. Entrez :

   ```
   ps -ef | grep slatt
   ```
   Relevez le numéro des process associés à la commande **slattach**.

2. Pour chaque numéro de process, entrez :

   ```
   kill process_number
   ```
   N'utilisez pas l'indicateur **–9** de la commande **kill**.

   Si l'indicateur **–9** est malencontreusement associé à la commande **slattach**, un verrou slip est susceptible d'être resté dans /etc/locks. Supprimez-le pour le nettoyage post-**slattach**.

Suppression d’un TTY

Pour retirer un tty, vous disposez du raccourci Web-based System Manager **wsm** ou du raccourci SMIT **smit rminet**.

Protocole asynchrone point-à-point (PPP)

Le sous–système asynchrone PPP (Point–to–Point Protocol) offre une alternative à SLIP. en proposant une méthode standard pour le transport des datagrammes multiprotocoles au travers de supports point–à–point. PPP se compose de trois couches principales :

1. Une méthode d'encapsulage des datagrammes multiprotocoles. PPP prend en charge les protocoles de couche réseau TCP/IP.
2. Un protocole LCP (Link Control Protocol) qui établit, configure et teste la connexion de liaison de données. PPP l'implémente via des extensions de noyau.
3. Une famille de protocoles NCP (Network Control Protocols) pour établir et configurer différents protocoles de couche réseau. PPP prend en charge le protocole IPCP (Internet Protocol Control Protocol) pour négocier une connexion TCP/IP.

PPP prend en charge les RFC (Request for Comments) suivants :

- RFC 1661, *"The Point-to-Point Protocol, LCP."*
- RFC 1332, *"The PPP Internet Protocol Control Protocol (IPCP)."*
- RFC 1662, *"PPP in HDLC-like Framing."*
- RFC 1334, *"PPP Authentication Protocols."*
- RFC 1990, *PPP Multilink*

PPP différencie client et serveur. Ce système peut être à la fois client et serveur : la distinction n'a pour but que de simplifier la configuration. Les serveurs PPP tentent d'affecter un pool d'adresses IP parmi les connexions en cours d'établissement. Il existe une corrélation entre les unités de support : PPP la rompt. Toutes les connexions PPP serveur sont affectées sur la base du " premier disponible ", ceci pour faciliter la séparation entre PPP et le support. Le processus de raccordement doit demander à être lié au type de liaison adéquat.

Processus utilisateur

Le protocole asynchrone PPP sur ce système d'exploitation utilise trois niveaux de processus utilisateur :

1. Un démon de contrôle (**pppcontrold**) exécuté par la racine sous le contrôleur SRC (System Resource Controller) (**startsrc -s pppcontrold**). Ce démon assure le chargement et la configuration de toutes les extensions de noyau associées au sous-système. Il reste actif aussi longtemps que PPP est requis par le système d'exploitation.
2. Un processus de liaison (**pppattachd**) qui associe un flot TTY à une instance du protocole de contrôle de liaison NPC (Network Control Protocol), et un protocole datagramme. Il existe une instance de **pppattachd** pour chaque connexion PPP active dans le système. Tout utilisateur du processus de liaison doit appartenir au groupe **uucp** et comporter **/usr/sbin** dans sa variable d'environnement **PATH**.
3. Un processus de numérotation (**pppdial**) qui établit une connexion sortante. Le numéroteur est conçu pour être exécuté avec **pppattachd** comme programme connecteur. Son objet est d'interagir au travers de l'unité asynchrone avant la négociation PPP. Cette interaction est définie de la même façon que le format du dialogue chat UUCP. Le numéroteur aide à établir une connexion avec un système distant. L'établissement effectif de la session est hors de la portée de PPP.

Configuration du protocole asynchrone PPP

Vous pouvez configurer le protocole asynchrone PPP à l'aide de Web-based System Manager ou de SMIT. Toutes les opérations de configuration sont indiquées dans le tableau ci-après. Ces opérations sont accessibles à l'utilisateur racine.

Pour la configuration initiale de votre système, vous aurez à :

- ajouter une configuration de liaison
- ajouter une interface de serveur (si vous définissez la machine en tant que serveur PPP),
- ajouter une interface de demande
  (si vous souhaitez que la machine accepte les connexions à la demande),
- manipuler les utilisateurs/mots de passe PAP ou CHAP(si vous désirez que la machine gère l'authentification PPP),
- lancer PPP pour prendre les modifications en compte
  (ou arrêter puis relancer PPP si ce protocole est actif).

Configuration du protocole PPP asynchrone		
Tâche	*Raccourci SMIT*	Web-based System Manager Management Environment
Création d'une configuration de contrôle de liaison	**smit ppplcp**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link**.
Ajouter une configuration de liaison	**smit addlcp**	
Modifier/afficher une configuration de liaison	**smit chglcp**	
Supprimer une configuration de liaison[1]	**smit rmlcp**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link** —> **Link Configuration** —> **Remove Link Configuration**.
Créer des interfaces IP PPP	**smit pppip**	
Ajouter une interface serveur	**smit addpppserver**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link** —> **Server Interfaces** —> **Add/Change Interface**.
Modifier/afficher une interface serveur	**smit listserver**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link** —> **Server Interfaces** —> **Add/Change Interface**.

Supprimer une interface serveur[1]	**smit rmlistserver**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link** —> **Server Interfaces** —> **Delete Interface**.
Ajouter une interface de demande	**smit addpppdemand**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link** —> **Demand Interfaces** —> **Add/Change Interface**.
Modifier/afficher une interface de demande	**smit listdemand**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link** —> **Demand Interfaces** —> **Add/Change Interface**.
Supprimer une interface de demande[1]	**smit rmlistdemand**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Configure the Point–to–Point Link** —> **Demand Interfaces** —> **Delete Interface**.
Manipuler les utilisateurs/mots de passe PAP	**smit ppppap**	
Ajouter un utilisateur PAP	**smit addpapuser**	
Modifier/afficher un utilisateur PAP	**smit listpapuser**	
Supprimer un utilisateur PAP	**smit rmpapuser**	
Manipuler les utilisateurs/mots de passe CHAP	**smit pppchap**	
Ajouter un utilisateur CHAP	**smit addchapuser**	
Modifier/afficher un utilisateur CHAP	**smit listchapuser**	
Supprimer un utilisateur CHAP	**smit rmchapuser**	
Lancer PPP[2]	**smit startppp**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Start the PPP Subsystem**.
Arrêter PPP[3]	**smit stopppp**	Software —> **Network** —> **TCPIP (IPv4 et IPv6)** —> **Point–to Point (PPP)** —> **Stop the PPP Subsystem**.

Remarques :

1. Sélectionner cette opération fait disparaître les informations existantes.

2. Lancer le protocole PPP est également possible avec la commande **startsrc –s pppcontrold**. En outre, via l'interface SMIT, vous pouvez demander que ce protocole soit lancé à l'amorçage du système.
3. Arrêter le protocole PPP est aussi possible avec la commande **stopsrc –s pppcontrold**. Par ailleurs, via SMIT, vous pouvez demander que ce protocole ne soit pas démarré à l'amorçage du système.

Protocoles PPP et SNMP

L'interaction de PPP avec le démon TCP/IP SNMP permet d'obtenir les informations relatives à la configuration de la couche de liaison PPP, et celles concernant les interfaces LCP (Link Control Protocol). Dans la mesure où la configuration de TCP/IP SNMP et du logiciel de gestion de SNMP est correcte, PPP SNMP peut :

- rechercher les informations sur la configuration de la liaison PPP (Maximum Receive Unit size, Asynchronous Character Mapping, etc.)
- définir les informations de configuration de la liaison PPP ;
- rechercher les informations relatives à l'interface LCP pour les liaisons LCP actives ;
- passer l'état des liaisons LCP actives à "down" en définissant l'objet **ifAdminStatus** approprié dans la base MIB (Management Information Base).

Tous les objets définis dans le RFC 1471 pour la MIB PPP ne sont pas pris en charge. Seul le tableau **pppLink** s'applique au sous-système PPP : les parties **pppLqr** et **pppTests** ne sont donc pas prises en charge. La partie **pppLink** est prise en charge, excepté les objets suivants :

- L'objet **pppLinkConfigMagicNumber** est accessible seulement en lecture. Dans PPP, la négociation de numéros magiques est toujours exécutée et ne peut être désactivée.
- L'objet **pppLinkConfigFcsSize** est accessible seulement en lecture. PPP n'accepte que la taille 16 pour FCS sous ce système.

Activation de PPP SNMP

SNMP pour PPP est désactivé par défaut. Vous pouvez activer PPP SNMP via le raccourci Web-based System Manager **wsm** ou via la procédure suivante. Cette procédure est accessible à l'utilisateur racine.

Remarque : La configuration de liaison PPP est supposée définie avant d'entamer cette procédure. Sinon, exécutez la procédure décrite à la section Configuration du protocole asynchrone PPP page 4-206 avant d'activer PPP SNMP.

1. Lancez l'interface SMIT et affichez l'écran Change/Show a Link Configuration avec la commande :

   ```
   smit chglcp
   ```

2. Basculez sur yes le champ Enable PPP SNMP subagent.
3. Validez vos modifications et quittez SMIT.

PPP SNMP sera activé au prochain redémarrage du protocole PPP.

- Si PPP est en cours d'utilisation :

1. Arrêtez-le avec le raccourci **smit stopppp** (voir le tableau à la section Configuration du protocole asynchrone PPP, page 4-206).
2. Vérifiez régulièrement où en est l'arrêt complet du sous-système, via la commande :

   ```
   lssrc –s pppcontrold
   ```

   Le temps que prend l'arrêt complet dépend du nombre de liaisons défini dans la configuration PPP. L'état `inoperative` renvoyé par la commande ci-dessus signifie que le sous-système est complètement arrêté.
3. Démarrez-le avec le raccourci **smit startppp** (voir le tableau à la section Configuration du protocole asynchrone PPP, page 4-206).

- Si PPP n'est pas en cours d'utilisation, lancez-le via le raccourci **smit startppp** (voir le tableau à la section Configuration du protocole asynchrone PPP, page 4-206).

Normes QoS (Qualité du service) TCP/IP

Les normes QoS (Quality of Service) sont une famille de normes Internet qui offrent un mode de traitement préférentiel de certains types de trafic IP. Ces normes peuvent réduire les délais d'attente variables et la congestion ayant pour effet de limiter les performances du réseau. Le système d'exploitation offre une prise en charge des normes QoS au niveau de l'hôte afin de répartir le trafic vers l'extérieur en classes de service distinctes. Ces normes permettent également d'indiquer et de faire des réservations de ressources telles que l'exigent les applications clients.

Les normes QoS peuvent être utilisées par un organisme pour déployer et mettre en place des politiques de gestion du réseau régissant l'utilisation de la largeur de bande du réseau. Avec les normes QoS, un hôte peut procéder aux opérations suivantes :

- Réguler le volume d'un certain type de trafic au sein du réseau ;
- Marquer des paquets sélectionnés en fonction d'un certain type de politique afin que les routeurs puissent par la suite fournir le service demandé ;
- Prendre en charge des services, tels que le service virtuel de lignes spécialisées avec une prise en charge appropriée des normes QoS le long de la route ;
- Participer aux requêtes de réservation de ressources des destinataires et annoncer les sessions expéditrices disponibles pour ces requêtes.

La prise en charge des normes QoS offre les fonctions suivantes :

- Services différenciés tels que définis dans la norme RFC 2474
- Politique de gestion du trafic
- Marquage des paquets à l'intérieur et hors du profil
- Conception du trafic
- Mesure
- Services intégrés pour applications client et serveur tels que définis dans la norme RFC 1633
- Signalisation RSVP (RFC 2205)
- Service garanti (RFC 2212)
- Service de contrôle de charge (RFC 2211)
- Mise en réseau conformément à la politique en vigueur
- Bibliothèque RAPI partagée destinée aux applications

Le sous–système de normes QoS se compose de quatre éléments :

Extension du noyau QoS (/usr/lib/drivers/qos)
L'extension du noyau QoS réside dans le répertoire **/usr/lib/drivers/qos** ; elle est chargée et déchargée à l'aide des méthodes de configuration **cfgqos** et **ucfgqos**. Cette extension de noyau permet la prise en charge QoS.

Agent de politique (/usr/sbin/policyd)
L'agent de politique est un démon de niveau utilisateur qui réside dans **/usr/sbin/policyd**. Il prend en charge la gestion de la politique et sert d'interface avec l'extension du noyau QoS afin d'installer, de modifier et de supprimer les règles de politique. Les règles de politique peuvent être définies dans le fichier de configuration local (**/etc/policyd.conf**), récupérées dans le serveur de réseau central à l'aide de LDAP, ou les deux.

Agent RSVP (/usr/sbin/rsvpd)

L'agent RSVP est un démon de niveau utilisateur qui réside dans **/usr/sbin/rsvpd**. Il met en œuvre la sémantique de protocole de signalisation RSVP.

Bibliothèque partagée RAPI (/usr/lib/librapi.a)

Les applications peuvent utiliser la RSVP API (RAPI) pour une meilleure qualité de service telle que définie par le modèle QoS Internet de services intégrés (Integrated Services Internet QoS model). Cette bibliothèque dialogue avec l'agent RSVP local afin de diffuser la requête QoS le long du chemin emprunté par le flux de données à l'aide du protocole RSVP. Cette API est une norme ouverte.

Remarque: Cette mise en œuvre de QoS est basée sur un ensemble de normes Internet en constante évolution et des ébauches de normes en cours d'élaboration par l'Internet Engineering Task Force (IETF) ainsi que ses divers groupes de travail. Les efforts de normalisation au sein de l'IETF permettront d'améliorer la cohérence et la définition de cette technologie. Il est également à noter que la QoS est une nouvelle technologie Internet récemment déployée au sein de ce réseau. Elle présente de nombreux avantages à tous les stades de son déploiement. Toutefois, les services bout en bout ne peuvent être offerts qu'avec une prise en charge totale de la technologie QoS.

Modèles QoS

Les modèles QoS pour Internet sont des normes ouvertes définies par l'IETF. Deux de ces modèles sont en cours de normalisation au sein de l'IETF : *Services intégrés* et *Services différenciés*. Ceux–ci renforcent le modèle traditionnel de service optimisé décrit dans la norme RFC 1812.

Services intégrés

Le service IS (Services intégrés) est un modèle dynamique de réservation des ressources pour Internet, tel que décrit dans la norme RFC 1633. Les hôtes utilisent un protocole de signalisation appelé Resource ReSerVation Protocol (RSVP) pour demander au réseau, de manière dynamique, une qualité de service spécifique. Les paramètres QoS sont acheminés dans ces messages RSVP et chaque nœud de réseau le long du chemin installe ces paramètres afin de disposer de la qualité de service requise. Ces paramètres QoS décrivent l'un des deux services actuellement définis, à savoir le service garanti et le service de contrôle de charge. L'IS se caractérise par le fait que cette signalisation porte sur chaque flux de trafic et que les réservations s'appliquent à chaque bond sur le chemin. Bien que ce modèle soit tout à fait à même de répondre à l'évolution constante des applications, il persiste encore certains problèmes en termes d'évolutivité qui empêchent son déploiement sur des réseaux au sein desquels des routeurs uniques gèrent plusieurs flux simultanés.

Services différenciés

Le service DS (Services différenciés) résout les problèmes d'évolutivité par flux et par bond par le biais d'un mécanisme simple de classification des paquets. Plutôt qu'une approche de signalisation dynamique, le service DS privilégie l'utilisation de bits dans l'octet TOS (type de service) IP afin de répartir les paquets en classes. Ce modèle de bit particulier dans l'octet TOS IP est appelé point de code DS. Il est utilisé par les routeurs pour définir la qualité de service fournie au niveau de ce bond en particulier, se rapprochant par là–même de leur mode d'acheminement IP par le biais de la consultation des tables de routage. Le traitement d'un paquet avec un point de code DS particulier s'appelle le PHB (per–hop behavior). Il est géré indépendamment à chaque nœud de réseau. La concaténation de ces différents PHB indépendants offre un service bout en bout.

Les services différenciés sont en cours de normalisation par un groupe de travail IETF, qui a défini trois PHB : le PHB avec acheminement expéditif (EF : abréviation de Expedited Forwarding), le groupe PHB avec acheminement assuré (AF : abréviation de Assured

Forwarding) et le PHB par défaut (DE : abréviation de par défaut). Le EF PHB peut être utilisé pour la mise en œuvre d'un service bout en bout, telle qu'une ligne spécialisée (VLL) offrant un délai d'attente et un taux d'instabilité faibles ainsi que des pertes réduites. L'AF est une famille de PHB, appelé groupe de PHB. Il est utilisé pour classer des paquets en fonction des différents niveaux de priorité. Le niveau de priorité attribué à un paquet détermine son importance relative au sein de la classe AF. Par ce biais, il est possible de bénéficier du service dit *Olympique*, à savoir bronze, argent et or. Le DE PHB est le modèle traditionnel de service optimisé tel que normalisé par la RFC 1812.

Normes prises en charge et ébauches de normes

Les ébauches de normes Internet et les RFC suivants traitent des normes sur lesquelles s'appuie cette mise en œuvre des modèles QoS.

RFC 2474	Définition du champ Services différenciés (champ DS) dans les en–têtes IP versions 4 et 6
RFC 2475	Architecture des services différenciés
RFC 1633	Présentation des services intégrés au sein de l'architecture Internet
RFC 2205	Protocole de réservation des ressources (RSVP)
RFC 2210	Utilisation du RSVP avec les services intégrés IETF
RFC 2211	Spécification du service d'éléments réseau avec contrôle de charge
RFC 2212	Spécification de la qualité de service garantie
RFC 2215	Paramètres de définition généraux des éléments réseau des services intégrés
draft–ietf–diffserv–framework–01.txt, octobre 1998	Cadre des services différenciés
draft–ietf–diffserv–rsvp–01.txt, novembre 1998	Cadre d'utilisation du RSVP avec des réseaux DIFF–serv
draft–ietf–diffserv–phb–ef–01.txt	Groupe PHB d'acheminement expéditif
draft–ietf–diffserv–af–04.txt	Groupe PHB d'acheminement assuré
draft–rajan–policy–qosschema–00.txt, octobre 1998	Schéma des services différenciés et intégrés au sein des réseaux
draft–ietf–rap–framework–01.txt, novembre 1998	Cadre pour contrôle d'admission [25] basé sur des règles de politique
draft–ietf–rap–rsvp–ext–01.txt, novembre 1998	Extensions RSVP pour contrôle de politique

Remarque : QoS est une technologie Internet émergente. Elle présente de nombreux avantages à tous les stades de son déploiement. Toutefois, les services bout en bout ne peuvent être offerts qu'avec une prise en charge totale de la technologie QoS.

Installation de QoS

QoS est livré avec **bos.net.tcp.server**. L'installation de ces fichiers est indispensable pour utiliser QoS. Pour utiliser la bibliothèque RAPI partagée, installez aussi **bos.adt.include**.

Configuration de QoS

Arrêt et démarrage du sous–système QoS

QoS peut être lancé ou arrêté avec le raccourci SMIT (**smit qos**) ou les commandes **mkqos** et **rmqos**.

Pour désactiver dès à présent le sous–système QoS et lors du prochain redémarrage du système, procédez comme suit :

```
/usr/sbin/rmqos –B
```

Pour activer le sous–système QoS pour la période en cours seulement, procédez comme suit :

```
/usr/sbin/mkqos –N
```

Reportez–vous à la description des commandes **mkqos** et **rmqos** pour le lancement et le retrait des indicateurs de commande.

Les démons **policyd** et **rsvpd** sont configurés par les fichiers de configuration **/etc/policyd.conf** et **/etc/rsvpd.conf**. Ces fichiers *doivent* être édités afin de personnaliser le sous–système QoS en fonction de l'environnement local. QoS ne fonctionne pas correctement avec les configurations type fournies.

Configuration de l'agent RSVP

L'agent RSVP est nécessaire si l'hôte doit prendre en charge le protocole du même nom. Utilisez le fichier de configuration **/etc/rsvpd.conf** pour configurer l'agent RSVP. La syntaxe de ce fichier est précisée dans le fichier de configuration type installé dans **/etc/rsvpd.conf**.

Configuration type

```
interface 1.2.30.1
interface 1.20.2.3 disabled
interface 1.2.3.3 disabled
interface 1.2.3.4
    {
      trafficControl
    }

rsvp 1.2.30.1
    {
      maxFlows 64
    }

rsvp 1.2.3.4
    {
      maxFlows 100
    }
```

L'exemple ci–dessus illustre une possibilité de configuration RSVP au sein de laquelle l'hôte a 4 interfaces (virtuelles ou physiques) représentées par les 4 adresses IP : `1.2.3.1, 1.2.3.2, 1.2.3.3` et `1.2.3.4`.

L'interface `1.2.3.1` a été activée pour le RSVP. Toutefois, la fonction de contrôle du trafic n'a pas été spécifiée et les messages RESV RSVP entrants n'entraînent pas la réservation des ressources au sein du sous–système TCP. Cette interface peut prendre en charge un maximum de 64 sessions RSVP simultanées.

Les interfaces `1.2.3.2` et `1.2.3.3` ont été désactivées. L'agent RSVP ne peut pas utiliser cette interface pour transmettre ni recevoir des messages RSVP.

L'interface `1.2.3.4` a été activée pour le RSVP. En outre, elle peut procéder à des réservations de ressources au sein du sous–système TCP en réponse à un message RESV RSVP. Cette interface peut prendre en charge jusqu'à 100 sessions RSVP.

Toutes les autres interfaces existantes sur l'hôte mais non reprises de manière explicite dans **/etc/rsvpd.conf** sont désactivées.

Configuration de l'agent de politique

L'agent de politique est un composant indispensable du sous–système QoS. Utilisez le fichier **/etc/policyd.conf** pour configurer l'agent de politique. La syntaxe de ce fichier est précisée dans le fichier de configuration type installé dans **/etc/policyd.conf**.

Vous pouvez configurer l'agent de politique en éditant /etc/policyd.conf. En outre, les commandes suivantes sont fournies pour faciliter la configuration des politiques :

- **qosadd**
- **qosmod**
- **qoslist**
- **qosremove**

Configurations type

Dans l'exemple suivant, une catégorie de service de qualité est créée et utilisée dans la règle de politique tcptraffic. Cette catégorie de service a une vitesse de transmission maximum de 110 000 Kbps, une profondeur de compartiment à jeton de 10 000 bits et une valeur TOS IP sortante de 11100000 en système binaire. La règle de politique tcptraffic offre ce service de qualité à l'ensemble du trafic pour lequel l'adresse IP source est fournie par `1.2.3.6`, avec l'adresse de destination `1.2.3.3` et le port de destination compris entre 0 et 1024.

```
ServiceCategories  premium
{
 PolicyScope     DataTraffic
 MaxRate         110000
 MaxTokenBucket  10000
 OutgoingTOS     11100000
}

ServicePolicyRules   tcptraffic
{
 PolicyScope    DataTraffic
 ProtocolNumber 6  # tcp
 SourceAddressRange      1.2.3.6-1.2.3.6
 DestinationAddressRange 1.2.3.3-1.2.3.3
 DestinationPortRange    0-1024
 ServiceReference        premium
}
```

Les instructions suivantes définissent une catégorie de service par défaut et l'utilisent pour réduire le trafic UDP entre les interfaces `1.2.3.1` à `1.2.3.4` et les adresses IP `1.2.3.6` à `1.2.3.10`, port 8000.

```
ServiceCategories  default
{
   MaxRate         110000
   MaxTokenBucket  10000
   OutgoingTOS     00000000
}
ServicePolicyRules   udptraffic
{
   ProtocolNumber 17  # udp
   SourceAddressRange      1.2.30.1-1.2.30.4
   DestinationAddressRange 1.20.60.10-1.2.3.3
   DestinationPortRange    8000-8000
   ServiceReference        default
}
```

La configuration type ci–après peut être utilisée pour télécharger des règles à partir d'un serveur LDAP à l'aide du nom de sous–arborescence spécifique,

```
ReadFromDirectory
{
 LDAP_Server      1.2.3.27
 Base             ou=NetworkPolicies,o=myhost.mydomain.com,c=fr
}
```

Identification des problèmes au niveau du QoS

La commande **qosstat** peut être utilisée pour afficher des informations d'état relatives aux politiques actives installées dans le sous–système QoS. Ces informations peuvent vous être utiles afin de détecter la présence d'un problème lors du débogage de la configuration QoS. Utilisez **qosstat** pour produire le rapport suivant.

```
Action:
  Token bucket rate (B/sec): 10240
  Token bucket depth (B): 1024
  Peak rate (B/sec): 10240
  Min policied unit (B): 20
  Max packet size (B): 1452
  Type: IS-CL
  Flags: 0x00001001 (POLICE,SHAPE)

  Statistics:
    Compliant packets: 1423 (440538 bytes)

  Conditions:
    Source address        Dest address       Protocol
    192.168.127.39:8000   192.168.256.29:35049  tcp       (1
connection)

Action:
  Token bucket rate (B/sec): 10240
  Token bucket depth (B): 1024
  Peak rate (B/sec): 10240
  Outgoing TOS (compliant): 0xc0
  Outgoing TOS (non-compliant): 0x00
  Flags: 0x00001011 (POLICE,MARK)
  Type: DS

  Statistics:
    Compliant packets: 335172 (20721355 bytes)
    Non-compliant packets: 5629 (187719 bytes)

  Conditions:
    Source address        Dest address       Protocol
    192.168.127.39:80     *:*                tcp (1 connection)
    192.168.127.40:80     *:*                tcp (5 connections)
```

Spécification de politiques

Cette section décrit les classes et attributs d'objets utilisés par l'agent de politique pour spécifier les politiques de qualité du service (QoS) sur le trafic sortant. Elle définit les classes et attributs d'objets puis donne des instructions relatives au marquage, à la gestion du trafic et à la conception.

Les conventions suivantes sont utilisées dans les explications ci–dessous :

```
p   : Choisissez l'un des paramètres autorisés
B   : Valeur d'entier d'un octet (0 =< B =< 255)
b   : Chaîne de bit commençant par le bit le plus à gauche
      (101 équivaut à 10100000 dans un champ d'octet)
i   : Valeur d'entier
s   : Une chaîne de caractères
a   : Format d'adresse IP B.B.B.B
(R) : Paramètre requis
(O) : Paramètre facultatif
```

ReadFromDirectory

Cette instruction spécifie les paramètres permettant d'établir une session LDAP. Elle est utilisée dans le fichier **/etc/policyd.conf** pour établir la session LDAP.

```
ReadFromDirectory
{
 LDAP_Server    a   # Adresse IP du serveur de répertoires
                      exécutant LDAP
 LDAP_Port      i   # Numéro du port écouté par le serveur LDAP
 Base     s         # Nom spécifique pour l'utilisation LDAP
 LDAP_SelectedTag s # Balise correspondant à SelectorTag dans
                     les classes d'objets
}
```

où

```
LDAP_Server (R) : Adresse IP du serveur LDAP
LDAP_Port   (0) : Numéro de port unique, le port
                  par défaut est 389
Base        (R) : Exemple : o=ibm, c=fr où o est votre
                  organisation et c le pays
LDAP_SelectedTag (R) : Chaîne unique correspondant à l'attribut
                  SelectorTag dans la classe d'objets
```

ServiceCategories

Cette instruction spécifie le type de service qu'un flux de paquets IP (d'une connexion TCP ou de données UDP par exemple) doit recevoir de bout en bout lors de son passage sur le réseau. Les instructions `ServiceCategories` peuvent être répétées, chacune ayant un nom différent pour qu'il soit possible d'y faire référence à un stade ultérieur. Un objet `ServiceCategories` exige une instruction `ServicePolicyRules` pour que la définition de politique soit complète.

```
ServiceCategories   s
{
 SelectorTag    s     # Balise requise pour la recherche LDAP
 MaxRate        i     # Vitesse cible du trafic dans cette classe
                        de service
 MaxTokenBucket i     # La profondeur du compartiment
 OutgoingTOS    b     # Valeur TOS du trafic sortant pour cette
                        classe de service
 FlowServiceType   p  # Type de trafic

}
```

où

```
s           (R)     : est le nom de cette catégorie de service
SelectorTag (R)     : Requis uniquement pour la recherche LDAP
                      dans les classes d'objets
MaxRate     (O)     : En Kbps (K bits par seconde), la valeur par
                      défaut est 0
MaxTokenBucket(O)   : En Kb, la valeur par défaut est définie par
                      le système au maximum
OutgoingTOS (O)     : La valeur par défaut est 0
FlowServiceType  (O) : ControlledLoad | Guaranteed, la valeur
                       par défaut est ControlledLoad
```

ServicePolicyRules

Cette instruction spécifie les caractéristiques des paquets IP pour la correspondance avec une catégorie de service donnée. En d'autres termes, elle définit un jeu de datagrammes IP qui doivent recevoir un service. Les instructions `ServicePolicyRules` sont associées à des instructions `ServiceCategories` via l'attribut `ServiceReference`. Si deux règles font référence à la même `ServiceCategory`, chacune d'entre elles est associée à une instance unique de celle–ci.

```
ServicePolicyRules   s
{
 SelectorTag       s   # Balise requise pour la recherche LDAP
 ProtocolNumber    i   # Id du protocole de transport de la règle
                         de politique
 SourceAddressRange    a1-a2
 DestinationAddressRange a1-a2
 SourcePortRange       i1-i2
 DestinationPortRange  i1-i2
 PolicyRulePriority  i       # La valeur la plus élevée est
                               utilisée en premier
 ServiceReference      s  # Nom de la catégorie de service de
                            cette règle de politique
}
```

où

```
s           (R) : est le nom de cette règle de politique
SelectorTag (R) : Requis uniquement pour la recherche LDAP dans
                  la classe d'objet
ProtocolNumber     (R) : La valeur par défaut est 0
(aucune correspondance n'est trouvée). Spécification explicite requise
SourceAddressRange (O): de a1 à a2 où a2 >= a1, la valeur par
                        défaut est 0, n'importe quelle adresse source
SourcePortRange    (O) : de i1 à i2 où i2 >= i1, la valeur par
défaut est 0, n'importe quel port source
DestinationAddressRange (O) : Voir SourceAddressRange
DestinationPortRange    (O) : Voir SourcePortRange
PolicyRulePriority      (O) : Spécification importante en présence
                              de politiques superposées
ServiceReference   (R) : catégorie de service utilisée
                         par cette règle
```

Instructions relatives aux environnements DiffServ

Les instructions ci–dessous se rapportent à la spécification de politiques pour le marquage, la conception et/ou la gestion du trafic dans un environnement DiffServ.

1. **Marquage uniquement**

```
OutgoingTOS     : Type de service souhaité
FlowServiceType : ControlledLoad
MaxRate         : Utilisez la valeur par défaut 0
```

2. **Conception uniquement**

```
OutgoingTOS     : Utilisez la valeur par défaut 0
FlowServiceType : Guaranteed
MaxRate         : Vitesse cible souhaitée pour le trafic sous
                  la forme d'un entier positif
```

3. **Marquage et gestion (Voir remarque)**

```
OutgoingTOS     : Type de service souhaité
FlowServiceType : ControlledLoad
MaxRate         : Vitesse cible souhaitée pour le trafic sous
                  la forme d'un entier positif
```

4. **Marquage et conception**

```
OutgoingTOS     : Type de service souhaité
FlowServiceType : Guaranteed
MaxRate         : Vitesse cible souhaitée pour le trafic sous
                  la forme d'un entier positif
```

Remarque: Le type de service des paquets ne correspondant pas au profil est défini sur zéro pour la gestion.

Fichier de configuration policyd exemple

L'exemple ci–dessous reprend un fichier de configuration **/etc/policyd.conf** complet.

Fichier de configuration policyd

```
#loglevel  511    # Verbose logging

#########################################################################
#
#
# Mark rsh traffic on TCP source ports 513 and 514.
ServiceCategories       tcp_513_514_svc
{
        MaxRate                 0               # Mark only
        OutgoingTOS             00011100        # binary
        FlowServiceType         ControlledLoad
}

ServicePolicyRules      tcp_513_514_flt
{
        ProtocolNumber          6  # TCP
        SourceAddressRange      0.0.0.0-0.0.0.0 # Any IP src addr
        DestinationAddressRange 0.0.0.0-0.0.0.0 # Any IP dst addr
        SourcePortRange         513-514
        DestinationPortRange    0-0             # Any dst port
        ServiceReference        tcp_513_514_svc
}
#
#########################################################################
```

```
#
#
# Shape connected UDP traffic on source port 9000.
ServiceCategories       udp_9000_svc
{
        MaxRate                 8192     # kilobits
        MaxTokenBucket          64       # kilobits
        FlowServiceType         Guaranteed
}

ServicePolicyRules      udp_9000_flt
{
        ProtocolNumber          17  # UDP
        SourceAddressRange      0.0.0.0-0.0.0.0 # Any IP src addr
        DestinationAddressRange 0.0.0.0-0.0.0.0 # Any IP dst addr
        SourcePortRange         9000-9000
        DestinationPortRange    0-0             # Any dst port
        ServiceReference        udp_9000_svc
}
#
#########################################################################
#
#
# Mark and police finger traffic on TCP source port 79.
ServiceCategories       tcp_79_svc
{
        MaxRate                 8         # kilobits
        MaxTokenBucket          32        # kilobits
        OutgoingTOS             00011100  # binary
        FlowServiceType         ControlledLoad
}

ServicePolicyRules      tcp_79_flt
{
        ProtocolNumber          6  # TCP
        SourceAddressRange      0.0.0.0-0.0.0.0 # Any IP src addr
        DestinationAddressRange 0.0.0.0-0.0.0.0 # Any IP dst addr
        SourcePortRange         79-79
        DestinationPortRange    0-0             # Any dst port
        ServiceReference        tcp_79_svc
}
#
#########################################################################
#
#
# Mark and shape ftp-data traffic on TCP source port 20.
ServiceCategories       tcp_20_svc
{
        MaxRate                 81920        # kilobits
        MaxTokenBucket          128          # kilobits
        OutgoingTOS             00011101     # binary
        FlowServiceType         Guaranteed
}

ServicePolicyRules      tcp_20_flt
{
        ProtocolNumber          6  # TCP
        SourceAddressRange      0.0.0.0-0.0.0.0 # Any IP src addr
        DestinationAddressRange 0.0.0.0-0.0.0.0 # Any IP dst addr
        SourcePortRange         20-20
        DestinationPortRange    0-0             # Any dst port
        ServiceReference        tcp_20_svc
}
#
```

```
#########################################################################
#
#
# LDAP server entry.
#ReadFromDirectory
#{
#   LDAP_Server             9.3.33.138  # IP address of LDAP server
#   Base                    o=ibm,c=us  # Base distinguished name
#   LDAP_SelectedTag        myhost      # Typically client hostname
#}
#
#########################################################################
#
```

Chargement de politiques dans le serveur de répertoires SecureWay Directory

Si le démon de politique est utilisé avec le serveur de répertoires LDAP SecureWay Directory, utilisez le schéma ci–dessous comme guide pour mettre à jour **/etc/ldapschema/V3.modifiedschema** avant de démarrer le serveur LDAP. Pour plus d'informations, reportez–vous à Planification et configuration pour la résolution de noms LDAP (Schéma de répertoire SecureWay) page 4-89

Schéma LDAP

```
objectClasses {
( ServiceCategories-OID NAME 'ServiceCategories' SUP top MUST
( objectClass $ SelectorTag $ serviceName ) MAY
( description $ FlowServiceType $ MaxRate $ MaxTokenBucket $ OutgoingTos
) )
( ServicePolicyRules-OID NAME 'ServicePolicyRules' SUP top MUST
( objectClass $ PolicyName $ SelectorTag ) MAY
( description $ DestinationAddressRange $ DestinationPortRange $
ProtocolNumber $ ServiceReference $ SourceAddressRange $ SourcePortRange
) )
}
attributeTypes {
( DestinationAddressRange-OID NAME 'DestinationAddressRange' SYNTAX
1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
( DestinationPortRange-OID NAME 'DestinationPortRange' SYNTAX
 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
( FlowServiceType-OID NAME 'FlowServiceType'
 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
( MaxRate-OID NAME 'MaxRate' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )
( MaxTokenBucket-OID NAME 'MaxTokenBucket' SYNTAX
1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
( OutgoingTos-OID NAME 'OutgoingTos' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )
( PolicyName-OID NAME 'PolicyName' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )
( ProtocolNumber-OID NAME 'ProtocolNumber' SYNTAX
1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
( SelectorTag-OID NAME 'SelectorTag' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )
( ServiceReference-OID NAME 'ServiceReference' SYNTAX
 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
( SourceAddressRange-OID NAME 'SourceAddressRange' SYNTAX
 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
( SourcePortRange-OID NAME 'SourcePortRange' SYNTAX
1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
}

IBMattributeTypes {
( DestinationAddressRange-OID DBNAME ( 'DestinationAddressRange'
'DestinationAddressRange' ) )
( DestinationPortRange-OID DBNAME ( 'DestinationPortRange'
'DestinationPortRange' ) )
( FlowServiceType-OID DBNAME ( 'FlowServiceType' 'FlowServiceType' ) )
( MaxRate-OID DBNAME ( 'MaxRate' 'MaxRate' ) )
( MaxTokenBucket-OID DBNAME ( 'MaxTokenBucket' 'MaxTokenBucket' ) )
( OutgoingTos-OID DBNAME ( 'OutgoingTos' 'OutgoingTos' ) )
( PolicyName-OID DBNAME ( 'PolicyName' 'PolicyName' ) )
( ProtocolNumber-OID DBNAME ( 'ProtocolNumber' 'ProtocolNumber' ) )
( SelectorTag-OID DBNAME ( 'SelectorTag' 'SelectorTag' ) )
( ServiceReference-OID DBNAME ( 'ServiceReference' 'ServiceReference' ) )
( SourceAddressRange-OID DBNAME ( 'SourceAddressRange'
'SourceAddressRange' ) )
( SourcePortRange-OID DBNAME ( 'SourcePortRange' 'SourcePortRange' ) )
}

ldapSyntaxes {
}

matchingRules {
}
```

Configuration du système

Politiques superposées

Les politiques qui se superposent sont installées dans QoS Manager dans un ordre non déterminant. En présence de politiques superposées, l'attribut `PolicyRulePriority` de l'instruction `ServicePolicyRules` doit être spécifié pour déterminer l'ordre d'application des politiques. Cet attribut prend un entier comme paramètre. En présence de politiques superposées, la règle dont la valeur d'entier est la plus élevée est mise en application.

Utilisation de sockets UDP

Seules les sockets UDP connectées sont prises en charge pour QoS.

Conflits de politiques avec des réservations RSVP

Les agents de politiques et les agents RSVP sont indépendants. Il convient donc de prendre garde de ne pas spécifier une politique en conflit avec une réservation RSVP existante ou couverte par celle–ci. En présence de conflits, le système accepte la première politique ou réservation et enregistre une violation pour les autres.

Spécification de la profondeur du compartiment à jeton

Pour un fonctionnement correct, l'attribut `MaxTokenBucket` doit être défini au moins sur le MTU maximum de toutes les interfaces configurées dans le système.

Modification de politiques

Les modifications de politiques sont gérées par l'agent de politique via la suppression automatique des politiques existantes et l'installation de nouvelles politiques. Elles peuvent entraîner une courte période durant laquelle le trafic correspondant reçoit le service par défaut (en général l'effort maximal).

Conformité aux normes

Cette version est compatible avec les normes IETF actuelles pour les services différenciés (DiffServ) et les services intégrés (IntServ) sur Internet.

Modèle IntServ

Les normes RFC suivantes décrivent les divers composants du modèle IntServ :

- Utilisation du RSVP avec les services intégrés IETF (RFC 2210)
- Spécification du service d'éléments réseau avec contrôle de charge (RFC 2211)
- Spécification de la qualité de service garantie (RFC 2212)

Modèle DiffServ

Les normes RFC suivantes décrivent les divers composants du modèle DiffServ :

- Définition du champ Services différenciés (champ DS) dans les en–têtes IP versions 4 et 6 (RFC 2474)
- Architecture des services différenciés (RFC 2475)

La norme RFC suivante expose l'utilisation actuelle de l'octet TOS IP :

- Type de service dans la suite Internet Protocol (RFC 1349)

Les normes RFC suivantes exposent les pratiques futures relatives à l'utilisation de l'octet TOS IP :

- Définition du champ Services différenciés (champ DS) dans les en–têtes IP versions 4 et 6 (RFC 2474)
- Groupe PHB d'acheminement assuré (RFC 2597)
- Groupe PHB d'acheminement expéditif (RFC 2598)

Prise en charge de IPv6

QoS for AIX 5.2 prend en charge uniquement IPv4. IPv6 n'est pas pris en charge.

Contrôle du démon de politique

Le démon de politique peut être contrôlé à l'aide de SRC (contrôleur des ressources système). Par exemple, la commande :

```
startsrc -s policyd -a "-i 60"
```

lance l'agent de politique avec un intervalle d'actualisation de 60 secondes.

La commande

```
stopsrc -s policyd
```

arrête le démon de politique.

Remarque : L'arrêt du démon de politique be supprime pas les politiques installées dans le noyau. Lorsque vous redémarrez le démon de politique, les anciennes politiques (déjà installées dans le noyau) sont supprimées et les politiques définies dans le fichier **/etc/policyd.conf** sont réinstallées.

Référence QoS

Pour des mises à jour importantes de cette documentation, consultez le fichier README dans **/usr/samples/tcpip/qos**.

Commandes

- **qosadd**
- **qoslist**
- **qosmod**
- **qosremove**
- **qosstat**
- **mkqos**
- **rmqos**

Méthodes

- **cfgqos**
- **ucfgqos**

Identification des incidents TCP/IP

Cette section traite du diagnostic des incidents courants en environnement TCP/IP (Transmission Control Protocol/Internet Protocol).

La commande **netstat** est très utile pour localiser un incident. Une fois la zone en cause isolée, vous disposez d'outils plus précis : commandes **netstat –i** et **netstat –v** pour déterminer la présence d'un problème au niveau d'une interface matérielle, puis programmes de diagnostic pour mieux cerner les causes de l'incident. Ou bien, si la commande **netstat -s** a détecté des erreurs de protocole, vous pouvez utiliser la commande **trpt** ou **iptrace.**

Cette section traite des points suivants :

Incidents de communication

Si vous ne parvenez pas à communiquer avec un hôte de votre réseau :

- Essayez de contacter l'hôte à l'aide de la commande **ping**. Lancez la commande **ping** sur l'hôte local pour vérifier que l'interface locale reliée au réseau est opérationnelle et active.
- Tentez de résoudre le nom de l'hôte avec la commande **host**. Si vous n'y parvenez pas, vous avez un problème de résolution de noms. Pour en savoir plus, reportez–vous à Problèmes de résolution de noms.

Si le nom est résolu et que l'hôte à contacter se trouve sur un autre réseau, il s'agit peut–être de difficultés de routage. Pour en savoir plus, reportez–vous à Problèmes de routage.

- Sur un réseau en anneau à jeton, vérifiez si l'hôte cible réside sur un autre anneau. Dans l'affirmative, il est probable que le champ `allcast` soit mal renseigné. Pour accéder au menu des interfaces de réseau, vous pouvez utiliser **wsm**, Web-based System Manager ou le raccourci SMIT **smit chinet**. Spécifiez ensuite **no** dans le champ Confine Broadcast to Local Ring to, de la boîte de dialogue pour la définition de l'anneau à jeton.
- Si un grand nombre de paquets ARP transitent sur le réseau, vérifiez que votre masque du sous–réseau est correctement défini. Faute de quoi, vous vous trouvez en présence d'un conflit de diffusion pouvant affecter les performances de votre système.

Incidents de résolution de noms

Les routines de résolution exécutées sur des hôtes TCP/IP tentent de résoudre les noms en faisant appel aux sources suivantes et dans l'ordre suivant :

1. au serveur de noms DOMAIN (**named**),
2. NIS (Network Information Services),
3. au fichier **/etc/hosts** local.

Lors de l'installation de NIS+, les préférences de recherche sont définies dans le fichier **irs.conf**. Pour plus d'informations, reportez–vous au *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide*.

Hôte client

En cas d'échec de résolution d'un nom d'hôte avec le fichier **/etc/hosts** (réseau plat), vérifiez que ce fichier contient le nom d'hôte et l'adresse IP correcte.

En cas d'échec de résolution d'un nom d'hôte avec un serveur de noms :

1. Vérifiez que le fichier **resolv.conf** contient le nom du domaine et l'adresse Internet d'un serveur de noms.
2. Vérifiez que le serveur de noms local est opérationnel en émettant la commande **ping** avec l'adresse IP du serveur (relevée dans le fichier **resolv.conf** local).
3. Si le serveur de noms est opérationnel, vérifiez que le démon **named** sur votre serveur de noms local est actif en émettant la commande **lssrc –s named** sur le serveur de noms.
4. Si vous exécutez **syslogd**, recherchez les éventuels messages d'erreur journalisés. (la sortie des messages est définie dans le fichier **/etc/syslog.conf**).

Si ces opérations ne permettent pas d'identifier l'incident, examinez l'hôte serveur de noms.

Hôte serveur de noms

En cas d'échec de résolution d'un nom d'hôte :

1. Vérifiez que le démon **named** est actif :

   ```
   lssrc –s named
   ```

2. Vérifiez que l'adresse de l'hôte cible existe dans la base de données du serveur de noms et qu'elle est correcte. Envoyez un signal SIGINT au démon **named** pour placer un cliché de la base de données et de la mémoire cache dans le fichier **/var/tmp/named_dump.db**. Vérifiez que l'adresse que vous tentez de résoudre s'y trouve et est correcte.

   Ajoutez ou corrigez les informations de résolution nom-adresse dans le fichier de données hôte **named** du serveur de noms maître du domaine. Puis, exécutez la commande SRC ci-dessous pour relire les fichiers de données :

   ```
   refresh –s named
   ```

3. Vérifiez que les demandes de résolution de noms ont été traitées. Pour ce faire, lancez le démon **named** à partir de la ligne de commande et spécifiez le niveau de mise au point (de 1 à 9) sachant que plus le niveau est élevé, plus le mécanisme de mise au point consigne d'informations.

   ```
   startsrc –s named –a "–d DebugLevel"
   ```

4. Recherchez d'éventuelles erreurs de configuration dans les fichiers de données **named**. Pour en savoir plus, reportez–vous à Serveur de noms – Généralités on page 4-73. Vous pouvez aussi consulter "DOMAIN Data File Format,", "DOMAIN Reverse Data File Format," "DOMAIN Cache File Format," et "DOMAIN Local Data File Format" dans le manuel *AIX 5L Version 5.2 Files Reference*.

   Remarque : le plus souvent, les erreurs proviennent d'une mauvaise utilisation du point (.) et de l'arrobas (@) dans les fichiers de données DOMAIN.

Si des utilisateurs externes ne peuvent accéder à vos domaines :

- Vérifiez que tous vos serveurs de noms non maîtres (esclave, cache) sont définis avec les mêmes délais TTL dans les fichiers de données DOMAIN.

Si vos serveurs sont continuellement sollicités par des routines de résolution externes :

- Assurez–vous que vos serveurs diffusent des fichiers de données DOMAIN avec des délais TTL suffisants. Si la valeur TTL est nulle ou négligeable, le délai accordé aux données transférées s'écoule très rapidement. Pour y remédier, prévoyez au moins une semaine comme valeur minimum dans vos enregistrements SOA.

Incidents de routage

Si vous ne parvenez pas à accéder à un hôte de destination, contrôlez les points suivants :

- Si vous recevez le message `Network Unreachable`, vérifiez la route vers l'hôte passerelle. Lancez la commande **netstat -r** pour afficher la liste des tables de routage noyau.
- Si vous recevez le message `No route to host` (hôte sans route), vérifiez que l'interface de réseau local est opérationnelle en lançant la commande **ifconfig** nom_interface. Le résultat doit indiquer "up". Lancez la commande **ping** pour tenter d'atteindre un autre hôte du réseau.
- Si vous recevez le message `Connection timed out` :
  - Vérifiez que la passerelle locale est opérationnelle à l'aide de la commande **ping** assortie du nom ou de l'adresse Internet de la passerelle.
  - Vérifiez qu'une route vers l'hôte passerelle a été correctement définie. Lancez la commande **netstat -r** pour afficher la liste des tables de routage noyau.
  - Vérifiez que l'hôte qui vous intéresse dispose d'une entrée de table de routage renvoyant à votre machine.
- Si vous utilisez le routage statique, assurez–vous qu'une route vers l'hôte cible et l'hôte passerelle a été définie. Lancez la commande **netstat -r** pour afficher la liste des tables de routage noyau.

Remarque : L'hôte qui vous intéresse doit disposer d'une entrée de table de routage renvoyant à votre machine.

- En routage dynamique, vérifiez, à l'aide de la commande **netstat –r,** que la passerelle est répertoriée dans les tables de routage noyau et qu'elle est correcte.
- Si l'hôte passerelle utilise le protocole RIP avec le démon **routed**, vérifiez qu'une route statique d'accès à l'hôte cible est définie dans le fichier **/etc/gateways.**

Remarque : Cette opération n'est requise que si le démon de routage ne parvient pas à identifier la route vers l'hôte distant en interrogeant les autres passerelles.

- Si l'hôte passerelle utilise RIP avec le démon **gated**, vérifiez qu'une route statique d'accès à l'hôte cible est définie dans le fichier **gated.conf**.
- En routage dynamique avec le démon **routed** :
  - Si **routed** ne parvient pas à identifier la route par le biais de demandes (par exemple, si l'hôte cible n'exécute pas le protocole RIP), vérifiez qu'une route d'accès à l'hôte cible est définie dans le fichier **/etc/gateways**.
  - Vérifiez que les passerelles chargées d'expédier les paquets à l'hôte sont opérationnelles et exécutent RIP. Sinon, vous devez définir une route statique.
  - Exécutez le démon **routed** avec l'option de mise au point pour journaliser les anomalies (réception de paquets erronés par exemple). Appelez le démon à partir de la ligne de commande, comme suit :

```
startsrc –s routed –a "–d"
```

  - Exécutez le démon **routed** avec l'indicateur **–t** pour envoyer tous les paquets entrants et sortants vers la sortie standard. Exécuté dans ce mode, **routed** reste sous le contrôle du terminal qui l'a lancé. Et il peut être arrêté depuis ce terminal.

- En routage dynamique avec le démon **gated** :
  - Vérifiez que le fichier **/etc/gated.conf** est correctement configuré et que vous exécutez les protocoles adéquats.
  - Vérifiez que les passerelles sur les réseaux source et cible utilisent le même protocole.
  - Vérifiez que la machine que vous tentez de contacter dispose d'une route retour vers votre machine hôte.
  - Vérifiez que les noms de passerelle des fichiers **gated.conf** et **/etc/networks** correspondent.
- Si vous utilisez le protocole RIP ou HELLO et que les routes d'accès ne peuvent pas être identifiées par des demandes de routage, vérifiez qu'une route d'accès à l'hôte cible est définie dans le fichier **gated.conf**. Il est conseillé de définir des routes statiques si :
  - L'hôte de destination n'exécute pas le même protocole que l'hôte source et ne peut donc pas échanger d'informations de routage.
  - L'accès à l'hôte doit se faire par une passerelle distante (c'est-à-dire sur un autre système autonome que l'hôte source). Le protocole RIP peut être utilisé uniquement entre des hôtes d'un même système autonome.

Autres possibilités

Si aucune des solutions proposées n'aboutit, vous pouvez activer le suivi du démon de routage (**routed** ou **gated**). Exécutez la commande SRC **traceson** à partir de la ligne de commande ou envoyez un signal au démon pour spécifier différents niveaux de suivi. Pour en savoir plus, reportez-vous au démon **gated** ou **routed.**

Incidents SRC

- Si les modifications apportées au fichier **/etc/inetd.conf** ne sont pas prises en compte :

  Mettez à jour le démon **inetd** via la commande **refresh –s inetd** ou **kill –1** *InetdPID*.

- Si **startsrc –s** [*sous_système*] renvoie le message :

  ```
  0513-00  The System Resource Controller is not active.
  ```

  Le sous-système SRC (System Resource Controller) n'a pas été activé. Lancez la commande **srcmstr &** pour lancer SRC, puis à nouveau la commande **startsrc**.

  Vous pouvez tenter de lancer le démon à partir de la ligne de commande sans SCR.

- Si **refresh –s** [*sous_système*] ou **lssrc –ls** [*sous_système*] renvoie le message :

  ```
  [nom sous système] does not support this option.
  ```

  Le sous-système ne prend pas en charge l'option SRC émise. Consultez la documentation relative au sous-système.

- Si le message ci-dessous s'affiche :

  ```
  SRC was not found, continuing without SRC support.
  ```

  Un démon a été appelé directement à partir de la ligne de commande et non via la commande **startsrc**. Ceci ne constitue pas un incident. Toutefois les commandes SRC (telles que **stopsrc** et **refresh)** ne peuvent pas être utilisées pour manipuler un sous-système appelé directement.

Le démon **inetd** est installé, s'exécute correctement et le service approprié ne présente pas de problème, mais la connexion est impossible ; analysez le démon **inetd** à l'aide d'un programme de mise au point.

1. Arrêtez temporairement le démon **inetd** en tapant :

   ```
   stopsrc –s inetd
   ```

   La commande **stopsrc** arrête les sous–systèmes tel que le démon **inetd**.

2. Modifiez le fichier **syslog.conf** pour ajouter une ligne relative à la mise au point à la fin. Par exemple :

   ```
   vi /etc/syslog.conf
   ```

   a. Ajoutez la ligne "`*.debug /tmp/monfichier`" à la fin du fichier, puis quittez.

   b. Le fichier spécifié doit être un fichier existant (`/tmp/monfichier` dans cet exemple). Pour valider l'existence du fichier, vous pouvez utiliser la commande **touch**.

3. Rafraîchissez le fichier :

   – Si vous utilisez SRC, entrez :

   ```
   refresh -s syslogd
   ```

   – Si vous n'utilisez pas SRC, arrêtez le démon **syslogd** :

   ```
   kill -1 `ps -e | grep /etc/syslogd | cut -c1-7`
   ```

4. Lancez la sauvegarde du démon **inetd** en y associant l'activation de la mise au point :

   ```
   startsrc -s inetd -a "-d"
   ```

   L'indicateur **–d** active la mise au point.

5. Exécutez une connexion pour détecter les erreurs dans le fichier de mise au point `/tmp/monfichier`. Par exemple :

   ```
   tn bastet
    Trying...
    connected to bastet
    login:>
    Connection closed
   ```

6. Examinez le fichier de mise au point à la recherche d'éventuels problèmes. Par exemple :

   ```
   tail -f /tmp/monfichier
   ```

Incidents liés à telnet ou rlogin

Voici quelques indications sur les incidents liés aux commandes **telnet** et **rlogin**.

Distorsion de l'écran

Si vous rencontrez des problèmes de distorsion d'écran dans des applications plein écran :

1. Vérifiez la variable d'environnement **TERM**, via la commande :

   ```
   env
   ```

   OU

   ```
   echo $TERM
   ```

2. Vérifiez que la valeur de **TERM** concorde avec le type d'écran de terminal utilisé.

Mise au point par telnet

Les sous-commandes **telnet** qui peuvent vous aider à résoudre des incidents sont :

display	Affiche les valeurs définies et les valeurs de commutation.
toggle	Affiche toutes les données réseau en hexadécimal.
toggle options	Change l'affichage des options internes du process **telnet.**

Mise au point du démon telnetd

Si le démon **inetd** exécute le service **telnet**, alors que vous n'êtes toujours pas en mesure de vous connecter à l'aide de la commande **telnet**, cela signifie qu'il y a un problème au niveau de l'interface **telnet**.

1. Vérifiez que **telnet** utilise le type de terminal correct.

   a. Vérifiez la variable **$TERM** sur votre machine :

   ```
   echo $TERM
   ```

   b. Connectez–vous à la machine que vous souhaitez relier et vérifiez la variable **$TERM** :

   ```
   echo $TERM
   ```

2. Utilisez les fonctions de mise au point de l'interface **telnet** en utilisant la commande **telnet** sans indicateur.

   ```
   telnet
    tn>
   ```

   a. Entrez `open hôte` (*hôte* est le nom de la machine).

   b. Appuyez sur Ctrl–T pour rappeler l'invite `tn%gt;`.

   c. A l'invite `tn>`, entrez `debug` pour accéder au mode mise au point.

3. Essayez de vous connecter à une autre machine en utilisant l'interface **telnet** :

   ```
   telnet bastet
   Trying...
    Connected to bastet
    Escape character is '^T'.
   ```

   Observez le défilement des différentes commandes à l'écran. Par exemple :

   ```
    SENT do ECHO
    SENT do SUPPRESS GO AHEAD
    SENT will TERMINAL TYPE (reply)
    SENT do SUPPORT SAK
    SENT will SUPPORT SAK (reply)
    RCVD do TERMINAL TYPE (don't reply)
    RCVD will ECHO (don't reply)
    RCVD will SUPPRESS GO AHEAD (don't reply)
    RCVD wont SUPPORT SAK (reply)
    SENT dont SUPPORT SAK (reply)
    RCVD do SUPPORT SAK (don't reply)
    SENT suboption TELOPT_NAWS Width 80, Height 25
    RCVD suboption TELOPT_TTYPE SEND
    RCVD suboption TELOPT_TTYPE aixterm
    ...
   ```

4. Vérifiez la définition de `aixterm` dans les répertoires **/etc/termcap** ou **/usr/lib/terminfo**. Par exemple :

   ```
   ls –a /usr/lib/terminfo
   ```

5. Si la définition de `aixterm` est manquante, ajoutez–la en créant le fichier **ibm.ti**. Par exemple :

   ```
   tic ibm.ti
   ```

   La commande **tic** est un compilateur d'informations de terminal.

Programmes utilisant la bibliothèque curses étendue

Certains problèmes peuvent apparaître au niveau des touches de fonction et des touches fléchées si vous utilisez les commandes **rlogin** et **telnet** avec des programmes faisant appel à la bibliothèque curses étendue. En effet, ces touches génèrent des séquences d'échappement, qui peuvent être dissociées si le temps imparti ne suffit pas à la séquence complète. Après un certain délai, la bibliothèque curses décide si Echap doit être interprété seul ou comme le début d'une séquence d'échappement multi-octets générée par d'autres touches (touches fléchées, touches de fonction ou touche Action).

Si, dans le temps imparti, la touche Echap n'est suivie d'aucune donnée valide, curses l'interprète comme la touche Echap seule et fractionne la séquence de touches. Le délai associé aux commandes **rlogin** ou **telnet** dépend du réseau. C'est en fonction de sa vitesse que les touches de fonction et les touches fléchées fonctionnent normalement ou non. Pour résoudre efficacement le problème, attribuez une valeur élevée (entre 1000 et 1500) à la variable d'environnement **ESCDELAY**.

Incidents de configuration

Une fois la carte installée, les interfaces de réseau sont automatiquement configurées au premier lancement du système. Il reste toutefois certaines valeurs initiales à définir pour TCP/IP, telles que le nom de l'hôte, l'adresse Internet et le masque de sous–réseau. Pour cela, vous pouvez utiliser **wsm**, Web-based System Manager ou l'interface SMIT comme suit :

- Servez-vous du raccourci **smit mktcpip** pour définir les valeurs initiales pour le nom d'hôte, l'adresse Internet et le masque de sous–réseau.
- Cette commande **smit mktcpip** permet également de spécifier un serveur pour la résolution de noms. Cependant, **smit mktcpip** ne configure qu'une seule interface de réseau.
- Pour définir d'autres attributs de réseau, utilisez le raccourci **smit chinet**.

Si vous souhaitez mettre en place des routes statiques pour que l'hôte puisse acheminer des informations de transmission, par exemple une route d'accès à la passerelle locale, définissez–les de façon permanente dans la base de configuration, avec Web-based System Manager, **wsm**, ou le raccourci SMIT **smit mkroute**.

Si vous rencontrez d'autres difficultés, reportez–vous à Configuration d'une liste de contrôle du réseau TCP/IP.

Incidents courants sur les interfaces de réseau

Une fois la carte installée, les interfaces de réseau sont automatiquement configurées au premier lancement du système. Il reste toutefois certaines valeurs initiales à définir pour TCP/IP. Par exemple, il est possible de définir le nom d'hôte et l'adresse Internet à l'aide de **wsm** de Web-based System Manager ou du raccourci **smit mktcpip** de SMIT.

Si vous passez par SMIT, ayez recours au raccourci **smit mktcpip** pour définir ces valeurs de façon permanente dans la base de configuration. Pour les modifier dans le système actif, utilisez les raccourcis **smit chinet** et **smit hostname**. Le raccourci **smit mktcpip** permet une configuration minimale de TCP/IP. Pour ajouter des cartes, passez par le menu Further Configuration, accessible via le raccourci **smit tcpip**.

Si, malgré la validité des valeurs définies, vous avez toujours des difficultés à recevoir et envoyer des données :

- Vérifiez que votre carte réseau dispose d'une interface de réseau en exécutant la commande **netstat –i**. La sortie doit mentionner une interface, par exemple **tr0**, dans la colonne `Name`. Dans le cas contraire, créez une interface de réseau via Web-based System Manager ou via le raccourci **smit mkinet** de SMIT.
- Vérifiez que l'adresse IP de l'interface est correcte, en exécutant **netstat –i**. La sortie doit afficher l'adresse IP dans la colonne Network. Si l'adresse est incorrecte, modifiez-la via Web-based System Manager ou via le raccourci SMIT **smit chinet**.

- Utilisez la commande **arp** pour vérifier que l'adresse IP de la machine cible est complète. Par exemple :

  ```
  arp –a
  ```

  La commande **arp** recherche l'adresse physique de la carte. Cette commande risque de renvoyer une adresse incomplète. Par exemple :

  ```
  ? (192.100.61.210) à (incomplète)
  ```

  Les raisons peuvent être les suivantes : une machine débranchée, une adresse isolée sans machine à l'adresse spécifique ou un problème matériel (par ex. une machine en mesure de se connecter et de recevoir des paquets mais qui ne peut pas les renvoyer).

- Recherchez les erreurs au niveau de la carte. Par exemple :

  ```
  netstat –v
  ```

  La commande **netstat –v** affiche les données statistiques au niveau des pilotes des périphériques Ethernet, Token Ring, X.25 et 802.3. Elle révèle également les données relatives aux connexions au réseau et aux erreurs de connexion pour tous les pilotes de périphériques actifs sur une interface, y compris :
  `No Mbufs Errors`,`No Mbuf Extension Errors`,
  `Packets Transmitted et Adapter  Errors Detected`.

- Consultez le journal des erreurs, en lançant la commande **errpt,** pour vérifier qu'aucun incident de carte n'a été détecté.

- Vérifiez que la carte est fiable en exécutant les programmes de diagnostics. Pour cela, utilisez l'application Devices de Web-based System Manager, le raccourci **smit diag** ou la commande **diag**.

Si ces étapes ne permettent pas d'identifier le problème, reportez–vous à Problèmes d'interface réseau SLIP page 4-232,

Problèmes d'interface réseau Ethernet page 4-233, ou Problème d'interface réseau en anneau jeton.

Incidents sur une interface de réseau SLIP

En général, la méthode la plus efficace pour résoudre ce type d'incident consiste à vérifier pas à pas la configuration de votre système. Vous pouvez également :

- Vérifiez que le process **slattach** s'exécute sur le port tty approprié, via la commande **ps –ef**. Si ce n'est pas le cas, lancez la commande **slattach**. Pour la syntaxe à utiliser, reportez–vous à Configuration de SLIP pour modem, ou à Configuration de SLIP pour câble null– modem.

- Vérifiez les adresses point-à-point spécifiées via la commande **smit chinet**.

  Sélectionnez l'interface SLIP. Vérifiez l'adresse Internet et l'adresse de destination.

Si le modem ne fonctionne pas correctement :

- Vérifiez son installation. Reportez-vous au manuel opérateur du modem.

- Vérifiez que les contrôles de flux que le modem peut effectuer sont désactivés.

Si le tty ne fonctionne pas correctement, vérifiez le débit (en bauds) correspondant ainsi que les caractéristiques du modem, dans la base de données de configuration, via la commande **smit tty**.

Incidents sur l'interface de réseau Ethernet

Si une interface réseau est initialisée, les adresses définies et la carte installée correcte :

- Vérifiez qu'un connecteur en T est directement branché sur l'émetteur-récepteur (intégré ou non).
- Assurez–vous que vous utilisez un câble Ethernet. Le câble Ethernet est de 50 OHM.
- Assurez–vous que vous utilisez des terminaisons Ethernet. Les terminaisons Ethernet sont de 50 OHM.
- Les cartes Ethernet peuvent fonctionner avec un émetteur-récepteur interne ou externe. Un cavalier, installé sur la carte, définit le type d'émetteur-récepteur utilisé. Vérifiez la position de ce cavalier (reportez-vous à la documentation de la carte).
- Vérifiez le type de connecteur utilisé (BNC pour câble fin et DIX pour câble épais). Si vous changez ce type de connecteur, vous devez définir le champ Apply Change to Database Only (appliquer les modifications uniquement à la base de données) ; pour ce faire, utilisez **wsm** de Web-based System Manager ou le raccourci **smit chgenet** de SMIT. Ce champ doit être coché dans Web-based System Manager ou défini à **yes** dans SMIT. Réamorcez ensuite la machine pour appliquer la nouvelle configuration. (Reportez-vous à "Configuration et gestion des cartes", page 4-37.)

Incidents liés à une interface de réseau en anneau à jeton

Si vous ne parvenez pas communiquer avec certaines machines, alors que l'interface de réseau est initialisée, les adresses convenablement définies et la carte installée correcte :

- Vérifiez si les machines en cause se trouvent sur un autre anneau. Si tel est le cas, utilisez **wsm** de Web-based System Manager ou le raccourci **smit chinet** de SMIT pour cocher le champ Confine BROADCAST to Local Token–Ring (limiter la diffusion à l'anneau à jeton local). Ce champ *ne doit pas* être coché dans Web-based System Manager ou défini à **no** dans SMIT.
- Vérifiez que la carte de réseau en anneau à jeton est configurée pour fonctionner à la bonne vitesse d'anneau. S'il est configuré incorrectement, utilisez l'application réseau Web-based System Manager ou SMIT pour modifier l'attribut de la vitesse de l'anneau à jeton (reportez–vous à Configuration et gestion des cartes). Une fois TCP/IP redémarré, la vitesse d'anneau de la carte de réseau en anneau à jeton sera identique à celle du réseau.

Incidents avec un pont anneau à jeton/Ethernet

Si la communication entre un réseau en anneau à jeton et un réseau Ethernet reliés par un pont est défaillante alors que le pont fonctionne normalement, il est probable que la carte Ethernet rejette des paquets. Ce rejet a lieu lorsque le nombre de paquets entrants (en-têtes compris) est supérieur à la valeur MTU (Maximum Transmission Unit) de la carte. Par exemple, un paquet de 1500 octets envoyé par une carte en anneau à jeton via un pont, totalise 1508 octets, avec un en-tête LLC de 8 octets. Si la valeur MTU de la carte Ethernet est fixée à 1500, le paquet est rejeté.

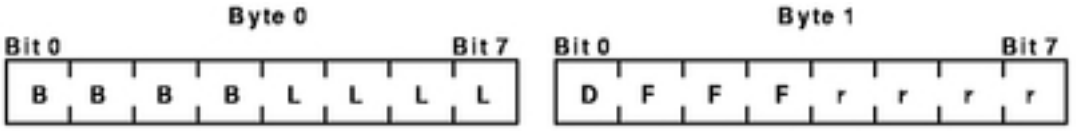
Vérifiez les valeurs MTU (Maximum Transmission Unit) des deux cartes de réseau. Pour autoriser l'adjonction, par la carte en anneau à jeton, d'en-têtes LLC de 8 octets aux paquets sortants, la valeur MTU de cette carte doit être inférieure d'au moins 8 octets à celle de la carte Ethernet. Par exemple, pour qu'une carte en anneau à jeton puisse communiquer avec une carte Ethernet avec une MTU de 1500 octets, sa MTU doit être fixée à 1492.

Incidents sur un pont reliant deux réseaux en anneau à jeton

Lorsque la communication transite par un pont, la valeur MTU par défaut (de 1500 octets) doit être ramenée à 8 octets en-dessous de la valeur maximum I-frame déclarée par le pont dans le champ de contrôle de routage.

Pour retrouver la valeur du contrôle de routage, exécutez le démon **iptrace** qui permet d'examiner les paquets entrants. Les bits 1, 2 et 3 de l'octet 1 constituent les bits de trames maximales (Largest Frame Bit). Ils déterminent le maximum d'informations transmissibles entre deux stations de communication sur une route spécifique. Pour le format du champ de contrôle de routage, consultez la figure ci-dessous :

Figure 26. Champ de contrôle de routage Cette illustration représente l'octet 0 et l'octet 1 d'un champ de contrôle de routage. Les 8 bits de l'octet zéro sont B, B, B, B, L, L, L, L. Les 8 bits de l'octet 1 sont T D, F, F, F, r, r, r, r.



Les valeurs possibles des bits de trames maximales sont :

000 516 octets maximum dans le champ d'information.

001 1500 octets maximum dans le champ d'information.

010 2052 octets maximum dans le champ d'information.

011 4472 octets maximum dans le champ d'information.

100 8144 octets maximum dans le champ d'information.

101 Réservé.

110 Réservé.

111 Utilisé dans les trames de diffusion générale (toute route).

Par exemple, si la valeur de "maximum I-frame" est 5 212.08 cm dans le champ de contrôle de routage, celle de MTU doit être fixée à 2044 (pour les interfaces anneau à jeton seulement).

Remarque : Lorsque vous utilisez **iptrace**, le fichier de sortie *ne doit pas* résider sur un système de fichiers NFS.

Incidents de livraison de paquets

Communication avec un hôte distant

Si vous ne parvenez pas à établir la communication avec un hôte distant :

- Lancez la commande **ping** sur l'hôte local pour vérifier que l'interface locale reliée au réseau est opérationnelle et active.
- Appliquez la commande **ping** successivement aux hôtes et passerelles par lesquelles l'information transite, pour localiser la défaillance.

Si vous constatez des pertes de paquet ou des retards de livraison :

- Lancez la commande **trpt** pour effectuer un suivi des paquets au niveau socket.
- Lancez la commande **iptrace** pour effectuer le suivi de toutes les couches de protocole.

Si vous ne parvenez pas à établir la communication entre un réseau en anneau à jeton et un réseau Ethernet reliés par un pont qui fonctionne normalement :

- Vérifiez les valeurs MTU (Maximum Transmission Unit) des deux cartes de réseau. Elles doivent être compatibles pour autoriser la communication. En effet, si la taille du paquet entrant (en-têtes compris) est supérieure à la valeur MTU (Maximum Transmission Unit) de la carte, la machine rejette le paquet. Par exemple, un paquet de 1500 octets envoyé via un pont récupère un en-tête LLC de 8 octets pour atteindre une taille de 1508 octets. Si la valeur MTU de la machine réceptrice est fixée à 1500, le paquet est rejeté.

Réponses snmpd

Si **snmpd** ne répond pas et qu'aucun message d'erreur n'est transmis, il est probable que la taille du paquet est trop grande pour le gestionnaire de paquets UDP noyau. Dans ce cas, augmentez la valeur des variables du noyau **udp_sendspace** et **udp_recvspace** :

```
no –o udp_sendspace=64000
 no –o udp_recvspace=64000
```

La taille maximum d'un paquet UPD est de 64 Ko. La requête est rejetée si elle dépasse 64 Ko. Pour éviter ce type d'incident, le paquet doit être fractionné.

Incidents au niveau du protocole DHCP

Si vous ne pouvez pas obtenir une adresse IP ou d'autres paramètres de configuration :

- Assurez–vous qu'une interface à configurer a été spécifiée. Pour cela, éditez le fichier **/etc/dhcpcd.ini** à l'aide de l'application Network de Web-based System Manager ou à l'aide du raccourci **smit dhcp** de SMIT.
- Vérifiez qu'il existe un serveur sur le réseau local ou un agent relais configuré pour acheminer vos requêtes hors du réseau local.
- Vérifiez que le programme **dhcpcd** est actif. Dans la négative, lancez-le via la commande **startsrc –s dhcpcd**.

Informations de référence TCP/IP

Les thèmes relatifs au protocole TCP/IP abordés dans cette section sont les suivants :

Liste des commandes TCP/IP

chnamsv	Modification sur un hôte de la configuration du service de noms TCP/IP.
chprtsv	Modification de la configuration d'un service d'impression sur une machine client ou serveur.
hostent	Manipulation directe des entrées d'équivalence d'adresse dans la base de données de configuration du système.
ifconfig	Configuration/affichage des paramètres d'interface d'un réseau TCP/IP.
mknamsv	Configuration sur un hôte du service de noms TCP/IP pour un client.
mkprtsv	Configuration sur un hôte d'un service d'impression TCP/IP.
mktcpip	Définition des valeurs requises pour le lancement de TCP/IP sur un hôte.
no	Configuration des options de réseau.
rmnamsv	Déconfiguration sur un hôte du service de noms TCP/IP.
rmprtsv	Déconfiguration de la configuration d'un service d'impression sur une machine client ou serveur.
slattach	Raccordement des lignes série comme interfaces de réseau.
arp	Affichage/modification des tables de traduction d'adresse Internet en adresse matérielle, utilisées par ARP (Address Resolution Protocol).
gettable	Récupération à partir d'un hôte des tables d'hôte au format NIC.
hostid	Définition ou affichage de l'identificateur de l'hôte local courant.
hostname	Définition ou affichage du nom du système hôte courant.
htable	Conversion des fichiers hôtes au format utilisé par les routines de bibliothèque de réseau.
ipreport	Génération d'un rapport de suivi de paquet à partir du fichier spécifié.
iptrace	Suivi des paquets au niveau interface pour les protocoles Internet.
lsnamsv	Affichage des informations du service de noms stockées dans la base de données.
lsprtsv	Affichage des informations du service d'impression stockées dans la base de données.
mkhosts	Génération du fichier de tables hôte.

namerslv	Manipulation directe des entrées de serveur de noms de domaine pour les routines de résolution dans la base de données de configuration.
netstat	Affichage de l'état du réseau.
route	Manipulation directe des tables de routage.
ruser	Manipulation directe des entrées de trois bases de données système distinctes contrôlant l'accès des hôtes étrangers aux programmes locaux.
ruptime	Affichage de l'état de chaque hôte d'un réseau.
securetcpip	Activation de la fonction de sécurité réseau.
setclock	Définition de la date et de l'heure d'un hôte sur un réseau.
timedc	Informations sur le démon **timed**.
trpt	Suivi des prises TCP (Transmission Control Protocol).

Liste des démons TCP/IP

fingerd	Affichage des informations sur un utilisateur distant.
ftpd	Fonction serveur pour le protocole FTP (File Transfer Protocol) d'Internet.
gated	Apport des fonctions de routage de passerelles aux protocoles RIP (Routing Information Protocol), HELLO, EGP (Exterior Gateway Protocol), BGP (Border Gateway Protocol) et SNMP (Simple Network Management Protocol).
inetd	Gestion du service Internet pour un réseau.
named	Apport de la fonction serveur au protocole DOMAIN.
rexecd	Apport de la fonction serveur à la commande **rexec.**
rlogind	Apport de la fonction serveur à la commande **rlogin.**
routed	Gestion des tables de routage de réseau.
rshd	Apport de la fonction serveur pour l'exécution de commandes à distance.
rwhod	Apport de la fonction serveur aux commandes **rwho** et **ruptime.**
syslogd	Lecture et consignation des messages système.
talkd	Apport de la fonction serveur à la commande **talk.**
telnetd	Apport de la fonction serveur au protocole TELNET.
tftpd	Assure la fonction serveur pour le protocole TFTP (Trivial File Transfer Protocol).
timed	Appel au démon **timeserver** au lancement du système.

Liste des méthodes

Les méthodes d'unité sont des programmes associés à une unité qui exécutent des opérations de base de configuration d'unité. Pour en savoir plus sur les méthodes TCP/IP, reportez-vous à la section "List of TCP/IP Programming References" *AIX 5L Version 5.2 Communications Programming Concepts*.

Liste des fichiers TCP/IP

/etc/rc.bsdnet

Pour des informations sur les fichiers TCP/IP et les formats de fichier, reportez-vous à la section ”List of TCP/IP Programming References” *AIX 5L Version 5.2 Communications Programming Concepts*.

Liste des RFC

Pour connaître la liste des RFC (Request for Comments) pris en charge par ce système, reportez-vous à la section ”List of TCP/IP Programming References” *AIX 5L Version 5.2 Communications Programming Concepts*.

- RFC 1359 *Connecting to the Internet: What connecting institutions should anticipate*
- RFC 1325 *FYI on questions and answers: Answers to commonly asked ’new Internet user’ questions*
- RFC 1244 *Site Security Handbook*
- RFC 1178 ”Choosing a Name for Your Computer”
- RFC 1173 *Responsibilities of host and network managers: A summary of the ‘oral tradition’ of the Internet*

Chapitre 5. Administration du réseau

Administrer un réseau consiste à gérer globalement des réseaux systèmes, via le protocole SNMP, permettant aux hôtes d'échanger des informations de gestion. SNMP (Simple Network Management Protocol) est un protocole conçu pour les interréseaux basés sur TCP/IP. Ce chapitre traite des points suivants :

- Administration de réseau avec SNMP page 5-2
- SNMPv3 page 5-3
- SNMPv1 page 5-13

Lorsque AIX 5.2 est installé, la version non chiffrée de SNMPv3 est installée par défaut et démarrée lors de l'amorçage du système. Si vous avez configuré vos propres communautés, vos interruptions et vos entrées smux dans le fichier **/etc/snmpd.conf**, vous devez les faire migrer manuellement vers le fichier **/etc/snmpdv3.conf**. Pour plus d'informations sur la migration des communautés, reportez–vous à Migration de SNMPv1 vers SNMPv3 page 1-9.

Vous pouvez également consulter ces informations dans SNMP Overview for Programmers dans le manuel *AIX 5L Version 5.2 Communications Programming Concepts*.

Administration de réseau avec SNMP

L'administration de réseau SNMP repose sur le modèle client/serveur, largement exploité dans les applications basées sur TCP/IP. Chaque hôte à gérer exécute un processus appelé un *agent*. L'agent est un processus serveur qui maintient la base de données MIB (Management Information Base) pour l'hôte. Les hôtes impliqués dans les décisions d'administration du réseau peuvent exécuter un processus appelé un gestionnaire. Un *gestionnaire* est une application client qui génère les requêtes d'informations à la MIB et traite les réponses. Un gestionnaire peut en outre envoyer des requêtes aux serveurs de l'agent pour modifier les informations MIB.

SNMP sous AIX prend en charge les RFC suivants :

RFC 1155	Structure et identification des données de gestion (SMI) pour les interréseaux TCP/IP
RFC 1157	SNMP (Simple Network Management Protocol)
RFC 1213	Base MIB pour l'administration des interréseaux basés sur TCP/IP : MIB–II
RFC 1227	Protocole SNMP (Simple Network Management Protocol), protocole SMUX (single multiplexer) et base MIB (Management Information Base).
RFC 1229	Extensions à l'interface générique MIB (Management Information Base)
RFC 1231	MIB (Management Information Base) anneau à jeton IEEE 802.5
RFC 1398	Définitions des objets gérés pour les types d'interface Ethernet
RFC 1512	Base MIB FDDI
RFC 1514	Base MIB des ressources hôte
RFC 1592	SNMP–DPI (Simple Network Management Protocol–Distributed Program Interface) Version 2
RFC 1905	Opérations de protocole pour la Version 2 de Simple Network Management Protocol (SNMPv2)
RFC 1907	Base MIB pour la Version 2 de Simple Network Management Protocol (SNMPv2)
RFC 2572	Traitement et envoi des messages pour Simple Network Management Protocol (SNMP)
RFC 2573	Applications SNMP
RFC 2574	Modèle de sécurité utilisateur USM (User–based Security Model) pour la version 3 de Simple Network Management Protocol (SNMPv3)
RFC 2575	Modèle VACM (View–based Access Control Model) pour Simple Network Management Protocol (SNMP)

SNMPv3

Les informations de cette section concernent uniquement SNMPv3. Cette section traite des points suivants :

- Présentation de SNMPv3 page 5-4
- Architecture de SNMPv3 page 5-5
- Clés d'utilisateur SNMPv3 page 5-7
- Envoi de requêtes SNMPv3 page 5-10
- Identification des incidents SNMPv3 page 5-11

En outre, les tâches SNMPv3 suivantes sont aussi traitées :

- Migration de SNMPv1 vers SNMPv3 page 1-9
- Création d'utilisateurs dans SNMPv3 page 1-13
- Mise à jour dynamique des clés d'authentification et de confidentialité dans SNMPv3 page 1-18

Présentation de SNMPv3

Avant la version AIX 5.2, SNMPv1 était la seule version disponible de SNMP pour AIX. SNMPv3 est fourni avec AIX 5.2. SNMPv3 fournit une structure puissante et souple à la sécurité des messages et au contrôle de sécurité. La sécurité des message fournit les fonctionnalités suivantes :

- Le contrôle d'intégrité des données qui évite la corruption des données en transit
- La vérification de l'origine des données qui garantit que la requête ou la réponse est bien envoyée par la source revendiquée
- La vérification des dates des messages et facultativement, la confidentialité des données contre les accès non autorisées

L'architecture SNMPv3 met en œuvre le modèle USM (User–based Security Model) pour la sécurité des messages et le modèle VACM (View–based Access Control Model) pour le contrôle des accès. L'architecture prend en charge l'utilisation simultanée de différents modèles de sécurité, de contrôle des accès et de traitement des messages. Par exemple, la sécurité basée sur la communauté peut être utilisée simultanément avec USM.

USM fait appel au concept d'un utilisateur pour lequel les paramètres de sécurité (niveaux de sécurité, authentification et protocoles de confidentialité, et clés) sont configurés sur l'agent et le gestionnaire. Les messages envoyés via USM sont mieux protégés que ceux utilisant une sécurité basée sur la communauté, dans lesquels les mots de passe sont envoyés en clair et affichés dans les fichiers de traces. Avec USM, les messages échangés entre le gestionnaire et l'agent bénéficient de l'intégrité des données et de l'authentification de l'origine des données. Les retards et les renvois répétés de messages (hormis leurs occurrences normales en cas de protocole de transmission sans connexion) sont évités grâce à des indicateurs d'horodatage et des ID de requête. La confidentialité des données, ou le chiffrement, est également disponible, si cela est autorisé, sous la forme d'un produit installable séparément. La version SNMP chiffrée est fournie avec AIX Expansion Pack.

L'utilisation de VACM consiste à définir des collections de données (appelées des vues), des groupes d'utilisateurs des données et des instructions d'accès qui définissent les vues qu'un groupe d'utilisateurs particulier peut employer pour une opération de lecture, d'écriture ou de réception dans une interruption.

SNMPv3 permet également de configurer dynamiquement l'agent SNMP en lançant les commandes SNMP SET sur les objets MIB représentant la configuration de l'agent. Cette configuration dynamique prend en charge l'ajout, la suppression et la modification des entrées de configuration, localement ou à distance.

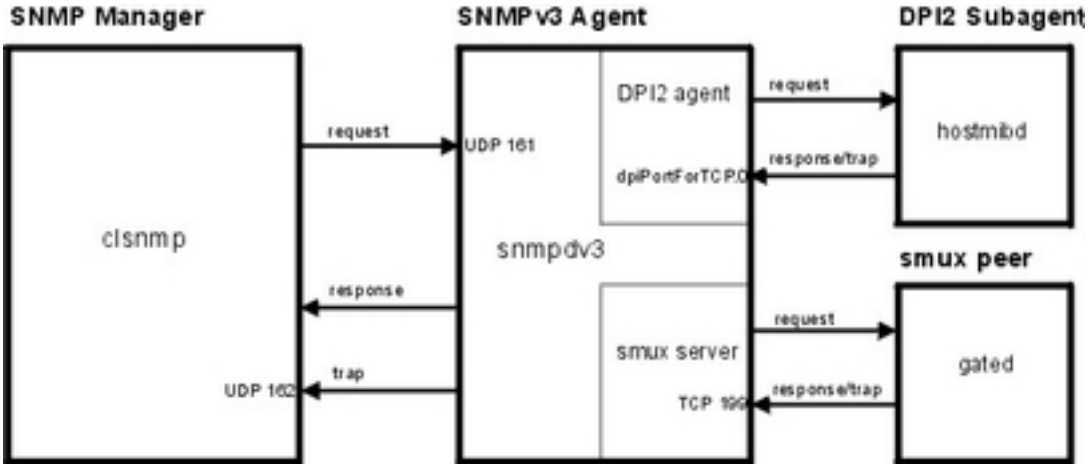
Les politiques d'accès SNMPv3 et les paramètres de sécurité sont spécifiés dans le fichier **/etc/snmpdv3.conf** sur l'agent SNMP et le fichier **/etc/clsnmp.conf** dans le gestionnaire SNMP. Vous trouverez un scénario sur la configuration des fichiers dans Création des utilisateurs dans SNMPv3 page 1-13. Vous pouvez aussi consulter les formats des fichiers **/etc/snmpdv3.conf** et **/etc/clsnmp.conf** dans *AIX 5L Version 5.2 Files Reference*.

Architecture SNMPv3

L'architecture SNMPv3 comporte quatre parties comme le montre le graphique suivant. Cette section décrit la façon dont ces systèmes interagissent l'un avec l'autre pour fournir les données requises.

Figure 27. Les principaux éléments de l'architecture SNMPv3

Cette illustration représente un exemple de l'architecture SNMPv3. Le sous–agent DPI2, l'homologue smux, le gestionnaire SNMP et l'agent SNMP sont représentés. La communication entre eux est également représentée.



Agent SNMP

L'agent SNMP reçoit des requêtes du gestionnaire SNMP et lui répond. En outre, l'agent SNMP communique avec tous les agents DPI2 et les homologues smux sur le système. L'agent SNMP gère certaines variables MIB et tous les sous–agents DPI2 et homologues smux enregistrent leurs variables MIB avec l'agent SNMP.

Lorsque **clsnmp** (le gestionnaire SNMP) émet une requête, elle est envoyée à UDP 161 sur l'agent SNMP. Si la requête est une requête SNMPv1 ou SNMPv2c, l'agent SNMP vérifie le nom de communauté et traite la requête. Si la requête est une requête SNMPv3, l'agent SNMP tente d'authentifier l'utilisateur demandant les données et vérifie qu'il possède les droits d'accès requis pour exécuter la requête en utilisant les clés d'authentification, et, si la version chiffrés s'exécute, les clés de confidentialité. Si l'agent SNMP ne peut pas authentifier l'utilisateur, ou si l'utilisateur n'a pas les droits d'accès adéquats pour exécuter la requête, l'agent SNMP n'honore pas la requête. Pour savoir comment créer des utilisateurs dans SNMPv3, reportez–vous à Création d'utilisateurs dans SNMPv3.

Si l'utilisateur est authentifié et possède les droits d'accès adéquats, l'agent SNMP exécute la requête. L'agent SNMP recherche les variables MIB demandées. Si l'agent SNMP lui–même gère les variables MIB demandées, il traite la requête et envoie une réponse au gestionnaire SNMP. Si un sous–agent DPI2 ou un homologue smux gère les variables MIB demandées, l'agent SNMP retransmet la requête au sous–agent DPI2 ou à l'homologue smux sur lequel les variables MB sont gérées, lui permet de traiter la requête et répond alors au gestionnaire SNMP.

Sous–agents DPI2

Un sous–agent tel que **hostmibd** communique avec l'agent DPI2 qui, dans SNMPv3, fait partie de l'agent SNMP. Le sous–agent DPI2 envoie des réponses et des interruptions à l'agent DPI2 via dpiPortForTCP.0. Comme il ne s'agit pas d'un port bien connu, le sous–agent DPI2 doit d'abord lancer une requête pour connaître le numéro de port de dpiPortForTCP.0. La requête est envoyée à UDP 161 sur l'agent SNMP, qui répond au sous–agent DPI2 en indiquant le numéro de port de dpiPortForTCP.0. Une fois le numéro de port reçu, le sous–agent DPI2 établit une connexion à l'agent DPI2 en utilisant le numéro

de port indiqué. Le sous–agent DPI2 enregistre alors ses sous–arborescences MIB avec l'agent DPI2.

Une fois la connexion établie et les sous–arborescences MIB enregistrées, le sous–agent DPI2 est prêt à répondre aux requêtes reçues de l'agent DPI2. Lorsqu'une requête est reçue, le sous–agent DPI2 traite la requête et renvoie les informations requises.

Le sous–agent DPI2 est également prêt à envoyer des interruptions si nécessaire. Lorsqu'une interruption est envoyée, l'agent SNMP vérifie son fichier **/etc/snmpdv3.conf** pour déterminer la ou les adresses IP à laquelle l'interruption doit être envoyée, et leur envoie l'interruption.

Homologues smux

Un homologue smux tel que **gated**, une fois démarré, établit la connexion à TCP 199 et initialise l'association smux. Suite à l'initialisation, l'homologue smux enregistre les sous–arborescences MIB qu'il va gérer.

Après l'enregistrement, l'homologue smux est prêt à accepter toute requête entrante du serveur smux et à renvoyer des réponses. Lorsque l'homologue smux reçoit une requête, il la traite et renvoie une réponse au serveur smux.

L'homologue smux peut aussi envoyer une interruption au serveur smux. Lorsqu'une interruption est envoyée, l'agent SNMP vérifie son fichier **/etc/snmpdv3.conf** pour déterminer la ou les adresses IP à laquelle l'interruption doit être envoyée, et leur envoie l'interruption.

Gestionnaire SNMP

Le gestionnaire SNMP exécute **clsnmp**, qui est compatible avec SNMPv1, SNMPv2c, et SNMPv3. Avec la commande **clsnmp**, un utilisateur peut envoyer une requête, par exemple `get`, `get-next`, `get-bulk`, ou `set` . La requête est envoyée à UDP 161 sur l'agent SNMP, puis attend la réponse de l'agent SNMP.

Il peut aussi écouter les interruptions SNMP sur UDP 162. Le gestionnaire SNMP reçoit les interruptions si l'adresse IP si cela est précisé dans le fichier **/etc/snmpdv3.conf** de l'agent SNMP.

Variables MIB

Pour plus d'informations sur les variables MIB, reportez–vous à Management Information Base, Terminology Related to Management Information Base Variables, Working with Management Information Base Variables, et Management Information Base Database dans le manuel *AIX 5L Version 5.2 Communications Programming Concepts*.

Si vous souhaitez configurer vote propre sous–agent DPI2 ou un homologue smux, reportez–vous aux répertoires **/usr/samples/snmpd/smux** et **/usr/samples/snmpd/dpi2**.

Clés utilisateur SNMPv3

Clés d'authentification

L'authentification est en général requise pour permettre le traitement des requêtes SNMPv3 (sauf si le niveau de sécurité demandé est `noAuth`). Lors de l'authentification d'une requête, l'agent SNMP vérifie que la clé d'authentification envoyée dans une requête SNMPv3 peut être utilisée pour créer un résumé de message correspondant à celui créé par la clé d'authentification définie par l'utilisateur.

Lorsque le gestionnaire SNMP envoie une requête, la commande **clsnmp** utilise la clé d'authentification trouvée dans une entrée dans le fichier **/etc/clsnmp.conf** du gestionnaire SNMP. Il doit établir une corrélation avec la clé d'authentification indiquée dans une entrée `USM_USER` de cet utilisateur dans le fichier **/etc/snmpdv3.conf** de l'agent SNMP. Les clés d'authentification sont générées à l'aide de la commande **pwtokey**.

La clé d'authentification est générée à partir de deux éléments :

- Le mot de passe indiqué
- L'identification de l'agent SNMP sur lequel la clé sera utilisée. Si l'agent est un agent IBM, et que son engineID (ID moteur) a été généré avec une formule engineID spécifique au fournisseur, l'agent peut être identifié par une adresse IP ou un nom d'hôte. Sinon, l'engineID doit être fourni en tant qu'identification de l'agent.

Une clé incorporant l'identification de l'agent sur lequel il sera utilisé est appelée une clé localisée. Elle peut uniquement être utilisée sur cet agent. Une clé n'incorporant pas l'engineID de l'agent sur lequel il sera utilisé est appelée une clé non localisée.

Les clés stockées dans le fichier de configuration de la commande **clsnmp**, **/etc/clsnmp.conf**, sont normalement des clés non localisées. Les clés stockées dans le fichier configuration de l'agent SNMP, **/etc/snmpdv3.conf**, peuvent être localisées ou non localisées, mais il est considéré comme plus sûr d'utiliser des clés localisées.

Au lieu de stocker les clés d'authentification dans le fichier de configuration client, la commande **clsnmp** permet de stocker les mots de passe utilisateur. Sil a commande **clsnmp** est configurée avec un mot de passe, le code génère une clé d'authentification (et une clé de confidentialité si nécessaire, et si la version chiffrée est installée) pour l'utilisateur. Ces clés doivent produire les mêmes valeurs d'authentification que les clés configurées pour `USM_USER` dans le fichier **/etc/snmpdv3.conf** de l'agent ou être configuré dynamiquement avec les commandes SNMP SET. Toutefois, l'utilisation de mots de passe dans le fichier de configuration client est considérée comme moins sûre que celles de clés dans le fichier de configuration.

Clés de confidentialité

Le chiffrement est proposé en tant que produit distinct dans AIX Expansion Pack lorsque la législation sur l'exportation l'autorise. Les clés utilisées pour le chiffrement sont générées avec les mêmes algorithmes que ceux utilisés pour l'authentification. Toutefois, les longueurs de clés diffèrent. Par exemple, une clé d'authentification HMAC–SHA a une longueur de 20 octets, mais une clé de chiffrement localisée utilisée avec HMAC–SHA n'a que 16 octets.

La version chiffrée est activée automatiquement après l'installation. Pour revenir à la version non chiffrée, lancez la commande **snmpv3_ssw**.

Génération de clés

AIX utilise la commande **pwtokey** pour générer des clés d'authentification et, le cas échéant, des clés de confidentialité. La commande **pwtokey** permet de convertir les mots de passe en clés d'authentification et de confidentialité localisées et non localisées. La procédure **pwtokey** choisit un mot de passe et un ID pour l'agent et génère des clés d'authentification et de confidentialité. Comme la procédure utilisée par la commande **pwtokey** est le même algorithme utilisé par la commande **clsnmp**, la personne configurant l'agent SNMP peut générer des clés appropriées d'authentification (et de confidentialité) à

placer dans le fichier **/etc/clsnmp.conf** sur le gestionnaire SNMP pour un utilisateur, avec un mot de passe et l'adresse IP sur laquelle la cible va s'exécuter.

Lorsque vous avez généré les clés d'authentification (et de confidentialité si vous exécutez la version chiffrée), vous devez entrer ces clés dans le fichier **/etc/snmpdv3.conf** sur l'agent SNMP et dans le fichier **/etc/clsnmp.conf** sur le gestionnaire SNMP.

Dans SNMPv3, il existe neuf configurations d'utilisateur possibles. Chaque configuration possible, ainsi qu'un exemple, est indiquée ci–après. Ces clés particulières ont été générées avec le mot de passe `defaultpassword` et l'adresse IP `9.3.149.49` . La commande suivante a été utilisée :

```
pwtokey –u all –p all defaultpassword 9.3.149.49
```

Les clés d'authentification et de confidentialité suivantes ont été générées :

```
Affichage de la clé d'authentification 16 octets HMAC–MD5 authKey :

   18a2c7b78f3df552367383eef9db2e9f

 Affichage de la clé d'authentification 16 octets HMAC–MD5 localized
authKey :
   a59fa9783c04bcbe00359fb1e181a4b4

 Affichage de la clé de confidentialité 16 octets HMAC–MD5 privKey :

   18a2c7b78f3df552367383eef9db2e9f

 Affichage de la clé de confidentialité localisée 16 octets HMAC–MD5
privKey :
   a59fa9783c04bcbe00359fb1e181a4b4

 Affichage de la clé d'authentification 20 octets HMAC–SHA authKey :

   754ebf6ab740556be9f0930b2a2256ca40e76ef9

 Affichage de la clé d'authentification localisée 20 octets HMAC–SHA
localized authKey :
   cd988a098b4b627a0e8adc24b8f8cd02550463e3

 Affichage de la clé de confidentialité 20 octets HMAC–SHA privKey :

   754ebf6ab740556be9f0930b2a2256ca40e76ef9

 Affichage de la clé de confidentialité localisée 16 octets HMAC–SHA :
   cd988a098b4b627a0e8adc24b8f8cd02
```

Ces entrées apparaîtront dans le fichier **/etc/snmpdv3.conf**. Les neuf configurations possibles sont les suivantes :

- Clés d'authentification et de confidentialité localisées utilisant le protocole HMAC–MD5 :

  ```
  USM_USER user1 – HMAC–MD5 a59fa9783c04bcbe00359fb1e181a4b4 DES
  a59fa9783c04bcbe00359fb1e181a4b4 L – –
  ```

- Clés d'authentification et de confidentialité non localisées utilisant le protocole HMAC–MD5 :

  ```
  USM_USER user2 – HMAC–MD5 18a2c7b78f3df552367383eef9db2e9f DES
  18a2c7b78f3df552367383eef9db2e9f N  – –
  ```

- Clé d'authentification localisée utilisant le protocole HMAC–MD5 :

  ```
  USM_USER user3 – HMAC–MD5 a59fa9783c04bcbe00359fb1e181a4b4 – – L –
  ```

- Clé d'authentification non localisée utilisant le protocole HMAC–MD5 :

  ```
  USM_USER user4 – HMAC–MD5 18a2c7b78f3df552367383eef9db2e9f – – N –
  ```

- Clés d'authentification et de confidentialité localisées utilisant le protocole HMAC–SHA :

  ```
  USM_USER user5 – HMAC–SHA cd988a098b4b627a0e8adc24b8f8cd02550463e3 DES
  cd988a098b4b627a0e8adc24b8f8cd02 L –
  ```

- Clés d'authentification et de confidentialité non localisées utilisant le protocole HMAC–SHA :

```
USM_USER user6 – HMAC–SHA 754ebf6ab740556be9f0930b2a2256ca40e76ef9 DES
754ebf6ab740556be9f0930b2a2256ca40e76ef9 N –
```

- Clé d'authentification localisée utilisant le protocole HMAC–SHA :

```
USM_USER user7 – HMAC–SHA cd988a098b4b627a0e8adc24b8f8cd02550463e3 – – L
–
```

- Clé d'authentification non localisée utilisant le protocole HMAC–SHA :

```
USM_USER user8 – HMAC–SHA 754ebf6ab740556be9f0930b2a2256ca40e76ef9 – – N
–
```

- Aucune clé d'authentification ou de confidentialité utilisée (SNMPv1)

```
USM_USER user9 – none – none – – –
```

La configuration des utilisateurs dans SNMPv3 nécessite la configuration des deux fichiers **/etc/snmpdv3.conf** et **/etc/clsnmp.conf**. Pour consulter un scénario sur la génération des clés utilisateur et l'édition des fichiers de configuration requis, reportez–vous à Création d'utilisateurs dans SNMPv3. En outre, reportez–vous aux descriptions des commandes **pwtokey** et **clsnmp** dans *AIX 5L Version 5.2 Commands Reference*, et aux formats de fichier des fichiers **/etc/clsnmp.conf** et **/etc/snmpdv3.conf** dans *AIX 5L Version 5.2 Files Reference*. Vous pouvez aussi vous reporter aux exemples de fichiers de configuration **snmpdv3.conf** et **clsnmp.conf** situés dans le répertoire **/usr/samples/snmpdv3**.

Mise à jour des clés

SNMPv3 offre la possibilité de mettre à jour dynamiquement des clés utilisateur en fonction des nouveaux mots de passe. Pour ce faire, vous devez lancer la commande **pwchange** pour générer de nouvelles clés utilisateur basées sur un mot de passe mis à jour, puis lancer la commande **clsnmp** pour mettre à jour dynamiquement la clé utilisateur dans le fichier **/etc/snmpdv3.conf** et éditer le fichier **/etc/clsnmp.conf** en indiquant les nouvelles clés. Pendant ce processus, le nouveau mot de passe n'est jamais communiqué entre les machines.

Pour obtenir des instructions pas à pas sur la mise à jour des clés utilisateur, reportez–vous à Mise à jour dynamique des clés d'authentification et de confidentialité dans SNMPv3 page 1-18. En outre, reportez–vous aux descriptions des commandes **pwchange** et **clsnmp** dans *AIX 5L Version 5.2 Commands Reference* et aux formats de fichiers **/etc/clsnmp.conf** et **/etc/snmpdv3.conf** dans *AIX 5L Version 5.2 Files Reference*

Emission de requêtes SNMPv3

La commande **clsnmp** permet d'envoyer des requêtes SNMP aux agents SNMP sur les hôtes locaux ou distants. Il peut s'agir de requêtes SNMPv1, SNMPv2c ou SNMPv3. Pour que les requêtes puissent être traitées, le fichier **/etc/clsnmp.conf** doit être configuré.

La commande **clsnmp** peut lancer des requêtes get, getnext, getbulk, set, walk, et findname. Chacune de ces requêtes est décrite brièvement ci–dessous :

get	permet à l'utilisateur de collecter les données d'une variable MIB
getnext	indique la variable MIB suivante dans la sous–arborescence MIB
getbulk	indique toutes les variables MIB de plusieurs sous–arborescences MIB
set	permet à l'utilisateur de définir une variable MIB
walk	indique toutes les variables MIB d'une seule sous–arborescence
findname	établit une équivalence entre l'OID et le nom de la variable
trap	permet à clsnmp d'écouter les interruptions sur le port 162

Pour plus de détails sur l'émission de requêtes clsnmp, reportez–vous à la description de commande **clsnmp** dans *AIX 5L Version 5.2 Commands Reference*.

Identification des incidents SNMPv3

Les problèmes suivants peuvent se présenter.

- Après une mise à niveau pour passer d'une ancienne version d'AIX à AIX 5.2, SNMP ne fonctionne plus de la même façon qu'avant la migration.

  Vous devez faire migrer les entrées de communauté et smux définies dans le fichier **/etc/snmpd.conf** vers le fichier **/etc/snmpdv3.conf**. Pour plus d'informations sur la migration de ces informations, reportez–vous à Migration de SNMPv1 vers SNMPv3 page 1-9.

- Mes requêtes ne reçoivent aucune réponse.

  La cause la plus probable de ce problème est une erreur de configuration dans les fichiers **/etc/snmpdv3.conf** ou **/etc/clsnmp.conf** ou les deux. Examinez soigneusement ces fichiers pour vérifier que toutes les informations sont entrées correctement. Pour plus d'informations sur l'édition de ces fichiers lors de la création de nouveaux utilisateurs, reportez–vous à Création d'utilisateurs dans SNMPv3 page 1-13.

- J'ai configuré un nouvel utilisateur à l'aide de clés d'authentification et de confidentialité, mais j'obtiens un message d'erreur lorsque je l'utilise.

  La cause la plus probable est que vous n'exécutez pas la version chiffrée de SNMPv3. Procédez comme suit pour déterminer la version que vous exécutez :

  1. Exécutez `ps -e|grep snmpd`.
     - Si vous n'avez pas reçu de résultat, démarrez le démon **snmpd**. Exécutez `startsrc -s snmpd`.
     - Si votre résultat comprend `snmpdv1`, vous exécutez SNMPv1. Vous pourrez lancer des requêtes SNMPv1 lors de l'exécution de cette version.
     - Si votre résultat comprend `snmpdv3ne`, vous exécutez la version non chiffrée de SNMPv3. Une fois installé AIX 5.2, cette version fonctionnera correctement. Elle ne vous permet pas d'utiliser les clés de confidentialité.
     - Si votre résultat inclut `snmpdv3e`, vous exécutez la version chiffrée de SNMPv3, qui est un produit installable séparément. La version chiffrée de SNMPv3 est disponible dans AIX Expansion Pack si cela est autorisé. La version chiffrée de SNMPv3 permet d'utiliser des clés de confidentialité.
  2. Déterminez si la version que vous exécutez est la version voulue. Si ce n'est pas le cas, la commande **snmpv3_ssw** pour modifier la version comme suit :
     - `snmpv3_ssw -1` bascule vers SNMPv1
     - `snmpv3_ssw -n` bascule vers la version non chiffrée de SNMPv3
     - `snmpv3_ssw -e` bascule vers la version chiffrée de SNMPv3 si elle est installée

- J'ai modifié le fichier **/etc/snmpdv3.conf** et régénéré le démon, mais mes modifications ne sont pas appliquées.

  Après avoir modifié le fichier **/etc/snmpdv3.conf**, arrêtez et démarrez le démon SNMP. La régénération du démon ne donne pas de résultat. Utilisez la procédure suivante :

  1. Arrêtez le démon SNMP en exécutant `stopsrc -s snmpd`.
  2. Démarrez le démon SNMP en exécutant `startsrc -s snmpd`.

- Le sous–agent DPI2 est démarré mais je ne peux pas interroger les variables MIB à partir de celui–ci.

  La cause la plus probable est que la communauté `public` n'est pas configuré dans le fichier **/etc/snmpdv3.conf**. Par défaut, le sous–agent DPI2 livré avec AIX utilise le nom de communauté `public` pour se connecter à l'agent SNMP. La communauté `public` est configurée dans le fichier **/etc/snmpdv3.conf** par défaut. Si vous avez supprimé la

communauté `public` du fichier **/etc/snmpd.conf**, ajoutez les lignes suivantes au fichier :

```
VACM_GROUP group1 SNMPv1  public  –
VACM_VIEW defaultView 1.3.6.1.4.1.2.2.1.1.1.0 – included –
 VACM_ACCESS  group1 – – noAuthNoPriv SNMPv1  defaultView – defaultView –
COMMUNITY public    public     noAuthNoPriv 0.0.0.0     0.0.0.0         –
```

`1.3.6.1.4.1.2.2.1.1.1.0` est l'OID de dpiPortForTCP.0.

- Je ne peux pas interroger des variables MB gérées par l'homologue smux alors que cela était possible avant la migration.

  Vérifiez que votre entrée smux est présente dans les fichiers **/etc/snmpdv3.conf** et **/etc/snmpd.peers**. Si vous configurez de nouveaux homologues smux, vérifiez qu'ils sont entrés aussi dans ces deux fichiers.

- J'ai implémenté mon propre groupe de variables MIB, mais je ne peux pas les inclure ou les exclure des vues des utilisateurs.

  Dans l'entrée VACM_VIEW du fichier **/etc/snmpdv3.conf**, vous devez spécifier l'OID de la variable MIB à la place du nom de variable MIB.

- Je ne reçois pas d'interruptions.

  Vérifiez que vous avez correctement configuré les entrées d'interruption dans le fichier **/etc/snmpdv3.conf**. En outre, si l'interruption est une interruption SNMPv3, le fichier **/etc/clsnmp.conf** doit aussi être configuré. Pour plus d'instructions sur la configuration des interruptions, reportez–vous à Création d'utilisateurs dans SNMPv3 page 1-13.

  En outre, vérifiez que la machine devant recevoir les interruptions (dans le fichier **/etc/snmpdv3.conf**) est à leur écoute. Vous pouvez démarrer ce processus en exécutant `clsnmp trap` sur la ligne de commande.

- Pourquoi le serveur DPI2 ne s'exécute–t–il pas dans l'environnement SNMPv3 ?

  Dans l'architecture SNMPv3, l'agent SNMPv3 exécute lui–même le serveur DPI2. Pour plus d'informations, reportez–vous à Architecture SNMPv3 page 5-5.

SNMPv1

Les informations de cette section sont spécifiques à SNMPv1.

Politiques d’accès SNMPv1

Lorsque SNMPv1 est utilisé, l’agent **snmpd** utilise un schéma d’authentification simple pour connaître les stations du gestionnaire SNMP (Simple Network Management Protocol) peuvent accéder à ses variables MIB (Management Information Base). Le schéma d’authentification suppose de spécifier les politiques d’accès SNMP pour SNMPv1. Une politique d’accès SNMP est un ensemble de relations administratives impliquant une association au sein d’une communauté SNMP, un mode d’accès et une vue MIB.

On appelle *communauté SNMP* un groupe d’hôtes doté d’un nom. Un nom de communauté est une chaîne d’octets qu’un gestionnaire SNMP doit imbriquer dans un paquet de requêtes SNMP à des fins d’authentification.

Le *mode d’accès* spécifie l’accès accordé aux hôtes de la communauté, en ce qui concerne la récupération et la modification des variables MIB à partir d’un agent SNMP spécifique. Le mode d’accès peut être : *none*, *read–only*, *read–write* ou *write–only*.

Une *vue MIB* définit une ou plusieurs sous–arborescences MIB accessibles par une communauté SNMP donnée. Il peut s’agir de toute l’arborescence MIB ou d’un sous–ensemble de cette arborescence.

Lorsque l’agent SNMP reçoit une requête, il compare le nom de la communauté à l’adresse IP de l’hôte demandeur pour savoir si ce dernier est un membre de la communauté SNMP. Si oui, il détermine ensuite si l’hôte demandeur a le droit d’accès spécifié aux variables MIB voulues, tel que défini dans la politique d’accès associée à cette communauté. Si tous les critères sont vérifiés, l’agent SNMP tente de répondre à la demande. Sinon, il génère une interruption d’échec d’authentification (*authenticationFailure*) ou envoie un message d’erreur à l’hôte demandeur.

Les politiques d’accès de SNMPv1 pour l’agent **snmpd** sont configurables par l’utilisateur et sont spécifiées dans le fichier **/etc/snmpd.conf**. Pour configurer les politiques d’accès SNMP pour l’agent **snmpd**, reportez–vous au fichier **/etc/snmpd.conf**.

Démon SNMP

Le démon SNMP (Simple Network Management Protocol) est un processus serveur d'arrière-plan exécutable sur n'importe quel hôte station de travail TCP/IP (Transmission Control Protocol/Internet Protocol). Ce démon, qui sert d'agent SNMP, reçoit, authentifie et traite les requêtes SNMP issues des applications du gestionnaire. Pour en savoir plus, reportez-vous aux sections "Simple Network Management Protocol," "How a Manager Functions" et "How an Agent Functions" *AIX 5L Version 5.2 Communications Programming Concepts*.

Remarque : Les termes démon SNMP, agent SNMP et agent sont synonymes.

Pour une configuration minimale, il faut que l'interface TCP/IP de boucle soit active pour le démon **snmpd**. Avant de lancer TCP/IP, entrez la commande :

```
ifconfig lo0 loopback up
```

Configuration du démon SNMP

Le démon SNMP (Simple Network Management Protocol) tente de lier les sockets à certain ports UDP (User Datagram Protocol) et TCP (Transmission Control Protocol) identifiés, qui doivent être définis dans le fichier **/etc/services**, comme suit :

```
snmp            161/udp
snmp-trap       162/udp
smux            199/tcp
```

Le service `snmp` doit être affecté du port `161`, conformément à RFC 1157. Le fichier **/etc/services** assigne les ports 161, 162 et 199 à ces services. Si le fichier **/etc/services** est mis à disposition à partir d'une autre machine, ces ports assignés doivent être rendus disponibles dans le fichier **/etc/services** servi pour que le démon SNMP puisse s'exécuter.

Le démon **SNMP** lit le fichier de configuration sur la version SNMP en cours d'exécution au lancement et lors de l'émission d'une commande **refresh** (si le démon **snmpd** est appelé sous le contrôle SRC) ou d'un signal **kill–1**.

fichier /etc/snmpd.conf

Le fichier de configuration **/etc/snmpd.conf** spécifie les noms de communauté et les vues et droits d'accès associés, les hôtes pour la notification d'interruption, les attributs de connexion, les paramètres spécifiques de **snmpd** et les configurations SMUX (single multiplexer) pour le démon **SNMP**. Pour en savoir plus, consultez le fichier **/etc/snmpd.conf**.

Fonctionnement du démon SNMP

Le démo, SNMP (Simple Network Management Protocol) traite les requêtes SNMP issues des applications du gestionnaire. Pour en savoir plus, consultez les sections ”Simple Network Management Protocol (SNMP),” ”How a Manager Functions” et ”How an Agent Functions” *AIX Communications Programming Concepts*.

Traitement d’un message et authentification

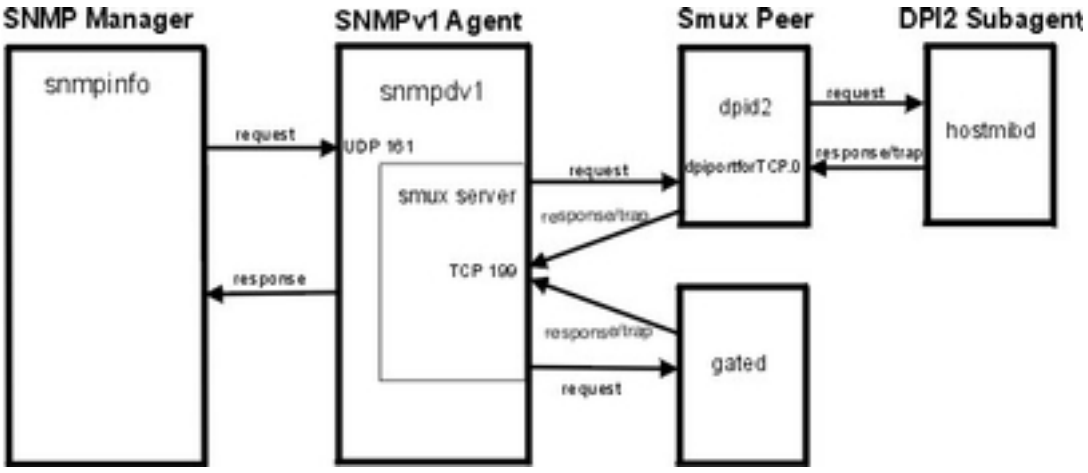
Toutes les requêtes, interruptions et réponses sont transmises sous la forme de messages codés en ASN.1. Un message, tel que défini par RFC 1157, a la structure suivante :

Version Communauté PDU

Version étant la version de SNMP (actuellement la version 1), *Communauté,* le nom de la communauté et *PDU*, l’unité des données de protocole contenant les données de requête, de réponse ou d’interruption SNMP. Un PDU est également codé selon les règles ASN.1.

Figure 28. Les principaux éléments de l’architecture SNMPv1

Cette illustration représente un exemple de l’architecture SNMPv1. Le sous–agent DPI2, le pair smux, le gestionnaire SNMP et l’agent SNMP sont représentés. La communication entre eux est également représentée.



Le démon SNMP reçoit et transmet tous les messages du protocole SNMP via UDP (User Datagram Protocol) TCP/IP (Transmission Control Protocol/Internet Protocol). Les requêtes sont acceptées sur le port identifié 161. Les interruptions sont transmises aux hôtes répertoriés dans les entrées d’interruption du fichier **/etc/snmpd.conf** qui écoutent le port identifié 162.

A réception d’une requête, l’adresse IP source et le nom de la communauté sont comparés à la liste des adresses IP, des noms de communauté, des droits et des vues, spécifiés dans le fichier **/etc/snmpd.conf**. L’agent **snmpd** lit ce fichier au lancement et à l’émission d’une commande **refresh** ou d’un signal **kill –1**. En l’absence d’entrée correspondante, la requête est ignorée. Dans le cas contraire, l’accès est accordé, en fonction des droits spécifiés pour cette association (adresse IP, communauté et nom de vue) dans le fichier **/etc/snmpd.conf**. Le message et le PDU doivent être codés conformément aux règles ASN.1.

Ce schéma d’authentification n’est pas censé garantir une sécurité totale. Si le démon SNMP n’est utilisé que pour les requêtes ”get” et ”get-next”, la sécurité n’est pas forcément très importante. En revanche, si des requêtes ”set” sont autorisées, il est possible de restreindre le privilège ”set”.

Pour en savoir plus, consultez le fichier **/etc/snmpd.conf**. Pour en savoir plus, reportez-vous à ”Management Information Base (MIB)” *AIX Communications Programming Concepts*.

Traitement d'une requête

Le démon SNMP peut recevoir trois types de requêtes PDU. Les types de requêtes, définies dans RFC 1157, et les PDU ont tous le format suivant :

Format de PDU de requête			
ID requête	**état-erreur**	**index-erreur**	**liaisons-variable**
GET	0	0	*VarBindList*
GET–NEXT	0	0	*VarBindList*
SET	0	0	*VarBindList*

Le champ ID-requête indique la nature de la requête ; les champs état-erreur et index-erreur sont inutilisés et doivent être définis à 0 (zéro) ; le champ liaisons-variable contient une liste de longueur variable des ID d'instance, au format numérique, dont les valeurs sont demandées. Si la valeur du champ ID requête est SET, le champ liaisons-variable est une liste de paires ID d'instance/valeur.

Pour en savoir plus, consultez la section "Using the Management Information Base (MIB) Database" *AIX Communications Programming Concepts*.

Traitement d'une réponse

Les PDU de réponse ont presque le même format que les PDU de requête :

Format de PDU de réponse			
ID requête	**état-erreur**	**index-erreur**	**liaisons-variable**
GET–RESPONSE	*ErrorStatus*	*ErrorIndex*	*VarBindList*

Si la requête a abouti, la valeur des champs état-erreur et index-erreur est 0 (zéro), et le champ liaisons-variable contient la liste complète des paires ID d'instance/valeur.

Si un ID d'instance du champ liaisons-variable du PDU de requête n'a pas abouti, l'agent SNMP interrompt le traitement, entre l'index de l'ID d'instance défaillant dans le champ index-erreur, enregistre un code d'erreur dans le champ état-erreur et copie la liste de résultats partiellement complétée dans le champ liaisons-variable.

RFC 1157 définit les valeurs suivantes pour le champ état-erreur :

Valeurs du champ état-erreur		
Valeur	**Valeur**	**Explication**
noError	0	Traitement réussi (index d'erreur = 0).
tooBig	1	La taille du PDU de réponse dépasse une limite définie par l'implémentation (index d'erreur = 0).
noSuchName	2	Un ID d'instance n'existe pas dans la vue MIB appropriée pour les types de requête GET et SET ou n'a pas de successeur dans l'arborescence MIB dans la vue MIB appropriée pour les requêtes GET-NEXT (index d'erreur différent de zéro).
badValue	3	Pour les requêtes SET uniquement, une valeur spécifiée est syntaxiquement incompatible avec l'attribut de type de l'ID d'instance correspondant (index d'erreur différent de zéro).

readOnly	4	Non défini.
genErr	5	Une erreur définie par l'implémentation s'est produite (index d'erreur différent de zéro) ; par exemple, une tentative d'assignation d'une valeur dépassant les limites d'implémentation.

Traitement d'une interruption

Les PDU d'interruption sont définis par RFC 1157 de façon à avoir le format suivant :

Format de PDU d'interruption					
entreprise	**agent adresse**	**générique interruption**	**spécifique interruption**	**horodate**	**variable liaisons**
ID Objet	*Entier*	*Entier*	*Entier*	*Tics d'horloge*	*VarBindList*

Les champs sont utilisés comme suit :

entreprise
: Identificateur d'objet assigné au fournisseur implémentant l'agent. Valeur de la variable **sysObjectID**, unique pour chaque metteur en oeuvre d'un agent SNMP. La valeur assignée à cette implémentation de l'agent est **1.3.6.1.4.1.2.3.1.2.1.1.3** ou **risc6000snmpd.3**.

adresse-agent
: Adresse IP de l'objet générateur de l'interruption.

interruption générique
: Entier, comme suit :

  0 *coldStart*

  1 *warmStart*

  2 *linkDown*

  3 *linkUp*

  4 *authenticationFailure*

  5 *egpNeighborLoss*

  6 *enterpriseSpecific*

interruption spécifique
: Inutilisé, réservé à un usage ultérieur.

horodate
: Temps écoulé, en centièmes de seconde, depuis la dernière réinitialisation de l'agent jusqu'à l'événement générant l'interruption.

liaisons-variable
: Informations supplémentaires, fonction du type d'*interruption-générique*.

Les valeurs d'interruption générique suivantes indiquent que certains événements système ont été détectés :

coldStart	L'agent est en cours de réinitialisation. Les données de configuration et/ou la valeur des variables MIB peuvent avoir changé. Les epochs de mesure doivent être relancés.
warmStart	L'agent est en cours de réinitialisation, mais les données de configuration ou la valeur des variables MIB n'ont pas changé. Dans cette mise en oeuvre de l'agent SNMP, une interruption *warmStart* est générée à la relecture du fichier **/etc/snmpd.conf**. Les informations de configuration dans le fichier **/etc/snmpd.conf** concernent la configuration de l'agent sans effets sur les bases de données du gestionnaire SNMP. Les epochs de mesure ne doivent pas être relancés.
linkDown	L'agent a détecté qu'une interface de communication identifiée a été désactivée.
linkUp	L'agent a détecté qu'une interface de communication identifiée a été activée.
authenticationFailure	Un message reçu n'a pu être authentifié.
egpNeighborLoss	Un neighbor EGP (Exterior Gateway Protocol) est perdu. Cette valeur n'est générée que lorsque l'agent s'exécute sur un hôte exécutant le démon **gated**, avec le protocole EGP (Exterior Gateway Protocol).
enterpriseSpecific	Non implémenté, réservé à un usage ultérieur.

Les interruptions *linkDown* et *linkUp* contiennent une paire ID d'instance/valeur unique dans la liste des liaisons de variable. L'ID d'instance identifie l'**ifIndex** de la carte désactivée ou activée, et la valeur est celle de **ifIndex**. L'interruption pour *egpNeighborLoss* contient également une liaison consistant en l'ID d'instance et la valeur de *egpNeighAddr* pour le voisin perdu.

Support du démon SNMP pour la famille EGP de variables MIB

Si l'hôte de l'agent exécute le démon **gated** alors que le protocole EGP (Exterior Gateway Protocol) est activé, plusieurs variables MIB (Management Information Base) du groupe EGP sont acceptées par le démon **gated** et accessibles par l'agent **snmpd**.

Les variables MIB EGP suivantes ont une instance unique :

egpInMsgs	Nombre de messages EGP reçus sans erreur.
egpInErrors	Nombre de messages EGP reçus avec erreur.
egpOutMsgs	Nombre total de messages EGP transmis par le démon `gated` actif sur l'hôte de l'agent.
egpOutErrors	Nombre de messages EGP qui n'ont pas pu être envoyés au démon `gated` de l'hôte de l'agent, par suite de limitations des ressources.
egpAs	Numéro système autonome du démon `gated` de l'hôte de l'agent.

Les variables MIB EGP suivantes ont une instance pour chaque homologue ou voisin EGP acquis par le démon **gated** de l'hôte de l'agent :

egpNeighState	État de cet homologue EGP : 1 idle 2 acquisition 3 down 4 up 5 cease
egpNeighAddr	Adresse IP de cet homologue EGP.
egpNeighAs	Numéro système autonome de cet homologue EGP. Zéro (0) indique que ce numéro n'est pas encore connu.
egpInNeighMsgs	Nombre de messages EGP reçus sans erreur de cet homologue EGP.
egpNeighInErrs	Nombre de messages EGP reçus avec erreur de cet homologue EGP.
egpNeighOutMsgs	Nombre de messages EGP générés localement pour cet homologue EGP.
egpNeighOutErrs	Nombre de messages EGP générés en local, non envoyés à cet homologue EGP par suite de limitations des ressources.
egpNeighInErrMsgs	Nombre de messages d'erreur définis par EGP reçus de cet homologue EGP.
egpNeighOutErrMsgs	Nombre de messages d'erreur définis par EGP envoyés à cet homologue EGP.
egpNeighStateUp	Nombre de transitions de l'état EGP jusqu'à l'état UP avec cet homologue EGP.
egpNeighStateDowns	Nombre de transitions de l'état EGP à partir de l'état UP jusqu'à n'importe quel état avec cet homologue EGP.
egpNeighIntervalHello	Intervalle entre les retransmissions de la commande Hello d'EGP, en centièmes de seconde.

egpNeighIntervalPoll Intervalle entre les retransmissions de la commande d'interrogation d'EGP, en centièmes de seconde.

egpNeighMode Mode d'interrogation de cet homologue EGP. Il peut être actif (1) ou passif (2).

egpNeighEventTrigger Une variable de contrôle déclenche des événements de lancement et d'arrêt initiés par l'opérateur sur cet homologue EGP. Cette variable MIB peut alors être définie pour le lancement (1) ou l'arrêt (2).

Si le démon **gated** n'est pas actif, que le démon **gated** n'est pas configuré pour communiquer avec l'agent **snmpd** ou que le démon **gated** n'est pas configuré pour EGP, les requêtes get et set pour les valeurs de ces variables renvoient le code d'erreur *noSuchName*.

Le fichier de configuration du démon **gated**, **/etc/gated.conf**, doit contenir l'instruction :

```
snmp        yes;
```

Le démon *gated* est configuré en interne pour être un homologue du protocole SMUX (SNMP multiplexing), ou un agent mandataire (proxy) du démon **snmpd**. A son lancement, le démon **gated** enregistre l'arborescence de la variable MIB *ipRouteTable* avec l'agent **snmpd**. Si le démon **gated** est configuré pour EGP, le démon *gated* enregistre également l'arborescence de la variable MIB EGP. Une fois l'enregistrement terminé, un gestionnaire SNMP peut envoyer des requêtes à l'agent **snmpd** concernant les variables MIB *ipRouteTable* d'un EGP, prises en charge par le démon **gated** de l'hôte de cet agent. Ainsi, lorsque le démon **gated** s'exécute, toutes les informations de routage MIB sont obtenues via le démon *gated*. Dans ce cas, les requêtes set pour *ipRouteTable* ne sont pas autorisées.

La communication SMUX entre les démons **gated** et **snmpd** s'effectue via le port TCP (Transmission Control Protocol) identifié 199. Si le démon **gated** doit s'arrêter, **snmpd** désenregistre immédiatement les arborescences précédemment enregistrées par **gated**. Si **gated** démarre avant **snmpd**, **gated** contrôle régulièrement le démon **snmpd** jusqu'à établissement de l'association SMUX.

Pour configurer l'agent **snmpd** pour qu'il reconnaisse et autorise l'association SMUX avec le client du démon **gated**, il faut ajouter une entrée SMUX dans le fichier */etc/***snmpd.conf**. L'identificateur et le mot de passe de l'objet client spécifiés dans cette entrée SMUX pour le démon **gated** doivent correspondre à ceux du fichier **/etc/snmpd.peers**.

L'agent **snmpd** prend en charge les requêtes set pour les variables en lecture-écriture MIB I et MIB II suivantes :

sysContact Identification textuelle de la personne à contacter pour l'hôte de cet agent. Cette information contient le nom de la personne et comment la contacter : par exemple, "Bob Smith, 555–5555, ext 5." La valeur est limitée à 256 caractères. Si, pour une requête set, cette chaîne dépasse 256 caractères, l'agent **snmpd** renvoie l'erreur *badValue*, et l'opération set n'est pas exécutée. La valeur initiale de *sysContact* est définie dans **/etc.snmp.conf**. Valeur par défaut : chaîne nulle.

Instance	Valeur	Action
0	"chaîne"	La variable MIB est définie comme "chaîne".

atN0etAddress Adresse IP correspondant à l'adresse matérielle ou physique spécifiée dans *atPhysAddress*. Il s'agit de la même variable MIB que *ipNetToMediaNetAddress*.

Instance	Valeur	Action
f.1.n.n.n.n	m.m.m.m	Pour l'interface avec *ifIndex f,* une entrée de table ARP existante pour l'adresse IP n.n.n.n est remplacée par l'adresse IP m.m.m.m.

ipForwarding Indique si l'hôte de l'agent achemine les datagrammes.

Instance	Valeur	Action
0	1	Si l'hôte de l'agent possède plusieurs interfaces actives, le noyau TCP/IP est configuré pour l'acheminement des paquets. S'il ne possède qu'une seule interface active, la requête set échoue.
	2	Le noyau TCP/IP sur l'hôte de l'agent est configuré de sorte qu'il n'achemine pas les paquets.

ipDefaultTTL Durée de vie (TTL) par défaut, insérée dans l'en-tête IP des datagrammes générés par l'hôte de l'agent.

Instance	Valeur	Action
0	n	La valeur de durée de vie par défaut, utilisée par le support de protocole IP, est définie comme l'entier n.

ipRouteDest Adresse IP de destination d'une route dans la table des routes.

Instance	Valeur	Action
n.n.n.n	m.m.m.m	La route de destination pour la route n.n.n.n est définie à l'adresse IP m.m.m.m.

ipRouteNextHop
Passerelle par laquelle une adresse IP de destination peut être atteinte par l'hôte de l'agent (entrée de la table des routes).

Instance	Valeur	Action
n.n.n.n	m.m.m.m	Une entrée de la table des routes pour atteindre le réseau n.n.n.n via la passerelle m.m.m.m est ajoutée à la table des routes. La portion hôte de l'adresse IP n.n.n.n doit être égale à 0 pour indiquer une adresse de réseau.

sysName Nom de l'hôte de cet agent. Il s'agit généralement du nom qualifié complet du domaine. La valeur est limitée à 256 caractères. Si, pour une requête set, cette chaîne dépasse 256 caractères, l'agent **snmpd** renvoie l'erreur *badValue*, et l'opération set n'est pas exécutée.

Instance	Valeur	Action
0	"chaîne"	La variable MIB est définie comme "chaîne".

sysLocation Chaîne textuelle indiquant l'emplacement physique de la machine sur laquelle se trouve l'agent **snmpd** : par exemple, "Site Austin, building 802, lab 3C–23." La valeur est limitée à 256 caractères. Si, pour une requête set, cette chaîne dépasse 256 caractères, l'agent **snmpd** renvoie l'erreur *badValue*, et l'opération set n'est pas exécutée. La valeur initiale de *sysLocation* est définie dans **/etc/snmp.conf**. Valeur par défaut : chaîne nulle.

Instance	Valeur	Action
0	"chaîne"	La variable MIB est définie comme "chaîne".

ifAdminStatus État souhaité d'une carte d'interface sur l'hôte de l'agent. Les états possibles sont **actif/inactif**. Un état "test" peut également être défini, mais cette valeur est sans effet sur l'état effectif de l'interface.

Instance	Valeur	Action
f	1	La carte d'interface avec `ifIndex f` est activée.

Remarque : Il est possible que, même si l'état *ifAdminStatus* est défini comme actif ou inactif, le changement effectif d'état n'ait pas eu lieu. Dans ce cas, une requête get de *ifAdminStatus* peut indiquer un état *up* (actif), et un *ifOperStatus* un état *down* (inactif) pour cette interface. Il faut alors que l'administrateur de réseau réémette une requête set de passage de *ifAdminStatus* à l'état actif pour retenter l'opération.

atPhysAddress

Partie matérielle de l'adresse d'une liaison de table d'adresses sur l'hôte de l'agent (entrée de la table ARP (Address Resolution Protocol)). Même variable MIB que *ipNetToMediaPhysAddress.*

Instance	Valeur	Action
f.1.n.n.n.n	hh:hh:hh:hh:hh:hh	Pour l'interface avec **ifIndex f**, toute liaison de table ARP existante pour l'adresse IP n.n.n.n est remplacée par la liaison (n.n.n.n, hh:hh:hh:hh:hh:hh). S'il n'y en a pas, la nouvelle liaison est ajoutée. hh:hh:hh:hh:hh:hh est une adresse matérielle hexadécimale à douze chiffres.

ipRouteType Etat d'une entrée de la table des routes sur l'hôte de l'agent (utilisé pour supprimer des entrées).

Instance	Valeur	Action
h.h.h.h	1	Toute route à destination de l'adresse IP de l'hôte h.h.h.h est supprimée.
n.n.n.n	2	Toute route à destination de l'adresse IP de l'hôte n.n.n.n est supprimée.

ipNetToMediaPhysAddress

Partie matérielle de l'adresse d'une liaison de table d'adresses sur l'hôte de l'agent (entrée de la table ARP). Même variable MIB que *atPhysAddress*.

Instance	Valeur	Action
f.1.n.n.n.n	hh:hh:hh:hh:hh:hh	Pour l'interface avec **ifIndex f**, toute liaison de table ARP existante pour l'adresse IP n.n.n.n est remplacée par la liaison (n.n.n.n, hh:hh:hh:hh:hh:hh). S'il n'y en a pas, la nouvelle liaison est ajoutée. hh:hh:hh:hh:hh:hh est une adresse matérielle hexadécimale à douze chiffres.

ipNetToMediaNetAddress

Adresse IP correspondant à l'adresse matérielle ou physique spécifiée dans *ipNetToMediaPhysAddress*. Même variable MIB que *atNetAddress*.

Instance	Valeur	Action
f.1.n.n.n.n	m.m.m.m	Pour l'interface avec **ifIndex f,** une entrée de table ARP existante pour l'adresse IP n.n.n.n est remplacée par l'adresse IP m.m.m.m.

ipNetToMediaType

Type de mappage de l'adresse IP vers l'adresse physique.

Instance	Valeur	Action
f.1.n.n.n.n	1	Pour l'interface avec **ifIndex f,** pour une liaison ARP existante de l'adresse IP vers l'adresse physique, le type de mappage a la valeur 1, ou autre.
	2	Pour l'interface avec **ifIndex f,** pour une liaison ARP existante de l'adresse IP vers l'adresse physique, le type de mappage a la valeur 2, ou n'est pas valide. Un effet secondaire est que l'entrée correspondante de **ipNetMediaTable** est invalidée, c'est-à-dire que l'interface est dissociée de cette entrée **ipNetToMediaTable.**
	3	Pour l'interface avec **ifIndex f,** pour une liaison ARP existante de l'adresse IP vers l'adresse physique, le type de mappage a la valeur 3, ou dynamique.
	4	Pour l'interface avec **ifIndex f,** pour une liaison ARP existante de l'adresse IP vers l'adresse physique, le type de mappage a la valeur 4, ou statique.

snmpEnableAuthenTraps

Indique si l'agent **snmpd** est configuré de façon à générer des interruptions *authenticationFailure*.

Instance	Valeur	Action
0	1	L'agent **snmpd** générera des interruptions "authentication failure".
	2	L'agent **snmpd** ne générera pas d'interruptions "authentication failure".

smuxPstatus Etat d'un homologue de protocole SMUX (utilisé pour supprimer des homologues SMUX).

Instance	Valeur	Action
n	1	L'agent **snmpd** ne fait rien.
	2	L'agent **snmpd** arrête de communiquer avec l'homologue SMUX n.

smuxTstatus Etat d'une arborescence SMUX (utilisé pour supprimer des montages d'arborescence MIB).

Instance	Valeur	Action
l.m.m.m._ _ _ .p	1	L'agent **snmpd** ne fait rien.
	2	Démonte le montage SMUX de l'arborescence MIB m.m.m... avec / comme longueur d'une instance d'arborescence MIB et **p** la valeur de smuxTpriority.

Les variables ci–après sont définissables via le démon **snmpd**, conformément à RFC 1229. L'unité sous–jacente peut ne pas autoriser leur définition. Vérifiez ce qui est admis dans chaque cas.

ifExtnsPromiscuous

Etat du mode promiscuous sur une unité. Cette opération permet d'activer ou de désactiver le mode promiscuous sur une unité donnée. L'action **snmpd** est finalisée et terminée. Lorsque **snmpd** est instruit de s'arrêter, le mode promiscuous est complètement désactivé, quelles que soient les autres applications sur la machine.

Instance	Valeur	Action
n	1	Active le mode promiscuous pour l'unité n.
	2	Désactive le mode promiscuous pour l'unité n.

ifExtnsTestType

Variable d'initiation de test. Lorsqu'elle est définie, le test approprié est lancé pour cette unité. La valeur de cette variable est un identificateur d'objet. La valeur spécifique dépend du type d'unité et du test à exécuter. Actuellement, FullDiplexLoopBack est le seul test défini que **snmpd** sait exécuter.

Instance	Valeur	Action
n	oid	Lance le test spécifié par oid.

ifExtnsRcvAddrStatus

Variable d'état d'adresse. Lorsqu'elle est définie, l'adresse spécifiée est créée avec un niveau de durée approprié. **snmpd** permet la définition d'une adresse temporaire uniquement, car il est incapable de définir des enregistrements ODM d'unité et qu'il n'est autorisé qu'à définir des adresses multidestinataires/multidiffusion.

Instance	Valeur	Action
n.m.m.m.m.m.m	1	Ajoute l'adresse à titre ni temporaire ni permanent.
	2	Empêche l'utilisation de l'adresse.
	3	Ajoute l'adresse à titre temporaire.
	4	Ajoute l'adresse à titre permanent.

Les variables ci–après sont définissables via le démon **snmpd**, conformément à RFC 1231. L'unité sous–jacente peut ne pas autoriser leur définition. Vérifiez ce qui est admis dans chaque cas.

dot5Commands

Commande que l'unité token-ring doit exécuter.

Instance	Valeur	Action
n	1	Ne fait rien. Renvoyé.
	2	Demande à l'unité token-ring de s'ouvrir.
	3	Demande au token-ring de se réinitialiser.
	4	Demande à l'unité token-ring de se fermer.

dot5RindSpeed

Vitesse ou largeur de bande de l'anneau actuel.

Instance	Valeur	Action
n	1	Vitesse inconnue.
	2	Vitesse d'anneau de 1 mégabits.
	3	Vitesse d'anneau de 4 mégabits.
	4	Vitesse d'anneau de 16 mégabits.

dot5ActMonParticipate

L'objet indique si l'unité doit participer ou non au processus de sélection active du moniteur.

Instance	Valeur	Action
n	1	Doit participer.
	2	Ne doit pas participer.

dot5Functional

Masque fonctionnel permettant à l'unité token-ring de spécifier les adresses à partir desquelles elle recevra des trames.

Instance	Valeur	Action
n	m.m.m.m.m.m	Masque fonctionnel à définir.

Les variables suivantes sont définies dans la consigne RFC comme étant en lecture seule, mais nous vous conseillons de leur affecter des droits en lecture-écriture. Elles concernent des manipulations d'horloge complexes. Etudiez-les attentivement dans RFC pour bien comprendre leurs interactions. **snmpd** permet au demandeur de les définir, mais l'unité ne le pourra peut-être pas. Pour plus d'informations, consultez la documentation relative au pilote de l'unité. Les variables sont :

- dot5TimerReturnRepeat
- dot5TimerHolding
- dot5TimerQueuePDU
- dot5TimerValidTransmit

- dot5TimerNoToken
- dot5TimerActiveMon
- dot5TimerStandbyMon
- dot5TimerErrorReport
- dot5TimerBeaconTransmit
- dot5TimerBeaconReceive

Les variables ci–après sont définissables via le démon SNMP conformément à RFC 1512. Le démon se sert de la norme de protocole FDDI Station Management (SMT) 7.2 pour obtenir des informations. Ceci est déterminé au niveau du microcode. Contrôlez le microcode dans la documentation FDDI pour vérifier que le microcode SMT 7.2 est utilisé.

fddimibSMTUserData

Variable contenant 32 octets d'informations utilisateur.

Instance	Valeur	Action
n	chaîne	Stocke 32 octets d'informations utilisateur.

fddimibSMTConfigPolicy

Etat des politiques de configuration, notamment l'utilisation de la politique "hold" de maintien en l'état.

Instance	Valeur	Action
n	0	Ne pas utiliser la politique "hold".
	1	Utiliser la politique "hold".

fddimibSMTConnectionPolicy

Etat des politiques de connexion dans le noeud FDDI. Voir RFC 1512 pour plus d'informations sur les valeurs définissables spécifiques.

Instance	Valeur	Action
n	k	Définit les politiques de connexion.

fddimibSMTTNotify

Horloge, exprimée en secondes, utilisée dans le protocole Neighbor Notification. Sa valeur est comprise entre 2 et 30 secondes (30 secondes par défaut).

Instance	Valeur	Action
n	k	Définit la valeur de l'horloge.

fddimibSMTStatRptPolicy

Etat de la génération de trames de compte rendu d'état.

Instance	Valeur	Action
n	1	Le noeud génère des trames de compte rendu d'état pour les événements implémentés.
	2	Le noeud ne crée pas de trames de compte rendu d'état.

fddimibSMTTraceMaxExpiration

Cette variable définit la valeur maximale d'expiration de l'horloge pour le suivi.

Instance	Valeur	Action
n	k	Définit l'expiration maximale de l'horloge (en millisecondes).

fddimibSMTStationAction

Cette variable provoque l'exécution par l'entité SMT d'une action spécifique. Pour en savoir plus, voir la RFC.

Instance	Valeur	Action
n	k	Définit une action sur l'entité SMIT. Valeurs comprises entre 1 et 8.

fddimibMACRequestedPaths

Définit les chemins dans lesquels le MAC (medium access control) doit être inséré.

Instance	Valeur	Action
n.n	k	Définit le chemin demandé pour le MAC.

fddimibMACFrameErrorThreshold

Seuil au-delà duquel un compte rendu d'état du MAC doit être généré. Définit le nombre d'erreurs à partir duquel générer un compte rendu.

Instance	Valeur	Action
n.n	k	Définit le nombre d'erreurs à partir duquel générer un compte rendu d'état MAC.

fddimibMACMAUnitdataEnable

Cette variable détermine la valeur de l'indicateur **MA_UNITDATA_Enable** dans RMT. La valeur initiale et par défaut de cet indicateur est "vrai" (1).

Instance	Valeur	Action
n.n	1	Marque l'indicateur MA_UNITDATA_Enable comme vrai (true).
	2	Marque l'indicateur MA_UNITDATA_Enable comme faux (false).

fddimibMACNotCopiedThreshold

Seuil déterminant à quel moment est généré un compte rendu de condition de MAC.

Instance	Valeur	Action
n.n	k	Définit le nombre d'erreurs à partir duquel générer un compte rendu de condition de MAC.

Les trois variables suivantes, interdépendantes, concernent l'horloge. Avant de les modifier, assurez-vous que vous avez bien assimilé leur fonction, telle que définie dans **RFC 1512**.

- fddimibPATHTVXLowerBound
- fddimibPATHTMaxLowerBound
- fddimibPATHMaxTReq

fddimibPORTConnectionPolicies

Spécifie les politiques de connexion pour le port spécifié.

Instance	Valeur	Action
n.n	k	Définit les politiques de con–nexion pour le port spécifié.

fddimibPORTRequestedPaths

Cette variable est la liste des chemins permis du port. Le premier octet correspond à "aucun", le deuxième, à "arborescence", et le troisième, à "homologue".

Instance	Valeur	Action
n.n	ccc	Définit les chemins du port.

fddimibPORTLerCutoff

Estimation du taux d'erreur de liaison au-delà duquel une connexion de liaison sera rompue. La valeur est comprise entre 10**-4 et 10**-15, et est rapportée comme la valeur absolue du logarithme à base 10 (valeur par défaut : 7).

Instance	Valeur	Action
n.n	k	Définit le LerCutoff du port.

fddimibPORTLerAlarm

Estimation du taux d'erreur de liaison au-delà duquel une connexion de liaison génère une alarme. La valeur est comprise entre 10**-4 et 10**-15 et est rapportée comme la valeur absolue du logarithme à base 10 de l'estimation (valeur par défaut : 8).

Instance	Valeur	Action
n.n	k	Définit le LerAlarm du port.

fddimibPORTAction

Cette variable entraîne l'exécution d'une action spécifique par le PORT. Pour en savoir plus, voir la RFC.

Instance	Valeur	Action
n	k	Définit une action sur le port défini. Valeurs comprises entre 1 et 6.

Remarque : RFC 1213 décrit toutes les variables des tables *atEntry* et *ipNetToMediaEntry* comme étant en lecture-écriture. Le support de set n'est assuré que pour les *variables atEntry* aux adresses *atPhysAddress* et *atNetAddress*, et pour les variables *ipNetToMediaEntry* aux adresses *ipNetToMediaPhysAddress*, *ipNetToMediaNetAddress*, et de type *ipNetToMediaType.* Les requêtes set acceptées qui spécifient les autres attributs non acceptés dans ces deux tables sont : *atIfIndex* et *ipNetToMediaIfIndex.* Aucune réponse d'erreur n'est renvoyée à l'émetteur de la requête set, mais la requête get suivante montrera que les valeurs originales sont retenues.

RFC 1213 décrit toutes les variables de la table *ipRouteEntry* comme étant en lecture-écriture, sauf *ipRouteProto*. Comme mentionné ci-dessus, le support de set n'est assuré que pour les variables *ipRouteDest*, *ipRouteNextHop* et ipRouteType. Pour accepter des requêtes set pouvant spécifier plusieurs attributs de route non pris en charge, les requêtes set pour les autres variables de la table *ipRouteEntry* sont acceptées : *ipRouteIfIndex*, *ipRouteMetric1*, *ipRouteMetric2*, *ipRouteMetric3*, *ipRouteMetric4*, *ipRouteMetric5*, *ipRouteAge* et *ipRouteMask*. Aucune réponse d'erreur n'est renvoyée à l'émetteur de la requête set, mais la requête get suivante montrera que les valeurs originales sont retenues. Le démon **snmpd** ne coordonne pas le routage avec le démon **routed**. Si le démon **gated** s'exécute et a enregistré la variable *ipRouteTable* avec le démon **snmpd**, les requêtes set sur *ipRouteTable* ne sont pas autorisées.

RFC 1229 décrit les variables définissables ; **snmpd** permet leur définition. Pour les exceptions, reportez-vous aux entrées précédentes.

Exemples

Les exemples suivants utilisent la commande *snmpinfo*. Le nom de communauté par défaut de snmpinfo, public, est supposé avoir accès en lecture-écriture à la sous-arborescence MIB correspondante :

```
snmpinfo -m set sysContact.0="Primary contact: Bob Smith, office phon
e: 555-5555,
beeper: 9-123-4567. Secondary contact: John Harris, phone: 555-1234."
```

Cette commande affecte à `sysContact.0` la valeur de la chaîne spécifiée. S'il existe déjà une entrée pour `sysContact.0`, elle est remplacée.

```
snmpinfo -m set sysName.0="bears.austin.ibm.com"
```

Cette commande affecte à `sysName.0` la valeur de la chaîne spécifiée. S'il existe déjà une entrée pour `sysName.0`, elle est remplacée.

```
snmpinfo -m set sysLocation.0="Austin site, building 802, lab 3C-23
, southeast
 corner of the room."
```

Cette commande affecte à `sysLocation.0` la valeur de la chaîne spécifiée. S'il existe déjà une entrée pour `sysLocation.0`, elle est remplacée.

```
snmpinfo -m set ifAdminStatus.2=2
```

Désactive la carte d'interface réseau dont l'ifIndex a la valeur 2. Si la valeur affectée est égale à 1, la carte d'interface est activée.

```
snmpinfo -m set atPhysAddress.2.1.192.100.154.2=02:60:8c:2e:c2:00
snmpinfo -m set ipNetToMediaPhysAddress.2.1.192.100.154.2=02:60:8c:
2e:c2:00
```

Changent l'adresse matérielle dans l'entrée de la table ARP de `192.100.154.2` en `02:60:8c:2e:c2:00`. Elles affectent la même entrée de table ARP. La variable MIB *atPhysAddress* est une variable dépréciée, remplacée par la variable MIB *ipNetToMediaPhysAddress*. Donc, *atPhysAddress* et *ipNetToMediaPhysAddress* ont accès à la même structure dans la table ARP du noyau TCP/IP.

```
snmpinfo -m set atNetAddress.2.1.192.100.154.2=192.100.154.3
 snmpinfo -m set ipNetToMediaNetAddress.2.1.192.100.154.2=192.100.154.3
```

Changent l'adresse IP dans l'entrée de la table ARP de `192.100.154.2` en `192.100.154.3`. Elles affectent la même entrée de table ARP. La variable MIB *atNetAddress* est une variable dépréciée, remplacée par la variable MIB *ipNetToMediaNetAddress*. Ainsi, *atNetAddress* et *ipNetToMediaNetAddress* ont accès à la même structure dans la table ARP du noyau TCP/IP.

```
snmpinfo -m set ipForwarding.0=1
```

Définit le noyau TCP/IP de sorte qu'il puisse acheminer les paquets si l'hôte de l'agent a plusieurs interfaces actives. S'il n'en a qu'une, la requête set échoue et l'agent **snmpd** renvoie l'erreur *badValue*.

```
snmpinfo -m set ipDefaultTTL=50
```

Permet à un datagramme IP utilisant la durée de vie (TTL) par défaut de passer par des passerelles (50 maximum) avant d'être rejeté. A chaque traitement du datagramme par une passerelle, cette dernière décrémente de 1 le champ de durée de vie. En outre, chaque passerelle décrémente ce champ du nombre de secondes qu'a attendu le datagramme pour être traité avant d'être transmis à la destination suivante.

```
snmpinfo -m set ipRouteDest.192.100.154.0=192.100.154.5
```

Définit l'adresse IP de destination de la route associée à `192.100.154.0` comme étant `192.100.154.5` (en supposant que la route `192.100.154` existait déjà).

```
snmpinfo -m set ipRouteNextHop.192.100.154.1=129.35.38.47
```

Définit une route vers l'hôte `192.100.154.1` via la passerelle hôte `129.35.38.47` (en supposant que la route `192.100.154.1` existait déjà).

```
snmpinfo -m set ipRouteNextHop.192.100.154.0=192.100.154.7
```

Définit une route vers le serveur de classe C `192.100.154` via la passerelle hôte `192.100.154.7` (en supposant que la route `192.100.154.0` existait déjà). Remarquez que la partie hôte de l'adresse doit être 0 pour indiquer une adresse de réseau.

```
snmpinfo -m set ipRouteType.192.100.154.5=2
```

Supprime toute route pour l'hôte `192.100.154.5`.

```
snmpinfo -m set ipRouteDest.129.35.128.1=129.35.128.1
                        ipRouteType.129.35.128.1=3
                    ipRouteNextHop.129.35.128.1=129.35.128.90
```

Crée une nouvelle route depuis l'hôte `129.35.128.90` jusqu'à `129.35.128.1` comme passerelle.

```
snmpinfo -m set ipNetToMediaType.2.1.192.100.154.11=4
```

Définit l'entrée de la table ARP en `192.100.154.11` comme statique.

```
snmpinfo -m set snmpEnableAuthenTraps=2
```

Indique à l'agent **snmpd** sur l'hôte spécifié de ne pas générer d'interruptions de type authenticationFailure.

```
snmpinfo -m set smuxPstatus.1=2
```

Annule la validité de l'homologue SMUX 1. L'effet secondaire est que la connexion entre l'agent **snmpd** et cet homologue SMUX prend fin.

```
snmpinfo -m set smuxTstatus.8.1.3.6.1.2.1.4.21.0=2
```

Annule la validité ou supprime le montage de l'arborescence SMUX `1.3.6.1.2.1.4.21`, la table *ipRoute*. Le premier nombre de l'instance indique le nombre de niveaux dans l'identificateur d'arborescence SMUX. Le dernier nombre indique la priorité smuxTpriority. Dans cet exemple, il y a 8 niveaux dans l'identificateur d'arborescence SMUX : `1.3.6.1.2.1.4.21`. La priorité, 0, est la plus haute.

```
snmpinfo -m set ifExtnsPromiscuous.1=1 ifExtnsPromiscuous.2=2
```

Active le mode "promiscuous" pour la première unité de la table d'interfaces et le désactive pour la deuxième unité.

```
snmpinfo -m set ifExtnsTestType.1=testFullDuplexLoopBack
```

Lance le test testFullDuplexLoopBack sur l'interface 1.

```
snmpinfo –m set ifExtnsRcvAddrStatus.1.129.35.128.1.3.2=2
```

Indique à l'interface 1 de supprimer l'adresse physique `129.35.128.1.3.2` de la liste des adresses acceptables.

```
snmpinfo –m set dot5Commands.1=2
```

Demande à la première interface d'exécuter une ouverture.

```
snmpinfo –m set dot5RingSpeed.1=2
```

Indique à la première interface de définir sa vitesse d'anneau à 1 mégabit.

```
snmpinfo –m set dot5ActMonParticipate.1=1
```

Indique à la première interface de participer au processus de sélection du moniteur actif.

```
snmpinfo –m set dot5Functional.1=255.255.255.255.255.255
```

Définit le masque d'adresse fonctionnel de sorte que tout soit autorisé.

```
snmpinfo –m set fddimibSMTUserData.1="Greg's Data"
```

Définit les données utilisateur sur la première entité SMT comme "Greg's Data".

```
snmpinfo –m set fddimibMACFrameErrorThreshold.1.1=345
```

Définit le seuil des erreurs de trame à 345 sur le premier MAC de la première entité SMT.

Remarque : Toutes les variables décrites sont définissables par l'une ou l'autre des méthodes indiquées précédemment.

Reportez–vous à Protocole de résolution d'adresses et à adresses Internet pour plus d'informations sur les protocoles et les adresses Internet.

Identification et résolution des incidents liés au démon SNMP

Si l'agent **snmpd** ne se comporte pas comme il le devrait, voici quelques indices pour vous aider à diagnostiquer et corriger le problème. Il est fortement recommandé de démarrer l'agent **snmpd** en spécifiant une journalisation. En cas d'incidents suite à l'appel du démon **snmpd**, il est vivement recommandé de configurer le démon **syslogd** pour une journalisation au niveau de l'utilitaire du démon et de la gravité DEBUG. Reportez–vous à la commande **snmpd** et au fichier **snmpd.conf** pour plus d'informations sur la journalisation **snmpd**.

Interruption prématurée

Si le démon **snmpd** s'arrête dès son appel :

- La cause de l'arrêt est enregistrée dans le fichier journal **snmpd** ou **syslogd** configuré. Consultez ce fichier pour prendre connaissance du message d'erreur **FATAL**.

  Solution : Corrigez le problème et relancez le démon **snmpd**.

- La syntaxe de la ligne de commande **snmpd** est incorrecte. Si vous avez appelé la commande **snmpd** sans SRC (System Resource Controller), la syntaxe requise s'affiche à l'écran. Si vous avez appelé le démon **snmpd** sous SRC (System Resource Controller), la syntaxe requise ne s'affiche pas à l'écran. Consultez le fichier journal pour connaître la syntaxe appropriée.

  Solution : Corrigez la syntaxe de la commande **snmpd** .

- Seul l'utilisateur racine doit appeler le démon **snmpd**. L'agent **snmpd** n'est pas exécuté s'il n'est pas appelé par l'utilisateur racine.

  Solution : Ouvrez une session utilisateur racine et relancez le démon **snmpd**.

- Le fichier **snmpd.conf** doit appartenir à l'utilisateur racine. L'agent **snmpd** vérifie la propriété du fichier de configuration. Si le fichier n'appartient pas à l'utilisateur racine, l'agent **snmpd** s'arrête, ceci étant considéré comme une erreur fatale.

  Solution : Vérifiez que vous êtes connecté en tant qu'utilisateur racine, changez le propriétaire du fichier de configuration et relancez le démon **snmpd**.

- Le fichier **snmpd.conf** doit exister. Si vous n'avez pas spécifié le fichier de journalisation sur la ligne de commande **snmpd** via l'indicateur **–c**, c'est le fichier **/etc/snmpd.conf** qui doit exister. Si vous avez accidentellement supprimé le fichier **/etc/snmpd.conf**, réinstallez l'image **bos.net.tcp.client** ou reconstituez le fichier avec les entrées de configuration adéquates, telles que définies dans la page man du fichier **snmpd.conf**. Si vous aviez vraiment spécifié le fichier de configuration sur la ligne de commande **snmpd** via l'indicateur **–c**, vérifiez que ce fichier existe et qu'il appartient à l'utilisateur racine. Vous devez spécifier le chemin d'accès complet et le nom du fichier de configuration si vous ne voulez pas utiliser le fichier **/etc/snmpd.conf** par défaut.

  Solution : Assurez-vous de l'existence du fichier de configuration spécifié et de son appartenance à l'utilisateur racine. Relancez le démon **snmpd**.

- Il y a déjà une liaison avec le **port udp 161**. Vérifiez que le démon **snmpd** n'est pas déjà en cours d'exécution. Lancez la commande **ps –eaf | grep snmpd** pour déterminer si un processus du démon **snmpd** est déjà en cours. Un seul agent **snmpd** peut effectuer la liaison au **port udp 161**.

  Solution : Tuez l'agent **snmpd** existant ou n'essayez pas de démarrer un autre processus du démon **snmpd**.

Défaillance du démon

Si le démon **snmpd** échoue lorsque vous émettez un signal **refresh** ou **kill -1** :

- La cause de l'arrêt est enregistrée dans le fichier journal **snmpd** ou **syslogd** configuré. Recherchez dans l'un ou l'autre le message d'erreur FATAL.

  Solution : Corrigez le problème et relancez le démon **snmpd**.

- Vérifiez que vous avez spécifié le chemin d'accès complet et le nom du fichier de configuration à l'appel du démon **snmpd**. Le démon **snmpd** "bifurque", passant au répertoire racine lorsqu'il est appelé. Si vous n'avez pas spécifié le nom complet du fichier, l'agent **snmpd** ne peut pas le trouver lors d'un rafraîchissement. Il s'agit d'une erreur fatale qui entraîne l'arrêt prématuré de l'agent **snmpd**.

  Solution : Spécifiez le chemin d'accès complet et le nom du fichier de configuration **snmpd**. Vérifiez qu'il appartient à l'utilisateur racine. Relancez le démon **snmpd**.

- Vérifiez que le fichier de configuration du **snmpd** existe encore. Il peut avoir été malencontreusement supprimé après l'appel de l'agent **snmpd**. Si l'agent **snmpd** ne peut pas l'ouvrir, l'agent **snmpd** s'arrête prématurément.

  Solution : Recréez le fichier de configuration **snmpd**, assurez-vous qu'il appartient à l'utilisateur racine et relancez le démon **snmpd**.

Accès impossible aux variables MIB

Si l'agent **snmpd** ne peut accéder aux variables MIB, ou s'il s'exécute mais que l'application du gestionnaire SNMP (Simple Network Management Protocol) dépasse le délai d'attente d'une réponse de l' agent **snmpd** :

- Vérifiez la configuration réseau de l'hôte sur lequel s'exécute l'agent **snmpd** à l'aide de la commande **netstat –in**. Vérifiez que l'unité lo0, en boucle, est active. Si l'unité n'est pas active, un * (astérisque) est affiché en regard de lo0. Pour que l'agent **snmpd** serve les requêtes, lo0 doit être active.

  Solution : Emettez la commande suivante pour démarrer l'interface de boucle :

  ```
  ifconfig lo0 inet up
  ```

- Vérifiez que le démon **snmpd** a une route conduisant à l'hôte sur lequel vous avez émis les requêtes.

  Solution : Sur l'hôte sur lequel s'exécute le démon **snmpd**, ajoutez une route conduisant à l'hôte sur lequel la requête SNMP a émis la commande **route add**. Reportez-vous à la commande **route**.

- Vérifiez que le nom de l'hôte et son adresse IP sont les mêmes.

  Solution : Redéfinissez le nom de l'hôte pour le faire correspondre à son adresse IP.

- Vérifiez si *localhost* (hôte local) est défini comme adresse IP de lo0.

  Solution : Définissez que *localhost* est à la même adresse que celle utilisée par l'adresse IP de lo0 (généralement `127.0.0.1`).

Accès impossible aux variables MIB dans une entrée de communauté

Si une entrée de communauté est spécifiée dans le fichier de configuration avec un nom de vue MIB, mais qu'il est impossible d'accéder aux variables MIB :

- Vérifiez l'entrée de communauté. Si vous y avez indiqué un nom de vue, tous les champs de cette entrée sont obligatoires.

  Solution : Spécifiez tous les champs de l'entrée de la communauté dans le fichier de configuration. Rafraîchissez l'agent **snmpd** et relancez la requête.

- Assurez-vous que le mode d'accès défini dans l'entrée de la communauté correspond à votre type de requête. Si vous émettez une requête **get** ou **get–next**, vérifiez que la communauté est dotée de droits en lecture seule ou en lecture-écriture. Si vous émettez une requête **set**, vérifiez qu'elle est dotée de droits en lecture-écriture.

  Solution : Corrigez le mode d'accès dans l'entrée de la communauté. Rafraîchissez l'agent **snmpd** et relancez la requête.

- Assurez-vous que vous avez spécifié une entrée de vue correspondant au nom de vue indiqué dans l'entrée de communauté. Faute de quoi, l'agent **snmpd** interdit l'accès à cette communauté. Il est impératif de spécifier une entrée de vue pour une entrée de communautée dans le fichier de configuration.

  Solution : Spécifiez une entrée de vue correspondant au nom de vue indiqué dans l'entrée de la communauté. Rafraîchissez l'agent **snmpd** et relancez la requête.

- Si vous avez spécifié `iso` comme sous-arborescence MIB pour votre entrée de vue, assurez-vous d'avoir indiqué `iso.3`. L'instance de `3` est requise pour que l'agent **snmpd** ait accès à la portion `org` de l'arborescence `iso`.

  Solution : Spécifiez `iso.3` comme sous-arborescence MIB dans l'entrée de vue. Rafraîchissez l'agent **snmpd** et relancez la requête.

- Vérifiez l'*adresse IP* et le *masque de réseau* dans l'entrée de la communauté. Vérifiez que l'hôte à partir duquel vous émettez la requête SNMP est inclus dans la communauté spécifiée.

  Solution : Modifiez les champs *IP address* (adresse IP) et *network mask* (masque de réseau) dans l'entrée de communauté du fichier de configuration pour y inclure l'hôte à partir duquel vous émettez la requête SNMP.

Absence de réponse de l'agent

Si l'*adresse IP* de la communauté est 0.0.0.0, mais que l'agent **snmpd** ne répond pas :

- Vérifiez le champ *network mask* (masque de réseau) dans l'entrée de la communauté. Pour donner un accès général à ce nom de communauté, le champ *network mask* doit avoir la valeur **0.0.0.0**. Si vous avez affecté au champ *network mask* la valeur **255.255.255.255**, vous avez configuré l'agent **snmpd** de façon à interdire toute requête avec le nom de communauté spécifié.

  Solution : Donnez la valeur 0.0.0.0 au champ *network mask* (masque de réseau) de l'entrée de la communauté. Rafraîchissez l'agent **snmpd** et relancez la requête.

- Assurez-vous que le mode d'accès défini dans l'entrée de la communauté correspond à votre type de requête. Si vous émettez une requête **get** ou **get–next**, vérifiez que la communauté est dotée de droits en lecture seule ou en lecture-écriture. Si vous émettez une requête **set**, vérifiez qu'elle est dotée de droits en lecture-écriture.

  Solution : Corrigez le mode d'accès dans l'entrée de la communauté. Rafraîchissez l'agent **snmpd** et relancez la requête.

Message noSuchName

Si, lors d'une tentative de définition d'une variable MIB que l'agent **snmpd** est censé prendre en charge, le message d'erreur **noSuchName** est renvoyé :

La requête set émise n'incluait peut-être pas de nom de communauté correspondant à une communauté autorisée avec un accès en écriture. Le protocole SNMP spécifie qu'une requête set mentionnant une communauté avec des droits d'accès inadéquats doit recevoir en réponse le message d'erreur **noSuchName**.

Solution : Emettez la requête **set** avec le nom d'une communauté dotée de droits d'accès en écriture et comprenant l'hôte à partir duquel est émise la requête **set**.

Chapitre 6. Système de fichiers réseau et SMBFS

Ce chapitre fournit des informations sur NFS (Network File System), mécanisme de stockage des fichiers sur le réseau. Il traite des points suivants :

Pour plus d'informations sur la sécurité NFS, reportez–vous à Network File System (NFS) Security dans le manuel *AIX 5L Version 5.2 Security Guide*.

Système de fichiers NFS : généralités

Le système NFS (Network File System) est un système de fichiers distribués, donnant aux utilisateurs accès aux fichiers et répertoires sur des ordinateurs distants - ils ont ainsi la possibilité de les traiter comme s'il s'agissait de fichiers et répertoires locaux. L'utilisateur dispose des commandes du système d'exploitation pour créer, supprimer, lire, écrire ou définir les attributs de ces répertoires et de ces fichiers.

Le module NFS contient les commandes et démons de NFS, NIS (Network Information Service) et autres services. Mais, bien qu'ils soient installés simultanément, NFS et NIS constituent deux modules distincts, configurés et administrés indépendamment. Reportez–vous au *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide* pour plus de détails sur NIS et NIS+.

Ce système d'exploitation prend en charge les dernières mises à jours du protocole NFS, NFS Version 3 et fournit également une version 2 de NFS client et serveur. Il garantit de la compatibilité ascendante avec les bases d'installation clients et serveurs NFS existantes.

Cette section traite des points suivants :

Services NFS

Les services NFS sont fournis via une relation client-serveur. Les ordinateurs qui rendent leurs *systèmes de fichiers*, leurs *répertoires* et d'autres ressources accessibles à distance sont appelés des *serveurs.* Le fait de rendre ces ressources disponibles est appelé *exportation*. Les ordinateurs, ou les processus qu'ils exécutent, qui utilisent les ressources d'un serveur sont dits *clients*. Lorsqu'un client *monte* un système de fichiers exporté par un serveur, il a accès aux fichiers du serveur (l'accès aux répertoires peut être limité à certains clients).

Les principaux services NFS sont les suivants :

Service Mount	Via le démon **/usr/sbin/rpc.mountd** sur le serveur et la commande **/usr/sbin/mount** sur le client.
Remote File access	Via le démon **/usr/sbin/nfsd** sur le serveur et la commande **/usr/sbin/biod** sur le client.
Service Remote execution	Via le démon **/usr/sbin/rpc.rexd** sur le serveur et la commande **/usr/sbin/on** sur le client.
Service Remote System Statistics	A partir du démon **/usr/sbin/rpc.rstatd** sur le serveur et la commande **/usr/bin/rup** sur le client.
Service Remote User Listing	A partir du démon **/usr/lib/netsvc/rusers/rpc.rusersd** sur le serveur et la commande **/usr/bin/rusers** sur le client.

Service Boot Parameters	Fournit des paramètres d'amorçage aux clients sans disque SunOS via le démon **/usr/sbin/rpc.bootparamd** sur le serveur.
Service Remote Wall	À partir du démon **/usr/lib/netsvc/rwall/rpc.rwalld** sur le serveur et de la commande **/usr/sbin/rwall** sur le client.
Service Spray	Envoie un flot unilatéral de paquets RPC via le démon **/usr/lib/netsvc/spray/rpc.sprayd** sur le serveur et la commande **/usr/sbin/spray** sur le client.
Service PC authentication	Fournit un service d'authentification utilisateur pour PCNFS via le démon **/usr/sbin/rpc.pcnfsd** sur le serveur.

Remarque : Un ordinateur peut être simultanément serveur NFS et client NFS.

Un serveur NFS est *sans état*. C'est-à-dire qu'il n'a à mémoriser aucune information concernant les transactions de ses clients. En d'autres termes, les transactions NFS sont atomiques : une transaction NFS correspond à une et une seule opération complète sur un fichier. C'est le client qui doit mémoriser les informations requises pour les usages ultérieurs de NFS.

Listes de contrôle d'accès (ACL) sous NFS

NFS prend en charge les listes de contrôle d'accès (ACL), mais ceci n'est plus défini par défaut. Pour utiliser les listes de contrôle d'accès avec NFS, spécifiez l'option **acl** avec l'indicateur NFS **–o**, comme illustré dans l'exemple suivant :

```
mount –o acl
```

La prise en charge des ACL est gérée par un programme RPC qui assure l'échange des informations sur ces listes entre clients et serveurs. Le support ACL n'a pas d'incidence sur les spécifications du protocole NFS : il s'agit d'une fonction distincte.

Le système d'exploitation ajoute les ACL au système de fichiers standard. Le protocole NFS standard ne les prenant pas en charge, les ACL ne sont pas visibles des clients NFS standard. Des surprises sont ainsi possibles. Un utilisateur d'un client NFS peut, par exemple, présumer qu'il a accès à un fichier, au vu des bits d'octroi de droits, et se retrouver interdit d'accès car les ACL associées au fichier ont modifié les droits. Les droits sur un serveur étant octroyés selon l'ACL associée au serveur, un utilisateur sur une machine cliente peut donc se voir notifier une erreur relative aux droits d'accès.

Lorsqu'un client tente un premier accès à un système de fichiers monté distant, il commence par essayer de contacter le programme RPC ACL sur le serveur.

S'il s'agit d'un serveur version 3.2, le client consulte l'ACL associée au fichier avant d'accorder le droit d'accès au programme sur le client. Le client réagit alors comme il se doit lorsque la demande est envoyée vers le serveur. En outre, les commandes **aclget**, **aclput** et **alcedit** sont disponibles sur le client pour manipuler les ACL.

Système de fichiers cache (CacheFS)

Le système de fichiers cache (CacheFS) est un mécanisme de cache qui améliore les performances et l'évolutivité du serveur NFS en réduisant la charge du réseau et du serveur. Conçu comme un système de fichiers en couches, CacheFS permet de cacher un système sur un autre. Dans un environnement NFS, CacheFS augmente le taux client-par-serveur, réduit la charge du serveur et du réseau et améliore les performances des liaisons client lentes, telles que le protocole PPP (Point-to-Point Protocol).

Vous créez un cache sur le client de sorte que l'accès aux systèmes de fichiers définis pour être montés dans le cache s'effectue localement et non par le réseau. Lorsqu'un utilisateur demande pour la première fois accès à ces fichiers, ils sont placés dans le cache. Le cache

reste vide tant qu'un utilisateur ne demande pas l'accès à un (ou plusieurs) fichier(s). Les premières requêtes d'accès peuvent sembler lentes, mais les accès suivants au(x) même(s) fichier(s) sont plus rapides.

Remarques :

1. Vous ne pouvez pas cacher les systèmes de fichier **/** (racine) et **/usr**.
2. Vous ne pouvez monter que des systèmes de fichiers partagés. (Reportez-vous à la commande **exportfs**.)
3. Cacher un système de fichiers disque JFS local (Journaled File System) n'apporte aucun gain de performances.
4. Les tâches du tableau suivant sont réservées aux utilisateurs détenant les droits racine ou système.

Tâches CacheFS			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Définir un cache	**cachefs_admin_create**	**cfsadmin –c** *MountDirectoryName*[1]	Logiciel —> **Systèmes de fichiers** —> **Systèmes de fichiers cache** —> **Nouveau système de fichiers cache**.
Spécification des fichiers à monter	**cachefs_mount**	**mount –F cachefs –o backfstype=***FileSysType***,cachedir=***CacheDirectory*[,*options*] *BackFileSystem MountDirectoryName*[2] ou edit **/etc/filesystems**	Logiciel —> **Systèmes de fichiers** —> **Aperçu et tâches** —> **Monter un système de fichiers**.
Modification du cache	**cachefs_admin_change**	supprime le cache, puis le recrée avec les options adéquates de la commande **mount**	Logiciel —> **Systèmes de fichiers** —> **Systèmes de fichiers cache** —> **Sélectionné** —> **Supprimer**. Configure un cache comme à la première rangée de ce tableau.
Affichage des informations du cache	**cachefs_admin_change**	**cfsadmin –l** *MountDirectoryName*	Logiciel —> **Systèmes de fichiers** —> **Systèmes de fichiers cache** —> **Sélectionné** —> **Propriétés**.

Suppres–sion d'un cache	**cachefs_admin_ remove**	1Démontage du système de fichiers **umount** *MountDirectoryName* 2Détermination de l'ID du cache : **cfsadmin –l** *MountDirectoryName* 3Suppression du système de fichiers **cfsadmin –d** *CacheID CacheDirectory*	Logiciel —> **Systèmes de fichiers** —> **Systèmes de fichiers cache** —> **Sélectionné** —> **Supprimer**.
Vérification de l'intégrité du système de fichiers	**cachefs_admin_check**	**fsck_cachefs***Cache Directory*[3]	Logiciel —> **Systèmes de fichiers** —> **Systèmes de fichiers cache** —> **Sélectionné** —> **Vérifier l'intégrité du cache**.

Remarques :

1. Une fois le cache créé, n'exécutez aucune opération à l'intérieur du répertoire cache lui-même (**cachedir**). Vous provoqueriez un conflit à l'intérieur du logiciel CacheFS.
2. Si vous utilisez la commande **mount** pour spécifier les fichiers à monter, vous devez relancer la commande chaque fois que le système est réamorcé.
3. Associez l'option **–m** ou **–o** à la commande **fsck_cachefs** pour vérifier les systèmes de fichiers sans effectuer aucune réparation.

Mappage de fichiers sous NFS

Le mappage de fichiers NFS donne aux programmes d'un client accès à un fichier comme s'il se trouvait en mémoire. Via la sous–routine **shmat**, les utilisateurs peuvent mapper des zones d'un fichier dans leur espace d'adressage : lorsqu'un programme lit ou écrit dans cet espace mémoire, le fichier est copié en mémoire à partir du serveur ou mis à jour sur le serveur.

Le mappage de fichiers sous NFS est limité :

- Le partage des informations entre clients est mal assuré.
- Les modifications apportées à un fichier sur un client via un fichier mappé ne sont pas visibles sur un autre client.
- Verrouiller et déverrouiller des régions d'un fichier est inefficace quant à la coordination des données entre clients.

Si un fichier NFS doit servir au partage de programmes de différents clients, il convient de verrouiller les enregistrements et d'exécuter les sous-routines standard **read** et **write.**

Plusieurs programmes sur un client peuvent partager des données via un fichier mappé. Un verrouillage astucieux d'enregistrement peut coordonner les mises à jour sur le fichier sur le client, sous réserve que l'intégralité du fichier soit verrouillé. Plusieurs clients ne peuvent partager des données via des fichiers mappés que s'il s'agit de données immuables (base de données statique, par exemple).

Types de montage

Il existe trois types de montage :

1. Prédéfini,
2. Explicite
3. Automatique.

Les montages *prédéfinis* sont spécifiés dans le fichier **/etc/filesystems**. Chaque strophe (entrée) de ce fichier définit les caractéristiques d'un montage : elle comprend des données telles que le nom de l'hôte, le chemin d'accès à distance, le chemin d'accès local, etc. Adoptez des montages prédéfinis si l'exploitation d'un client requiert toujours le même type de montage.

Les montages *explicites* sont l'apanage de l'utilisateur racine. Généralement limités à de courtes périodes, ils permettent de répondre à un besoin occasionnel, non planifié. Ils permettent également d'effectuer un montage pour une tâche spéciale, lequel est généralement inaccessible au client NFS. Ces montages sont généralement entièrement qualifiés sur la ligne de commande via l'instruction **mount** assortie de toutes les informations requises. Les montages explicites ne requièrent pas la mise à jour du fichier **/etc/filesystems**. Les systèmes de fichiers explicitement montés le restent tant qu'ils ne sont pas explicitement démontés via la commande **umount** ou que le système n'est pas réinitialisé.

Les montages *automatiques* sont contrôlés par le démon **automount** ; l'extension de noyau **AutoFS** surveille alors l'activité des répertoires spécifiés. Si un programme ou un utilisateur tente d'accéder à un répertoire non monté, le démon **AutoFS** intercepte la demande, monte le système de fichiers, puis répond à la demande.

Processus de montage NFS

Pour accéder aux fichiers du serveur, les clients commencent par monter les répertoires exportés du serveur, sans effectuer une copie de ces répertoires. Le processus de montage utilise en revanche une série d'appels de procédure à distance pour donner à un client accès aux répertoires du serveur de façon transparente. Le processus de montage est le suivant :

1. Lorsque le serveur démarre, le script **/etc/rc.nfs** exécute la commande **exportfs,** laquelle lit le fichier **/etc/exports** du serveur et informe le noyau des répertoires à exporter et des restrictions d'accès qu'ils requièrent.
2. Le démon **rpc.mountd** et plusieurs démons **nfsd** (8, par défaut) sont ensuite lancés par le script **/etc/rc.nfs.**
3. Lorsque le client démarre, le script **/etc/rc.nfs** lance plusieurs démons **biod** (8, par défaut), qui acheminent les demandes de montage client vers le serveur concerné.
4. Le script **/etc/rc.nfs** exécute ensuite la commande **mount**, qui lit les systèmes de fichiers répertoriés dans le fichier **/etc/filesystems.**
5. **mount** repère le(s) serveur(s) exportant les informations demandées par le client et établit la communication avec ce(s) serveur(s). Ce processus est appelé *liaison.*
6. La commande **mount** demande ensuite qu'un ou plusieurs serveurs autorisent le client à accéder aux répertoires inscrits dans le fichier **/etc/filesystems.**
7. Le démon **rpc.mountd** du serveur reçoit les demandes de montage client, et les accorde ou les refuse. Si le répertoire demandé est accessible, **rpc.mountd** envoie au noyau du client un identificateur appelé *descripteur de fichier.*
8. Le noyau client attache ce descripteur au point de montage (répertoire) en enregistrant des informations dans un *enregistrement de montage.*

Une fois le système de fichiers monté, le client peut travailler sur les fichiers. Lorsque le client exécute une opération sur un fichier, le démon **biod** envoie le descripteur du fichier au serveur, où le fichier est lu par l'un des démons **nfsd** pour traiter la demande. Si le client est autorisé à exécuter l'opération demandée, le démon **nfsd** renvoie ensuite les informations requises au démon **biod** du client.

Fichier /etc/exports

Le fichier **/etc/exports** recense tous les répertoires exportés par un serveur à ses clients. Chaque ligne spécifie un seul répertoire. Le serveur exporte automatiquement les répertoires de la liste à chaque lancement du serveur NFS. Ces répertoires exportés peuvent ensuite être montés par les clients. La syntaxe d'une ligne du fichier **/etc/exports** est la suivante :

```
directory     -options[,option]
```

`directory` est le chemin d'accès complet au répertoire. Options désigne soit un indicateur simple, tel que `ro`, soit une liste de noms hôte. Reportez-vous à la documentation du fichier **/etc/exports** et de la commande **exportfs** pour la liste complète des options et leur description. Le script **/etc/rc.nfs** ne lance pas les démons **nfsd** ou le démon **rpc.mountd** si le fichier **/etc/exports** n'existe pas.

Exemple d'entrées d'un fichier **/etc/exports** :

```
/usr/games    -ro,access=ballet:jazz:tap
 /home         -root=ballet,access=ballet
 /var/tmp
 /usr/lib      -access=clients
```

La première entrée indique que le répertoire `/usr/games` peut être monté par les systèmes `ballet`, `jazz` et `tap`. Ces systèmes sont habilités à lire des données et à exécuter des programmes du répertoire, mais ne peuvent y écrire.

La deuxième entrée spécifie que le répertoire `/home` peut être monté par le système `ballet` et que l'accès racine y est autorisé.

La troisième entrée spécifie que n'importe quel client peut monter le répertoire `/var/tmp`. (Notez l'absence de liste d'accès.)

La quatrième entrée spécifie une liste d'accès désignée par le groupe réseau `clients`. En d'autres termes, ces machines désignées comme appartenant au groupe réseau `clients` peuvent monter le répertoire `/usr/lib` à partir de ce serveur. (Un *groupe réseau* est groupe à l'échelle du réseau, ayant accès à certains ressources du réseau à des fins de sécurité ou d'organisation. Les Netgroups sont contrôlés à l'aide du NIS ou du NIS+. Pour plus d'informations, reportez–vous au *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide*.

Fichier /etc/xtab

Le format du fichier **/etc/xtab** est identique à celui du fichier **/etc/exports.** Ce fichier donne la liste des répertoires exportés. A chaque exécution de la commande **exportfs**, le fichier **/etc/xtab** est modifié : vous pouvez ainsi exportez temporairement un répertoire sans avoir à modifier le fichier **/etc/exports**. Si vous annulez l'exportation du répertoire, il est retiré du fichier **/etc/xtab.**

Remarque : Le fichier **/etc/xtab,** dont la mise à jour est automatique, ne doit pas être édité.

Implémentation de NFS

NFS peut être implémenté sur nombre de types de machines, de systèmes d'exploitation et d'architectures réseau. Cette autonomie lui est conférée par le protocole RPC (*Remote Procedure Call*).

Protocole RPC (Remote Procedure Call)

RPC est une bibliothèque de procédures. Ces procédures permettent à un processus (client) de commander à un autre (processus serveur) l'exécution d'appels de procédures, comme s'il les exécutait dans son propre espace d'adressage. Les processus client et serveur étant distincts, ils n'ont pas besoin de résider sur le même système (bien qu'ils le puissent).

NFS est implémenté comme un ensemble d'appels RPC, le serveur prenant en charge certains types d'appels client. Le client lance ces appels sur la base des opérations sur systèmes de fichiers effectuées par le processus client. En ce sens, NFS est une application RPC.

Les processus serveur et client pouvant résider sur des systèmes d'architectures complètement différentes, RPC doit prendre en compte le fait que les données ne sont peut-être pas représentées de la même manière des deux côtés. D'où son adoption du protocole de représentation XDR (eXternal Data Representation).

Protocole XDR (eXternal Data Representation)

XDR est une spécification assurant une représentation standard de différents types de données. Un programme utilisant cette norme ne risque pas de mal interpréter des données, même provenant d'un système doté d'une architecture totalement différente.

Dans la pratique, la plupart des programmes n'utilisent pas XDR en interne. Ils adoptent plutôt la représentation propre à l'architecture du système concerné. Lorsque le programme doit communiquer avec un autre, il convertit ses données au format XDR avant de les envoyer. De même, lorsqu'il reçoit des données, il les convertit du format XDR dans sa propre représentation de données.

Démon portmap

Chaque application RPC est associée avec un numéro de programme et un numéro de version. Ces numéros servent à communiquer avec une application serveur sur un système. Le client, effectuant une demande à partir d'un serveur, doit connaître le numéro du port sur lequel le serveur reçoit les demandes. Ce numéro de port est associé au protocole UDP (User Datagram Protocol) ou TCP (Transmission Control Protocol) utilisé par le service. Le client connaît le numéro du programme, le numéro de version et le nom du système (ou celui de l'hôte sur lequel réside le service). Le client doit pouvoir faire correspondre la paire numéro de programme/numéro de version au numéro de port de l'application serveur. Cette opération est effectuée à l'aide du démon **portmap**.

Le démon **portmap** est exécuté sur le même système que l'application NFS. Lorsque le serveur la lance, il l'enregistre avec **portmap.** Par le biais de cet enregistrement, il fournit son numéro de programme, son numéro de version et son numéro de port UDP ou TCP. Le démon **portmap** maintient une table des applications serveur. Lorsque le client émet une demande vis-à-vis du serveur, il contacte d'abord le démon **portmap** (sur un port identifié) pour connaître le port utilisé par le serveur. Le démon **portmap** répond au client en lui indiquant le port en question. Le client est alors à même d'émettre ses demandes directement à l'application serveur.

Contrôle de NFS

Les démons NFS, NIS et NIS+ sont surveillés par le contrôleur SRC (System Resource Controller). Cela signifie que vous devez utiliser les commandes telles que **startsrc**, **stopsrc** et **lssrc** pour lancer, arrêter et vérifier l'état des démons NFS, NIS et NIS+.

Certains démons NFS ne sont pas contrôlés par SRC, à savoir : **rpc.rexd**, **rpc.rusersd**, **rpc.rwalld** et **rpc.rsprayd**. Ils sont lancés et arrêtés par le démon **inetd**.

Le tableau suivant répertorie les démons et sous-systèmes contrôlés par SRC.

Démons et sous-systèmes associés		
Chemin d'accès au fichier	*Sous-système*	*Groupe*
/usr/sbin/nfsd	**nfsd**	nfs
/usr/sbin/biod	**biod**	nfs
/usr/sbin/rpc.lockd	**rpc.lockd**	nfs
/usr/sbin/rpc.statd	**rpc.statd**	nfs
/usr/sbin/rpc.mountd	**rpc.mountd**	nfs

/usr/lib/netsvc/yp/ypserv	**ypserv**	yp
/usr/lib/netsvc/yp/ypbind	**ypbind**	yp
/usr/lib/netsvc/rpc.yppasswdd	**yppasswdd**	yp
/usr/lib/netsvc/rpc.ypupdated	**ypupdated**	yp
/usr/sbin/keyserv	**keyserv**	keyserv
/usr/sbin/portmap	**portmap**	portmap

Les démons NIS+ sont traités dans *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide*. Chacun de ces démons peut être défini dans les commandes SRC à l'aide de leur nom de sous–système ou de leur nom de groupe approprié. Aucun de ces démons ne prend en charge la liste des fonctions, ni les commandes de suivi SRC.

Pour en savoir plus, reportez-vous à "Contrôleur SRC" dans *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices* .

Modification du nombre de démons biod et nfsd

Pour modifier le nombre de démons **biod** ou **nfsd** actifs, lancez la commande **chnfs**. Ainsi, pour limiter à 10 le nombre de démons **nfsd,** et à 4 le nombre de démons **biod,** entrez :

```
chnfs -n 10 -b 4
```

Cette commande arrête temporairement les démons actifs, modifie le code de la base de données SRC et relance les démons.

Remarque : Dans l'implémentation NFS, le nombre de démons **biod** n'est contrôlable que par point de montage via l'option **biod –o**. La spécification qui utilise **chnfs** n'est maintenue que pour des raisons de compatibilité et n'a pas d'effet sur le nombre réel de routine exécutant les E/S.

Modification des arguments des démons contrôlés par SRC

Nombre de démons NFS, NIS et NIS+ peuvent être assortis d'arguments, sur la ligne de commande, spécifiés une fois le démon activé. Ces démons n'étant pas eux-mêmes activés directement via la ligne de commande, vous devez mettre à jour la base de données SRC pour que les démons puissent être correctement activés. Pour ce faire, lancez la commande **chssys**. La commande **chssys** a le format :

```
chssys -s Daemon -a 'NewParameter'
```

Par exemple :

```
chssys -s nfsd -a '10'
```

modifie le sous-système **nfsd** de sorte que, à l'activation du démon, la ligne de commande soit semblable à `nfsd 10`. La modification induite par la commande **chssys** ne prend effet qu'une fois le sous-système arrêté puis relancé.

Lancement des démons NFS au démarrage du système

Par défaut, les démons NFS ne sont pas activés au cours de l'installation. Celle-ci achevée, tous les fichiers sont placés sur le système, mais les étapes d'activation de NFS ne sont pas effectuées. Vous pouvez lancer les démons NFS au démarrage du système via :

- The Web-based System Manager, **wsm**
- Le raccourci SMIT, **smit mknfs**
- La commande **mknfs**.

Quelle que soit la méthode choisie, une entrée est intégrée au fichier **inittab** de façon que le script **/etc/rc.nfs** soit exécuté à chaque redémarrage du système. A son tour, ce script lance tous les démons requis par un système donné.

Lancement des démons NFS

La taille maximale des fichiers situés sur un serveur NFS est définie par l'environnement du processus au démarrage de **nfsd.** Pour modifier cette valeur, éditez le fichier **/etc/rc.nfs** et insérez-y une commande **ulimit,** indiquant la nouvelle limite, avant la commande **startsrc** relative à **nfsd**.

Les démons NFS peuvent être lancés individuellement ou tous à la fois. Pour les lancer individuellement :

```
startsrc -s Daemon
```

Daemon étant l'un des démons contrôlés par SRC. Ainsi, pour lancer les démons **nfsd** :

```
startsrc -s nfsd
```

Pour les lancer tous simultanément :

```
startsrc -g nfs
```

Remarque : Si le fichier **/etc/exports** n'existe pas, les démons **nfsd** et **rpc.mountd** ne sont pas lancés. Vous pouvez créer un fichier **/etc/exports** vide via la commande **touch /etc/exports** : les démons **nfsd** et **rpc.mountd** seront lancés, mais aucun système de fichiers ne sera exporté.

Arrêt des démons NFS

Les démons NFS peuvent être arrêtés individuellement ou tous à la fois. Pour les arrêter individuellement :

```
stopsrc -s Daemon
```

Daemon étant l'un des démons contrôlés par SRC. Ainsi, pour arrêter **rpc.lockd** :

```
stopsrc -s rpc.lockd
```

Pour les arrêter tous simultanément :

```
stopsrc -g nfs
```

Etat des démons NFS

Vous pouvez afficher l'état de démons NFS spécifiques ou de tous les démons à la fois. Pour afficher l'état d'un démon, entrez :

```
lssrc -s Daemon
```

Daemon étant l'un des démons contrôlés par SRC. Ainsi, pour obtenir l'état de **rpc.lockd** :

```
lssrc -s rpc.lockd
```

Pour afficher simultanément l'état des tous les démons :

```
lssrc -a
```

Installation et configuration de NFS

Pour en savoir plus sur l'installation de NFS (Network File System), reportez-vous à *AIX 5L Version 5.2 Installation Guide*.

Etapes de configuration de NFS

Une fois le logiciel NFS installé sur vos systèmes, il faut le configurer.

1. Déterminez les systèmes du réseau qui seront serveurs, et ceux qui seront clients (un système peut être à la fois serveur et client).
2. Pour chaque système (client ou serveur), suivez les instructions indiquées à "Lancement des démons NFS au démarrage du système".
3. Pour chaque serveur NFS, suivez les instructions indiquées à "Configuration d'un serveur NFS".
4. Pour chaque client NFS, suivez les instructions indiquées à "Configuration d'un client NFS".
5. Si vous souhaitez donner aux PC du réseau accès aux serveurs NFS (outre leur capacité à monter des systèmes de fichiers), configurez PC-NFS comme indiqué à "PC-NFS".

Configuration d'un serveur NFS

Procédez comme suit :

1. Lancez NFS comme indiqué à "Lancement des démons NFS à l'aide de SRC", page 6-10.
2. Créez le fichier **/etc/exports.**

Configuration d'un client NFS

1. Vérifiez que NFS est le système de fichiers distant par défaut. (Sinon, il vous faudra assortir la commande **mount** de l'indicateur **–v nfs**.) A l'aide d'un éditeur de votre choix, ouvrez le fichier **/etc/vfs** et recherchez les entrées suivantes :

   ```
   #%defaultvfs jfs nfs
   #nfs 2 /sbin/helpers/nfsmnthelp none remote
   ```

   Si des signes dièse (#) apparaissent en tête de ligne, effacez-les.
2. Lancez NFS comme indiqué à "Lancement des démons NFS".
3. Définissez le point de montage local via la commande **mkdir**. La réussite d'un montage NFS suppose la présence d'un répertoire servant de point de montage. Ce répertoire doit être vide. La création de ce point de montage ne diffère en rien de celle de n'importe quel répertoire, et aucun attribut particulier ne doit être spécifié.

   Remarque : Les points de montage doivent exister préalablement à tout montage à une exception près : si vous utilisez le démon **automount**, il n'est parfois pas nécessaire de créer des points de montage. Reportez-vous à la documentation **automount**.
4. Etablissez les montages prédéfinis comme indiqué à "Etablissement de montages NFS prédéfinis".

Exportation d'un système de fichiers NFS

Vous pouvez exporter un système de fichiers NFS via l'application Web-based System Manager Network, ou en utilisant l'une des procédures suivantes.

- Via SMIT :

1. Vérifiez que NFS est actif en lançant la commande `lssrc –g nfs`. La sortie doit indiquer que les démons **nfsd** et **rpc.mountd** sont actifs. Dans la négative, lancez NFS comme indiqué à "Lancement des démons NFS", page 6-10.
2. Utilisez

   ```
   smit mknfsexp
   ```

3. Renseignez les zones Chemin d'accès du répertoire à exporter, Mode d'accès au répertoire exporté et Export répert maintenant, init-syst. ou les deux.
4. Modifiez les autres caractéristiques ou acceptez les valeurs par défaut.
5. Vos changements terminés, SMIT met à jour le fichier **/etc/exports**. Si le fichier **/etc/exports** n'existe pas, il est créé.
6. Répétez les étapes 3 à 5 pour chaque répertoire à exporter.

- Pour exporter un système de fichiers NFS via un éditeur :

1. Ouvrez le fichier **/etc/exports** sous votre éditeur favori.
2. Créez une entrée pour chaque répertoire à exporter, en indiquant le chemin d'accès complet du répertoire. Répertoriez tous les répertoires à exporter en commençant à la marge gauche. Ne spécifiez pas de répertoire qui en contient un autre déjà exporté. Pour en savoir plus sur la syntaxe des entrées dans le fichier **/etc/exports**, reportez-vous à la documentation du fichier **/etc/exports**.
3. Sauvegardez et fermez le fichier **/etc/exports**.
4. Si NFS est actif, entrez :

   ```
   /usr/sbin/exportfs –a
   ```

   L'indicateur **–a** indique à la commande **exportfs** d'envoyer au noyau toutes les informations du fichier **/etc/exports.** Si NFS n'est pas actif, lancez-le comme indiqué à "Lancement des démons NFS", page 6-10.

- Pour exporter temporairement un système de fichiers NFS (c'est-à-dire sans modifier le fichier **/etc/exports**), entrez :

  ```
  exportfs –i /dirname
  ```

  dirname étant le nom du système de fichiers à exporter. La commande **exportfs –i** spécifie de ne pas rechercher le répertoire dans le fichier **/etc/exports**, et que toutes les options sont directement issues de la ligne de commande.

Annulation de l'exportation d'un système de fichiers NFS

Vous pouvez annuler l'exportation d'un système de fichiers NFS via l'application Web-based System Manager Network, ou en utilisant l'une des procédures suivantes.

- Via SMIT :

1. Entrez :

   ```
   smit rmnfsexp
   ```

2. Entrez le chemin d'accès dans la zone `Chemin d'accès` du répertoire exporté devant être retiré.

   Le répertoire est supprimé du fichier **/etc/exports** et son exportation annulée.

- Pour annuler l'exportation d'un fichier via un éditeur :

1. Ouvrez le fichier **/etc/exports** sous votre éditeur favori.
2. Repérez l'entrée correspondant au répertoire concerné et effacez la ligne.
3. Sauvegardez et fermez le fichier **/etc/exports**.
4. Si NFS est actif, entrez :

```
exportfs –u dirname
```

dirname étant le chemin d'accès complet au répertoire que vous venez de supprimer du fichier **/etc/exports.**

Modification d'un système de fichiers exporté

Vous pouvez exporter un système de fichiers NFS via l'application Web-based System Manager Network, ou en utilisant l'une des procédures suivantes.

- Via SMIT :

  1. Annulez l'exportation du système de fichiers, **Entrez :**

  ```
  exportfs –u /dirname
  ```

  dirname étant le nom du système de fichiers à modifier.

  2. **Entrez :**

  ```
  smit chnfsexp
  ```

  3. Entrez le chemin d'accès approprié dans la zone Chemin d'accès répert exporté.
  4. Effectuez les modifications souhaitées.
  5. Quittez SMIT.
  6. Réexportez le système de fichiers :

  ```
  exportfs /dirname
  ```

  dirname étant le nom du système de fichiers que vous venez de modifier.

- Pour modifier un système de fichiers via un éditeur, **Entrez :**

  1. Annulez l'exportation du système de fichiers :

  ```
  exportfs –u /dirname
  ```

  dirname étant le nom du système de fichiers à modifier.

  2. Ouvrez le fichier **/etc/exports** sous votre éditeur favori.
  3. Effectuez les modifications souhaitées.
  4. Sauvegardez et fermez le fichier **/etc/exports**.
  5. Réexportez le système de fichiers :

  ```
  exportfs /dirname
  ```

  dirname étant le nom du système de fichiers que vous venez de modifier.

Activation de l'accès racine à un système de fichiers exporté

Lorsque vous exportez un système de fichiers, vous pouvez accorder à l'utilisateur racine les droits d'accès racine à ce système, sur une machine donnée. Par défaut, ces droits ne sont pas accordés. Lorsqu'une personne, connectée en tant qu'utilisateur racine sur un hôte, demande l'accès à un fichier NFS, son ID utilisateur est comparé (par NFS) à l'ID de l'utilisateur `nobody` (nobody étant l'un des noms d'utilisateur inscrits dans le fichier **/etc/password**). Les droits d'accès de l'utilisateur `nobody` sont les mêmes que les droits publics (*autres*) affectés à un fichier donné. Par exemple, si *autres* n'a que le droit d'exécution sur un fichier, `nobody` ne peut qu'exécuter ce fichier.

Pour activer les droits racine sur un système de fichiers exporté, suivez les instructions indiquées dans Modification d'un système de fichiers exporté. Si vous passez par SMIT ou par Web-based System Manager, indiquez dans la zone Hôtes ayant un accès racine, le nom de l'hôte pour lequel vous souhaitez accorder les droits racine. Si vous faites appel à un éditeur, ajoutez le qualificateur `-root=hostname` à l'entrée correspondant au système de fichiers. Par exemple :

```
/usr/tps -root=hermes
```

spécifie que l'utilisateur racine sur l'hôte `hermes` détient des droits d'accès racine au répertoire `/usr/tps`.

Montage explicite d'un système de fichiers NFS

Pour monter explicitement un répertoire NFS, utilisez le raccourci Web-based System Manager **wsm** ou la procédure suivante.

1. Vérifiez que le serveur NFS a exporté le répertoire :

   ```
   showmount -e ServerName
   ```
   `ServerName` étant le nom du serveur NFS. Cette commande affiche le nom des répertoires exportés du serveur NFS. Si le répertoire à monter ne s'y trouve pas, exportez-le.

2. Définissez le point de montage local via la commande **mkdir**. La réussite d'un montage NFS suppose la présence d'un répertoire servant de point de montage. Ce répertoire doit être vide. La création de ce point de montage ne diffère en rien de celle de n'importe quel répertoire, et aucun attribut particulier ne doit être spécifié.

3. Entrez :

   ```
   mount ServerName:/remote/directory /local/directory
   ```
   `ServerName` étant le nom du serveur NFS, `/remote/directory`, le répertoire du serveur NFS que vous souhaitez monter et `/local/directory` le point de montage sur le client NFS.

4. Sur la machine cliente, entrez :

   ```
   smit mknfsmnt
   ```

5. Modifiez les champs suivants en fonction de la configuration de votre réseau. Vous n'aurez peut-être pas à renseigner tous les champs de cet écran.

   Remarque : Si vous utilisez l'interface SMIT, appuyez sur la touche de tabulation pour modifier la valeur d'un champ, mais *n'appuyez pas* sur Entrée avant d'avoir terminé l'étape 7.

   – Chemin d'accès point de montage.
   – Chemin d'accès du répertoire distant.
   – Hôte sur lequel réside le répertoire distant.
   – MONTAGE immédiat, ajout /etc/filesystems ou les 2 ?
   – L'entrée /etc/filesystems entraîne le montage du répertoire lors de l'init-système.
   – Mode d'accès à ce système de fichiers NFS.

6. Conservez les valeurs par défaut ou modifiez-les en fonction de votre configuration NFS.
7. Une fois modifiés les champs requis, SMIT monte le système de fichiers NFS.
8. Lorsque le champ `Commande :` affiche l'état OK, quittez SMIT.

Le système de fichiers NFS est prêt.

Montage automatique d'un système de fichiers à l'aide de AutoFS

AutoFS fait appel à la commande **automount** pour communiquer les informations de configuration pour le montage automatique à l'extension de noyau AutoFS et lance le démon **automountd**. L'extension est alors en mesure de monter automatiquement et de manière transparente le système de fichiers dès qu'un fichier ou un répertoire de ce système de fichiers est ouvert. L'extension informe le démon **autmountd** des requêtes de montage et de démontage, et c'est le **autmountd** qui exécute véritablement le service demandé.

La liaison nom-emplacement étant dynamique dans le démon **automount**, les mises à jour d'une mappe utilisée par le démon **automount** sont transparentes pour l'utilisateur. De ce fait, il est inutile de prémonter les systèmes de fichiers partagés pour les applications dotées de références aux fichiers et aux répertoires codées matériellement. Il est également inutile de maintenir des enregistrements indiquant quels hôtes doivent être montés pour quelles applications.

AutoFS permet de monter les systèmes de fichiers à la demande. Ainsi, ceux montés avec NFS n'ont pas besoin de l'être en permanence.

Pour monter automatiquement un répertoire NFS :

1. Vérifiez que le serveur NFS a exporté le répertoire :

   `showmount –e` *`ServerName`*

   `ServerName` étant le nom du serveur NFS. Cette commande affiche le nom des répertoires exportés du serveur NFS.

2. Création d'un fichier de mappe **AutoFS**. **AutoFS** monte et démonte les répertoires indiqués dans ce fichier de mappe. Supposons par exemple, que vous souhaitez utiliser **AutoFS** pour monter les répertoires `/usr/local/dir1` et `/usr/local/dir2`, comme demandé par le serveur `serve1`, sur les répertoires `/usr/remote/dir1` et `/usr/remote/dir2` respectivement. Le nom du fichier de mappe est ici `/tmp/mount.map`.

   ```
   dir1            –rw              serve1:/usr/local/dir1
    dir2            –rw              serve1:/usr/local/dir2
   ```

3. Vérifiez que l'extension de noyau **AutoFS** est chargée et que le démon **automountd** est en cours d'exécution. Vous disposez pour ce faire de deux méthodes :

   a. Via **SRC** : Lancez **lssrc –s automountd**. Si le sous–système **automountd** ne fonctionne pas, lancez **startsrc –s automountd**.

   b. Via la commande **automount** : Lancez **/usr/bin/automount –v**.

   Définissez le fichier de mappe à l'aide de l'interface de ligne de commande, en tapant :

   `/usr/sbin.automount  /usr/remote  /tmp/mount.map`

   `/usr/remote` étant le point de montage **AutoFS** sur le client. Dès lors, si un utilisateur exécute la commande **cd /usr/remote/dir1**, l'extension de noyau **AutoFS** va intercepter l'accès à ce répertoire et lancer un appel de procédure distante vers le démon **automountd**, qui montera le répertoire`/usr/remote/dir1` et permettra l'exécution de la commande **cd**.

   `/usr/sbin/automount /usr/remote /tmp/mount.map`

   `/usr/remote` étant le point de montage sur le client NFS. Si un utilisateur lance alors la commande **cd /usr/remote/dir1,** le démon **automount** monte le répertoire `/usr/remote/dir1`, permettant l'aboutissement de la commande **cd**.

4. Pour arrêter le démon **automount**, exécutez la commande **stopsrc -s automountd**.

Si, pour une raison quelconque, le démon **automountd** a été lancé sans passer par **SRC** :

kill *automountd_PID*

automountd_PID étant l'ID de processus du démon **automountd**. (Entrez **ps –e** pour afficher l'ID de processus du démon **automountd**.) La commande **kill** envoie un signal SIGTERM au démon **automountd**.

Etablissement de montages NFS prédéfinis

Vous pouvez définir des montages NFS prédéfinis via l'application Web-based System Manager Network, ou en utilisant l'une des procédures suivantes.

Attention : Spécifiez les options **bg** (background) et **intr** (interruptible) dans le fichier **/etc/filesystems** lorsque vous établissez un montage prédéfini à effectuer lors du démarrage du système. Les montages non interruptibles exécutés à l'avant-plan peuvent déconnecter le client pour peu que le réseau ou le serveur soit hors fonction au démarrage du système client. Si un client ne peut accéder au réseau ou à un serveur, l'utilisateur doit relancer la machine en mode maintenance et modifier en conséquence les demandes de montage.

- Pour établir des montages NFS prédéfinis via SMIT :

  1. Entrez :

  ```
  smit mknfsmnt
  ```

  2. Renseignez les champs de cet écran pour chaque montage que vous souhaitez prédéfinir. Vous devez renseigner les champs obligatoires (signalés par un astérisque (*) dans la marge gauche). Pour les autres champs, spécifiez des valeurs ou conservez les valeurs par défaut. Une entrée correspondante est créée dans le fichier **/etc/filesystems**, puis le montage est tenté.

- Pour établir des montages NFS prédéfinis en éditant le fichier **/etc/filesystems** :

  1. Ouvrez le fichier **/etc/filesystems** sous votre éditeur favori.
  2. Insérez-y une entrée pour chaque système de fichiers distant que vous souhaitez monter au démarrage du système. Par exemple :

```
/home/jdoe :
 dev = /home/jdoe
 mount = false
 vfs = nfs
 nodename = mach2
 options = ro,soft
 type = nfs_mount
```

Cette strophe commande au système de monter le répertoire distant `/home/jdoe` sur le point de montage local de même nom. Le système de fichiers est monté en mode lecture seule (`ro`). Etant également monté comme `soft`, une erreur est émise si le serveur ne répond pas. Si vous spécifiez `nfs_mount` pour le paramètre `type`, le système tente de monter le fichier `/home/jdoe` (avec les autres systèmes de fichiers spécifiés dans le groupe `type` = **nfs_mount**) au moment de l'émission de la commande **mount -t nfs_mount**.

L'exemple ci-après commande au système de monter le système de fichiers`/usr/games` au démarrage. Si le montage échoue, le système tente l'opération en arrière-plan.

```
/usr/games :
 dev = /usr/games
 mount = true
```

```
vfs = nfs
nodename = gameserver
options = ro,soft,bg
type = nfs_mount
```

Voici les paramètres requis dans les strophes relatives aux montages NFS :

dev=*filesystem_name*	Chemin d'accès au système de fichiers distant à monter.
mount=[*true*|*false*]	Si *true*, spécifie que le système de fichiers NFS sera monté à l'amorçage du système. Si *false*, spécifie que le système de fichiers NFS ne sera pas monté à l'amorçage du système.
nodename=*hostname*	Hôte sur lequel réside le système de fichiers distant.
vfs=nfs	Le système de fichiers virtuel en cours de montage est de type NFS.

Voici les paramètres facultatifs dans les strophes relatives aux montages NFS :

type= type_name	Définit le système de fichiers en cours de montage et qui appartient au groupe de montage type_name. Ce paramètre est associé à la commande **mount –t**, qui monte simultanément des groupes de systèmes de fichiers.
options= options	Définit un ou plusieurs des paramètres d'option suivants : biods= N Indique le nombre de démons **biod** à démarrer. La valeur par défaut, 6. N, est un entier. bg Indique que le montage doit être relancé en arrière–plan si la première tentative échoue.
	fg Indique que le montage doit être relancé en avant–plan si la première tentative échoue. noacl Désactive, pour le seul montage en cours, la prise en charge des listes ACL, assurée par le système de fichiers journalisé de NFS. Utilisé entre deux systèmes, NFS prend en charge les listes de contrôle des accès (ACL). Si l'option noacl est spécifiée au montage d'un système de fichiers, NFS ne se sert pas des ACL. Les conséquences de l'option noacl sont équivalentes à celles d'un client NFS d'un système qui tente un montage à partir d'un serveur NFS qui ne prend pas les ACL en charge. Pour en savoir plus sur les ACL, reportez–vous à Listes de contrôle d'accès (ACL) sous NFS page 6-3.
	retry= n Nombre de tentatives de montage. rsize= n Définit à n octets la taille du tampon de lecture. wsize= n Définit à n octets la taille du tampon d'écriture.

	`timeo=` n Définit la durée NFS à partir des dixièmes de secondes spécifiées par n. Utilisez cette variable pour éviter les situations pouvant se produire dans les réseaux où la charge du serveur peut causer des temps de réponse inadaptés. `retrans=` n Définit à n le nombre de retransmissions NFS. `port=` n Définit à n le numéro du port du serveur. `soft` Renvoie une erreur si le serveur ne répond pas.

	`hard`
	Relance la requête jusqu'à ce que le serveur réponde.
	Remarque :
	Si vous spécifiez un montage `hard`, il se peut que le processus se bloque pendant l'attente d'une réponse. Pour pouvoir interrompre le processus et le terminer à partir du clavier, spécifiez `intr` dans les paramètres de montage.
	`intr`
	Permet les interruptions à partir du clavier.
	`ro`
	Définit la variable lecture seule.

	`rw`
	Définit la variable en lecture–écriture. Associée à la variable `hard`, elle évite des erreurs de conflit avec des applications si un montage `soft` est tenté en lecture/écriture. Pour en savoir plus sur les incidents liés aux montages hard et soft, reportez–vous à Détermination des problèmes NFS page 6-27.
	`secure`
	Indique qu'un protocole plus sûr doit être utilisé pour les transactions NFS.
	`actimeo=` n
	Augmente de n secondes le délai avant nettoyage du cache, pour les fichiers et les répertoires standard.
	Remarque :
	Le paramètre du cache maintient les attributs de fichier sur le client. Ces attributs sont dotés d'un délai, au bout duquel ils sont effacés. Si un fichier est modifié avant expiration, le délai est augmenté du temps écoulé depuis la dernière modification (les fichiers récemment modifiés sont supposés pouvoir l'être à nouveau rapidement). Un minimum et un maximum sont définis pour l'extension de ce délai (pour les fichiers et les répertoires standard).

	`acregmin=` n Maintient les attributs en mémoire cache au moins n secondes après la modification du fichier. `acregmax=` n Maintient les attributs en mémoire cache au maximum n secondes après la modification du fichier. `acdirmin=` n Maintient les attributs en mémoire cache au moins n secondes après la modification du répertoire. `acdirmax=` n Maintient les attributs en mémoire cache au maximum n secondes après la modification du répertoire.

	Remarque :
	Si vous ne spécifiez pas les options suivantes, le noyau leur affecte une valeur par défaut :
	`biods=6`
	`fg`
	`retry=10000`
	`rsize=8192`
	`wsize=8192`
	`timeo=7`
	`retrans=5`
	`port=NFS_PORT`
	`hard`
	`secure=off`
	`acregmin=3`
	`acregmax=60`
	`acdirmin=30`
	`acdirmax=60`

1. Supprimez les entrées correspondant aux répertoires que vous ne souhaitez pas monter au démarrage du système.
2. Sauvegardez et fermez le fichier.
3. Lancez la commande **mount –a** pour monter tous les répertoires du fichier **/etc/filesystems.**

Démontage d'un système de fichiers monté explicitement ou automatiquement

Entrez :

```
umount /directory/to/unmount
```

Suppression de montages NFS prédéfinis

Vous pouvez supprimer un montage NFS prédéfini via l'application Web-based System Manager Network, ou en utilisant l'une des procédures suivantes.

- Via SMIT :
  1. Entrez :

  ```
  smit rmnfsmnt
  ```

- Pour supprimer un montage NFS prédéfini en éditant le fichier **/etc/filesystems** :
  2. Entrez la commande : umount /directory/to/unmount.
  3. Ouvrez le fichier **/etc/filesystems** sous votre éditeur favori.
  4. Repérez l'entrée correspond au répertoire démonté et effacez-la.

  – Sauvegardez et fermez le fichier.

PC–NFS

PC–NFS est un programme destiné aux ordinateurs personnels, qui leur permet de monter des systèmes de fichiers exportés par un serveur NFS (Network File System). L'ordinateur personnel a également la possibilité de demander à ce serveur NFS des adresses réseau et des noms d'hôte. Par ailleurs, si le serveur NFS exécute le démon **rpc.pcnfsd**, l'ordinateur personnel peut bénéficier des services d'authentification et d'impression différée.

Vous pouvez configurer le démon **rpc.pcnfsd** sur les matériels suivants :

- systèmes exécutant des services d'authentification d'utilisateur ;
- systèmes offrant des fonctions d'impression en différé ;
- tous les serveurs NIS maître et esclaves.

Remarque : Comme la configuration des réseaux NIS prévoit généralement que PC–NFS puisse sélectionner n'importe quel serveur NIS comme serveur par défaut, il est important que tous les serveurs soient dotés du programme **rpc.pcnfsd**. Si l'exécution de ce programme sur tous les serveurs NIS n'est pas envisageable, ou si vous souhaitez confiner les requêtes vers un serveur spécifique, ajoutez une commande **net pcnfsd** dans le fichier **autoexec.bat** de chaque ordinateur personnel, afin de l'obliger à faire appel à un serveur NIS spécifique. Pour plus d'informations, reportez–vous à *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide*.

Service d'authentification PC–NFS

Par défaut, PC–NFS se présente aux serveurs NFS comme étant l'utilisateur `nobody`. Avec les privilèges `nobody`, tous les fichiers des utilisateurs de l'ordinateur personnel sont détenus par `nobody`, et il est impossible de faire la distinction entre les différents utilisateurs. Les fonctions d'authentification du démon **rpc.pcnfsd** permettent de surveiller les ressources système et la sécurité, en autorisant la reconnaissance des différents utilisateurs et l'affectation de différents privilèges.

Lorsque le démon **rpc.pcnfsd** est en cours d'exécution, un utilisateur PC–NFS peut lancer la commande **net name** à partir d'un ordinateur personnel pour ouvrir une session PC–NFS de la même manière qu'un utilisateur se connecte sur ce système d'exploitation. Le nom de l'utilisateur et le mot de passe sont vérifiés par le démon **rpc.pcnfsd**. Cette procédure d'authentification ne rend pas le serveur plus sûr mais elle autorise un meilleur contrôle des accès aux fichiers disponibles via NFS.

Service d'impression en différé PC–NFS

Le service d'impression en différé du démon **rpc.pcnfsd** permet aux ordinateurs personnels exécutant PC–NFS d'imprimer sur des imprimantes qui ne leur sont pas directement raccordées. Plus précisément, PC–NFS redirige les fichiers destinés aux imprimantes de l'ordinateur personnel vers un fichier placé sur un serveur NFS. Ces fichier est placé dans un répertoire de spoulage sur le serveur NFS. Le démon **rpc.pcnfsd** appelle alors la fonction d'impression du serveur. (Le répertoire de spoulage doit se trouver dans un système de fichiers exporté afin que les clients PC–NFS puissent le monter.) Lorsque PC–NFS demande au démon **rpc.pcnfsd** d'imprimer le fichier, il fournit les informations suivantes :

- Nom du fichier à imprimer
- ID d'ouverture de session de l'utilisateur sur le client
- Nom de l'imprimante à utiliser

Configuration du démon rpc.pcnfsd

Pour configurer le démon **rpc.pcnfsd** :

1. Installez le programme PC–NFS sur votre ordinateur personnel.
2. Sélectionnez un emplacement pour le répertoire de spoulage sur le serveur NFS. Le répertoire de spoulage par défaut est **/var/tmp**. Ce répertoire doit disposer d'au moins 100 Ko de mémoire disponible.
3. Exportez le répertoire de spoulage. Ne définissez pas de restrictions d'accès sur le répertoire exporté afin de ne pas engendrer de problèmes d'accès à partir du réseau. Pour plus d'informations sur la procédure, reportez–vous à "Exportation d'un système de fichiers NFS".
4. Lancez le démon **rpc.pcnfsd** en suivant les instructions de "Lancement du démon rpc.pcnfsd".
5. Vérifiez que le démon **rpc.pcnfsd** est accessible en suivant les instructions de "Vérification de la disponibilité du démon rpc.pcnfsd".

Remarque : Les demandes de redirection d'impression laissent parfois dans les répertoires de spoulage PC–NFS des listings de fichiers de longueur nulle ; éliminez donc régulièrement ces entrées du répertoire de spoulage.

Lancement du démon rpc.pcnfsd

Pour lancer le démon **rpc.pcnfsd** à partir du répertoire de spoulage par défaut :

1. A l'aide de votre éditeur de texte favori, annulez la mise en commentaire de l'entrée suivante dans le fichier **/etc/inetd.conf** :

   ```
   pcnfsd sunrpc_udp udp wait root /usr/sbin/rpc.pcnfsd pcnfsd 150001 1
   ```

2. Sauvegardez le fichier et quittez l'éditeur de texte.

Pour lancer le démon **rpc.pcnfsd** à partir d'un répertoire autre que le répertoire par défaut :

1. A l'aide de votre éditeur de texte favori, ajoutez l'entrée suivante dans le fichier **/etc/rc.nfs** :

   ```
   if [ –f /usr/sbin/rpc.pcnfsd ] ; then
    /usr/sbin/rpc.pcnfsd –s spooldir ; echo ' rpc.pcnfsd\c'fi
   ```

   spooldir correspond au chemin d'accès complet du répertoire de spoulage.
2. Sauvegardez le fichier et quittez l'éditeur de texte.
3. A l'aide de votre éditeur de texte favori, mettez en commentaire l'entrée suivante dans le fichier **/etc/inetd.conf** :

   ```
   pcnfsd sunrpc_udp udp wait root /usr/sbin/rpc.pcnfsd pcnfsd 150001 1
   ```

   en insérant un signe dièse (#) au début de la ligne. Ceci évite que le démon **inetd** ne démarre le démon **rpc.pcnfsd** à partir du répertoire de spoulage par défaut.
4. Lancez le programme d'impression en différé du démon **rpc.pcnfsd** en entrant la commande suivante sur la ligne de commande :

   ```
   /usr/sbin/rpc.pcnfsd –s spooldir
   ```

   spooldir correspond au chemin d'accès complet du répertoire de spoulage.

Pour plus d'informations sur la mise à jour de la base de données de configuration **inetd**, reportez–vous à "Configuration du démon inetd", page 4-164.

Remarque : le répertoire par défaut utilisé par le démon **rpc.pcnfsd** ne peut être modifié à partir du fichier **inetd.conf**.

Vérification de la disponibilité du démon rpc.pcnfsd

Pour vérifier que le démon **rpc.pcnfsd** est accessible, entrez :

```
rpcinfo -u host 150001
```

host correspond au nom de l'hôte du système sur lequel vous configurez **rpc.pcnfsd**, et 15001 est le numéro de programme RPC du démon **rpc.pcnfsd**. Après avoir entré la commande, vous devez recevoir un message signalant que le programme est prêt et en attente.

WebNFS

Le système d'exploitation fournit des fonctions de serveur NFS pour WebNFS. Défini par Sun Microsystems, WebNFS est un simple prolongement du protocole NFS permettant de faciliter l'accès aux serveurs et clients par l'intermédiaire de pare–feu Internet.

Un navigateur Web WebNFS peut utiliser les adresses URL universelles NFS pour accéder directement aux données à partir du serveur. Voici un exemple d'URL NFS :

```
nfs://www.  VotreSociété.com/
```

WebNFS fonctionne en conjonction avec les protocoles Web existants afin de mettre les données à la disposition des clients.

WebNFS bénéficie également de l'évolutivité des serveurs NFS.

Gestionnaire NLM (Network Lock Manager)

Le gestionnaire NLM (network lock manager) est un utilitaire qui, associé à NFS, fournit un fichier de consultation et un verrouillage des enregistrements sur le réseau à la manière de System V. Les démons du gestionnaire NLM (**rpc.lockd**) et du contrôleur d'état du réseau (**rpc.statd**) sont les démons de service réseau. Le démon **rpc.statd** est un processus de niveau utilisateur alors que le démon **rpc.lockd** est implémenté comme un ensemble de routines de noyau (semblable au serveur NFS). Les deux démons sont indispensables pour assurer la capacité du noyau à fournir les services réseau fondamentaux.

Remarque : Les verrous obligatoires ou forcés ne sont pas admis sur NFS.

Architecture du gestionnaire NLM

Le gestionnaire NLM comporte des fonctions serveur et des fonctions client. Les fonctions client traitent les demandes émises par les applications et envoient les demandes au gestionnaire NLM sur le serveur. Les fonctions serveur sont chargées d'accepter les requêtes de verrouillage émises par les clients et de générer les appels de verrouillage correspondants sur le serveur. Le serveur répond ensuite à la requête de verrouillage du client.

Contrairement à NFS, qui est "sans état", le gestionnaire NLM est doté d'un état implicite. Autrement dit, il doit mémoriser certaines informations sur le client, à savoir si le client est actuellement verrouillé. Le contrôleur d'état du réseau, **rpc.statd**, implémente un protocole simple qui permet au gestionnaire de verrous de contrôler l'état des autres machines du réseau. Grâce à la précision des informations d'état, le gestionnaire NLM parvient à maintenir un état cohérent dans l'environnement "sans état" de NFS.

Verrouillage des fichiers du réseau

Lorsqu'une application souhaite obtenir un verrou sur un fichier local, elle en adresse la demande au noyau via la sous-routine **lockf**, **fcntl** ou **flock**. Le noyau traite alors la demande. Toutefois, si une application sur un client NFS demande un verrou sur un fichier distant, Le client NFS génère un RPC (Remote Procedure Call) à destination du serveur pour qu'il prenne la requête en charge.

Lorsque le client reçoit la requête de verrou distant pour la première fois, il l'enregistre dans le serveur via le démon **rpc.statd** du client. Il est de même pour le contrôleur de verrouillage du réseau sur le serveur. Il enregistre la première requête d'un client sur le client avec le contrôleur d'état du réseau.

Processus de reprise

Chaque démon **rpc.statd** d'une machine notifie de ses activités les démons **rpc.statd** des autres machines. Lorsque le démon **rpc.statd** d'une machine apprend qu'une machine est en panne ou qu'elle a repris ses activités, il en avertit son démon **rpc.lockd.**

Si un serveur tombe en panne, les clients avec des fichiers verrouillés doivent pouvoir recouvrer leurs verrous. Si un client tombe en panne, son serveur doit conserver ses verrous jusqu'à reprise du client. En outre, pour préserver la transparence globale de NFS, le recouvrement doit s'effectuer sans intervention des applications elles-mêmes.

La procédure de reprise est simple. Si une anomalie est relevée sur un client, le serveur libère les verrous du client en question, en présumant que l'application client les redemandera au besoin. Si une anomalie est relevée sur un serveur, le gestionnaire des verrous client retransmet toutes les demandes de verrous précédemment accordées par le serveur. Ces informations retransmises sont exploitées par le serveur pour reconstituer son état de verrouillage pendant une période dite de grâce. (La période de grâce, de 45 secondes par défaut, est le délai pendant lequel un serveur autorise les clients à réclamer leurs verrous.)

Le démon **rpc.statd** se sert des noms hôte conservés dans **/etc/sm** et dans **/etc/sm.bak** pour garder trace des hôtes à informer lorsque la machine doit recouvrer ses opérations.

Lancement du gestionnaire NLM

Par défaut, le script **/etc/rc.nfs** lance les démons **rpc.lockd** et **rpc.statd** avec les autres démons NFS. Si NFS est déjà actif, vérifiez si **rpc.lockd** et **rpc.statd** sont actifs, comme indiqué à "État des démons NFS", page6-10. Ces deux démons doivent être à l'état *actif*. Si les démons **rpc.lockd** et **rpc.statd** ne sont pas actifs, procédez comme suit :

1. A l'aide de votre éditeur favori, ouvrez le fichier **/etc/rc.nfs.**
2. Repérez les lignes suivantes :

```
if [ -x /usr/sbin/rpc.statd ]; then
    startsrc -s rpc.statd
fi
 if [ -x /usr/sbin/rpc.lockd ]; then
    startsrc -s rpc.lockd
fi
```

3. Si certaines lignes commencent par le signe dièse (#), effacez-le, puis sauvegardez le fichier et quittez l'éditeur. Lancez ensuite successivement les démons **rpc.statd** et **rpc.lockd**, comme indiqué à "Lancement des démons NFS", page 6-10.

   Remarque : L'ordre de lancement est important. Commencez toujours par **statd**.

4. Si NFS est actif et que les entrées du fichier **/etc/rc.nfs** sont correctes, arrêtez puis relancez **rpc.statd** et **rpc.lockd**, comme indiqué à "Arrêt des démons NFS", page 6-10 et à "Lancement des démons NFS", page 6-10.

   Remarque : L'ordre de lancement est important. Commencez toujours par **statd**.

Si les démons **rpc.statd** et **rpc.lockd** ne sont toujours pas actifs, reportez-vous à "Dépannage du gestionnaire NLM".

Dépannage du gestionnaire NLM

Si vous recevez sur un client un message du style :

```
clnttcp_create: RPC: Remote System error - Connection refused
 rpc.statd:cannot talk to statd at {server}
```

c'est que la machine suppose qu'il y a une autre machine qui doit être informée qu'elle doit prendre des mesures de recouvrement. Lorsqu'une machine est réamorcée, ou que **rpc.lockd** et **rpc.statd** sont arrêtés puis relancés, les noms de machine sont déplacés de **/etc/sm** vers **/etc/sm.bak** et **rpc.statd** tente d'informer chaque machine correspondant à chaque entrée de **/etc/sm.bak** que des procédures de reprise s'imposent.

Si **rpc.statd** parvient à atteindre la machine, son entrée dans **/etc/sm.bak** est supprimée. Si **rpc.statd** ne parvient pas à atteindre la machine, il poursuit sa tentative à intervalles réguliers. Chaque fois que la machine échoue à répondre, le message ci-dessus est généré à l'issue du délai de dépassement. Dans l'intérêt de l'intégrité du verrouillage, le démon poursuit ses tentatives, mais ceci peut avoir l'effet inverse sur les performances du verrouillage. La gestion est différente, selon que la machine cible ne répond simplement pas ou se trouve de façon semi-permanente hors état productif. Pour éliminer le message :

1. Vérifiez que les démons **statd** et **lockd** sur le serveur sont lancés, comme indiqué dans la section État des démons NFS. (Ces deux démons doivent être à l'état *actif*.)
2. Si ce n'est pas le cas contraire, lancez les démons **rpc.statd** et **rpc.lockd** sur le serveur, comme indiqué dans la section Lancement des démons NFS.

   Remarque : L'ordre de lancement est important. Commencez toujours par **statd**.

Une fois tous les démons relancés, n'oubliez pas le délai de grâce. Une fois tous les démons relancés, n'oubliez pas le délai de grâce : pendant ce délai, les démons **lockd** autorisent les autres clients à réclamer les verrous conservés précédemment par le serveur. L'obtention d'un nouveau verrou n'est donc pas instantanée.

Pour éliminer le message, vous pouvez également procéder comme suit :

1. Arrêtez **rpc.statd** et **rpc.lockd** sur le client, comme indiqué à "Arrêt des démons NFS".
2. Sur le client, supprimez l'entrée de la machine cible de **/etc/sm.bak**. Entrez :

   ```
   rm /etc/sm.bak/TargetMachineName
   ```

   Ceci action empêche la machine cible d'être informée qu'elle doit peut-être participer au recouvrement du verrouillage : ne l'effectuez que si vous êtes sûr que les applications actives sur cette machine ne participent pas au verrouillage réseau avec la machine affectée.
3. Lancez **rpc.statd** et **rpc.lockd** sur le client, comme indiqué à "Lancement des démons NFS".

Si vous ne parvenez pas à obtenir un verrou d'un client :

1. Lancez la commande **ping** pour vérifier si client et serveur peuvent s'atteindre et se reconnaître. Si les deux machines fonctionnent et que le réseau est intact, vérifiez, dans le fichier **/etc/hosts**, le nom d'hôte de chaque machine. Pour que les machines puissent se reconnaître, ces noms doivent être exactement les mêmes pour le serveur et pour le client. Si un serveur de noms est utilisé pour la conversion des noms d'hôte, vérifiez que les informations hôte sont identiques à celles du fichier **/etc/hosts.**
2. Vérifiez que les démons **rpc.lockd** et **rpc.statd** sont lancés sur le client et sur le serveur, comme indiqué dans la section État des démons NFS. Ces deux démons doivent être à l'état *actif*.
3. S'ils ne le sont pas, lancez les démons **rpc.statd** et **rpc.lockd**, comme indiqué dans la section Lancement des démons NFS.
4. S'ils sont actifs, vous devrez peut-être les réinitialiser sur les clients et les serveurs. Pour ce faire, arrêtez les applications qui demandent un verrou.
5. Ensuite, arrêtez **rpc.statd** et **rpc.lockd** sur le client et sur le serveur, comme indiqué à "Arrêt des démons NFS", page 6-10.
6. Relancez ensuite **rpc.statd** et **rpc.lockd**, d'abord sur le serveur, puis sur le client, comme indiqué à "Lancement des démons NFS".

   Remarque : L'ordre de lancement est important. Commencez toujours par **statd**.

Si le problème de verrou persiste, exécutez le démon **lockd** en mode débogage :

1. Arrêtez **rpc.statd** et **rpc.lockd** sur le client et sur le serveur, comme indiqué à "Arrêt des démons NFS".
2. Lancez le démon **rpc.statd** sur le client et sur le serveur, comme indiqué à "Lancement des démons NFS", page 6-10.
3. Lancez **rpc.lockd** sur le client et sur le serveur :

   ```
   /usr/sbin/rpc.lockd –d1
   ```

   Appelé avec l'indicateur **–d1**, le démon **lockd** génère des messages de diagnostic vers syslog. Les premiers messages concernent le délai de grâce : attendez qu'ils s'effacent. Exécutez ensuite l'application problématique et vérifiez qu'une demande de verrou est bien transmise du client au serveur et du serveur au client.

Identification des incidents NFS

Les machines utilisant NFS, comme tout service de réseau, ne sont pas à l'abri d'incidents. Pour résoudre ces défaillances, il faut être en mesure de les identifier, d'interpréter les messages d'erreur et de déterminer la méthode de résolution appropriée. Il s'agit dans un premier temps de localiser le dysfonctionnement sur l'un des trois principaux éléments : serveur, client ou réseau.

Remarque : Reportez-vous à "Dépannage du gestionnaire NLM", page 6-25.

Inaccessibilité des fichiers en montage fixe ou logiciel

En cas de défaillance du réseau ou serveur, les programmes ne parviennent plus à accéder aux fichiers distants, mais ce mécanisme diffère selon que le fichier fait l'objet d'un montage fixe ou logiciel.

Dans le cas d'un montage fixe, NFS signale l'échec du serveur par le message :

```
NFS server hostname not responding, still trying
```

Les systèmes de fichiers distants en montage fixe mettent les programmes en attente jusqu'à la réponse du serveur de sorte que le client peut relancer la demande de montage jusqu'à obtenir satisfaction. Pour un montage fixe, associez l'indicateur **–bg** à la commande **mount** pour que le client puisse tenter le montage en arrière-plan si le serveur ne répond pas.

Dans le cas d'un montage logiciel, NFS signale l'échec du serveur par le message :

```
Connection timed out
```

Passé un certain délai, en cas de tentatives infructueuses, les systèmes de fichiers distants en montage logiciel renvoient un message d'erreur. Mais un grand nombre de programmes ne vérifient pas le résultat des opérations sur systèmes de fichiers. Ce message d'erreur ne vous est alors pas communiqué au moment d'accéder aux fichiers en montage logiciel. Il est toutefois affiché à la console.

Liste de contrôle pour l'identification des incidents NFS

En cas d'incident sur un client NFS :

1. Vérifiez les connexions au réseau.
2. Vérifiez que les démons **inetd**, **portmap** et **biod** sont exécutés sur le client comme indiqué dans la section "État des démons NFS", page 6-10.
3. Vérifiez qu'un point de montage valide est disponible pour le système de fichiers en cours de montage. Pour plus d'informations, reportez-vous à "Configuration d'un client NFS", page 6-11.
4. Vérifiez que le serveur fonctionne en exécutant, à partir de l'invite du shell, côté client, la commande suivante :

   ```
   /usr/bin/rpcinfo -p server_name
   ```

   Si le serveur fonctionne, la liste des programmes, versions, protocoles et ports s'affiche comme suit :

   ```
   program    vers  proto      port
   100000        2      tcp        111    portmapper
   100000        2      udp        111    portmapper
   100005        1      udp      1025    mountd
   100001        1      udp      1030    rstatd
   100001        2      udp      1030    rstatd
   100001        3      udp      1030    rstatd
   100002        1      udp      1036    rusersd
   100002        2      udp      1036    rusersd
   ```

```
100008        1       udp       1040    walld
100012        1       udp       1043    sprayd
100005        1       tcp         694    mountd
100003        2       udp       2049    nfs
100024        1       udp         713    status
100024        1       tcp         715    status
100021        1       tcp         716    nlockmgr
100021        1       udp         718    nlockmgr
100021        3       tcp         721    nlockmgr
100021        3       udp         723    nlockmgr
100020        1       udp         726    llockmgr
100020        1       tcp         728    llockmgr
100021        2       tcp         731    nlockmgr
```

Sinon, connectez-vous au serveur à partir de la console du serveur et vérifiez l'état du démon **inetd** comme indiqué à "État des démons NFS", page 6-10.

5. Vérifiez que les démons **mountd**, **portmap** et **nfsd** sont actifs sur le serveur NFS en spécifiant à partir de l'invite du shell les commandes :

   ```
   /usr/bin/rpcinfo  -u  server_name mount
    /usr/bin/rpcinfo  -u  server_name portmap
    /usr/bin/rpcinfo  -u  server_name nfs
   ```

   Si le démon est exécuté au niveau du serveur, les réponses renvoyées sont les suivantes :

   ```
   program  100005  version  1  ready  and  waiting
    program  100000  version  2  ready  and  waiting
    program  100003  version  2  ready  and  waiting
   ```

   Les numéros de programme correspondent aux commandes, comme indiqué dans l'exemple ci-dessus. Sinon, connectez-vous au serveur à partir de la console du serveur et vérifiez l'état des démons comme indiqué à "État des démons NFS", page 6-10.

6. Vérifiez que le fichier **/etc/exports** sur le serveur comporte le nom du système de fichiers que le client souhaite monter et que ce système de fichiers est exporté. Pour ce faire, entrez :

   ```
   showmount -e server_name
   ```

   Cette commande affiche la liste de tous les systèmes de fichiers exportés par *server_name*.

Erreurs d'écriture asynchrone

Lorsqu'un programme d'application inscrit des données dans un fichier appartenant à un système de fichiers monté NFS, l'écriture est planifiée pour être traitée en mode asynchrone par le démon **biod**. Si une erreur se produit au niveau du serveur NFS pendant l'écriture sur le disque, elle est signalée au client NFS et le démon **biod** la sauvegarde en interne dans les structures de données NFS. L'erreur enregistrée est ensuite renvoyée au programme d'application dès qu'il fait appel aux fonctions **fsync** ou **close**. Autrement dit, l'application n'est avertie de l'erreur que lorsque le programme ferme le fichier.
Cet événement survient généralement lors de la saturation d'un système de fichiers sur le serveur, toute tentative d'écriture entreprise par le client sur ce système étant vouée à l'échec.

Messages d'erreur NFS

Voici les messages d'erreur qui peuvent être générés lors de l'utilisation de NFS.

Message d'erreur nfs_server

Un nombre insuffisant de tampons de transmission peut entraîner le message d'erreur :

```
nfs_server:  bad  sendreply
```

Pour augmenter le nombre de tampons, vous disposez du raccourci Web-based System Manager **wsm** ou du raccourci SMIT **smit commodev**. Sélectionnez ensuite votre type de carte et augmentez le nombre de tampons de transmission.

Messages d'erreur mount

Plusieurs causes peuvent provoquer l'échec d'un montage à distance. Les messages d'erreur associés aux échecs de montage sont les suivants :

`mount: ... already mounted`
: Le système de fichiers que vous tentez de monter l'est déjà.

`mount: ... not found in /etc/filesystems`
: Le système de fichiers ou le répertoire spécifié est introuvable.

Si vous exécutez la commande **mount** en spécifiant soit un répertoire soit un système de fichiers, et non les deux, la commande recherche dans le fichier **/etc/filesystems** l'entrée qui comporte le système de fichiers ou le répertoire correspondant. Si la commande **mount** trouve une entrée de la forme :

```
/dancer.src:
                dev=/usr/src
     nodename  =  d61server
     type  =  nfs
     mount  =  false
```

elle exécute le montage comme si vous aviez spécifié sur la ligne de commande :

```
/usr/sbin/mount –n dancer –o rw,hard /usr/src /dancer.src
```

`... not in hosts database`
: Emis sur un réseau sans NIS, ce message indique que l'hôte spécifié à la commande **mount** ne figure pas dans le fichier **/etc/hosts**. En revanche, sur un réseau exécutant NIS, ce message indique que NIS n'a pas pu trouver le nom d'hôte dans la base de données **/etc/hosts** ou que le démon NIS **ypbind** de votre machine n'est plus actif. Si le fichier **/etc/resolv.conf** existe, la résolution des noms d'hôte se fait via le serveur de noms. Le problème peut alors provenir de la base de données **named**.
Reportez-vous à "Résolution de noms sur un serveur NFS", page6-34.

Vérifiez le libellé et la syntaxe de la commande **mount**. Si la commande est correctement spécifiée, vous pouvez en déduire que votre réseau n'exécute pas NIS et que ce message ne concerne que ce nom d'hôte. Vérifiez l'entrée correspondante dans le fichier **/etc/hosts.**

Si votre réseau exécute NIS, assurez–vous que le démon **ypbind** s'exécute en spécifiant à partir de la ligne de commande :

```
ps  –ef
```

Le démon **ypbind** doit apparaître dans la liste. Lancez la commande **rlogin** pour tenter de vous connecter à une autre machine distante ou la commande **rcp** pour effectuer une copie à distance sur une autre machine. Si ces opérations échouent, il est probable que votre démon **ypbind** est arrêté ou bloqué.

Si le message ne concerne que ce nom d'hôte, vérifiez l'entrée **/etc/hosts** sur le serveur NIS.

`mount: ... server not responding: port mapper failure - RPC timed out`
Soit le serveur à partir duquel le montage est effectué est hors service, soit le programme de mappage de ports est arrêté ou bloqué. Tentez de redémarrer le serveur pour activer les démons **inetd**, **portmap** et **ypbind.**

Si la connexion au serveur à distance avec la commande **rlogin** échoue, mais que le serveur fonctionne, testez la connexion réseau en vous connectant à une autre machine distante. Vérifiez également la connexion réseau du serveur.

`mount: ... server not responding: program not registered`
La commande **mount** a contacté le programme de mappage de port, mais le démon de montage NFS **rpc.mountd** NFS n'était pas répertorié.

`mount: accès refusé ...`
Le nom de votre machine ne figure pas dans la liste d'exportation du système de fichiers que vous tentez de monter à partir du serveur.

Pour obtenir la liste des systèmes de fichiers exportés du serveur, exécutez à partir de la ligne de commande:

`showmount -e` *`hostname`*

Si le système de fichiers recherché n'est pas répertorié ou que le nom de votre machine ou groupe de réseau ne figure pas dans la liste des utilisateurs du système de fichiers, connectez–vous au serveur et recherchez dans le fichier **/etc/exports** l'entrée correcte pour le système de fichiers. Un nom de système de fichiers apparaissant dans **/etc/exports** mais absent de la sortie de la commande **showmount**, révèle une défaillance du démon **mountd**. Soit le démon n'a pas pu analyser cette ligne dans le fichier, soit il n'a pas trouvé le répertoire, soit le répertoire spécifié n'a pas été monté localement. Si le fichier **/etc/exports** semble correct et que le réseau exécute NIS, vérifiez le démon **ypbind** sur le serveur. Il est peut–être arrêté ou bloqué. Pour plus d'informations, consultez le manuel *AIX 5L Version 5.2* Network Information Service *(NIS and NIS+) Guide*.

`mount: ...: Permission denied`
Ce message signale tout échec d'authentification sur le serveur. Il peut s'afficher, dans l'exemple précédent, si vous ne figurez pas dans la liste d'exportation, si le serveur n'a pas reconnu le démon **ypbind** sur votre machine ou si le serveur refuse l'identité que vous lui avez soumise.

Vérifiez le fichier **/etc/exports** du serveur et, le cas échéant, le démon **ypbind.** Dans ce cas, changez simplement votre nom d'hôte à l'aide de la commande **hostname** et relancez la commande **mount**.

`mount: ...: Not a directory`
Le chemin d'accès à distance ou local n'est pas un répertoire. Vérifiez le libellé de la commande et lancez l'exécution sur chaque répertoire.

`mount: ...: You are not allowed`
Vous devez disposer des droits d'utilisateur racine ou être membre du groupe système pour exécuter la commande **mount** sur votre machine, car cette commande affecte le système de fichiers pour tous les utilisateurs sur cette machine. Seuls les utilisateurs racine et les membres du groupe système sont habilités à effectuer des montages et des démontages NFS.

Problèmes de temps d'accès à NFS

Si l'accès aux fichiers distants semble anormalement lent, recherchez les causes possibles. Il peut s'agir par exemple d'un démon incontrôlable ou d'une ligne **tty** erronée.

Vérification des processus

Sur le serveur, entrez :

```
ps  -ef
```

Si le serveur fonctionne normalement et que d'autres utilisateurs obtiennent les réponses dans des délais satisfaisants, vérifiez que votre démon **biod** est actif. Procédez comme suit :

1. Exécutez la commande **ps –ef** et recherchez les démons **biod** dans la sortie.

   Si ces démons sont arrêtés ou bloqués, passez aux étapes 2 et 3.

2. Arrêtez les démons **biod** actifs :

   ```
   stopsrc -x biod -c
   ```

3. Lancez les démons **biod** :

   ```
   startsrc -s biod
   ```

Pour déterminer si les démons **biod** sont bloqués, exécutez plusieurs fois la commande **nfsstat –c** pendant la période qui correspond selon vous à un blocage des démons **biod**. If there is no noticeable change in the number of Remote Procedure Call (RPC) client reads or writes, one or more of the **biod** daemons are not performing their task. Vous pouvez constater qu'un ou plusieurs démon(s) **biod** est/sont bloqué(s), mais pas déterminer précisément lequel est inactif.

Vérification des connexions réseau

Si les démons **biod** fonctionnent, contrôlez les connexions au réseau. La commande **nfsstat** vérifie si des paquets sont perdus. Utilisez les commandes **nfsstat –c** et **nfsstat –s** pour savoir si un client ou un serveur retransmet des blocs de grande taille. En effet, des retransmissions sont toujours possibles lorsque des paquets ont été perdus ou que les serveurs sont occupés. Un taux de retransmission de 5 % est considéré comme élevé.

La probabilité de retransmissions peut être réduite en modifiant les paramètres des files d'attente de transmission des cartes de communication. Pour cette opération, vous pouvez utiliser SMIT (System Management Interface Tool).

Les valeurs recommandées pour les serveurs NFS sont :

Carte	UTM	File d'attente de transmission
anneau à jeton		
4 Mo	1500	50
16 Mo	3900	40 (Augmentez la valeur si la commande **nfsstat** dépasse le délai imparti.)
	1500	40 (Augmentez la valeur si la commande **nfsstat** dépasse le délai imparti.)
	8500	40 (Augmentez la valeur si la commande **nfsstat** dépasse le délai imparti.)
Ethernet	1500	40 (Augmentez la valeur si la commande **nfsstat** dépasse le délai imparti.)

Les tailles UTM maximales pour chaque vitesse d'anneau à jeton réduisent l'utilisation du processeur et favorisent considérablement les opérations de lecture/écriture.

Remarques :

1. Appliquez ces valeurs aux clients NFS si les retransmissions se poursuivent.
2. Tous les nœuds d'un réseau doivent utiliser la même taille UTM.

Taille UTM

Pour définir la taille MTU, utilisez le raccourci Web-based System Manager, **wsm**, ou le raccourci SMIT, **smit chif**. Sélectionnez la carte qui convient et indiquez une valeur UTM dans le champ Maximum IP Packet Size.

Vous pouvez également utiliser la commande **ifconfig** (qui est obligatoire pour fixer la taille UTM à 8500). Le format de la commande **ifconfig** est le suivant :

```
ifconfig tr n NodeName up mtu MTUSize
```

tr *n* étant le nom de votre carte, `tr0`, par exemple.

Enfin, vous pouvez combiner les deux méthodes : SMIT et la commande **ifconfig**.

1. Ajoutez la commande **ifconfig** pour les anneaux à jeton, comme indiqué dans l'exemple précédent, au fichier **/etc/rc.bsdnet**.
2. Entrez le raccourci **smit setbootup_option**. Faites basculer le champ Utilisation d'une configuration rc de style BSD sur **oui.**

Tailles de files d'attente de transmission

La taille des files d'attente de transmission des cartes de communication se définit à l'aide de SMIT. Entrez le raccourci **smit chgtok**, sélectionnez la carte concernée et indiquez une taille de file d'attente dans le champ Transmit.

Intervention sur programmes bloqués

Si des programmes bloquent au cours d'un travail sur un fichier, le serveur NFS est peut être arrêté. Dans ce cas, le message d'erreur suivant s'affiche :

```
NFS server hostname not responding, still trying
```

Le serveur NFS (`hostname`) est en panne. Ceci révèle un problème avec le serveur NFS, la connexion réseau ou le serveur NIS.

Vérifiez les serveurs à partir desquels vous avez monté les systèmes de fichiers si votre machine est complètement bloquée. Si un ou plusieurs d'entre eux sont hors service, il n'y a pas lieu de s'inquiéter. Les programmes se poursuivront automatiquement dès la remise en service des serveurs. Aucun fichier n'est détruit.

Si un serveur en montage logiciel expire, les autres travaux ne sont pas concernés. Les programmes qui dépassent le délai en tentant d'accéder aux fichiers distants en montage logiciel n'aboutissent pas et le message `errno` s'affiche. Mais vous pouvez toujours accéder aux autres systèmes de fichiers.

Si tous les serveur fonctionnent, déterminez si les autres personnes utilisant les mêmes serveurs rencontrent également des difficultés. Si plusieurs machines sont dans ce cas, l'anomalie provient des démons **nfsd** du serveur. Dans ce cas, connectez-vous au serveur et exécutez la commande **ps** pour voir si le démon **nfsd** s'exécute et accumule du temps CPU. Sinon, tentez d'arrêter et de relancer le démon **nfsd**. Si le problème persiste, réamorcez le serveur.

Si les autres systèmes semblent en service et fonctionner normalement, vérifiez votre connexion réseau et la connexion du serveur.

Droits d'accès et authentification

Une fois les montages correctement effectués, vous pouvez rencontrer des difficultés de lecture, écriture ou création de fichiers ou répertoires distants. Ce type de difficultés est généralement dû à des problèmes de droits ou d'authentification. La cause de ces problèmes de droit ou d'authentification dépendent du NIS utilisé ou de la protection appliquée aux montages.

Dans le cas de figure le plus simple, les montages ne sont pas sécurisés et NIS n'est pas utilisé. Dans ce cas, les ID utilisateur (UID) et les ID groupe (GID) sont alors mappés par le seul biais des fichiers serveur et clients **/etc/passwd** et **/etc/group**, respectivement. Dans ce cas, pour qu'un utilisateur appelé `john` soit identifié comme `john` sur le client et sur le serveur, l'utilisateur `john` doit disposer dans le fichier **/etc/passwd** du même ID utilisateur. En voici un contre-exemple :

```
User  john  is  uid  200  on  client  foo.
User  john  is  uid  250  on  server  bar.
User  jane  is  uid  200  on  server  bar.
```

Le répertoire `/home/bar` est monté à partir du serveur `bar` sur le client `foo`. Si l'utilisateur `john` édite des fichiers sur le système de fichiers distant `/home/bar` du client `foo`, la sauvegarde des fichiers générera des confusions.

Pour le serveur `bar`, les fichiers appartiennent à `user jane`, car l'ID de `jane` est `200` sur `bar`. Si `john` se connecte directement à `bar` par la commande **rlogin**, il est probable qu'il ne pourra pas accéder aux fichiers qu'il vient de créer en travaillant sur le système de fichiers monté à distance. En revanche, `jane` peut y accéder car les machines gèrent les droits d'accès par UID et non par nom.

La seule solution pour résoudre ce problème durablement est de réaffecter des UID cohérents sur les deux machines. Par exemple, attribuez à `john` l'UID 200 sur le serveur `bar` ou 250 sur le client `foo`. Il faut alors appliquer aux fichiers appartenant à `john` la commande **chown** pour que les nouveaux ID leur soient attribués sur les machines correspondantes.

En raison des problèmes de maintien de mappages UID et GID cohérents sur toutes les machines d'un réseau, NIS ou NIS+ est souvent utilisé pour pallier ce type de problème. Reportez–vous à *AIX 5L Version 5.2 NIS/NIS+ (Network Information Services) Guide* pour plus d'informations.

Résolution des noms sur un serveur NFS

Lorsqu'un serveur NFS prend en compte une requête de montage, il recherche le nom du client demandeur. Le serveur recherche, sur la base de l'adresse IP (Internet Protocol) du client, le nom hôte correspondant à cette adresse. Muni de ce nom, le serveur consulte la liste d'exportation du répertoire demandé et vérifie que le client est cité dans la liste d'accès au répertoire. S'il trouve une entrée relative au client qui corresponde exactement au résultat de la résolution de noms, l'authentification est accordée à ce niveau.

Si le serveur ne parvient pas à résoudre l'adresse IP en nom d'hôte, il rejette la demande de montage. Le serveur doit être en mesure de trouver une correspondance pour l'adresse IP du client demandeur. Si le répertoire est exporté avec une autorisation d'accès accordée à tous les clients, le serveur peut effectuer la résolution inverse pour accepter la demande de montage.

Le serveur doit également retrouver le nom exact du client. Considérons par exemple une entrée du fichier **/etc/exports** telle que :

```
/tmp      -access=silly:funny
```

A cette entrée correspondent, dans le fichier **/etc/hosts**, les entrées suivantes :

```
150.102.23.21        silly.domain.name.com
150.102.23.52        funny.domain.name.com
```

Remarquons que les noms ne correspondent pas exactement. Lorsque le serveur recherche les équivalences adresse IP– nom d'hôte pour les hôtes `silly` et `funny`, il ne retrouve pas exactement les mêmes chaînes de nom dans la liste d'accès d'exportation. Ce phénomène se produit généralement lorsque la résolution de noms est effectuée par le démon **named**. En effet, la plupart des bases de données du démon **named** contiennent des alias des noms complets des hôtes pour simplifier la tâche de l'utilisateur lors de la spécification des hôtes. Bien que des entrées noms hôte-adresses IP existent pour les alias, la recherche inversée peut ne pas être définie. La base de données pour la recherche inversée (adresse IP – nom hôte) contient généralement des entrées indiquant l'adresse IP et le nom complet de domaine (et non l'alias) de cet hôte. Parfois, les entrées d'exportation sont créées avec l'alias et le client rencontre des difficultés lorsqu'il tente d'effectuer un montage.

Limitation du nombre de groupes dans la structure NFS

Sur les systèmes qui utilisent NFS version 3.2, les utilisateurs ne peuvent pas, sans complications, être membres de plus de 16 groupes. (Les groupes sont définis via la commande **groups**.) Si un utilisateur dépend de 17 groupes, il ne sera pas autorisé à lire ou copier les fichiers du 17ème. Il faut alors modifier l'ordre de ces groupes pour lui permettre d'accéder à ces fichiers.

Montage à partir de serveurs équipés d'une version NFS antérieure

Lors du montage d'un système de fichiers à partir d'un serveur NFS de version antérieure à la 3 vers un client NFS version 3, l'utilisateur résidant sur le client rencontrera des difficultés de montage s'il est membre de plus de 8 groupes. En effet, certains serveurs, incapables de traiter correctement ce cas, refusent la demande de montage. La seule solution consiste à réduire le nombre de groupes auxquels appartient l'utilisateur concerné et de soumettre de nouveau la demande de montage. Le message d'erreur suivant est caractéristique de ce type de problème :

```
RPC:    Authentication  error;  why=Invalid  client  credential
```

Conséquences d'une extension de noyau NFS non chargée

Certaines commandes NFS ne s'exécutent pas correctement si l'extension de noyau NFS n'est pas chargée. Parmi ces commandes : **nfsstat**, **exportfs**, **mountd**, **nfsd** et **biod**. Lorsque NFS est installé sur le système, l'extension du noyau est placée dans le fichier **/usr/lib/drivers/nfs.ext**. Ce fichier est ensuite chargé comme extension de noyau lors de la configuration du système. Le script qui constitue cette extension charge le fichier **/etc/rc.net**. Ce script prévoit également, entre autres choses, le chargement de l'extension NFS. Remarquons à ce sujet, que l'extension de noyau TCP/IP (Transmission Control Protocol/Internet Protocol) doit être chargée avant l'extension de noyau NFS.

Remarque : La commande **gfsinstall** est utilisée pour charger l'extension NFS dans le noyau au démarrage initial du système. Cette commande peut être lancée plusieurs fois par amorçage sans générer de complications. Le système est actuellement livré avec la commande **gfsinstall**, présente à la fois dans le fichier **/etc/rc.net** et le fichier **/etc/rc.nfs**. Ce doublon est correct. Il est inutile de supprimer l'une ou l'autre de ces occurrences.

Informations de référence NFS

Liste des fichiers NFS (Network File System)

bootparams	Recense les clients qui peuvent être utilisés pour le démarrage des clients sans disque.
exports	Recense les répertoires qui peuvent être exportés vers des clients NFS.
réseau	Contient des informations sur les réseaux du réseau Internet.
Fichier de configuration pcnfsd.conf	Fournit les options de configuration du démon **rpc.pcnfsd**.
rpc	Contient les informations de base de données pour les programmes RPC (Remote Procedure Call).
xtab	Recense les répertoires actuellement exportés.
/etc/filesystems	Répertorie tous les systèmes de fichiers qui sont montés au démarrage du système.

Liste des commandes NFS

chnfs	Lance un nombre donné de démons **biod** et **nfsd**.
mknfs	Configure le système pour qu'il exécute NFS et lance les démons NFS.
nfso	Configure les options de réseau NFS.
automount	Monte automatiquement un système de fichiers NFS.
chnfsexp	Modifie les attributs d'un répertoire exporté vers NFS.
chnfsmnt	Modifie les attributs d'un répertoire monté sur NFS.
exportfs	Exporte et annule l'exportation de répertoires vers des clients NFS.
lsnfsexp	Affiche les caractéristiques des répertoires exportés avec NFS.
lsnfsmnt	Affiche les caractéristiques des systèmes NFS montés.
mknfsexp	Exporte un répertoire en utilisant NFS.
mknfsmnt	Monte un répertoire en utilisant NFS.
rmnfs	Arrête les démons NFS.
rmnfsexp	Supprime les répertoires NFS exportés de la liste des exportations d'un serveur.
rmnfsmnt	Supprime les systèmes de fichiers NFS montés de la liste des montages d'un client.

Liste des démons NFS

Verrouillage des démons

lockd	Traite les requêtes de verrouillage par le biais du module RPC.
statd	Fournit des fonctions de reprise pour les services de verrouillage sur NFS.

Utilitaires et démons de service réseau

biod	Envoie les requêtes de lecture et d'écriture du client au serveur.
mountd	Répond aux requêtes des clients pour le montage de systèmes de fichiers.
nfsd	Lance le démon qui traite une requête de client concernant les opérations sur les systèmes de fichiers.
pcnfsd	Traite les requêtes de service des clients PC–NFS.
nfsstat	Affiche les informations concernant les possibilités pour une machine de recevoir les appels.
on	Exécute des commandes sur des machines distantes.
portmap	Mappe les numéros de programme RPC et les numéros de port Internet.
rexd	Accepte les requêtes d'exécution de programmes à partir de machines distantes.
rpcgen	Génère le code C afin d'implémenter le protocole RPC.
rpcinfo	Rend compte de l'état des serveurs RPC.
rstatd	Renvoie les statistiques de performance obtenues du noyau.
rup	Affiche l'état d'un hôte distant sur le réseau local.
rusers	Recense les utilisateurs connectés sur des machines distantes.
rusersd	Répond aux requêtes de la commande **rusers**.
rwall	Envoie des messages à tous les utilisateurs du réseau.
rwalld	Gère les requêtes de la commande **rwall**.
showmount	Affiche la liste de tous les clients ayant monté des systèmes de fichiers distants.
spray	Envoie un nombre spécifique de paquets à un hôte.
sprayd	Reçoit les paquets envoyés par la commande **spray**.

Utilitaires et démons de sécurité du réseau

chkey	Modifie la clé de chiffrement de l'utilisateur.
keyenvoy	Fournit un intermédiaire entre les processus utilisateur et le serveur de clé.
keylogin	Décrypte et enregistre la clé privée de l'utilisateur.
keyserv	Enregistre les clés publiques et les clés privées.
mkkeyserv	Lance le démon **keyserv** et annule la mise en commentaire des entrées appropriées dans le fichier **/etc/rc.nfs**.
newkey	Crée une nouvelle clé dans le fichier **publickey**.
rmkeyserv	Arrête le démon **keyserv** et met en commentaire l'entrée correspondant au démon **keyserv** dans le fichier **/etc/rc.nfs**.
ypupdated	Met à jour les informations des mappes NIS (Network Information Service).

Pour plus d'informations sur la sécurité NFS, reportez–vous à Network File System (NFS) Security dans le manuel *AIX 5L Version 5.2 Security Guide*.

Support des clients sans disque Sun

bootparamd Fournit les informations nécessaires au démarrage des clients sans disque.

Sous–routines NFS

cbc_crypt, des_setparity ou ecb_crypt Implémente les routines DES (Data Encryption Standard).

SMBFS

Server Message Block Filesystem (SMBFS) permet d'accéder aux partages sur les serveurs SMB en tant que systèmes de fichiers sous AIX. Dans ce système de fichiers, l'utilisateur peut créer, supprimer, lire, écrire et modifier les durées d'accès aux fichiers et aux répertoires. Le propriétaire ou le mode d'accès des fichiers et des répertoires ne peut pas être modifié.

SMBFS peut être utilisé pour accéder à des fichiers sur un serveur SMB. Le serveur SMB est un serveur exécutant Samba, un serveur AIX exécutant AIX Fast Connect ou un serveur ou une station de travail Windows XP, Windows NT ou Windows 2000. Chacun de ces types de serveur permet d'exporter un répertoire en tant que partage. Ce partage peut ensuite être monté sur un système AIX exécutant SMBFS.

Installation de SMBFS

Pour installer SMBFS sur un système AIX, installez **bos.cifs_fs**.

Lorsque **bos.cifs_fs** est installé, le périphérique `nsmb0` est créé. Le périphérique permet à la commande **mount** d'établir une connexion entre le serveur SMB et le client.

Montage du système de fichiers

Le répertoire peut être monté de l'une des deux façon suivantes. Vous pouvez utiliser la commande AIX **mount**. Par exemple :

```
mount -v cifs -n pezman/user1/pass1 -o uid=201,fmode=750 /home /mnt
```

Pour plus d'informations sur la commande **mount** et pour obtenir des explications sur les indicateurs utilisés, reportez–vous à la commande **mount** dans le manuel *AIX 5L Version 5.2 Commands Reference*.

Vous pouvez aussi monter le système de fichiers avec l'utilitaire SMIT, `smit cifs_fs`, qui exécute la commande **mount** après avoir collecté toutes les informations nécessaires.

Pour monter un système de fichiers SMBFS, il es nécessaire de fournir un nom utilisateur et un mot de passe pour vous authentifier au serveur. Ce nom utilisateur et ce mot de passe sont utilisés pour exécuter toute les opérations sur les fichiers requises sur le serveur. Le champ **Password** dans le panneau smit n'est pas considéré comme requis. Cependant, si ce champ n'est pas complété, le système demande à l'utilisateur de fournir un mot de passe via l'invite de mot de passe standard d'AIX. De cette façon, l'utilisateur peut fournir un mot de passe sans qu'il soit visible.

Lorsqu'une commande de système de fichiers, par exemple une opération de lecture, est appelé pour un fichier dans le point de montage SMBFS, une requête de lecture du fichier est envoyée au serveur. Le nom utilisateur et le mot de passe sont envoyés avec cette requête afin que le serveur puisse déterminer si l'utilisateur possède des droits sur le serveur pour lire le fichier. Par conséquent, c'est le serveur qui décide en dernière instance si une opération sur un fichier est autorisée.

En revanche, l'option `fmode` permet à l'utilisateur root du système client de contrôler l'accès aux fichiers sur le serveur avant que ce dernier soit interrogé. Si l'option `fmode` n'est pas fournie par l'utilisateur, la valeur par défaut est `755`. Le tableau suivant montre comment l'option `fmode` utilise une requête en écriture :

Numéro de cas	utilisateur authentifié pour le serveur	utilisateur côté client souhaitant un accès en écriture	propriétaire du montage, groupe et mode	propriétaire, groupe et mode sur le serveur	accès autorisé
Cas 1	utilisateur1	utilisateur2	utilisateur1, personnel rwxr–xr–x	utilisateur1, personnel rwxrwxr–x	non
Cas 2	utilisateur1	root	utilisateur1, personnel rwxr–xr–x	utilisateur2, personnel rwxr–xr–x	non
Cas 3	utilisateur1	utilisateur1	utilisateur1, personnel rwxr–xr–x	utilisateur2, personnel rwxrwxr–x	oui
Cas 4	utilisateur1	utilisateur1	utilisateur, personnel rwxr–xr–x	root, système rwx———	non
Cas 5	utilisateur1	utilisateur1	utilisateur1, personnel rwxr–xr–x	root, système rwxrwxrwx	oui

Dans le Cas 1, l'accès a été refusé car le propriétaire, le groupe et le mode du montage sur le client n'ont pas accordé un accès en écriture à `utilisateur2`.

Dans le Cas 2, l'accès a été refusé bien que `root` ait accès à tout les éléments côté client, mais l'utilisateur authentifié par le serveur, `utilisateur1`, n'a pas accès au fichier sur le serveur.

Dans le Cas 3, l'accès a été accordé car `utilisateur1` était le propriétaire au montage, et `utilisateur1`, étant membre d'un groupe `personnel` sur le serveur, avait accès au fichier sur le serveur.

Dans le Cas 4, l'accès a été refusé car, bien que `utilisateur1` soit le propriétaire au montage, le fichier appartient à `root` sur le serveur sans accès par un groupe ou un autre membre.

Dans le Cas 5, l'accès a été accordé car `utilisateur1` était le propriétaire au montage, et `utilisateur1` avait accès au fichier sur le serveur via d'autres droits.

Identification des incidents SMBFS

Si la commande **mount** ou le raccourci **smit cifs_fs** renvoie une erreur, prenez en compte les points suivants :

- Vérifiez que le nom d'utilisateur et le mot de passe sont corrects. Le nom d'utilisateur et le mot de passe doivent accéder au partage sur le serveur.
- Vérifiez que le nom du serveur est correct. Si le nom du serveur est correct, utilisez le nom hôte entièrement qualifié au cas où le serveur ne fasse pas partie du même sous–réseau que le client. Vous pouvez aussi essayer d'utiliser l'adresse IP du serveur.
- Vérifiez que la commande `lsdev -L|grep nsmb` renvoie un nom d'unité. Si une unité `nsmb` n'est pas disponible, le client AIX ne pourra pas établir une connexion au serveur SMB.
- Vérifiez que le nom du partage est correct. Si le partage n'existe pas sur le serveur ou n'est pas accessible avec le nom d'utilisateur et le mot de passe indiqués, le serveur SMB rejette la demande de connexion.
- Utilisez l'ID d'événement 525 pour collecter les données de trace système pour SMBFS.

Chapitre 7. Unités TTY et communications série

Ce chapitre est consacré à la gestion des unités de terminal TTY. Les sujets abordés sont les suivants :

Généralités TTY

Une unité de terminal tty est une unité en mode caractère qui effectue des entrées-sorties caractère par caractère. La communication entre ces unités et les programmes qui y accèdent en lecture ou en écriture est contrôlée par l'interface tty. On trouve parmi les unités tty :

- Modems
- Terminaux ASCII
- Console Système (LFT)
- **aixterm** sous AIXwindows.

Il est possible d'ajouter, de supprimer, d'afficher ou de modifier des unités tty comme n'importe quelle autre unité du système, à l'aide de l'application Web-based System Manager Devices, via SMIT ou des commandes propres aux unités.

Variable TERM pour différents écrans et terminaux

Les informations relatives aux fonctions des terminaux sont stockées dans la base de données **terminfo**. Chaque terminal est décrit par la variable d'environnement **TERM** dans la base de données **terminfo**. Les programmes y trouvent toutes les données nécessaires à l'établissement de la communication avec une unité tty courante.

Valeurs TERM pour divers terminaux

Ecran/Terminal	**Valeur**
Terminal ASCII 3161	ibm3161
Terminal ASCII 3163	ibm3161
DEC VT100 (terminal)	vt100
DECVT220	vt220
Station écran ASCII 3151 ou 3161 avec cartouche	ibm3161-C
Station écran ASCII 3162	ibm3161
Station écran ASCII 3162 avec cartouche	ibm3162
Ecran 6091	lft
AIXwindows	aixterm

Pour des informations sur les entrées de la base de données **terminfo**, reportez-vous au format de fichier **terminfo**. Pour convertir les entrées **termcap** en entrées **terminfo**, reportez-vous à la commande **captoinfo**. (Le fichier **termcap** contient la description des terminaux des anciens systèmes Berkeley.)

Définition des caractéristiques de terminal TTY

Le *protocole de liaison* fournit une interface utilisateur indépendante du matériel entre l'ordinateur et une unité asynchrone. Ainsi, un utilisateur peut supprimer une simple ligne ou interrompre un processus en cours en entrant une séquence de caractères. Vous avez la possibilité de définir vous même ces séquences, ainsi que les caractéristiques des terminaux (vitesse de communication, par exemple), à l'aide de l'application Web-based System Manager Devices, la commande **chdev**, l'outil SMIT ou la commande **stty**.

Définition des attributs de l’unité TTY raccordée

L’établissement d’une communication entre l’hôte et l’unité tty raccordée exige :

- un câble de communication,
- Mise en concordance des attributs de communication (vitesse de liaison, longueur de mot, parité, bit d’arrêt et interface) identiques sur l’hôte et l’unité tty raccordée.

Gestion des unités TTY

Pour effectuer une tâche décrite dans le tableau suivant, une unité tty doit être installée.

Gestion des tâches liées aux unités TTY			
Tâche	*Raccourci SMIT*	*Commande ou fichier*	Web-based System Manager Management Environment
Liste des unités TTY définies	**smit lsdtty**	**lsdev –C –c tty –H**	Software —> **Devices** —> **All Devices**.
Ajout d'un terminal TTY	**smit mktty**	**mkdev –t tty**[1,2]	Software —> **Devices** —> **Overview and Tasks**.
Associer un terminal TTY à un autre port[3]	**smit movtty**	**chdev –l** *Nom* **–p** *Nom Parent* **–w Emplacement Connexion**[2,4]	
Modifier/afficher les caractéristiques d'un terminal TTY	**smit chtty**	**lsattr –l** *Nom* **–E** (afficher); **chdev –l** *Nom* (modifier)[4,5]	Software —> **Devices** —> **All Devices** —> **Selected** —> **Properties**.
Suppression d'un terminal TTY[3]	**smit rmtty**	**rmdev –l** *Nom*	Software —> **Devices** —> **All Devices** —> **Selected** —> **Delete**.
Configuration d'un terminal TTY défini (rendre disponible)	**smit mktty**	**mkdev –l** *Nom*	Software —> **Devices** —> **Overview and Tasks**.

Remarques :

1. D'autres indicateurs peuvent être utilisés pour définir plus précisément la nouvelle unité tty. Dans l'exemple ci dessous, il s'agit de définir et de configurer l'unité tty RS–232 connectée au port `0` sur la carte asynchrone 8 ports `sa3` avec un débit de `19200` (attribut `speed`), la valeur des autres attributs étant extraite du fichier `foo` :

   ```
   mkdev –t tty –s rs232 –p sa3 –w 0 –a speed=19200 –f foo
   ```

2. Les commandes **mkdev** et **chdev** prennent en charge les options incompatibles avec Web-based System Manager ou SMIT.
3. Désactivez l'unité tty avant d'effectuer cette tâche. Reportez-vous à la commande **pdisable**.
4. Utilisez des indicateurs pour modifier les caractéristiques spécifiques d'une unité tty à partir de la ligne de commande.
5. Vous pouvez sélectionner un débit en bauds Posix dans la liste ou saisir directement le débit en bauds non–Posix dans la zone de saisie. Si le débit en bauds sélectionné ne peut pas être pris en charge par le modem, le système affiche un message d'erreur.

Si vous ajoutez ou modifiez une unité tty à partir de la ligne de commande, reportez-vous à la liste ci-dessous pour rechercher le nom *d'attribut* que vous devez spécifier avec l'indicateur **–a** *Attribute=valeur* pour la caractéristique à définir. Par exemple, spécifiez `a speed=`*`valeur`* pour définir le débit en bauds de l'unité tty.

Caractéristique	Attribut
Activation de la CONNEXION	login
Vitesse de transmission (BAUDS)	rapidité
PARITE	parity
BITS par caractère	bpc
Nombre de BITS D'ARRET	stops
DELAI avant passage à déf. de port suivante	timeout
Etablissement de liaison XON-XOFF	xon
Type de TERMINAL	term
CONTROLE DE FLUX à utiliser	flow_disp
PROTOCOLE OUVERT à utiliser	open_disp
Attributs STTY pour le temps d'EXECUTION	runmodes
Attributs STTY pour la CONNEXION	logmodes
EXECUTION gestionnaire activité du shell	shell
Nom de CONNEXION	logger
ETAT de l'unité au moment de l'AMORCAGE	autoconfig
Nombre de mémoires tampons D'EMISSION	tbc
Niveau de déclenchement de la RECEPTION	rtrig
Modules STREAMS à ajouter à l'ouverture	modules
Fichier mappe d'ENTREE	imap
Fichier mappe de SORTIE	omap
Fichier mappe de JEU DE CODES	csmap
Caractère INTERRUPT	intr
Caractère QUIT	quit
Caractère ERASE	erase
Caractère KILL	kill
Caractère END OF FILE	eof
Caractère END OF LINE	eol
Deuxième caractère EOL	eol2
Caractère DELAY SUSPEND PROCESS	dsusp
Caractère SUSPEND PROCESS	susp
Caractère LITERAL NEXT	lnext
Caractère START	start
Caractère STOP	stop
Caractère WORD ERASE	werase
Caractère REPRINT LINE	reprint
Caractère DISCARD	discard

Utilitaire d'écran dynamique

L'utilitaire d'écran dynamique, ou commande **dscreen**, permet de connecter un terminal physique à plusieurs sessions de terminal virtuel (écrans) simultanément. Il a été conçu principalement pour les terminaux à pages de mémoire écran multiples (écrans 3151 modèle 310 ou 410 avec cartouche d'extension, par exemple). En effet, sur ce type de terminal, passer d'un écran virtuel à l'autre revient à changer de page écran de terminal physique, ce qui permet de sauvegarder et de restaurer chaque image d'écran virtuel. La commande **dscreen** peut également être appliquée à des terminaux sans pages de mémoire écran multiples pour passer d'une session écran virtuel à l'autre, mais dans ce cas, l'apparence de l'écran sera modifiée.

Remarque : Pour une prise en charge complète de **dscreen**, le terminal doit pouvoir sur commande passer d'une page écran interne à l'autre et mémoriser la position du curseur sur chaque page. **dscreen** peut être exploité sur des terminaux intelligents ou non, mais les images écran ne sont pas sauvegardées lors des changements d'écran sur les terminaux non intelligents.

Fichier de configuration de terminal dscreen

Le fichier **dsinfo**, fichier de configuration de terminal pour **dscreen**, sert à définir différents jeux de touches à utiliser avec cette commande, notamment lorsque les touches **dscreen** initialement définies ne sont pas compatibles avec une application exploitée sur le système.

Le type de terminal défini dans le fichier dsinfo admet par défaut une seule page de mémoire écran. Si le terminal utilisé en accepte davantage, ce fichier doit être modifié pour intégrer la séquence nécessaire au contrôle de la mémoire de page. Reportez-vous au manuel de référence du terminal pour connaître la séquence de contrôle spécifique.

Le fichier dsinfo par défaut est **/usr/lbin/tty/dsinfo**. Utilisez l'indicateur **–i** pour en spécifier un autre. Les informations développées dans cette section se rapportent au fichier par défaut mais elles sont valables pour n'importe quel autre fichier **dsinfo** créé.

Pour plus d'informations, reportez-vous à la section "Affectation d'écran dynamique" page 7-8.

Affectation de touches

L'exécution de **dscreen** ouvre un écran virtuel. Certaines touches du clavier ne sont pas transmises à cet écran : **dscreen** les intercepte et exécute, à leur activation, les actions suivantes :

Select (voir Select Keys page 7-7)	Sélectionne un écran.
Block (voir Block Keys page 7-7)	Bloque toute entrée et sortie.
New (voir New Keys page 7-7)	Ouvre une nouvelle session écran.
End (voir End and Quit Keys page 7-7)	Arrête l'utilitaire **dscreen**.
Quit (voir End and Quit Keys page 7-7)	Quitte l'utilitaire **dscreen**.
Previous (voir Previous Key page 7-7)	Revient à l'écran précédent.
List (voir List Key page 7-7)	Affiche les touches affectées à **dscreen** et leur fonction respective.

La fonction de chaque touche dépend du terminal et de sa définition dans le fichier **/usr/lbin/tty/dsinfo**.

Touche de sélection (Select)

A chaque écran virtuel créé est affectée une touche de sélection.
Lorsqu'elle est activée, elle :

- assure le basculement du terminal physique à la page vidéo associée à l'écran virtuel,
- réachemine les entrées-sorties entre le terminal physique et l'écran virtuel.

Une fois que toutes les touches de sélection définies dans le fichier **dsinfo** ont été associées à un écran virtuel, il n'est plus possible de créer d'écran. Les sessions écran individuelles sont fermées lorsque le processus shell initial s'arrête. La touche associée est alors libérée, à disposition d'un autre écran virtuel. **dscreen** s'arrête à la fermeture du dernier écran actif.

Touche de blocage (Block)

Les touches de blocage servent à arrêter les sorties (comme le fait la séquence de touche Ctrl–S en contrôle de flux IXON), permettant ainsi d'établir de façon transparente des sessions de terminal sur deux ordinateurs utilisant un terminal à deux ports série.

Touche de création d'écran (New)

Appuyer sur une touche de création d'écran définit un nouvel écran logique et lui affecte une touche de sélection. Chaque écran créé requiert :

- une des touches de sélection définies dans le fichier dsinfo,
- une pseudo unité de terminal **dscreen**,
- suffisamment de mémoire pour les diverses structures de suivi d'écran,
- un processus pour l'exécution du shell.

A défaut d'un de ces éléments, l'écran ne peut être créé. Un message s'affiche.

Touches d'arrêt et de sortie (End et Quit)

Un touche d'arrêt (end) provoque :

- la diffusion d'un signal **SIGHUP** à toutes les sessions écran,
- l'élimination des erreurs,
- la sortie à l'état 0.

Une touche de sortie (quit) entraîne les mêmes opérations, avec l'état de sortie 1.

Touche d'écran précédent (Previous)

Une touche d'écran précédent (Previous) bascule sur l'écran précédemment affiché.

Remarques :

1. Restez sur le même écran tant qu'une écriture est en cours. En effet, si une séquence d'échappement est tronquée, le terminal est placé dans un état inconnu.
2. Certains écrans de terminal peuvent mémoriser la position du curseur sur un écran sans enregistrer les modes (insertion, vidéo inverse, etc.). Dans ce cas, évitez d'utiliser ces modes en passant d'un écran à l'autre.

Touche de listage (List)

La touche de listage (List) affiche la liste des touches (reconnues par **dscreen**, avec leur fonction, sur l'écran du terminal). Lorsqu'un écran est créé via **dscreen**, le message `Press` *`KEY`* `for help` s'affiche (*`KEY`* est le nom de la touche de listage affichée sur le terminal). Ce message n'est émis *que* si une touche de listage a été définie.

Affectation d'écran dynamique

Dans le fichier **/usr/lbin/tty/dsinfo**, l'entrée de description du terminal comporte autant de touches de sélection d'écran que de pages écran physiques définies pour le terminal. Si le nombre de touches de sélection dépasse celui des pages écran physiques, **dscreen** affecte dynamiquement des pages écran physiques aux écrans virtuels.

Si un écran virtuel dépourvu de page de mémoire écran est sélectionné, **dscreen** lui affecte l'écran physique le moins récemment utilisé. Selon les spécifications mises à jour dans le fichier **/usr/lbin/tty/dsinfo**, il peut être indiqué que l'écran physique est connecté à un écran virtuel différent, par exemple, l'écran est effacé.vb

Description du fichier dsinfo

Le fichier **dsinfo** est une base de données de descriptions de terminal à l'usage de l'utilitaire d'écrans multiples **dscreen**. Ce fichier rassemble les informations suivantes :

- les touches **dscreen** avec leur fonction,
- le nombre de pages mémoire écran du terminal,
- les séquences de codes envoyées ou reçues pour l'utilisation des fonctions ci-dessus.

Dans le fichier **dsinfo** par défaut, les entrées sur le type de terminal se présentent sous la forme de données de terminal 3151 ASCII du type :

```
# The Cartridge for Expansion (pn: 64F9314) needed for this entry
 ibm3151|3151|IBM 3151,
 dsks=\E!a^M|Shift-F1|,            # Selects first screen
 dsks=\E!b^M|Shift-F2|,            # Selects second screen
 dsks=\E!c^M|Shift-F3|,            # Selects third screen
 dsks=\E!d^M|Shift-F4|,            # Selects fourth screen
 dskc=\E!e^M|Shift-F5|,            # Creates a new screen
 dske=\E!f^M|Shift-F6|\E pA\EH\EJ, # Go to screen 1 and end
 dskl=\E!g^M|Shift-F7|,            # Lists function keys (help)
 dskp=\E!h^M|Shift-F8|,            # Go to previous screen
 dskq=\E!i^M|Shift-F9|\E pA\EH\EJ, # Go to screen 1 and quit
 dsp=\E pA|\EH\EJ,               # Terminal sequence for screen 1
 dsp=\E pB|\EH\EJ,               # Terminal sequence for screen 2
 dsp=\E pC|\EH\EJ,               # Terminal sequence for screen 3
 dsp=\E pD|\EH\EJ,               # Terminal sequence for screen 4
 dst=10,                         # Allow 1 second timeout buffer
```

Format d'entrée

Les entrées du fichier **dsinfo** sont des champs séparés par une virgule. Le premier champ est constitué de la liste des noms possibles du terminal, séparés par une barre verticale (|). Tout texte précédé d'un astérisque (#) est un commentaire, ignoré par **dscreen**. Les autres champs sont des chaînes décrivant les fonctions du terminal à l'utilitaire **dscreen**. Les séquences d'échappement reconnues dans ces chaînes sont les suivantes :

Séquence Escape	Description
\E,\e	Echappement
\n,\l	Ligne suivante
\r	Retour chariot
\t	Tabulation
\b	Retour arrière
\f	Page suivante
\s	Espace

\ *nnn*	Valeur octale *nnn*
^ *x*	Ctrl– *x* pour toute valeur *x* appropriée

Tout autre caractère précédé d'une barre oblique inverse (\ ) génère le caractère lui-même. Les chaînes sont entrées sous la forme *type=chaîne*, *type* étant le type de chaîne et *chaîne*, sa valeur.

Dans le fichier **dsinfo**, veillez à séparer les champs par une virgule. Si la virgule est omise ou tronquée en fin d'entrée, le fichier est inexploitable par l'utilitaire **dscreen** et une erreur est envoyée à l'écran.

Types de chaîne

Voici les différents types de chaîne :

dskx

Les chaînes commençant par dsk décrivent une touche. La chaîne comporte 4 caractères. Le quatrième caractère, *x*, indique l'action de la touche. Les types de touches sont les suivants :

Type
Action

dsks
Bascule d'un écran à l'autre

dskb
Bloque les E/S

dske
Arrête **dscreen**

dskq
Quitte **dscreen** (avec l'état 1)

dskc
Crée un nouvel écran

dskp
Revient à l'écran précédent

dskl
Affiche la liste des touches et les actions correspondantes

Tout autre type de touche (c'est–à–dire un type de chaîne commençant par dsk*x* et suffixée par une autre lettre que s, b, e, q, c, p ou l ne génère aucune action **dscreen** interne, mais est répertoriée, reconnue et exécutée. Un type de dskn (n pour No operation) est préconisé pour signifier qu'aucune action interne de **dscreen** n'est souhaitée.

La chaîne des valeurs de chaque touche se compose de trois sous–chaînes séparées par une barre verticale (|).

Remarque :
Indiquez \| pour inclure | dans une des sous–chaînes.

La première sous–chaîne représente la séquence de caractères envoyée par le terminal lors de l'activation de la touche. La deuxième sous–chaîne est une étiquette destinée à la touche qui s'imprime lors de l'affichage d'une liste de touches. La troisième sous–chaîne est une séquence de caractères transmise par **dscreen** au terminal lorsque cette touche est activée avant l'exécution de l'action demandée par cette même touche.

dsp Chaîne décrivant un écran physique sur le terminal. Une chaîne dsp doit être spécifiée pour chaque écran physique du terminal. Cette chaîne se compose de deux sous-chaînes séparées par une barre verticale (|).

La première sous-chaîne est la séquence de caractères à envoyer au terminal pour l'affichage et la sortie sur la page physique du terminal.

La seconde sous-chaîne est envoyée au terminal lorsque la page est utilisée pour un nouvel élément. Elle correspond généralement à la séquence de vidage d'écran et est envoyée dans deux cas :

1. lors de la création d'une session de terminal virtuel,
2. lorsque le nombre de terminaux virtuels est supérieur au nombre d'écrans physiques. Si un terminal virtuel requiert de **dscreen** plusieurs utilisations d'un même écran physique, il envoie cette séquence à l'écran pour lui indiquer que son contenu ne concorde pas avec la sortie du terminal virtuel connecté.

Remarque :
Pour éviter toute confusion, il est déconseillé de travailler avec plus de terminaux virtuels que d'écrans physiques : ne définissez pas plus de touches de sélection d'écran (dsks=) que d'écrans physiques (dsp=) dans l'entrée dsinfo.

dst A Chaîne de type dst qui définit le délai d'attente (en dixièmes de secondes) en entrée de *dscreen*. La valeur de la chaîne est un nombre décimal (maximum 255 ; par défaut **1** [ou 0,1 seconde]).

Lorsque **dscreen** reconnaît un préfixe de séquence de touches d'entrée mais qu'il ne dispose pas de tous les caractères de la séquence, il attend les caractères manquants pour l'identifier. Passé le délai imparti, les caractères sont envoyés à l'écran virtuel et ne sont pas interprétés par **dscreen** comme partie intégrante d'une séquence de touches d'entrée.

Il peut être nécessaire d'augmenter le délai si une ou plusieurs des touches **dscreen** correspondent en fait à une série de touches (par exemple Ctrl–Z 1, Ctrl–Z 2, Ctrl–Z 3 ... pour la sélection d'écran, Ctrl–Z N pour un nouvel écran, etc.).

Exemple 1

L'entrée **/usr/lbin/tty/dsinfo** se rapporte à un Wyse–60 avec trois sessions d'écran :

```
wy60|wyse60|wyse model 60,
 dsks=^A'^M|Shift-F1|,
 dsks=^Aa^M|Shift-F2|,
 dsks=^Ab^M|Shift-F3|,
 dskc=\200|Ctrl-F1|,
 dske=\201|Ctrl-F2|\Ew0\E+,
 dskl=\202|Ctrl-F3|,
 dsp=\Ew0|\E+,
 dsp=\Ew1|\E+,
 dsp=\Ew2|\E+,
```

Cette entrée attribue :

- les séquences Maj–F1 à Maj–F3 à la sélection des écrans 1 à 3,
- la séquence Ctrl–F1 à la création d'un écran,
- Ctrl–F2 envoie : vers l'écran, de `Esc w 0 Esc +` (passage à la fenêtre 0 et vidage de l'écran) et à l'arrêt de **dscreen**,
- la séquence Ctrl–F3 à l'affichage des touches et de leur fonction.

Chaque fois qu'un écran physique est utilisé pour un nouvel écran, la séquence `Esc +` est envoyée au terminal, ce qui vide l'écran.

Exemple 2

Cet exemple concerne un Wyse–60 avec trois sessions d'écran, un des écrans se trouvant sur un second ordinateur communiquant via le second port série du terminal :

```
wy60-1|wyse60-1|wyse model 60 - first serial port
 dsks=^A‘^M|Shift-F1|,
 dsks=^Aa^M|Shift-F2|,
 dsks=^Ab^M|Shift-F3|\Ed#^Ab\r^T\Ee9,
 dskc=\200|Ctrl-F1|,
 dske=\201|Ctrl-F2|\Ed#\201^T\Ew0\E+,
 dskl=\202|Ctrl-F3|,
 dsp=\Ew0|\E+,dsp=\Ew1|\E+,
 wy60-2|wyse60-2|wyse model 60 - second serial port
 dsks=^A‘^M|Shift-F1|\Ed#^A‘\r^T\Ee8,
 dsks=^Aa^M|Shift-F2|\Ed#^Aa\r^T\Ee8,
 dsks=^Ab^M|Shift-F3|,
 dskc=\200|Ctrl-F1|,
 dske=\201|Ctrl-F2|\Ed#\201^T\Ew0\E+,
 dskl=\202|Ctrl-F3|,
 dsp=\Ew2|\E+,
```

dscreen doit être exécuté (avec l'option **t**) sur les deux ordinateurs, le premier équipé d'un terminal wy60–1 et le second d'un terminal wy60–2. L'entrée wy60–1 est examiné en premier.

Les deux premières entrées de touches ne sont pas modifiées par rapport à l'entrée wy60-2 initiale. La troisième, de type dskb, demande le blocage des entrées et des sorties. Lorsque cette touche est activée, la séquence :

```
Esc d # Ctrl-A b CR Ctrl-T Esc e 9
```

est envoyée au terminal. Dès lors, la sortie est bloquée et **dscreen** poursuit l'analyse des entrées de séquences de touches mais ignore les autres entrées.

La séquence `Esc d#` place le terminal en mode d'impression transparente (mode TPM) qui renvoie, jusqu'à réception d'un Ctrl–T, tous les caractères vers l'autre port série.

Les caractères `Ctrl-A b CR` sont envoyés vers l'autre port série, indiquant au process **dscreen** de l'autre ordinateur qu'il doit activer la fenêtre associée à la séquence Maj–F3.

La séquence Ctrl–T sort du mode TPM. Esc e 9 fait basculer le terminal sur l'autre port série AUX pour la communication des données.

Dès lors, l'autre ordinateur prend le relais et envoie une séquence `Esc w 2` pour basculer sur le troisième écran physique et reprendre la communication normale.

L'entrée wy60–2 observe le même format que pour les touches Maj–F1 et Maj–F2 :

- bascule en mode TPM,
- envoie la chaîne de touches de fonction à l'autre ordinateur,
- désactive le mode TPM,
- bascule sur l'autre port série.

La touche de fin Ctrl–F2 fonctionne de façon identique sur les deux ordinateurs : elle envoie la séquence de touches de fin à l'autre ordinateur par le biais du mécanisme d'impression transparente, fait passer le terminal dans la fenêtre 0, vide l'écran et quitte.

Modems

Les modems assurent les communications série via des lignes téléphoniques ordinaires. Cette section présente les normes relatives aux modems et explique comment installer et configurer les modems courants.

Généralités

Un *modem* est un dispositif qui permet de connecter deux ordinateurs via des lignes téléphoniques ordinaires. Le système téléphonique actuel est incapable de prendre en charge les variations de tension requises pour une connexion numérique directe. Le modem supprime cette contrainte en convertissant les informations numériques en fréquences transmissibles via une ligne téléphonique (modulation) et en rétablissant ces signaux en données numériques à réception (démodulation). Les modems sont couramment utilisés avec les protocoles BNU (Basic Network Utilities) et autres versions d'UUCP (UNIX–to–UNIX Copy Program). Un modem à haut débit (supérieur à 14 400 bps) peut être utilisé avec le protocole SLIP (Serial Line Internet Protocol) pour fournir en sus la connectivité TCP/IP.

On exprime souvent la vitesse des modems en *bauds* au lieu de bps (bits par seconde). Le baud est en fait l'unité de mesure du débit de modulation. Sur les modèles plus anciens, où un seul bit était codé à chaque changement de signal, le débit en bauds équivalait à la vitesse du modem. A des vitesses supérieures, les modems fonctionnent généralement à un débit de 2 400 (voire 1 200) bauds et codent deux ou plusieurs bits par changement de signal. La vitesse d'un modem en bps est le produit du nombre de bits de données par signal par le nombre de bauds (par exemple, 2 400 bauds x 6 bits par changement de signal = 14 400 bps). La plupart des modèles actuels peuvent communiquer à diverses vitesses (par exemple 14 400, 9 600, 7 800, 4 800 et 2 400 bps).

Normes de télécommunication

Sur les anciens modèles, les vitesses (300, 1 200 et 2 400 bps) étaient précisément définies. Mais pour atteindre des vitesses plus élevées, les constructeurs de modems ont commencé à développer diverses technologies, chacune selon des méthodes propriétaires incompatibles entre elles. De nos jours, ces communications à haut débit sont normalisées par le comité UIT–TSS (ex–CCITT : Consultative Committee for International Telephony and Telegraphy).

Même les modems à haut débit se révèlent bien plus lents que les autres systèmes de communication informatiques. Ainsi, un modem à haut débit peut fonctionner à 28 800 bps alors qu'une connexion Ethernet atteint 10 000 000 bps. Les modems à haut débit offrent généralement un ou plusieurs algorithmes de compression des données permettant d'accélérer le débit des données. Ces algorithmes permettent au modem d'atteindre des vitesses de 57 600 bps (si le débit est de 14 400 bps) ou de 115 200 bps (si le débit est de 28 800 bps). Notez que ces algorithmes de compression s'adaptent aux données transmises. Si les données ont déjà été compressées (par la commande **compress**, par exemple), les méthodes de compression des modems à haut débit offrent des avantages négligeables et peuvent même ralentir leur débit Pour l'exploitation d'un modem avec compression des données, la vitesse de la connexion ETTD/ETCD entre le terminal et le modem est égale ou supérieure au débit de données nominal de la connexion entre modems. Par exemple, le débit des données (la vitesse de communication par lignes téléphoniques) d'un modem V.32*bis* avec compression de données V.42*bis* est de 14400 bps. Si la compression V.42*bis* est active, le débit de données réel peut atteindre 57600 bps. Pour s'adapter au débit supérieur offert par la compression de données, la vitesse de la connexion DTE/DCE entre l'ordinateur et le modem doit être définie à 57600 bps.

Les termes ITU–TSS définissent une norme pour les communications à haut débit, y compris les algorithmes de compression des données. Les normes édictées sont généralement appelées V.*nn*, *nn* représentant un numéro. Il existe aussi une autre norme,

moins connue : le protocole MNP (Microcom Networking Protocol). Disponible dans les versions (ou classes) 1-9, il s'agit d'un protocole haut débit et haute performance qui a été mis en place très tôt et s'est imposé de facto comme norme avant l'avènement des normes CCITT.

Normes de communications UIT-TSS

Voici une liste non exhaustive des normes de communication courantes définies par l'UIT–TSS. Il en existe bien d'autres. Pour obtenir une liste complète, reportez–vous au site Web de International Telecommunication Union.

V.29 Norme ITU–TSS pour communication en semi-duplex 9600 bps.

V.32 Norme ITU–TSS pour communication en duplex intégral à 9600 bps.

V.32 *bis* Norme ITU–TSS pour communication à 14 400 bps. V.32 *bis* est une révision de la norme V.32.

V.34 Norme ITU–TSS pour communication à 33 600 bps. Cette norme vise un débit de données de 33 600 bps via un codage de bits multiple au lieu du schéma de compression de données utilisé par MNP Classe. Ex–norme V. *fast*.

V.42 Procédures ITU–TSS de correction des erreurs pour les DCE qui utilisent la conversion d'asynchrone à synchrone.

V.42 bis Norme révisée de compression de données ITU–TSS.

Normes de communication MNP

MNP classe 1 Méthode asynchrone, en semi-duplex, orientée octets pour le transfert des données avec 70 % d'efficacité. Peu courante sur les modems modernes.

MNP classe 2 Equivalent de la norme MNP classe 1 en duplex intégral. Peu courante sur les modems modernes.

MNP classe 3 Méthode synchrone, en duplex intégral, orientée bits pour le transfert des données avec 108 % d'efficacité. L'efficacité est supérieure à 100% du fait de l'élimination des bits de départ et d'arrêt requis pour une connexion asynchrone. La connexion DTE/DCE entre le modem et le système est cependant asynchrone.

MNP classe 4 Amélioration de la norme MNP classe 3 incluant un mécanisme de variation de la taille des paquets (assemblage adaptatif de paquets) et d'élimination des charges administratives redondantes (optimisation de la phase de données). Un modem conforme à MNP classe 4 offre environ 120 % d'efficacité.

MNP classe 5 Fonctions de la classe 4 complétées par la compression des données. Un modem conforme à MNP classe 5 offre 200 % d'efficacité.

MNP classe 6 Norme permettant l'incorporation dans un modem de plusieurs techniques de modulation incompatibles (négociation de liaison universelle). Les modems conformes à MNP classe 6 peuvent entamer la communication à basse vitesse et négocier une transition vers une vitesse supérieure. Inclut un schéma de duplexage statistique qui alloue dynamiquement l'utilisation de la modulation en semi-duplex pour simuler un service en duplex intégral. Englobe la totalité des fonctions de MNP classe 5.

MNP classe 7 Norme incorporant une méthode améliorée de compression de données. Combinée avec la classe 4, elle réalise une efficacité de 300 %.

MNP classe 8 Non applicable

MNP classe 9 Norme alliant à la technologie V.32 la compression de données améliorée pour atteindre un débit de 28 800 bps.

Configuration des modems génériques

Pour installer un modem :

1. Créez une unité TTY sur le serveur
2. Raccordez du modem avec les câbles appropriés
3. Ajoutez un TTY pour le modem
4. Configurez le modem.

Configuration d'une unité TTY sur le système d'exploitation

Définissez le port TTY pour la connexion de l'unité via SMIT. La plupart des paramètres correspondent aux unités courantes. Activation de la CONNEXION, seule zone concernant le modem, propose les valeurs suivantes :

DISABLE	Aucun processus getty n'est lancé sur le port. Paramètre réservé au port modem pour les appels sortants.
ENABLE	Aucun processus getty n'est lancé sur le port. Paramètre réservé au port modem pour les appels entrants.
SHARE	Le processus getty en cours sur le port autorise les programmes à effectuer des appels entrants et sortants sur le port. Il est inutile de modifier les paramètres en désactivation ou activation. Paramètre réservé au port bidirectionnel.
DELAY	Un processus getty est en cours sur le port en mode bidirectionnel ; aucun signal n'est envoyé jusqu'à émission d'une commande utilisateur.

Zones spécifiques de la carte asynchrone 128 ports :

Force Carrier or Ignore Carrier Detect (forcer la porteuse ou ignorer la détection de porteuse)	disable*
Perform Cooked Processing in Adapter (traitement préparé sur la carte)	disable

Remarque : Ce paramètre est signalé par un astérisque (*)et est désactivé si vous utilisez le connecteur 10 broches RJ–45. Ce paramètre doit être activé si vous utilisez le connecteur 8 broches RJ–45.

Raccordement du modem avec les câbles appropriés

Utilisez les câbles qui conviennent, dont voici les références et les descriptions :

6323741	Async Cable, EIA–232 ; permet de raccorder toutes les unités asynchrones. Parfois utilisé avec d'autres jeux de câbles.
59F3740	Connecteur D–shell de 10 à 25 broches utilisé pour raccorder le câble asynchrone 6323741 aux ports série natifs S1 et S2 comme le montre la figure suivante.

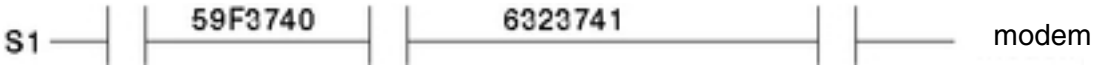
Figure 29. 10 à 25–Broches Cette illustration représente un connecteur de 10 à 25–broches.

59F3432 Câble P utilisé pour connecter le concentrateur à 16 ports. Le numéro de série inclut quatre câbles convertisseurs de RJ–45 à DB–25.

Voici des exemples de connexions de câble :

1. Pour raccorder un modem au port série natif S1, utilisez les câbles suivants :

   Figure 30. Connexion entre un modem et un câble de port série natif Cette illustration représente un câble 59F3740 sur le port série et un 6323741 sur le modem.



2. Pour raccorder un modem à un jeu de câbles d'interface d'une carte asynchrone à 8 ports (EIA–232), utilisez les câbles suivants :

   Figure 31. Connexion de l'interface à 8 ports aux câbles du modem Cette illustration représente une interface à 8 ports connectée à un modem avec un câble 6323741.

3. Pour raccorder un modem à un concentrateur à 16 ports sur une carte à 64 ports, utilisez les câbles suivants :

   Figure 32. Connexion de l'interface à 16 ports aux câbles du modem Cette illustration représente un câble 59F3432 sur le port série et un 6323741 sur le modem.

Ajout d'un TTY pour le modem

Assurez–vous que le système est sous tension et le modem hors tension. Utilisez Web–based System Manager, **wsm**, ou le raccourci SMIT **smit mktty**.

Configuration du modem

Utilisez l'une des deux méthodes de configuration présentées dans cette section. Si les utilitaires BNU (Basic Networking Utilities) sont installés, reportez–vous à Envoi de commandes AT avec la commande cu. Si les utilitaires BNU ne sont pas installés, installed, reportez–vous à Envoi de commandes AT avec un programme C. Pour plus d'informations sur l'installation des utilitaires BNU, reportez–vous à Utilitaires BNU (Basic Networking Utilities).

Envoi de commandes AT via la commande cu

Si BNU (Basic Network Utilities) est installé sur votre système, vous pouvez utiliser la commande **cu** comme suit pour configurer un modem : Les commandes et les paramètres détaillés dans cette section permettent la configuration d'un modem compatible Hayes pour une utilisation sur les ports série du serveur.

1. Ajoutez au fichier **/usr/lib/uucp/Devices** la ligne suivante à moins qu'elle n'y figure déjà. Remplacez le signe # par le numéro de votre port.

   ```
   Direct tty# - Any direct
   ```

2. Vérifiez que le TTY est désactivé en tapant ce qui suit :

   ```
   pdisable tty#
   ```

3. Entrez la commande suivante :

```
cu –ml tty#
```

Un message indiquant Connected s'affiche.

4. Vérifiez que le modem est connecté. Pour ce faire, tapez :

   ```
   AT
   ```

   Le modem répond par OK. Si ce n'est pas le cas, reportez–vous à Identification et résolution des problèmes de modem.

   Pour consulter les autres commandes **AT** et leurs descriptions, reportez–vous au Résumé des commandes AT.

5. Selon l'option getty sélectionnée, entrez l'une des commandes suivantes. Substituez le périphérique tty pour *n*.

   - `penable tty n`
   - `pshare tty n`
   - `pdelay tty n`
   - `pdisplay tty n`

   Le modem est configuré et intègre les commandes nécessaires à la plupart des communications série du système d'exploitation. En cas de problème, appelez **cu –dl** pour lancer un suivi de diagnostics sur la connexion.

Envoi de commandes AT via un programme C

Si la méthode précédente n'aboutit pas ou que BNU n'est pas installé, exécutez le programme C ci-après. Créez un fichier appelé **motalk.c** contenant le code ci-après. Sauvegardez le fichier. Compilez-le puis exécutez-le en suivant les indications données en commentaire dans le programme.

```
/*************************************************************/
 /*  MoTalk - A "C" program for modem setup.                  */
 /*           This program is meant as an aid only and is     */
 /*           not supported by IBM.                          */
 /*                  compile:  cc -o motalk motalk.c          */
 /*                  Usage:  motalk /dev/tty? [speed]         */
 /*************************************************************/
 #include  <errno.h>
 #include  <stdio.h>
 #include  <signal.h>
 #include  <fcntl.h>
 #include  <termio.h>
 FILE *fdr, *fdw;
 int fd;
 struct termio term_save, stdin_save;
 void Exit(int sig)
 {
  if (fdr) fclose(fdr);
  if (fdw) fclose(fdw);
  ioctl(fd, TCSETA, &term_save);
  close(fd);
  ioctl(fileno(stdin), TCSETA, &stdin_save);
  exit(sig);
 }
 main(int argc, char *argv[])
 {
  char *b, buffer[80];
  int baud=0, num;
  struct termio term, tstdin;
  if (argc < 2 || !strcmp(argv[1], "-?"))
```

```
  {
   fprintf(stderr, "Usage: motalk /dev/tty? [vitesse]\n");
   exit(1);
  }
  if ((fd = open(argv[1], O_RDWR | O_NDELAY)) < 0)
  {
   perror(argv[1]);
   exit(errno);
  }
  if (argc > 2)
  {
   switch(atoi(argv[2]))
   {
    case   300: baud = B300;
         break;
    case  1200: baud = B1200;
         break;
    case  2400: baud = B2400;
         break;
    case  4800: baud = B4800;
         break;
    case  9600: baud = B9600;
         break;
    case 19200: baud = B19200;
         break;
    case 38400: baud = B38400;
         break;
    default:    baud = 0;

        fprintf(stderr, "%s: %s is an unsupported baud\n", argv[0],
argv[2]);
         exit(1);
    }
   }
  /* Save stdin and tty state and trap some signals */
  ioctl(fd, TCGETA, &term_save);
  ioctl(fileno(stdin), TCGETA, &stdin_save);
  signal(SIGHUP, Exit);
  signal(SIGINT, Exit);
  signal(SIGQUIT, Exit);
  signal(SIGTERM, Exit);
  /*  Set stdin to raw mode, no echo */
  ioctl(fileno(stdin), TCGETA, &stdin_save);
  tstdin.c_iflag = 0;
  tstdin.c_lflag &= ~(ICANON | ECHO);
  tstdin.c_cc[VMIN] = 0;
  tstdin.c_cc[VTIME] = 0;
  ioctl(fileno(stdin), TCSETA, &tstdin);
  /*  Set tty state */
  ioctl(fd, TCGETA, &term);
  term.c_cflag |= CLOCAL|HUPCL;
  if (baud > 0)
  {
   term.c_cflag &= ~CBAUD;
   term.c_cflag |= baud;
  }
  term.c_lflag &= ~(ICANON | ECHO); /* to force raw mode */
  term.c_iflag &= ~ICRNL; /* to avoid non-needed blank lines */
```

```
term.c_cc[VMIN] = 0;
term.c_cc[VTIME] = 10;
ioctl(fd, TCSETA, &term);
fcntl(fd, F_SETFL, fcntl(fd, F_GETFL, 0) & ~O_NDELAY);
/*  Open tty for read and write */
if ((fdr = fopen(argv[1], "r")) == NULL )
{
 perror(argv[1]);
 exit(errno);
}
if ((fdw = fopen(argv[1], "w")) == NULL )
{
 perror(argv[1]);
 exit(errno);
}
/*  Talk to the modem */
puts("Ready... ^C to exit");
while (1)
{
 if ((num = read(fileno(stdin), buffer, 80)) > 0)
  write(fileno(fdw), buffer, num);
 if ((num = read(fileno(fdr), buffer, 80)) > 0)
  write(fileno(stdout), buffer, num);
 Exit (0);
}
}
```

Modems Hayes et compatibles

1. Modifiez la configuration de tty à l'aide de **wsm**, de Web-based System Manager ou du raccourci SMIT **smit chtty**. A titre d'exemple, vous pouvez modifier la champ LOGIN et lui associer **Share** ou **Enable**.
2. Ajoutez au fichier **/usr/lib/uucp/Systems** la ligne suivante :

   ```
   hayes Nvr HAYESPROG 2400
   ```

3. Ajoutez dans le fichier **/usr/lib/uucp/Devices** :

   ```
   # For programming the  hayes modem only:
    HAYESPROG tty0 - 2400 HayesProgrm2400
    #regular ACU entry:
    ACU tty0 - Any hayes
   ```

4. Ajoutez dans le fichier **/usr/lib/uucp/Dialers** :

   ```
   # This Entry is used to PROGRAM the modem ONLY:
    # the next 3 lines should be made into one:
   HayesProgrm2400     =,-,     "" \d\dAT\r\c OK AT&F\r\c OK ATM1\r\
   c OK
   AT&D3\r\c OK AT&K3&C1\r\c OK ATL0E0Q2\r\c OK ATS0=1\r\c OK AT&W\r
   \c
    OK
    hayes     =,-,     "" \dAT\r\c OK ATDT\T\d\r\c CONNECT
   ```

5. Pour programmer le modem, entrez la commande `cu -d hayes`. Cette commande utilise **cu** pour programmer le modem. Aucune connexion n'étant établie avec un autre système, la commande échoue. Le modem est programmé si `sendthem AT&W` et `OK got it` s'affichent en sortie.

Si vous n'effectuez pas de transfert de fichier binaire ou que vous n'utilisez pas BNU, sortez de **&K3**, et définissez XON pour le contrôle du flux. Toutefois, l'utilisation du contrôle du flux matériel est plus efficace (par rapport à l'établissement de liaison XON–XOFF). Pour ce faire, utilisez les paramètres et les entrées `Dialers` à partir de l'étape suivante.

6. Une fois le modem programmé, vous pouvez configurer le pilote pour l'utilisation du contrôle du flux matériel. Pour modifier le contrôle du flux sur RTS, utilisez **wsm** de Web-based System Manager ou SMIT (raccourci **smit chtty**). Consultez le manuel du modem pour vérifier si le contrôle du flux matériel est pris en charge.

Identification et résolution des problèmes de modem

Cette section identifie les incidents liés à l'utilisation d'un modem : Gardez à l'esprit les points suivants :

- Certains modems différencient les majuscules des minuscules. Saisissez la commande **AT** en majuscules.
- Lors d'opérations normales; il est préférable de relancer le modem au signal DTR (paramètre &D3). Cependant, à la première configuration du modem, il est recommandé de ne pas relancer le modem au signal DTR (paramètre &D2). Si le modem est relancé automatiquement, les paramètres qui n'ont pas été sauvegardés dans la mémoire système sont perdus.

  Le fait de ne pas réinitialiser le modem protège également les modifications lorsque &C1 est défini. La modification de l'état de détection de porteuse peut entraîner un basculement de la ligne de détection de porteuse sur certains modems, la commande **cu** perdant alors la ligne. Il est possible d'entrer le paramètre &D3 une fois la configuration terminée.
- Ces commandes sont standard sur la plupart des modems compatibles Hayes, mais peuvent ne pas fonctionner sur le modem que vous utilisez : consultez le manuel du modem.

Symptôme	Cause	Solution
Le modem (ou un autre dispositif raccordé au port série) ralentit progressivement le système ou le bloque. Sa mise hors tension permet généralement de revenir à la normale.	Sur un modem intelligent, CD est toujours positionné sur ON. Le système le détecte et envoie une annonce de connexion, que le modem tente d'interpréter comme une commande. N'y parvenant pas, le modem renvoie un écho au port tty du système. Ce cycle boucle à l'infini.	Appliquez au port tty un report de connexion (delay) sur le système pour qu'il n'y ait pas d'annonce de connexion émise. Ainsi, seul un retour chariot valide issu d'une connexion hôte générera une annonce de connexion. Vous pouvez également modifier le profil AT du modem pour que CD ne soit positionné sur ON qu'à détection d'un signal de porteuse valide sur la ligne téléphonique.

Questionnaire

Avant de faire appel à l'assistance technique, rassemblez les informations suivantes :

- Version du système d'exploitation ? Depuis quant l'utilisez–vous ?
- Le modem a-t-il déjà fonctionné ?
- Type de modem utilisé ? Type du modem à l'autre extrémité de la connexion téléphonique ?
- Type de carte (64 ports, 128 ports, S1...) raccordée au modem ?

- Numéro du port auquel le modem est connecté ?
- Numéro de tty auquel le modem est connecté ?
- Type de câblage utilisé ?
- Quelle configuration de connexion (share, delay, enable) ?
- Est-il possible de connecter votre modem à d'autres modems ?
- Est-il possible de connecter d'autres modems au vôtre ?
- Quelle sont les valeurs définies au niveau de Web-based System Manager, de SMIT, du modem ou du port, pour :
  - XON/XOFF ?
  - RTS/CTS ?
  - débit BPS ?
- Effectuez les vérifications suivantes :
  - Le port se verrouille-t-il par intermittence ?
  - Pouvez–vous composer un numéro d'appel ? Pouvez–vous recevoir des appels ?
  - Relevez-vous d'autres symptômes ?
- Constatez–vous des erreurs sur la console ? Lesquelles ?
- Ces erreurs figurent–elles dans le compte–rendu d'erreurs ? (**errpt** ou **errpt –a**)
- Quelle commande utilisez-vous pour composer un numéro d'appel ?
- Quels logiciels sont impliqués dans le système ?

Récapitulatif des commandes AT

Voici un récapitulatif du jeu de commandes Hayes Smartmodem. Ces commandes comprennent le jeu de commandes AT utilisé par un grand nombre de modems. Ces informations sont extraites de l'ouvrage Hayes Smartmodem 2400 *Quick Reference Card*, publié par Hayes Microcomputer Products, Inc. Pour en savoir plus sur les commandes AT principales, reportez–vous à la documentation du modem.

AT	Préfixe de commande, placé en tête de la ligne de commande.
<CR>	Retour chariot (ligne suivante), placé en fin de la ligne de commande.
A	Décroche, reste en mode commande.
A/	Répète la ligne de commande précédente. Commande ni précédée de **AT** ni suivie de **<CR>**/.
B0	Applique la norme CCITT V.22 pour les communications 1 200 bps.
B1	Applique la norme Bell 212A pour les communications 1 200 bps.
D	Entre en mode émission, compose le numéro qui suit et tente de passer en ligne. D est généralement suivi de T (tonalité) ou parfois de P (impulsion).
DS = *n*	Compose le numéro stocké à l'emplacement *n.*
E0	Désactive l'écho de caractère dans l'état "commande".
E1	Active l'écho de caractère dans l'état "commande".
H0	Décroche le téléphone.
H1	Fait fonctionner le support commutateur et le relais auxiliaire.
I0	Renvoie le code d'identification du produit.
I1	Calcule le total de contrôle sur le micrologiciel ROM et renvoie le résultat.

I2	Calcule le total de contrôle sur le micrologiciel ROM et renvoie en résultat OK ou ERROR.
L0	Haut-parleur désactivé.
L1	Règle le haut-parleur à un niveau sonore faible.
L2	Règle le haut-parleur à un niveau sonore moyen.
L3	Règle le haut-parleur à un niveau sonore élevé.
M0	Haut-parleur désactivé.
M1	Active le haut parleur jusqu'à détection d'un signal de porteuse.
M2	Haut-parleur toujours en service.
M3	Haut-parleur actif jusqu'à détection d'un signal de porteuse, sauf pendant la numérotation.
O0	Passe à l'état en ligne.
O1	Passe à l'état en ligne et lance une resynchronisation d'égalisation.
Q0	Le modem renvoie les codes de résultat.
Q1	Le modem ne renvoie pas les codes de résultat.
Sr	Positionne le pointeur sur le registre r.
Sr = *n*	Positionne le registre r à *n.*
V0	Affiche les codes de résultat sous forme numérique.
V1	Affiche les codes de résultat sous forme littérale (verbose).
X0	Active les fonctions représentées par les codes de résultat 0–4.
X1	Active les fonctions représentées par les codes de résultat 0–5, 10.
X2	Active les fonctions représentées par les codes de résultat 0–6, 10.
X3	Active les fonctions représentées par les codes de résultat 0–5, 7, 10.
X4	Active les fonctions représentées par les codes de résultat 0–7, 10.
Y0	Désactive la déconnexion long space.
Y1	Active la déconnexion long space.
Z	Réinitialise le modem.
&C0	Suppose la porteuse de données toujours présente.
&C1	Contrôle la présence de la porteuse de données.
&D0	Ignore le signal DTR.
&D1	Passe à l'état "commande" lors d'une désactivation de DTR.
&D2	Raccroche et passe à l'état "commande" lors d'une désactivation de DTR.
&D3	Réinitialise lors d'une désactivation de DTR.
&F	Réactive la configuration par défaut (d'usine).
&G0	Pas de tonalité de garde.
&G1	Tonalité de garde de 500 Hz.
&G2	Tonalité de garde de 1800 Hz.
&J0	Prise télécom RJ–11/RJ41/RJ45S.
&J1	Prise télécom RJ–11/RJ–13.
&P0	Numérote avec un rapport de conjonction/disjonction 39/61.
&P1	Numérote avec un rapport de conjonction/disjonction 33/67.
&Q0	Fonctionne en mode asynchrone.
&Q n	Fonctionne en mode synchrone *n.*

&R0 Contrôle la présence du signal CTS (prêt à émettre) en fonction du signal RTS (demande pour émettre).

&R1 Ignore le signal RTS et suppose la présence systématique d'un signal CTS.

&S0 Suppose la présence du signal DSR (modem prêt).

&S1 Contrôle la présence du signal DSR.

&T0 Met fin au test en cours.

&T1 Lance une boucle analogique locale.

&T3 Lance une boucle numérique.

&T4 Accepte une requête émise par un modem distant pour RDL.

&T5 Refuse une requête émise par un modem distant pour RDL.

&T6 Lance une boucle numérique distante.

&T7 Lance une boucle numérique distante avec autotest.

&T8 Lance une boucle analogique locale avec autotest.

&V Affiche la configuration active, les profils utilisateur et les numéros mémorisés.

&W n Sauvegarde les paramètres mémorisables de la configuration active comme profil utilisateur *n*.

&X0 Signal d'horloge de transmission émis par le modem.

&X1 Signal d'horloge de transmission émis par le terminal de données.

&X2 Signal d'horloge de transmission émis par la porteuse réceptrice.

&Y n Rappelle le profil utilisateur *n*.

&Z n= *x* Stocke le numéro de téléphone *x* à l'emplacement *n*.

Récapitulatif des registres S

Registre	Intervalle	Description
S0	0–255	Nombre de sonneries avant décrochage.
S1	0–255	Compteur de sonnerie (incrémenté à chaque sonnerie).
S2	0–127	Code ASCII du caractère d'échappement.
S3	0–127	Code ASCII du caractère de retour chariot.
S4	0–127	Code ASCII du caractère de ligne suivante.
S5	0–32, 127	Code ASCII du caractère de retour arrière.
S6	2–255	Délai, en secondes, entre le décroché et la numérotation.
S7	1–55	Délai, en secondes, entre la tonalité de porteuse et de numérotation.
S8	0–255	Durée, en secondes, de la pause marquée par une virgule.
S9	1–255	Délai minimum de réponse de détection de porteuse (en dixièmes de seconde).
S10	1–255	Temps entre la perte de porteuse et le raccrochage (en dixièmes de seconde).
S11	50–255	Durée/intervalle des tonalités (en millisecondes).
S12	50–255	Temps de garde avant et après la séquence d'échappement (en deux–centièmes de seconde).
S13	—	Réservé.
S14	—	Réservé.
S15	—	Réservé.

S16	—	Réservé – les fonctions de ce registre sont contrôlées par les commandes **&T**.
S17	—	Réservé.
S18	0–255	Durée du test du compteur (en secondes).
S19	—	Réservé.
S20	—	Réservé.
S21	—	Réservé.
S22	—	Réservé.
S23	—	Réservé.
S24	—	Réservé.
S25	0–255	Temps de détection de changement DTR (en centièmes de seconde).
S26	0–255	Délai entre une demande RTS et la réponse CTS (en centièmes de seconde).
S27	—	Réservé.

Récapitulatif des codes de résultat

Message numérique	Message littéral	Description
0	OK	Commande exécutée correctement.
1	CONNECT	Connexion établie à 0-300 bps.
2	RING	Détection d'une sonnerie.
3	NO CARRIER	Signal de porteuse perdu ou non détecté.
4	ERROR	Erreur de syntaxe dans la ligne de commande, ou commande, total de contrôle ou longueur invalide.
5	CONNECT 1200	Connexion établie à 1200 bps.
6	NO DIALTONE	Absence de tonalité d'invitation à numéroter.
7	BUSY	Détection d'une tonalité d'occupation.
8	NO ANSWER	Pas de réponse du système appelé.
9	CONNECT 2400	Connexion établie à 2400 bps.

Modificateurs de numérotation

Liste et description des modificateurs de numérotation :

0–9 # * A–D	Chiffres et caractères de numérotation
P	Réglage de l'impulsion
T	Réglage de la tonalité
,	Pause pour traitement du caractère suivant
!	Signal d'accroche
@	Attente d'un silence
W	Attente de tonalité
;	Retour à l'état "commande" après numérotation
R	Mode inversé
S= *n*	Compose le numéro stocké à l'emplacement *n.*

Définition des options de terminal avec stty–cxma

stty-cxma est un utilitaire qui définit et affiche les options du terminal pour les cartes PCI 8 et 128 ports. Il est situé dans le répertoire `/usr/lbin/tty`. Le format est le suivant :

```
stty-cxma [-a] [option(s)] [nomtty]
```

En l'absence d'options, **stty–cxma** affiche l'ensemble des définitions spéciales du pilote, les signaux du modem et tous les paramètres standard répertoriés par stty(1) pour l'unité tty désignée par l'entrée standard. Des options de commande sont fournies pour modifier les définitions du contrôle de flux, définir les options d'impression transparente, forcer les lignes de contrôle du modem et afficher les définitions du tty. Toute option non identifiée passe à stty(1) pour être interprétée. Voici les différentes options :

–a affiche l'ensemble des définitions de l'option de carte et toutes les définitions du tty standard fournies par la commande **stty –a**.

ttyname définit et affiche les options d'une unité tty donnée, et non de l'entrée standard. S'utilise avec le chemin d'accès à un tty précédé de `/dev/` ou avec un simple nom de tty précédé de tty. En l'absence de porteuse, peut aussi être utilisée sur une ligne de contrôle de modem.

Les options suivantes définissent des actions transitoires à exécuter immédiatement :

break émet un signal d'interruption de 250 ms sur la ligne tty.

flush indique un vidage immédiat des E/S tty.

flushin vide les entrées tty.

flushout vide les sorties tty.

Les options suivantes définissent des actions réinitialisées à la fermeture de l'unité. A l'ouverture suivante, l'unité prend en compte les valeurs par défaut.

stopout arrête les sorties comme à réception d'un caractère XOFF.

startout relance les sorties interrompues comme à réception d'un caractère XON.

stopin active le contrôle de flux pour arrêter les entrées.

startin désactive le contrôle de flux pour reprendre les entrées interrompues.

[–]dtr [drop] active la ligne de contrôle modem du DTR, excepté si le contrôle de flux matériel du DTR est sélectionné.

[–]rts [drop] active la ligne de contrôle modem RTS, excepté si le contrôle de flux matériel du RTS est sélectionné.

Les options suivantes restent effectives tant qu'elles ne sont pas modifiées ou jusqu'au réamorçage du système.

[–]fastcook traite les sorties en mode "préparé" sur la carte intelligente pour réduire l'utilisation du CPU hôte et améliorer les performances des entrées en mode brut.

[–]fastbaud modifie les tables des débits comme suit : 50 bauds deviennent 57 600 bauds ; 75, 76 800, 110, 115 200 et 200, 230 000 pour les unités prises en charge.

[–]rtspace active/désactive le contrôle de flux matériel en entrée du RTS, de sorte que la transmission à distance soit interrompue quand RTS est désactivé.

[–]ctspace active/désactive le contrôle de flux matériel en sortie du CTS, de sorte que la transmission locale soit interrompue quand CTS est désactivé.

[–]dsrpace active/désactive le contrôle de flux matériel en sortie du DSR, de sorte que la transmission locale soit interrompue quand DSR est désactivé.

[–]dcdpace active/désactive le contrôle de flux matériel en sortie du DCD, de sorte que la transmission locale soit interrompue quand DCD est désactivé.

[–]dtrpace active/désactive le contrôle de flux matériel en entrée du DTR, de sorte que la transmission à distance soit interrompue quand DTR désactivé.

[–]forcedcd désactive [réactive] la détection de porteuse pour permettre l'ouverture et l'exploitation du tty, même en l'absence de porteuse.

[–]altpin mappe le brochage du connecteur RJ–45 avec les valeurs du connecteur 8 broches ou celles par défaut du connecteur 10 broches. Quand altpin est activé (**enabled**), la position de DSR et DCD est activée pour que le signal DCD soit disponible avec un connecteur RJ–45 8 broches à la place du 10 broches. La valeur par défaut est **disable**.

Valeurs possibles :

enable (spécifie les valeurs du connecteur 8 broches)

disable (spécifie les valeurs du connecteur 10 broches)

startc *c* définit le caractère de contrôle du flux XON. Le caractère peut se voir associer un nombre hexadécimal, octal ou décimal. L'octal est identifié par le zéro de gauche et l'hexadécimal par 0x à gauche. Par exemple, le caractère standard XON, CTRL–Q, est 17 (décimal), 021 (octal) ou 0x11 (hexadécimal).

stopc *c* définit le caractère de contrôle de flux XOFF. Ce caractère peut se voir attribuer une valeur décimale, octale ou hexadécimale (pour le format octal ou décimal, se reporter à **startc**).

astartc *c* définit le caractère de contrôle de flux XON. Ce caractère peut se voir attribuer une valeur décimale, octale ou hexadécimale (pour le format octal ou décimal, se reporter à **startc**).

astopc *c* définit le caractère de contrôle de flux XON. Ce caractère peut se voir attribuer une valeur décimale, octale ou hexadécimale (pour le format octal ou décimal, se reporter à **startc**).

[–]aixon active le contrôle de flux auxiliaire, pour l'utilisation de deux caractères uniques XON et XOFF. Après réception des deux caractères XOFF, la transmission ne reprend qu'à réception des deux caractères XON.

[–]2200flow définit le mode de contrôle de flux 2200 sur le port. Les terminaux 2200 gèrent une imprimante connectée et utilisent quatre caractères de contrôle de flux : XON terminal (0xF8), XON imprimante (0xF9), XOFF terminal (0xFA) et XOFF imprimante (0xFB).

[–]2200print détermine le mode d'interprétation de ces caractères de contrôle de flux. Si 2200print est défini, exécutez un contrôle de flux indépendant pour les unités de terminal et d'impression transparente. Sinon, il est indissociable au niveau logique. A réception d'un caractère XOFF, toute sortie est interrompue jusqu'à réception du caractère XON correspondant.

maxcps *n* définit le débit maximal de transmission à l'unité d'impression transparente, exprimé en caractères par seconde (CPS). Un débit légèrement inférieur à la vitesse moyenne d'impression est préconisé. Un débit sous–estimé provoque un ralentissement de la vitesse d'impression. Avec un débit surestimé, l'imprimante effectue le contrôle de flux, ce qui peut ralentir l'entrée des données. La valeur par défaut est 100 caractères par seconde.

maxchar *n* définit le nombre maximal de caractères d'impression transparente placés par le pilote en file d'attente de sortie. Réduire la valeur accroît le temps système et l'augmenter accroît le temps de traitement des frappes de touches quand l'impression transparente est en cours d'exploitation. La valeur par défaut est 50 caractères.

bufsize *n* définit l'estimation par le pilote de la taille du tampon d'entrée d'impression transparente. Après une période d'inactivité, le pilote sature l'imprimante jusqu'à atteindre la taille impartie pour remplir le tampon puis ralentit jusqu'au débit maxcps. La valeur par défaut est 100 caractères.

onstr *s* définit la séquence d'échappement du terminal pour activer l'impression transparente. Les chaînes peuvent comporter des caractères ASCII d'impression et non imprimables. Les caractères de contrôle (non imprimables) doivent avoir une valeur octale de 3 chiffres précédés du symbole \. Par exemple, la valeur du caractère Echap, 33 en octal, doit être \033. Ainsi, avec la chaîne <Esc>[5i (norme ANSI) pour activer l'impression transparente, le paramètre doit avoir la valeur suivante : \033[5i

offstr *s* définit la séquence d'échappement du terminal pour désactiver l'impression transparente. Pour le format de la chaîne, reportez–vous à **onstr** *s*.

term *t* définit les chaînes d'activation/désactivation d'impression transparente aux valeurs données dans la table interne par défaut. Ces valeurs par défaut sont utilisées pour les terminaux suivants : adm31, ansi, dg200, dg210, hz1500, mc5, microterm, multiterm, pcterm, tvi, vp–a2, vp–60, vt52, vt100, vt220, wyse30, wyse50, wyse60 et wyse75. Pour tout autre type de terminal, ditty recherche dans terminfo l'entrée du type de terminal et définit les chaînes aux valeurs données par les attributs mc5/mc4 trouvés parmi les entrées terminfo.

Emulation ATE

L'émulation de terminal asynchrone ATE est un logiciel en option qui permet à un système d'émuler un terminal sur un système distant. ATE donne accès à la plupart des systèmes compatibles avec des terminaux asynchrones, y compris avec les connexions RS-232C ou RS-422A. Il est possible de configurer ATE de façon que le système distant perçoive votre terminal comme station de travail raccordée ou terminal DEC VT100.

Généralités sur la configuration d'ATE

Avant d'exécuter ATE, vous devez installer le logiciel et configurer les ports et connexions. ATE admet les connexions par câbles (directes) ou par modem. Les connexions RS-232C locales permettent de relier des machines distantes d'au maximum 15 mètres l'une de l'autre, les connexions RS-422A permettant d'aller jusqu'à 1 200 mètres.

Assurez–vous au préalable que l'unité tty à appeler (la vôtre ou celle du système distant) via ATE est prête à accepter l'appel. Assurez-vous au préalable que l'unité tty à appeler (la vôtre ou celle du système distant) via ATE est prête à accepter l'appel.

Pour en savoir plus sur l'installation et la configuration d'ATE, reportez–vous à la section "Configuration d'ATE".

Remarque : Vous ne pouvez utiliser ATE que si vous êtes membre d'un groupe UUCP (UNIX-to-UNIX Copy Program). Un superutilisateur (bénéficiant des droits d'accès root) peut utiliser Web-based System Manager ou définir ce type de groupe via SMIT.

Personnalisation d'ATE

Lors de la première exécution d'ATE, le programme crée un fichier **ate.def** par défaut dans le répertoire courant. Ce fichier regroupe les paramètres utilisés par ATE qui définissent :

- les caractéristiques de transmission de données,
- les fonctionnalités du système local,
- le fichier répertoire des numéros d'appel,
- Touches de contrôle

Pour modifier ces paramètres, éditez le fichier **ate.def**.

Si vous souhaitez disposer de plusieurs configurations d'ATE, conservez les versions correspondantes du fichier **ate.def** dans des répertoires distincts. Il suffit alors d'exécuter ATE à partir du répertoire ad hoc. Bien entendu, cette solution, qui nécessite plusieurs exemplaires du fichier **ate.def**, mobilise davantage d'espace de stockage système.

Pour plus d'informations sur l'édition du fichier **ate.def**, reportez-vous à la section "Edition du fichier par défaut d'ATE" dans le manuel *AIX 5L Version 5.2 System User's Guide: Communications and Networks* .

Pour changer temporairement la configuration sans modifier le fichier par défaut, vous disposez des sous–commandes **alter** et **modify**. Les modifications apportées par ce biais sont annulées dès que vous quittez le programme avec **quit**, et les valeurs du fichier **ate.def** sont de nouveau appliquées.

Une fois installé, ATE sollicite le fichier répertoire de numéros d'appel **/usr/lib/dir** du système. Vous pouvez le modifier temporairement – pour la durée de la connexion par modem : vous retrouvez alors les valeurs initiales dès la fin de la connexion et non à la sortie d'ATE. Un utilisateur racine peut ajouter dans le fichier **/usr/lib/dir** les numéros d'appel des modems utilisés par tous les utilisateurs. Chaque utilisateur a également la possibilité de créer ses propres fichiers répertoires et de modifier ses exemplaires du fichier **ate.def** pour permettre à ATE d'accéder à ces répertoires.

Pour en savoir plus sur l'utilisation d'ATE avec un répertoire de numéros d'appel personnalisé, reportez-vous à "Création d'un répertoire de numéros d'appel ATE" dans le manuel *AIX 5L Version 5.2 System User's Guide: Communications and Networks*.

Vous pouvez inclure dans le fichier répertoire les numéros d'appel fréquemment utilisés et modifier le débit, la longueur des données, les bits d'arrêt, la parité, l'écho et le retour de ligne d'un numéro d'appel. Pour établir la connexion avec un numéro non répertorié, lancez la sous-commande **connect**.

Remarque : Un fichier répertoire peut contenir jusqu'à 20 lignes (chacune correspondant à une entrée). Au-delà, les lignes sont ignorées par ATE.

Modification des caractéristiques ATE

Les caractéristiques d'ATE répertoriées ci-dessous peuvent être modifiées par l'utilisateur via la méthode indiquée.

Remarque : Toutes les caractéristiques d'ATE sont modifiables dans le fichier **ate.def**.

Modification des caractéristiques ATE

Caractéristique	**Via**
Touches de contrôle	Fichier **ate.def**
Longueur des données	**alter** ou entrée répertoire des numéros
Nom du répertoire de numéros d'appel	directory
Echo (activé ou non)	**modify** ou entrée répertoire des numéros
Nom du fichier de capture	modify
Suffixe de numérotation pour modem	alter
Préfixe de numérotation pour modem	alter
Retours de ligne	**modify** ou entrée répertoire des numéros
Nombre de tentatives de numérotation	alter
Nombre de bits d'arrêt	**alter** ou entrée répertoire des numéros
Parité (paire ou impaire)	**alter** ou entrée répertoire des numéros
Numéro de port (unité)	alter
Débit (bits/s)	**alter** ou entrée répertoire des numéros
Numéro de téléphone	entrée répertoire des numéros
Protocole de transfert (pacing ou xmodem)	alter
Espacement (caractère ou intervalle)	alter
Emulation VT100 (activée ou non)	modify
Délai entre deux tentatives de numérotation	alter
Capture des données entrantes	modify
Protocole Xon/Xoff (activé ou non)	modify

Configuration d'ATE

Cette section traite de la configuration d'ATE (Asynchronous Terminal Emulation)

Préalables

- Le programme ATE (Asynchronous Terminal Emulation) doit être installé sur votre système. ATE est un logiciel en option.
- Vous devez être utilisateur racine pour configurer le port de l'unité de communications.

Procédure

Effectuez les opérations suivantes pour préparer ATE à s'exécuter sur le système :

1. Installez une carte asynchrone dans l'unité centrale si le système n'est pas équipé d'un port série intégré.
2. Branchez le câble RS–232C ou RS–422A sur la carte ou le port série intégré.
3. Déclarez une unité tty pour le port de communication. Pour ce faire, utilisez le Gestionnaire système Web, **wsm**, ou entrez :

   ```
   smit mktty
   ```

4. Sélectionnez Ajout d'un TTY.
5. Sélectionnez le type tty.
6. Sélectionnez l'unité de carte parent.
7. Sélectionnez un port.
8. Sélectionnez l'option de désactivation **disable**du champ Enable LOGIN.
9. Définissez Type de terminal en tant que **HFT** ou **dumb**.
10. Modifiez l'environnement en conséquence, Les modifications les plus fréquentes portent sur la vitesse de la ligne, la parité, le nombre de bits par caractères et le mode de pilotage de la ligne (local ou distant). Indiquez `BPC 8` et `no parity` si le support de langue NLS est requis.
11. Configurez le port de l'unité.
    - Pour permettre des appels sortants via ATE, utilisez la commande **pdisable**. Par exemple, si le port est `tty1`, entrez :

      ```
      pdisable tty1
      ```

    - Pour permettre des appels entrants via ATE, utilisez la commande **penable**. Par exemple, si le port à appeler est `tty2`, entrez :

      ```
      penable tty2
      ```

12. Vérifiez que l'unité a été préalablement déclarée au système distant. Une fois l'unité définie, modifiez ATE en fonction de la configuration de l'unité du système distant. Pour personnaliser les paramètres par défaut, utilisez les sous–commandes **alter** et **modify**, ou édite le fichier **ate.def** par défaut. Pour une modification liée à une connexion téléphonique, utilisez une entrée du fichier répertoire des numéros d'appel.

Identification des incidents TTY

Cette section traite de l'identification des incidents du sous–système tty :

Régénération trop rapide

Le système enregistre le nombre de process getty générés pour un tty particulier dans un court laps de temps. Si ce nombre est supérieur à 5, l'erreur `Redémarrage trop rapide de la commande` s'affiche sur la console et le port est désactivé par le système.

Le tty reste désactivé environ 19 minutes ou jusqu'à ce que l'administrateur système le réactive. Au bout de ces 19 minutes, le système réactive automatiquement le port, générant un nouveau processus **getty**.

Causes possibles

- La configuration du modem est incorrecte
- Un port est défini et activé, mais aucune unité ou aucun câble ne lui est raccordé
- Le câblage est mauvais ou la connexion desserrée
- Il y a des bruits parasites sur la ligne de communication
- Les fichiers **/etc/environment** ou **/etc/inittab** sont altérés.
- La configuration de tty est erronée
- Le matériel est défectueux

Chacune de ces causes possible est expliquée dans Procédures de reprise page 7-30.

Procédures de reprise

- La configuration du modem est incorrecte :

  Vérifiez que le signal de détection de porteuse défini pour le modem *n'est* pas forcé à un niveau élevé.

  Remarque : Les instructions ci–dessous concernent les modems compatibles Hayes.

  1. Connectez le modem et examinez le profil actif.
  2. Le signal de détection de porteuse doit être positionné à **&C1** et non **&C0** (forcé à un niveau élevé). Utilisez les commandes AT ci–après pour définir ou modifier cet attribut :

     ```
     AT&C1
     AT&W
     Remarques :
     ```

     a. Envoi de commandes AT via la commande cu

     b. Pour plus d'informations à ce propos, reportez–vous au manuel du modem.

- Désactivez le tty, supprimez sa définition ou raccordez une unité au port :
  - Pour désactiver la définition tty, entrez la commande **chdev** comme suit :

    ```
    chdev -l   Nomtty   -a Login=disable
    ```

    Le tty *ne sera pas* réactivé au prochain démarrage du système.
  - Pour supprimer la définition tty :

1. Désactivez le port tty via la commande **pdisable** comme suit :

   ```
   pdisable  Nomtty
   ```

2. Supprimez la définition tty du système. Pour plus d'informations, reportez–vous à Gestion des unités TTY.

- Vérifiez les câbles et les connexions :
  1. Vérifiez les câbles. Resserrez les connexions et remplacez les connecteurs endommagés ou inadéquats.
  2. Vérifiez si le câblage vraisemblablement à l'origine de la défaillance est du type câble série référence 6323741 ou si les câbles répondent à la même norme. Remplacez les câbles endommagés ou inappropriés.
- Eliminez les bruits parasites sur la ligne :
  1. Vérifiez la longueur et l'impédance des câbles.
  2. Vérifiez que les bagues de serrage requises sur les câbles longs sont en place.
  3. Contrôlez le parcours des câbles, ils doivent être éloignés des lumières fluorescentes et des générateurs.
- Assurez–vous que les fichiers **/etc/environment** ou **/etc/inittab** ne sont pas endommagés ou n'ont pas été altérés :
  1. Si possible, comparez ces fichiers à des copies fiables.
  2. Faites une sauvegarde de ces fichiers et modifiez–les.
  3. Dans le fichier **/etc/environment**, supprimez les lignes *autres que* :
     - les lignes vierges,
     - les lignes de commentaire
     - *variable*=*valeur*.
  4. Vérifiez, dans le fichier **/etc/inittab**, les lignes relatives aux unités tty. Si tty est positionné sur off, le port tty n'est sans doute pas utilisé. S'il n'est pas utilisé, supprimez la définition tty ou attachez une unité au port.
- Supprimez les éléments de configuration de tty erronés :
  1. Supprimez la définition tty. Utilisez l'application Web-based System Manager Devices ou reportez–vous à Gestion des unités TTY pour plus d'informations.
  2. Pour effectuer une copie papier de cette définition avant de la supprimer, appuyez sur F8 ou Echap+8 (Image). une capture d'écran courant est réalisée et copiée dans le fichier **smit.log** de votre répertoire **$HOME**.
  3. Examinez la définition de tty. Consultez les instructions sur l'ajout d'un TTY à la section Gestion des unités TTY.
- Localisez le matériel défectueux :
  1. Exécutez les programmes de diagnostic à l'aide de la commande **diag**.
  2. Si vous décelez la moindre anomalie matérielle, suivez les procédures de résolution des incidents locaux.

Informations journalisées et identificateurs de journal TTY

Cette section présente les principaux fichiers et commandes de journalisation des erreurs ainsi que les messages d'erreur courants concernant les unités tty.

Fichiers et commandes de journalisation des erreurs

Commande : **errclear**

Cette commande supprime les entrées du journal d'erreur. Vous pouvez supprimer soit la totalité du journal avec la commande `errclear 0`, soit certaines entrées seulement en spécifiant des ID, une classe ou un type de messages.

Commande : **errpt**

Cette commande génère un compte rendu d'erreurs à partir des entrées du journal d'erreur du système. Le format le plus utilisé pour cette commande, `errpt -a | pg`, demande la génération d'un compte rendu détaillé avec, en tête, les erreurs les plus courantes.

Fichier : **/var/adm/ras/errlog**

Ce fichier stocke les occurrences d'erreurs et de défaillances détectées par le système. Le fichier **errlog** a tendance à s'allonger. S'il n'est pas régulièrement purgé, ce fichier peut mobiliser un espace disque important. Utilisez la commande **errclear** citée plus haut pour le purger.

Fichier : **/usr/include/sys/errids.h**

Le fichier d'en–tête **errids.h** fait la corrélation entre les ID d'erreur et leurs étiquettes.

Messages d'erreurs

Message	Description	Remarques
Core Dump	Arrêt anormal du programme	Cette erreur est consignée lors d'un arrêt anormal d'un programme entraînant un vidage de la mémoire. L'utilisateur n'a pas quitté proprement les applications, le système s'est arrêté en cours d'application, ou le terminal de l'utilisateur s'est bloqué et a interrompu l'application.
Errlog On	Activation du démon Err	Message consigné par le démon **error** dès le lancement de la journalisation. Le système désactive automatiquement la journalisation lors de l'arrêt du système (shutdown).
Lion Box Died	Perte de communication avec concentrateur 64 ports	Message consigné par le pilote du concentrateur 64 ports. Vérifiez l'horodateur pour déterminer si un utilisateur est à l'origine de ce message. Une série de messages de ce type peut révéler une défaillance de la carte 64 ports ou du matériel associé.
Lion Buffero	Saturation du tampon : concentrateur 64 ports	Le tampon matériel du concentrateur 64 ports est saturé. Si l'unité et le câblage le permettent, ajoutez une demande RTS au port et à l'unité et si possible, réduisez le débit en bauds.

Lion Chunknumc	Décompte erroné dans une tranche de mémoire : Contrôleur 64 ports	Le nombre de caractères compris dans une tranche de mémoire ne concorde pas avec les valeurs effectivement en mémoire tampon. Cette erreur peut signaler un incident matériel ; exécutez les diagnostics sur les unités.
Lion Hrdwre	Mémoire du contrôleur 64 ports inaccessible	Message consigné par le pilote du concentrateur 64 ports lorsqu'il ne parvient pas à accéder à la mémoire du contrôleur.
Lion Mem ADAP	Allocation de mémoire impossible : structure ADAP	Message consigné par le pilote du concentrateur 64 ports si la routine **malloc** pour la structure adap échoue.
Lion Mem List	Allocation de mémoire impossible : liste TTYP_T	Message consigné par le pilote du concentrateur 64 ports si la routine **malloc** pour la structure de liste *ttyp_t* échoue.
Lion Pin ADAP	Echec de la routine pin : structure ADAP	Message consigné par le pilote du concentrateur 64 ports si la routine **pin** pour la structure adap échoue.
SRC	Erreur du programme	Message consigné par le démon SRC (System Resource Controller) en cas de défaillance Les conditions anormales sont divisées en trois zones : des sous–systèmes, des communications et d'autres éléments.
Lion Unkchunk	Code d'erreur inconnu issu du concentrateur 64 ports	Code d'erreur : nombre de caractères reçus dans la tranche de mémoire.
TTY Badinput	Câble ou connexion défectueux	Le port génère une entrée plus rapidement que le système ne peut l'accepter et une partie de cette entrée est abandonnée. En général, l'entrée incorrecte est causée par un ou plusieurs signaux RS–232 qui changent d'état rapidement et à plusieurs reprises sur un court intervalle de temps, le système passant alors beaucoup de temps dans le gestionnaire d'interruptions. Les erreurs de signal sont en général causées par un connecteur lâche ou brisé, par un câble défectueux, non mis à la terre ou non blindé, ou par une liaison de communications parasitée.

TTY Overrun	Surcharge en entrée côté récepteur	La plupart des ports TTY ont un FIFO en entrée de 16 caractères et le paramètre par défaut indique qu'une interruption est publiée après la réception de 14 caractères. Cette erreur est signalée lorsque le gestionnaire d'interruption du pilote a effacé le FIFO en entrée et que des données ont été perdues. Les solutions possibles dépendent du matériel utilisé : • cartes 8 et 128 ports Vérifiez que le contrôle de flux est configuré correctement. Si c'est le cas, exécutez les diagnostics, et remplacez le matériel si nécessaire. • Ports natifs Si le problème se produit sur un système inactif, transférez la charge de travail sur un autre port. Si cela corrige le problème, mettez à niveau le microcode système. • Solutions générales – Réduisez la valeur du paramètre "RECEIVE trigger level" de ce port de 3 à 2 ou 1. – Réduisez la vitesse de ligne de ce port. – Examinez d'autres unités et processus afin d'essayer de réduire la durée consacrée par le système aux interruptions désactivées.
TTY TTYHOG	Saturation de TTYHOG	Cette erreur est en général causée par une absence de correspondance dans la méthode de contrôle de flux utilisée entre le transmetteur et le récepteur. Le pilote TTY a essayé à plusieurs reprises de demander au transmetteur de marquer une pause, mais l'entrée ne s'est pas arrêtée, et les données ont été abandonnées. Vérifiez les méthodes de contrôle de flux configurées à chaque extrémité pour vous assurer que la même méthode est utilisée sur chacune.

TTY Parerr	Erreur de parité/encadrement en entrée	Erreurs de parité sur les données entrantes au niveau des ports asynchrones, en mode caractère par caractère. Ceci est en général du à une absence de correspondance dans les paramètre de contrôle de ligne (parité, vitesse de ligne, taille de caractère, ou nombre de bits d'arrêt) entre le transmetteur et le récepteur. Les paramètres de contrôle de ligne doit être définis de façon identique des deux côtés pour pouvoir communiquer.
TTY Prog PTR	Erreur interne du pilote	Message consigné par le pilote tty si le pointeur *t_hptr* est nul.

Déblocage d'un port tty bloqué

Dans cet exemple, supposez que le port tty bloqué est `tty0`. Vous devez être utilisateur root.

1. Tapez ce qui suit pour déterminer si le tty gère actuellement des processus :

   ```
   ps –lt tty0
   ```

   La commande affiche un résultat semblable à ce qui suit :

   ```
        F S UID   PID  PPID   C PRI NI ADDR    SZ    WCHAN    TTY  TIME CMD
    240001 S 202 22566  3608   0  60 20 781a   444 70201e44   tty0  0:00 ksh
   ```

   L'ID de processus (PID) est ici 22566. Pour mettre fin à ce processus, tapez :

   ```
   kill 22566
   ```

   Vérifiez que le processus a été annulé en tapant la commande ps –lt tty0. Si le processus existe encore, ajoutez l'indicateur –9 flag à la commande kill comme le montre l'exemple suivant.

   Remarque : N'utilisez pas l'option –9 pour mettre fin à un processus slattach. Si vous mettez fin à un processus slattach avec l'indicateur –9, un verrouillage slip risque de rester dans le fichier **/etc/locks**. Supprimez ce fichier de verrouillage pour le nettoyage après slattach.

   ```
   kill –9 22566
   ```

2. Pour savoir si un processus tente d'utiliser le tty, tapez :

   ```
   ps –ef | grep tty0
   ```

   Remarque : Si **ps –ef | grep tty** renvoie un résultat du type suivant :

   ```
        root 19050      1    0    Mar 06       –  0:00 /usr/sbin/getty
       /dev/tty
   ```

   où le "–" s'affiche entre la date (Mar 06) et l'heure (0:00), ce tty n'a pas le câble adéquat. Ce statut indique que le processus de connexion du système (getty) tente d'ouvrir ce tty, et que le processus d'ouverture est bloqué parce que le signal RS–232 de détection de la porteuse de données (DCD) n'est pas déterminé. Pour résoudre ce problème, utilisez l'adaptateur null modem du câblage. Lorsque getty peut ouvrir le port tty, le "–" est remplacé par le nombre tty. Pour plus d'informations sur les câbles, reportez–vous à Connexion du modem avec les câbles adéquats.

Remarque : La commande suivante permet de désactiver le processus de connexion sur `tty0`.

```
pdisable tty0
```

Si le processus a été annulé mais que le tty ne répond toujours pas, passez à l'étape suivante.

3. Entrez la commande suivante :

   ```
   fuser -k /dev/tty0
   ```

   Ceci annule tous les processus qui s'exécutent sur le port et affiche le PID. Si le tty est toujours inutilisable, passez à l'étape suivante.

4. Utilisez la commande **strreset** pour vider les données en sortie du port bloqué par des données ne pouvant être transmises parce que la connexion à l'extrémité éloignée a été perdue.

   Remarque : Si la commande **strreset** répare le port bloqué, cela signifie que le port a un problème de câble de ou de configuration, car la perte de la connexion à l'extrémité éloignée a entraîné le vidage automatique des données en tampon.

   Vous devez d'abord déterminer les numéros d'unité principales et mineure du tty en tapant :

   ```
   ls -al /dev/tty0
   ```

   La commande affiche un résultat semblable à ce qui suit :

   ```
   crw-rw-rw-    1 root     system    18,    0 Nov  7  06:19 /dev/tty0
   ```

   Ceci indique que `tty0` a un numéro d'unité principale de 18 et un numéro d'unité mineure de 0. Indiquez ces chiffres lorsque vous utilisez la commande **strreset** comme suit :

   ```
   /usr/sbin/strreset -M 18 -m 0
    Si le tty est toujours inutilisable, passez à l'étape suivante.
   ```

5. Détachez et rattachez le câble du port tty bloqué. AIX utilise le signal Data Carrier Detect (DCD) pour détermine la présence d'un unité attachée au port. En abandonnant DCD, le fait de détacher et rattacher le câble, dans beaucoup de cas, annule les processus bloquées.

   Pour déterminer l'emplacement du port sur lequel le tty est configuré, tapez :

   ```
   lsdev -Cl tty0
   ```

   Le résultat est présenté de la manière suivante :

   ```
   tty0    Available   00-00-S1-00  Asynchronous Terminal
   ```

   La troisième colonne du résultat précédent indique le code d'emplacement du tty. dans cet exemple, `S1` indique que le port série est configuré pour le port série natif 1. Pour plus d'informations sur l'interprétaion des codes d'emplacement, reportez–vous au document Location Codes dans *AIX 5L Version 5.2 System Management Concepts: Operating System and Devices*.

   Si le tty est toujours inutilisable, passez à l'étape suivante.

6. Videz le port avec **stty–cxma**. Entrez la commande suivante :

   ```
   /usr/lbin/tty/stty-cxma flush tty0
   ```

   Cette commande est destinée aux ttys configurés sur les ports de cartes à 8 et 128 ports. Dan certains cas, cependant, elle permet de vider d'autres ports tty.

   Si le tty est toujours inutilisable, passez à l'étape suivante.

7. Sur le clavier du terminal bloqué, maintenez enfoncée la touche `Ctrl` et appuyez sur `Q`. Les résultats mis en suspens sont relancés par l'envoi d'un caractère **Xon**.

Si le tty est toujours inutilisable, passez à l'étape suivante.

8. Un programme ouvre parfois un port tty, modifie certains attributs et ferme le port sans réinitialiser l'état original des attributs. Pour corriger ce problème, attribuez l'état DEFINED au tty puis rendez–le disponible en tapant :

   ```
   rmdev –l tty0
   ```

   Cette commande conserve les informations sur le tty dans la base de données, mais rend le tty indisponible dans le système.

   La commande suivante réactive le tty :

   ```
   mkdev –l tty0
   ```

   Si le tty est toujours inutilisable, vous devez envisager de déplacer l'unité vers un autre port et de configurer un tty à cet emplacement jusqu'à ce que le système puisse être redémarré. Si le redémarrage n'efface pas le port, vous avez sans doute un problème matériel. Vérifiez le rapport d'erreur des problèmes matériels de port en entrant ce qui suit :

   ```
   errpt –a | pg
   ```

   Remarque : Certaines des commandes précédentes ne fonctionnent pas et donnent une erreur de méthode indiquant que l'unité est occupée. Ceci est du au processus s'exécutant sur le tty. Si aucune des étapes indiquées ci–dessus ne libèrent le tty bloqué, en dernier ressort, redémarrez le système AIX et videz le noyau afin d'éliminer le processus.

Chapitre 8. Protocole DLC

GDLC (Generic Data Link Control) est la définition d'une interface générique qui fournit aux utilisateurs niveau noyau et application, un jeu de commandes pour contrôler les gestionnaires d'unité DLC (Data Link Control) au sein du système d'exploitation. Pour la détermination des incidents, reportez–vous à GDLC Problem Determination dans le manuel *AIX 5L Version 5.2 Communications Programming Concepts*. Cette section traite des points suivants :

Environnement GDLC – généralités

GDLC (Generic Data Link Control) est la définition d'une interface générique qui fournit aux utilisateurs niveau noyau et application, un jeu de commandes pour contrôler les gestionnaires d'unité DLC (Data Link Control) au sein du système d'exploitation.

Pour en savoir plus sur l'environnement GDLC, reportez-vous à :

- Installation de l'interface GDLC page 8-5
- Installation de DLC, page 8-6
- Opérations ioctl sur l'interface GDLC, page 8-7
- Services spéciaux du noyau, page 8-10

L'interface GDLC indique les contraintes de définition des points d'entrée, les fonctions fournies et les structures de données pour tous les gestionnaires d'unité DLC. On trouve parmi les normes DLC conformes à l'interface GDLC :

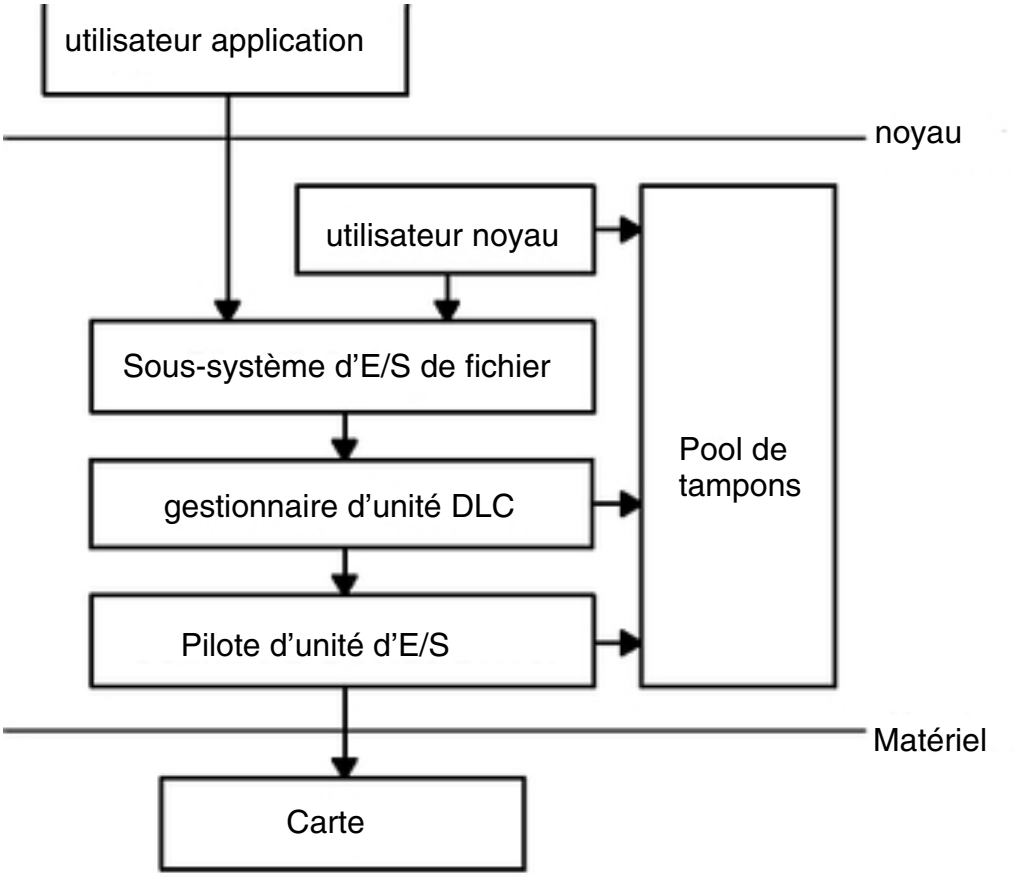
- 8023 (IEEE 802.3 pour Ethernet)
- ETHER (Ethernet standard)
- SDLC (Synchronous Data Link Control)
- TOKEN (anneau à jeton)
- FDDI (Fiber Distributed Data Interface)

Pour des performances optimales, les gestionnaires d'unité DLC sont implantés dans le noyau, mais ils appliquent des protocoles de la couche haute et des fonctions de portée plus étendue que celle d'un gestionnaire d'unité du noyau. Pour leurs requêtes d'E/S vers la carte, ils utilisent un gestionnaire d'unité du noyau. Quant aux utilisateurs DLC, ils se trouvent au sein ou au-dessus du noyau.

SDLC (Synchronous data link control) et IEEE 802.2 DLC sont des exemples de gestionnaires d'unité DLC. Chaque gestionnaire d'unité DLC fonctionne avec un pilote d'unité ou un groupe de pilotes d'unité spécifiques. Par exemple, SDLC fait fonctionner le pilote d'unité multiprotocole pour le produit du système et la carte associée.

La structure de base d'un environnement DLC est illustrée à la figure ci- dessous.
Les utilisateurs du noyau ont accès aux tampons de communications et appellent les points d'entrée add par les services du noyau fp. Les utilisateurs au dessus du noyau (niveau application) accèdent aux pilotes standard interface/noyau : le système de fichiers appelle les points d'entrée dd. Les données sont transférées de l'utilisateur à l'espace noyau.

Figure 33. Environnement du gestionnaire de périphérique DLC Cette illustration représente la liaison entre l'utilisateur de l'application et l'adaptateur (niveau matériel). Les zones intermédiaires sont Utilisateur du noyau, Sous–système d'E/S de fichiers, Gestionnaire de périphérique DLC, Piote d'unité d'E/S de communication, et Tampon. Ces entités intermédiaires se situent au niveau du noyau.



Composants de l'environnement du gestionnaire d'unité DLC :

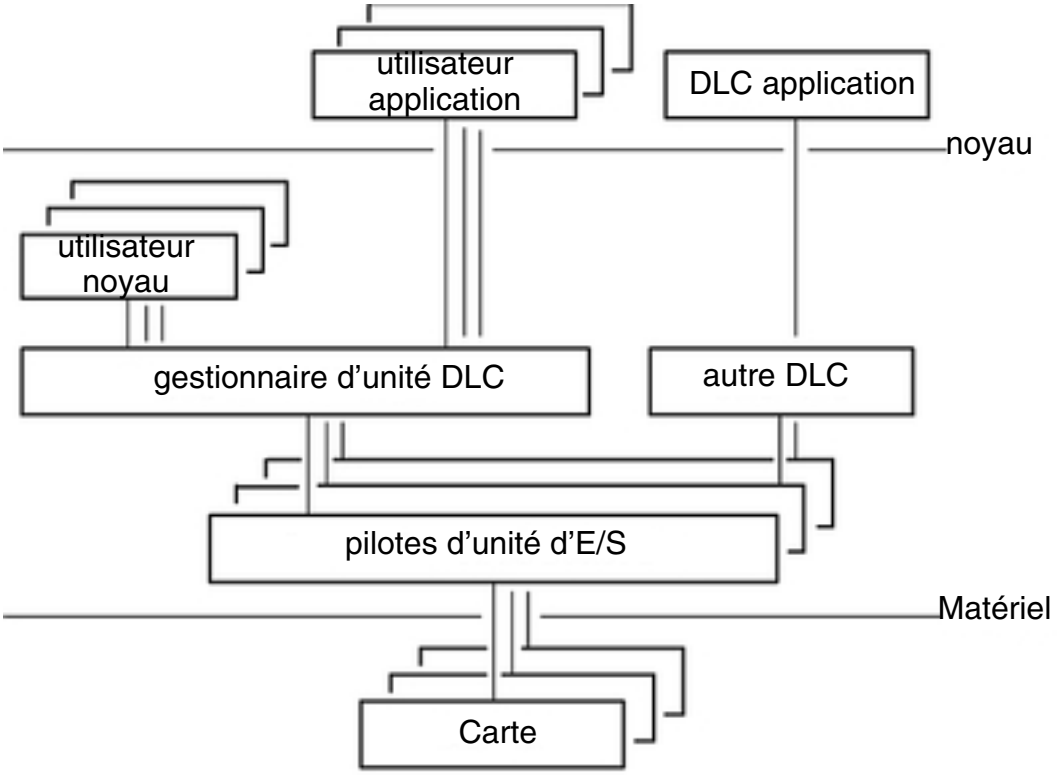
utilisateur application	Réside au-dessus du noyau comme application ou méthode d'accès.
utilisateur noyau	Réside dans le noyau comme process noyau ou gestionnaire d'unité.
Sous-système d'E/S de fichier	Dirige les routines de descripteur et de pointeur de fichiers vers les accès de pointeur de fichier de la table de localisation.
Pool de tampons	Fournit les services des tampons de données aux sous-systèmes de communication.
Pilote d'unité d'E/S	Contrôle les registres DMA et d'E/S de carte et achemine les paquets vers les différents DLC.
Carte	Se raccorde au support de communication.

Un gestionnaire d'unité conforme aux spécifications GDLC est compatible avec toute configuration matérielle du système d'exploitation comportant un pilote d'unité de communication et sa carte cible. Chaque gestionnaire d'unité peut prendre en charge plusieurs utilisateurs au-dessus, et plusieurs cartes et pilotes d'unité au-dessous. En général, les utilisateurs travaillent simultanément sur une seule carte, ou

individuellement sur plusieurs cartes. Les gestionnaires d'unité DLC varient en fonction de leurs contraintes de protocoles.

La figure ci-dessous illustre une configuration multi-utilisateur :

Figure 34. Configuration à utilisateurs et cartes multiples Le graphique ci–dessous est une autre vue du niveau noyau entre l'utilisateur de l'application et la carte. Les différentes entités représentent les différents utilisateurs.



Critères GDLC

Une interface GDLC doit présenter les caractéristiques suivantes :

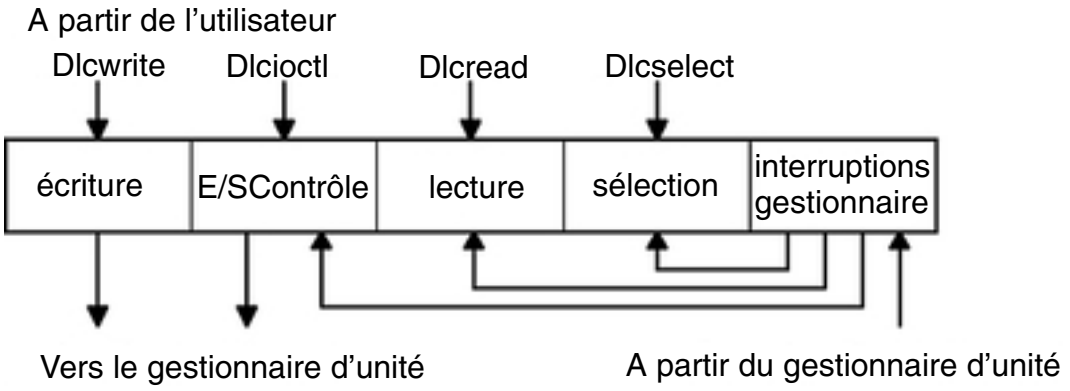
- souplesse et accessibilité aux utilisateurs niveau noyau et application,
- fonctions multi–utilisateurs et multicartes pour permettre aux protocoles d'exploiter les ports et sessions multiples,
- Support des services orientés connexion et sans connexion lorsque c'est possible.
- transfert des données transparent, dans le cas de contraintes spéciales dépassant la portée du gestionnaire d'unité DLC utilisé.

Mise en oeuvre de l’interface GDLC

Chaque gestionnaire d’unité DLC correspond à une entrée **/dev** standard qui fonctionne au niveau du noyau comme un gestionnaire de multiplexeur pour un protocole spécifique. Chaque sous-routine **open** soumise à un gestionnaire d’unité DLC pour une carte non utilisée par DLC créé un process noyau. Une sous-routine **open** est également transmise au gestionnaire d’unité de la carte cible. Au besoin, émettez des sous-routines **open** supplémentaires pour les divers ports de carte DLC du même protocole. Celles dirigées vers le même port ne créent pas de process noyau supplémentaires mais relient la sous–routine au process existant. On compte toujours un process noyau par port utilisé.

La structure interne d’un gestionnaire d’unité DLC est identique à la structure de base d’un gestionnaire d’unité du noyau, à la différence qu’un process noyau remplace le gestionnaire des interruptions pour les événements asynchrones. Le mécanisme de contrôle des E/S, l’écriture et la lecture, et les blocs de sélection sont illustrés à la figure ci-dessous :

Figure 35. Gestionnaire de périphérique de noyau standard Cette illustration représente la structure interne d’un gestionnaire de périphérique DLC. Cette structure se compose d’une écriture, d’un contrôle d’E/S, d’une lecture, d’une sélection et d’un gestionnaire d’interruption. Le gestionnaire de périphérique reçoit des informations de l’utilisateur lorsqu’elles sont transmises aux différentes zones au Gestionnaire de périphérique.



Installation de DLC

Vous pouvez installer les DLC séparément ou par groupe. Un gestionnaire d'unité DLC est automatiquement ajouté au noyau et rendu disponible pour chaque type de DLC installé. Pour vérifier l'installation, lancez la commande **lslpp** :

```
lslpp -h dlctype
```

en spécifiant pour *typedlc* l'un des DLC suivants :

bos.dlc.8023	DLC IEEE Ethernet (802.3)
bos.dlc.ether	DLC Standard Ethernet
bos.dlc.fddi	DLC FDDI
bos.dlc.sdlc	DLC SDLC
bos.dlc.token	DLC anneau à jeton

Vous pouvez afficher les informations relatives à un DLC installé via SMIT (System Management Interface Tool) ou à partir de la ligne de commande. Sur les ports de communication et systèmes très sollicités, il peut être nécessaire de modifier les attributs DLC pour optimiser les performances DLC. Si la réception est longue et que le journal des erreurs système signale une surcharge sur la file d'attente d'appels entre le DLC et son gestionnaire, augmentez sa capacité pour les données entrantes. Enfin, retirez un DLC installé à partir du noyau s'il est inutilisé pendant un certain temps. Il n'est pas supprimé du système, mais des ressources noyau sont libérées. Les instructions associées figurent à la section Gestion des pilotes d'unités DLC, page 8-12.

Opérations ioctl sur l'interface GDLC

L'interface GDLC prend en charge les opérations de sous-routines **ioctl** :

DLC_ENABLE_SAP	Active un point d'accès au service (SAP).
DLC_DISABLE_SAP	Désactive un SAP.
DLC_START_LS	Lance une station de liaison sur un SAP particulier comme appelant ou appelé.
DLC_HALT_LS	Interrompt une station de liaison.
DLC_TRACE	Suit l'activité d'une station de liaison (activités longues ou courtes).
DLC_CONTACT	Contacte une station distante pour une station de liaison locale particulière.
DLC_TEST	Teste la liaison vers une station distante pour une station de liaison locale particulière.
DLC_ALTER	Modifie les paramètres de configuration d'une station de liaison.
DLC_QUERY_SAP	Recherche les données statistiques d'un SAP.
DLC_QUERY_LS	Recherche les données statistiques d'une station de liaison.
DLC_ENTER_LBUSY	Passe en mode local-busy sur une station de liaison.
DLC_EXIT_LBUSY	Sort du mode local-busy sur une station de liaison.
DLC_ENTER_SHOLD	Passe en mode short- hold sur une station de liaison.
DLC_EXIT_SHOLD	Sort du mode short- hold sur une station de liaison.
DLC_GET_EXCEP	Renvoie des notifications d'exceptions asynchrones à l'utilisateur niveau application. **Remarque:** Cette opération de sous-routine **ioctl** n'est pas utilisée par l'utilisateur niveau noyau puisque toutes les conditions d'exception ont préalablement été filtrées par le gestionnaire d'exception.
DLC_ADD_GRP	Ajoute à un port un groupe ou une adresse de réception multi-destinataire.
DLC_DEL_GRP	Supprime d'un port un groupe ou une adresse de réception multi-destinataire.
DLC_ADD_FUNC_ADDR	Ajoute à un port un groupe ou une adresse de réception multi-destinataire.
DLC_DEL_FUNC_ADDR	Supprime d'un port un groupe ou une adresse de réception multi-destinataire.
IOCINFO	Renvoie une structure décrivant le gestionnaire d'unité GDLC. Pour en savoir plus, reportez-vous au format de fichier **/usr/include/sys/devinfo.h**.

Point d’accès au service

Un point d’accès au service (SAP) identifie un service utilisateur chargé d’envoyer et de recevoir une certaine classe de données. Ainsi, différentes classes de données peuvent être acheminées séparément vers leurs gestionnaires de service respectifs. Les DLC qui prennent en charge plusieurs SAP simultanément portent dans leur en-tête de paquet des adresses SAP source et destination. Ceux qui n’acceptent qu’un seul SAP n’ont pas besoin d’adressage SAP, mais l’activation du SAP est toujours requise. On compte généralement sur chaque port un SAP activé par utilisateur DLC.

La plupart des adresses SAP sont définies par les organismes de gestion de réseaux normalisés IEEE ou par les utilisateurs comme indiqué dans le manuel *Token–Ring Network Architecture Reference*. Voici quelques adresses SAP courantes :

Null SAP (0x00)	Permet de répondre à des noeuds distants même si aucun SAP n’est activé. Le SAP nul ne prend en charge que le service sans connexion et ne répond qu’aux LPDU (Link Protocol Data Unit) XID et TEST.
SNA Path Control (0x04)	Adresse SAP individuelle par défaut utilisée par les noeuds SNA.
PC Network NETBIOS (0xF0)	Utilisé pour toute communication DLC pilotée par émulation NETBIOS.
Discovery SAP (0xFC)	Utilisés par les services de noms LAN.
Global SAP (0xFF)	Identifie tous les SAP actifs.

Station de liaison

Une station de liaison (LS) identifie un raccordement entre deux noeuds pour une paire SAP. Cette liaison peut fonctionner comme service sans connexion (datagramme) ou orienté connexion (transfert intégralement suivi des données avec recouvrement des erreurs). Généralement, une station de liaison est lancée pour chaque téléraccordement.

Mode Local-Busy

En exploitation orientée connexion, une station de liaison doit arrêter l’émission des paquets en provenance de la station distante, en cas d’indisponibilité des ressources, par exemple. Il est alors possible d’avertir la station distante de faire passer la station locale en mode local–busy. Dès que les ressources sont de nouveau disponibles, la station locale en avertit la station distante qui peut reprendre l’émission des paquets. En mode local-busy, seuls les paquets d’informations séquencés sont arrêtés. Les autres types de données ne sont pas concernés.

Mode Short-Hold

Ce mode est particulièrement adapté aux réseaux de données pour lesquels :

- le délai d’établissement d’une connexion est court,
- le coût d’établissement de la connexion est faible par rapport à son coût d’utilisation.

En mode short-hold, la liaison entre deux stations est maintenue tant qu’il y a des données à transférer. Dès lors qu’il n’y a plus de données à envoyer, la liaison est interrompue (à l’expiration d’un délai défini) et n’est rétablie que lorsque des données sont de nouveau disponibles pour le transfert.

Test et suivi d’une liaison

Pour tester le raccordement de deux stations, demandez à une station de liaison d’émettre un paquet test à partir de la station locale. Ce paquet est renvoyé par la station distante si la liaison fonctionne correctement.

Certaines liaisons de données sont limitées par des contraintes de protocoles dans l’application de cette fonction. SDLC, par exemple, ne peut générer le paquet test qu’à partir de la station hôte ou principale. Néanmoins, la plupart des protocoles laissent toute latitude pour le choix de la station d’origine.

Pour suivre une liaison, les données de la ligne et les événements spéciaux (activation et fermeture d’une station, écoulement des délais, etc.) demandez à une station de liaison de consigner les canaux de suivi générique dans le dispositif de suivi générique de chaque station de liaison. Cette fonction permet de déterminer l’origine de certains incidents de raccordement. Les entrées de suivi longues et courtes sont toutes les deux prises en charge.

Statistiques

L’utilisateur GDLC dispose de deux services statistiques : les statistiques SAP, qui fournissent les informations et l’état SAP courants du gestionnaire d’unité ; et les statistiques LS, qui indiquent l’état courant de la station et des compteurs de fiabilité/disponibilité/maintenabilité (contrôlant l’activité de la station dès son lancement).

Services spéciaux du noyau

GDLC (Generic Data Link Control) met à la disposition de l'utilisateur noyau des services spéciaux. Le noyau doit cependant être doté d'un environnement sécurisé. A la différence du gestionnaire d'unité DLC qui copie les données des événements asynchrones dans un espace utilisateur, l'utilisateur noyau doit spécifier des pointeurs de fonction vers des routines spéciales appelées gestionnaires de fonction. Ces derniers sont appelés par le DLC lors de l'exécution, ce qui assure des performances maximales entre l'utilisateur noyau et les couches DLC. Il est demandé à chaque utilisateur noyau de limiter le nombre de gestionnaires de fonction à une longueur de chemin minimale et de suivre le schéma des tampons de mémoire de communication (mbuf).

Un gestionnaire de fonction ne doit jamais appeler une autre entrée DLC directement. En effet, les appels directs sont verrouillés, entraînant une mise en veille bloquante. Une seule exception à cette règle : l'utilisateur noyau peut appeler le point d'entrée **dlcwritex** pendant que ce dernier assure le service d'une des quatre fonctions de données de réception. Cet appel permet de générer immédiatement les réponses sans passer par un commutateur de tâche. Une logique spéciale est nécessaire dans le gestionnaire d'unité DLC pour contrôler l'identification de l'utilisateur sollicitant une opération en écriture. S'il s'agit d'un process DLC et que la capacité interne d'accueil en file d'attente DLC a été dépassée, l'écriture est renvoyée avec un code d'erreur (valeur retour EAGAIN) au lieu de mettre en veille le process appelant. La sous-routine du demandeur doit alors renvoyer une notification au DLC pour prévoir une nouvelle tentative du tampon récepteur.

Les gestionnaires de fonction disponibles sont les suivants :

Datagram Data Received Routine	Appelée chaque fois qu'un paquet datagramme arrive pour l'utilisateur noyau.
Exception Condition Routine	Appelée chaque fois qu'un événement asynchrone à signaler à l'utilisateur noyau se produit (SAP Closed ou Station Contacted, par exemple).
I–Frame Data Received Routine	Appelée chaque fois qu'un paquet normal de données séquencées arrive pour l'utilisateur noyau.
Network Data Received Routine	Appelée chaque fois qu'un paquet réseau spécifique arrive pour l'utilisateur noyau.
XID Data Received Routine	Appelée chaque fois qu'un paquet XID (exchange identification) arrive pour l'utilisateur noyau.

Les points d'entrée **dlcread** et **dlcselect** de DLC ne sont pas appelés par l'utilisateur noyau : les entrées asynchrones fonctionnelles sont appelées directement par le gestionnaire d'unité DLC. Normalement, la mise en file d'attente de ces événements doit intervenir dans le gestionnaire de fonction de l'utilisateur. Toutefois, si l'utilisateur noyau ne peut pas traiter un paquet, le gestionnaire d'unité DLC peut bloquer le dernier tampon reçu et passer dans l'un des deux modes user-busy :

User–Terminated Busy Mode (I–frame exclusivement)

Si l'utilisateur noyau ne peut pas traiter une trame–I reçue (suite à un incident tel un blocage au niveau de la file d'attente), un code DLC_FUNC_BUSY est renvoyé, et DLC bloque le pointeur de tampon et passe en mode local–busy pour interrompre l'émission des trames–I par la station distante. L'utilisateur du noyau doit appeler la fonction Exit Local Busy pour réinitialiser le mode local–busy et redémarrer la réception des trames–I. Seules les trames–I séquencées normales peuvent être arrêtées. Les données XID, de datagramme et de réseau ne sont pas concernées par le mode local–busy.

Timer–Terminated Busy Mode (tous types de trames)

Si l'utilisateur noyau ne peut pas traiter un paquet reçu et souhaite que DLC bloque un court instant le tampon de réception, puis rappelle la fonction de réception utilisateur, un code DLC_FUNC_RETRY est renvoyé au DLC. Si le paquet est une trame–I séquencée, la station passe en mode local–busy pendant ce délai. Dans tous les cas, un compte à rebours est lancé : à expiration, l'entrée fonctionnelle de données de réception est rappelée.

Gestion des pilotes d'unités DLC

Un DLC doit être installé pour pouvoir être ajouté au système. Chaque DLC installé est ajouté automatiquement après l'installation et à chaque redémarrage du système (voir Installation de GDLC page 8-6). Si un DLC a été supprimé sans être suivi d'un redémarrage, il peut être ajouté de nouveau.

Tâches de gestion des pilotes d'unité DLC

Tâche	**Raccourci SMIT**	**Commande ou fichier**	Web-based System Manager Management Environment [7]
Ajout d'un DLC installé	Choix possibles (par nom de pilote d'unité) : **smit cmddlc_sdlc** **smit cmddlc_token** **smit cmddlc_qllc** **smit cmddlc_ether** [1] **smit cmddlc_fddi** puis sélectionnez l'option **Add**	**mkdev** [2]	
Modification des attributs DLC[3,4]	Choix possibles (par nom de pilote d'unité) : **smit cmddlc_sdlc_ls** **smit cmddlc_token_ls** **smit cmddlc_qllc_ls** **smit cmddlc_ether_ls** [1] **smit cmddlc_fddi_ls**	**chdev** [2]	
Démarrage du suivi du moniteur LAN DLC[5]	**smit trace**	**trace –j *nnn***, la valeur *nnn* étant l'ID du point d'ancrage pour lequel un suivi est demandé.	
Arrêt du suivi du moniteur LAN DLC	**smit trcstop**	**trcstop** [2]	
Génération d'états sur le suivi du moniteur LAN DLC	**smit trcrpt**	**trcrpt –d *nnn***, la valeur *nnn* étant l'ID du point d'ancrage pour lequel un état est demandé.	

Affichage d'informations sur le DLC courant[3]	Choix possibles (par nom de pilote d'unité) : **smit cmddlc_sdlc_ls** **smit cmddlc_token_ls** **smit cmddlc_qllc_ls** **smit cmddlc_ether_ls** [1] **smit cmddlc_fddi_ls**	**lsdev** [2] ou **lsattr** @T>2	
Suppression d'un DLC[3,6]	Choix possibles (par nom de pilote d'unité) : **smit cmddlc_sdlc_rm** **smit cmddlc_token_rm** **smit cmddlc_qllc_rm** **smit cmddlc_ether_rm** [1] **smit cmddlc_fddi_rm**	**rmdev** [2]	

Remarques :

1. La commande SMIT d'ajout d'un gestionnaire d'unité Ethernet concerne aussi bien le gestionnaire standard que le gestionnaire IEEE 802.3.
2. Pour plus d'informations sur les options de ligne de commande, consultez les descriptions des commandes **mkdev**, **chdev**, **trace**, **trcstop**, **trcrpt**, **lsdev**, **lsattr**, ou **rmdev** dans le manuel *AIX 5L Version 5.2 Commands Reference*.
3. Un DLC doit être installé et ajouté avant que vous ne puissiez répertorier, afficher, modifier ou supprimer ses attributs (voir Installation de GDLC page 8-6). Modifier un attribut n'aboutit que si le DLC cible ne fait l'objet d'aucune ouverture active. Avant toute modification, il convient d'interdire à tous les services, tels que SNA, OSI ou NetBIOS, l'accès au DLC.
4. Modifier la taille de la file d'attente de réception a une incidence directe sur les ressources système. N'effectuez ce changement que si le DLC présente des failles au niveau de cette file d'attente (dégradation des performances ou surcharge entre le DLC et son gestionnaire, par exemple).
5. Soyez prudent si vous activez le suivi moniteur : cette fonction affecte directement les performances des DLC et unités associées.
6. Supprimer un DLC n'aboutit que si le DLC cible ne fait l'objet d'aucune ouverture active. Avant toute suppression, il convient d'interdire à tous les services, tels que SNA, OSI ou NetBIOS, l'accès au DLC.
7. Ces tâches ne sont pas disponibles dans l'environnement de gestion Web-based System Manager.

Chapitre 9. Utilitaires réseau (BNU)

Ce chapitre traite de l'installation, la configuration et la maintenance des utilitaires réseau (BNU). Il traite des points suivants :

Présentation de BNU

Les utilitaires BNU (Basic Networking Utilities) sont constitués d'un groupe de programmes, répertoires et fichiers, exploitables pour établir une communication avec un système UNIX sur lequel une version du programme UUCP (UNIXtoUNIX Copy Program) est active. Il s'agit de l'un des programmes de services étendus pouvant être installés avec le système d'exploitation de base.

BNU contient un groupe de commandes liées à UUCP, programme de communication UNIX vers UNIX développé par AT&T et modifié dans le cadre de la distribution Berkeley Software (BSD). BNU fournit des commandes, des processus et une base de données de support pour les connexions aux systèmes locaux et distants. Les réseaux de communication (tels TokenRing et Ethernet) servent à connecter des systèmes sur des réseaux locaux. Un réseau local peut être connecté à un système distant par un modem téléphonique ou un câble. Commandes et fichiers peuvent alors être échangés entre le réseau local et le système distant.

Cette section traite des points suivants :

- Fonctionnement de BNU
- Structure de répertoires et de fichiers BNU
- Sécurité de BNU
- Démons BNU

Les programmes BNU ne peuvent être exploités qu'une fois BNU installé et configuré.

BNU est contrôlé par un jeu de fichiers de configuration qui détermine si les systèmes distants peuvent se connecter au système local et ce qu'ils sont habilités à exécuter une fois la connexion établie. Ces fichiers de configuration doivent être configurés en fonction des impératifs et des ressources de votre système.

Pour la maintenance de BNU, vous devez lire et supprimer régulièrement les fichiers journaux, et vérifier les files d'attente BNU pour vous assurer que le transfert des travaux aux systèmes distants s'effectue correctement. Vous devez également mettre régulièrement à jour les fichiers de configuration pour y répercuter les modifications de votre système ou des systèmes distants.

Pour en savoir plus, reportez–vous à :

- Configuration de BNU
  - Configuration de BNU Informations préalables
- Maintenance de BNU
  - Fichiers journaux BNU
  - Commandes de maintenance BNU page 9-20.

Fonctionnement de BNU

BNU assure la communication entre systèmes via un ensemble de connexions matérielles et de logiciels. Une structure de répertoires et de fichiers suit à la trace les activités BNU. Cette structure intègre un jeu de répertoires publics, un groupe de répertoires et de fichiers administratifs, des fichiers de configuration et des fichiers de verrouillage. La plupart des répertoires BNU sont créés au cours de l'installation. Certains répertoires et fichiers administratifs sont créés par les différents programmes BNU.

A l'exception des commandes de connexion à distance, BNU fonctionne comme un système de traitement par lots. Lorsqu'un utilisateur demande qu'un travail soit envoyé à un système distant, BNU stocke les informations nécessaires. Cette opération s'appelle *mise en file d'attente* du travail. A des moments planifiés, ou à la demande d'un utilisateur, BNU contacte différents systèmes distants, transfère le travail en file d'attente et accepte d'autres travaux. Ces transferts sont contrôlés par les fichiers de configuration de votre système et par ceux du système distant.

Support NLS (National Language Support) pour les commandes BNU

Toutes les commandes BNU, sauf **uucpadm**, sont prises en charge par NLS (langue nationale). Les noms utilisateur ne doivent pas être forcément en caractères ASCII. Mais tous les noms de système doivent être en caractères ASCII. Si un utilisateur tente de planifier un transfert ou une exécution de commande à distance impliquant des noms système non ASCII, BNU renvoie un message d'erreur.

Structure des fichiers et répertoires BNU

Pour garder trace de ses activités, BNU a recours à une structure de répertoires et de fichiers. Cette structure inclut les éléments suivants :

- Répertoires BNU publics
- Fichiers de configuration BNU
- Répertoires et fichiers administratifs BNU
- Fichiers de verrouillage BNU

La plupart des répertoires BNU sont créés au cours de l'installation. Certains répertoires et fichiers administratifs sont créés au cours de l'exécution des différents programmes BNU.

Répertoires publics BNU

Après spécification, le répertoire public BNU (**/var/spool/uucppublic**) stocke les fichiers transférés sur le système local par d'autres systèmes. Les fichier restent en attente dans le répertoire public jusqu'à ce qu'un utilisateur vienne les chercher. Le répertoire public est créé au cours de l'installation de BNU. Dans le répertoire public, BNU crée autant de sous–répertoires que de systèmes distants envoyant des fichiers au système local.

Fichiers de configuration BNU

Les fichiers de configuration BNU, ou base de données de support BNU, se trouvent dans le répertoire **/etc/uucp**. Les fichiers doivent être configurés spécifiquement pour votre système. Ils sont propriété de l'ID de connexion uucp et ne peuvent être édités que par l'utilisateur racine. Les fichiers de configuration contiennent des informations sur :

- les systèmes distants accessibles,
- les unités permettant le contact avec les systèmes distants,
- les horaires d'accès aux systèmes distants,
- les actions autorisées aux systèmes distants sur votre système.

Certains fichiers de configuration spécifient également des limites aux activités de BNU pour éviter une surcharge de votre système.

Liste des fichiers de configuration :

Devices	Contient des informations sur les unités disponibles, notamment les modems et les connexions directes.
Dialcodes	Contient des codes de numérotation abrégés, permettant de raccourcir les numéros de téléphone dans le fichier **Systems**.
Dialers	Spécifie la syntaxe de commande d'appels pour un type de modem spécifique ("dialer").
Maxuuscheds	Limite les travaux programmés simultanément.
Maxuuxqts	Limite les exécutions simultanées de commandes à distance.
Droits d'accès	Contient les codes d'autorisation d'accès. Il s'agit du fichier de sécurité principal de BNU.
Poll	Définit les moments où le programme BNU doit demander aux systèmes distants de lancer les tâches.
Sysfiles	Répertorie les fichiers qui servent de fichiers **Systems**, **Devices** et **Dialers** pour la configuration BNU. Les fichiers par défaut sont **/etc/uucp/Systems**, **/etc/uucp/Devices** et **/etc/uucp/Dialers**.
Systems	Donne la liste des systèmes accessibles et des informations requises pour les contacter : unité à utiliser, combinaisons nom et mot de passe utilisateur requises pour la connexion, etc. Spécifie également les créneaux horaires pendant lesquels les systèmes peuvent être contactés.

Les fichiers de configuration se font mutuellement référence. Par exemple :

- Le fichier **Devices** contient un champ *Token* se rapportant aux entrées du fichier **Dialers**.
- Le fichier **Systems** contient une entrée par classe (Class) d'unité. Une unité de chaque *Class* mentionnée dans le fichier **Systems** doit être définie dans le fichier **Devices**.
- Le fichier **Poll** contient des entrées pour les systèmes appelés par le vôtre. Chaque système mentionné doit être défini dans le fichier **Systems**.

Les entrées des fichiers de configuration BNU dépendent du type des connexions entre votre système et chaque système distant. Par exemple, des entrées spéciales doivent être établies pour des connexions directes ou TCP/IP (Transmission Control Protocol/Internet Protocol). Si la connexion passe par des modems, ils doivent être définis dans le fichier **Dialers**.

Les fichiers **Systems**, **Devices** et **Permissions** doivent être configurés sur votre système pour que vous puissiez contacter des systèmes distants via BNU. D'autres fichiers de configuration donnent accès aux fonctions BNU telle l'interrogation automatique. La plupart de ces fichiers doivent être périodiquement modifiés pour refléter les changements opérés sur votre système ou sur les systèmes contactés. Le fichier **Sysfiles** peut servir à attribuer à d'autres fichiers le rôle des fichiers **Systems**, **Devices** et **Dialers**.

Répertoires et fichiers administratifs BNU

Les répertoires et fichiers administratifs BNU se trouvent dans des sous répertoires de **/var/spool/uucp**. Ils contiennent deux types d'informations :

- les données en attente de transfert vers d'autres systèmes,
- les informations de journalisation et d'erreur sur les activités BNU.

Dans le répertoire **/var/spool/uucp**, BNU crée les répertoires suivants :

.Admin	Contient quatre fichiers administratifs. • **audit** • **Foreign** • **errors** • **xferstats** Ces fichiers contiennent des informations de journalisation et d'erreur relatives aux activités BNU.
.Corrupt	Contient la copie des fichiers que le programme BNU ne peut pas traiter.
.Log et **.Old**	Contient les fichiers journaux issus des anciennes transactions BNU.
.Status	Prend date de la dernière tentative du démon **uucico** de communiquer avec les systèmes distants.
.Workspace	Contient les fichiers temporaires utilisés en interne par les programmes de transport de fichier.
.Xqtdir	Contient les fichiers exécutables avec les listes des commandes exécutables par les systèmes distants.
SystemName	Contient les fichiers utilisés par les programmes de transport de fichier. Ces fichiers sont : • Command (**C.***) • Data (**D.***) • Execute (**X.***) • Temporary (**TM.***) BNU crée un répertoire *SystemName* pour chaque système distant qu'il contacte.

Les répertoires dont le nom commence par un point sont *cachés*. Ils ne sont pas affichés par les commandes **ls** ou **li** sauf si elles sont assorties de l'indicateur **–a**. A son lancement, le démon **uucico** recherche dans le répertoire **/var/spool/uucp** les fichiers de travail et transfère les fichiers de tout répertoire non caché. Le démon **uucico** ne voit que les répertoires *SystemName*, à l'exclusion des autres répertoires administratifs.

Les fichiers des répertoires cachés sont propriété de l'ID de connexion uucp. Ils ne sont accessibles que par l'utilisateur racine ou via un ID de connexion dont l'UID est de 5.

Pour en savoir plus sur la maintenance des répertoires administratifs BNU, reportez-vous à "Maintenance de BNU", page 9-19.

Fichiers de verrouillage BNU

Ils sont stockés dans le répertoire **/etc/locks**. Lorsque BNU utilise une unité pour se connecter à un ordinateur distant, il place un fichier de verrouillage pour cette unité dans le répertoire **/var/locks**. Lorsqu'un autre programme (BNU ou non) a besoin de l'unité, il vérifie s'il existe un fichier de verrouillage dans **/var/locks**. Dans l'affirmative, le programme attend que l'unité soit disponible ou utilise une autre unité pour la communication.

En outre, le démon **uucico** place des fichiers de verrouillage dans le répertoire **/var/locks** pour les systèmes distants. Avant de contacter un système distant, le démon **uucico** vérifie la présence d'un fichier de verrouillage pour ce système dans **/var/locks**. Ces fichiers empêchent d'autres instances du démon **uucico** d'établir des connexions en double au même système distant.

Remarque : Outre BNU, d'autres logiciels, comme ATE (Asynchronous Terminal Emulation) et TCP/IP, utilisent le répertoire **/var/locks**.

Sécurité de BNU

D'autres systèmes prenant contact avec le vôtre pour se connecter, transférer des fichiers et lancer des commandes, BNU fournit des moyens d'assurer la sécurité. Les fonctions de sécurité BNU permettent de limiter les actions exécutables par les systèmes distants sur le système local (les utilisateurs des systèmes distants peuvent également limiter les actions que vous êtes habilité à effectuer). Pour ce faire, BNU exécute plusieurs démons et utilise les répertoires administratifs pour y stocker les fichiers dont il a besoin. Il conserve également un journal de ses propres activités.

La sécurité de BNU fonctionne à plusieurs niveaux. Lorsque vous configurez BNU, vous pouvez déterminer :

- les utilisateurs de votre système habilités à accéder aux fichiers BNU ;
- les systèmes distants accessibles par votre système ;
- le mode de connexion des utilisateurs distants à votre système ;
- les actions accessibles aux utilisateurs connectés à votre système.

ID de connexion uucp

A l'installation de BNU, tous les fichiers de configuration, les démons et nombre de commandes et de procédures shell appartiennent à l'ID de connexion uucp. L'ID de connexion uucp a un ID utilisateur (UID) de 5 et un ID de groupe (GID) de 5. Le démon **cron** lit le fichier **/var/spool/cron/crontabs/uucp** pour planifier les travaux automatiques pour BNU.

Il est en général interdit de se connecter comme utilisateur uucp. Pour modifier des fichiers appartenant à l'ID de connexion uucp, connectez–vous en tant qu'utilisateur racine.
Attention : Autoriser la connexion de systèmes distants au local system avec un ID de connexion uucp nuit gravement à la sécurité de votre système. En effet, les systèmes distants ainsi connectés peuvent afficher voire modifier (selon les droits d'accès définis dans l'entrée LOGNAME) les fichiers locaux **Systems** et **Permissions**. Il est donc vivement recommandé d'attribuer d'autres ID de connexion BNU aux systèmes distants et de réserver les ID UUCP aux administrateurs BNU du système local. Pour une sécurité optimale, chaque système distant appelé à communiquer avec le local system doit posséder un ID de connexion unique avec un UID unique.

Le système d'exploitation fournit un ID de connexion nuucp pour le transfert de fichiers.

ID de connexion BNU

Le shell de lancement pour les ID de connexion BNU est le démon **uucico** (**/usr/sbin/uucp/uucico**). Lorsque des systèmes distants appellent votre système, ils lancent automatiquement le démon **uucico** sur votre système. Les ID de connexion pour BNU ont un ID de groupe uucp de 5.

Les ID de connexion utilisés par les systèmes distants ont besoin de mots de passe. Pour éviter que, dans le cadre de la sécurité, un nouveau mot de passe soit demandé au nouvel ID de connexion BNU lorsque le système distant se connecte, vous devez définir ce mot de passe dès la création du compte. Pour ce faire, utilisez la commande **passwd** suivie de la commande **pwdadm**. Par exemple, pour définir un mot de passe pour l'ID de connexion `nuucp`, connectez–vous en tant qu'utilisateur racine et entrez les commandes :

```
passwd  nuucp


pwadm –f  NOCHECK
nuucp
```

Le système vous invite à indiquer un mot de passe pour l'ID de connexion `nuucp`. Mener à bien ces étapes permet au système distant de se connecter sans être immédiatement invité à entrer un nouveau mot de passe (que l'ID de connexion orientée traitement par lots `nuucp` ne peut fournir).

Après création de l'ID de connexion pour un système distant, communiquez le à l'administrateur de ce système et indiquez lui le mot de passe.

Création d'un ID de connexion administratif BNU

Un utilisateur racine peut définir un ID de connexion administratif BNU. Cette opération permet de déléguer des tâches d'administration BNU à un utilisateur ne détenant pas les droits racine. L'ID de connexion administratif BNU doit être sécurisé par mot de passe, doté d'un UID de5 et d'un GID (ID de groupe) uucp de5. Le shell de connexion administrative doit être le programme **/usr/bin/sh** (et non le démon **uucico**). Affecter à la connexion administrative BNU un UID de 5 lui confère les mêmes droits que l'ID de connexion uucp. C'est pourquoi il convient de ne pas accorder aux systèmes distants le droit de se connecter comme administrateur BNU.

Sécurité et fichiers Systems et remote.unknown

Sur la plupart des systèmes BNU, seuls les systèmes distants répertoriés dans le fichier **/etc/uucp/Systems** ou un de ses substituts (spécifié dans le fichier **Sysfiles**) peuvent se connecter au système local. Le script **/usr/sbin/uucp/remote.unknown** est exécuté chaque fois qu'un système inconnu tente d'appeler le système local. Ce script refuse la connexion du système inconnu et consigne l'heure de la tentative dans le fichier **/var/spool/uucp/.Admin/Foreign**.

Avec un privilège racine, ou en tant qu'administrateur BNU, vous pouvez modifier la procédure shell **remote.unknown** de façon à enregistrer plus d'informations sur le système distant ou à stocker les informations dans un fichier différent. Vous pouvez ainsi décider d'envoyer un courrier à l'administrateur BNU dès qu'un système inconnu essaie de se connecter.

En éliminant les droits d'exécution de la procédure shell **remote.unknown**, vous autorisez les machines inconnues à se connecter. Dans ce cas, ajoutez une entrée MACHINE=OTHER dans le fichier **/etc/uucp/Permissions** pour établir les droits des machines inconnues.

Votre système ne peut contacter que les systèmes distants figurant dans le fichier **Systems**. Ceci interdit aux utilisateurs de votre système de contacter des systèmes inconnus.

Sécurité et fichier Permissions

Le fichier **/etc/uucp/Permissions** détermine :

- les noms de connexion des utilisateurs distants pour la connexion au système local ;
- les commandes et les privilèges accordés aux systèmes distants se connectant au système local.

Le fichier **/etc/uucp/Permissions** contient deux types d'entrée :

LOGNAME Définit les noms de connexion et les privilèges associés. Les entrées LOGNAME prennent effet lorsqu'un système distant appelle le système local et essaie de se connecter.

MACHINE Définit les noms des machines et les privilèges associés. Les entrées MACHINE prennent effet lorsque le système distant essaie d'exécuter des commandes sur le système local.

Les options du fichier **Permissions** permettent d'établir différents niveaux de sécurité pour chaque système distant. Par exemple, si plusieurs systèmes distants partagent un même ID de connexion sur le système local, utilisez l'option VALIDATE pour obliger chaque système distant à utiliser un ID de connexion unique. Les options SENDFILES, REQUEST et CALLBACK spécifient le système qui détient le contrôle, en conservant au besoin le contrôle des transactions au système local.

Les options READ, WRITE, NOREAD et NOWRITE définissent l'accès à des répertoires spécifiques du système local. Ce sont elles également qui contrôlent l'endroit de votre système où les utilisateurs distants peuvent stocker les données. L'option COMMANDS limite le nombre de commandes exécutables sur le système local par les utilisateurs distants. L'option COMMANDS=ALL accorde tous les privilèges aux systèmes étroitement associés au vôtre.

Attention : L'option COMMANDS=ALL peut sérieusement menacer la sécurité de votre système.

Démons BNU

Le logiciel BNU comprend quatre démons, stockés dans le répertoire **/usr/sbin/uucp** :

uucico	Facilite les transferts de fichiers (voir Utilisation du démon uucico)
uusched	Facilite la planification des demandes de travail des fichiers mis en file d'attente dans le répertoire de spoulage local (voir Utilisation du démon uusched).
uuxqt	Facilite les exécutions de commande à distance (voir Utilisation du démon uuxqt)
uucpd	Facilite les communications via TCP/IP (voir Utilisation du démon uucpd)

Les démons **uucico**, **uusched** et **uuxqt** sont lancés par le démon cron, suivant la planification décidée par l'administrateur BNU. Avec un privilège d'utilisateur racine, vous pouvez également lancer ces démons manuellement. Dans ce cas, le démon **uucpd** doit être lancé par le démon **inetd** de TCP/IP.

Démon uucico

Le démon **uucico** transporte les fichiers nécessaires au transfert des données entre deux systèmes. Les commandes **uucp** et **uux** lancent le démon **uucico**, pour qu'il transfère les fichiers de commandes et de données, et qu'il exécute les fichiers sur le système désigné. Le démon **uucico** est également régulièrement lancé par le programmateur BNU, le démon **uusched**. Lorsqu'il est lancé par le démon **uusched**, **uucico** essaie de contacter d'autres systèmes et d'exécuter les instructions des fichiers de commandes.

Lancement du processus du démon

Pour exécuter les instructions des fichiers de commandes, le démon uucico recherche d'abord dans le fichier **/etc/uucp/Systems** (ou dans le(s) fichier(s) spécifié(s) dans **/etc/uucp/Sysfiles**) le système à appeler. Il consulte ensuite le fichier **Systems** pour connaître le créneau horaire défini. S'il se trouve dans un créneau horaire admis, le démon **uucico** vérifie les champs *Type* et *Class*, puis recherche dans le fichier **/etc/uucp/Devices** (ou le(s) fichier(s) spécifié(s) dans **/etc/uucp/Sysfiles**) une unité correspondante.

Une fois l'unité trouvée, le démon **uucico** explore le répertoire **/var/locks** à la recherche d'un fichier de verrouillage pour l'unité. S'il en trouve un, le démon recherche une autre unité du type et du débit requis.

Si aucune unité n'est disponible, le démon revient au fichier **Systems** pour trouver une autre entrée relative au système distant. S'il en existe une, le démon réitère le processus de recherche. Dans le cas contraire, il crée une entrée pour ce système dans le fichier **/var/spool/uucp/.Status/***SystemName*, puis passe à la requête suivante. Le fichier de commande reste dans la file d'attente. Le démon **uucico** tente à nouveau le transfert ultérieurement. Cette nouvelle tentative est appelée réessai.

Contact avec le système distant

Lorsque le démon **uucico** accède au système distant, il se sert des instructions des fichiers Systems pour établir la connexion. Ceci entraîne l'appel d'une instance du démon **uucico** sur le système distant également.

Les deux démons **uucico**, chacun sur un système, coopèrent pour effectuer le transfert. Le démon **uucico** du système appelant contrôle la liaison, en spécifiant les requêtes à effectuer. Le démon **uucico** du système distant vérifie si les autorisations en local permettent l'exécution de la requête. Si tel est le cas, le transfert de fichiers démarre.

Lorsque le premier démon **uucico** a fini de transférer toutes les requêtes pour le système distant, il envoie une demande de raccrochage. Si le deuxième démon **uucico** a des transactions à lui envoyer, il ignore cette demande, et inverse les rôles.

Remarque : Le fichier **/etc/uucp/Permissions** du système local ou le fichier **/etc/uucp/Permissions** du système distant peut interdire aux démons d'inverser leurs rôles. Dans ce cas, le système distant doit attendre, pour transférer ses fichiers, d'appeler le système local.

Lorsqu'il ne reste plus rien à transférer d'un côté ou de l'autre, les deux démons **uucico** raccrochent. A ce stade, le démon **uuxqt** daemon (Utilisation du démon uuxqt page 9-10) est appelé pour exécuter les demandes de commande à distance.

Pendant toute la durée du transfert, les démons **uucico** des deux systèmes enregistrent des messages dans les fichiers d'erreur et le fichier journal de BNU.

Démon uusched

Le démon **uusched** programme le transfert des fichiers en file d'attente dans le répertoire de spoulage du système local. Le répertoire de spoulage est **/var/spool/uucppublic**. Lorsque le démon **uusched** est appelé, il recherche dans ce répertoire les fichiers de commandes, puis les "randomise" et lance le démon **uucico**. Le démon **uucico** transfère les fichiers.

Démon uuxqt

Lorsqu'un utilisateur émet la commande **uux** pour exécuter une commande sur un système désigné, c'est le démon **uuxqt** qui exécute la commande. Après création des fichiers nécessaires, la commande **uux** lance le démon **uucico**, qui transfère ces fichiers dans le répertoire de spoulage public sur le système spécifié.

Le démon **uuxqt** explore régulièrement le répertoire de spoulage de chaque système connecté. Lorsqu'il trouve une requête, le démon **uuxqt** vérifie l'existence des droits et des fichiers requis. Puis, si l'action est autorisée, le démon exécute la commande spécifiée.

Démon uucpd

Le démon **uucpd** doit être actif sur le système distant pour que BNU puisse établir des communications avec l'ordinateur distant via TCP/IP (Transmission Control Protocol/Internet Protocol). Le démon **uucpd**, sous serveur du démon TCP/IP **inetd**, est lancé par le démon **inetd**.

Par défaut, le démon **uucpd** est commenté dans le fichier **inetd.conf**. Pour l'exécuter, supprimez les marques de commentaire et redémarrez **inetd**. Si toutefois, cette configuration a été modifiée sur votre système, il vous faudra reconfigurer le démon **inetd** en conséquence pour lancer le démon **uucpd**.

Configuration de BNU

Les procédures suivantes traitent de la configuration de BNU (Basic Network Utilities) pour les différents types de connexion : connexions par câble, par modem, via TCP/IP (Transmission Control Protocol/Internet Protocol), etc.

Prérequis

- BNU doit être installé sur votre système.
- Vous devez être utilisateur racine pour éditer les fichiers de configuration BNU.
- Si vous utilisez des connexions directes, les câbles appropriés entre votre système et les systèmes distants doivent être installés.
- Si vous utilisez des modems pour vos communications BNU, vous devez avoir installé et configuré chaque modem.
- Si une ou plusieurs de vos connexions utilisent le protocole TCP/IP, ce dernier doit être opérationnel entre votre système et les systèmes distants appropriés.
- Collectez les informations dont vous avez besoin pour configurer BNU (voir Informations à collecter avant la configuration de BNU). Doivent y figurer la liste des systèmes distants ainsi que celle des unités et des modems à utiliser pour la connexion aux systèmes.

Collecte des informations

Avant de configurer BNU, rassemblez les informations ci-dessous :

- Pour chaque système distant que votre système appellera :
  - le nom du système,
  - le nom de connexion que votre système doit utiliser sur le système distant,
  - le mot de passe pour le nom de connexion,
  - les invites de connexion et de mot de passe sur le système distant,
  - le type de connexion que vous utiliserez pour atteindre le système distant (TCP/IP, direct ou par téléphone).
- Si la connexion est directe :
  - le débit (en bits) de la connexion,
  - le port du système local auquel est rattachée la connexion.
- Si la connexion est téléphonique :
  - le numéro de téléphone du système distant,
  - la vitesse de votre modem, compatible avec celle du système distant.

    Remarque : Si l'un des systèmes distants appelle votre système, assurez-vous que l'administrateur BNU sur chacun des systèmes distants possède, à propos de votre système, toutes les informations citées ci-dessus.
- Pour chaque modem local utilisé :
  - le script chat pour le modem (consultez la documentation du modem),

    Remarque : Pour certains modems, le script chat se trouve déjà dans le fichier **/etc/uucp/Dialers**.
  - le port local pour le modem.

A l'aide de ces informations, dressez une liste pour chaque unité à connecter au système distant. Voici un exemple de liste pour le système local `morgan` :

```
direct:
hera 9600 tty5
zeus& 2400 tty2
ariadne 2400 tty1
hayes modem (tty3):  apollo, athena
TCP/IP:  merlin, arthur, percy
```

La connexion au système `hera` est de type `direct`, effectuée à la vitesse de `9600` à partir du port `tty5`. La connexion au système `apollo` passe par le modem `hayes` connecté au port `tty3`. La connexion aux systèmes `merlin`, `arthur` et `percy` fait appel à TCP/IP.

Procédure

Pour que BNU fonctionne correctement sur votre site, vous devez configurer les fonctions de communication à distance pour qu'elles :

- dressent la liste des unités utilisées pour établir une liaison par câble, téléphone ou modem ;
- dressent la liste des modems utilisés pour contacter les systèmes distants via le réseau téléphonique ;
- dressent la liste des systèmes distants accessibles ;
- dressent la liste des abréviations alphabétiques représentant les préfixes des numéros de téléphone utilisés pour contacter les systèmes distants spécifiés (en option) ;
- définissent les autorisations d'accès, spécifiant les modes de communication possibles entre les systèmes local et distants ;
- programment la surveillance des systèmes distants en réseau (en option).

Pour créer ces listes, autorisations, programmations et procédures :

- modifiez les fichiers de configuration BNU ;
- éditez le fichier **/var/spool/cron/crontabs/uucp** pour annuler la mise en commentaire (par #) des lignes planifiant les routines de maintenance automatique.

Vous devez également configurer les fichiers **Systems**, **Devices** et **Permissions**. Il n'est pas nécessaire de modifier les fichiers de configuration BNU dans un ordre particulier.

Pour configurer BNU sur votre système :

1. Assurez-vous que BNU est installé sur votre système :

   ```
   lslpp -h bos.net.uucp
   ```

   Si BNU est installé, `bos.net.uucp` apparaît sur la sortie. Sinon, installez–le à partir de la bande d'installation.

2. Définissez les ID de connexion et mots de passe pour les systèmes distants qui doivent communiquer avec votre système et indiquez–les à l'administrateur BNU ou UUCP (UNIX–to–UNIX Copy Program) de chaque système distant. Pour ce faire, éditez les fichiers **/etc/passwd**, **/etc/group**, **/etc/security/login.cfg** et **/etc/security/passwd**.

   Attention : Autoriser la connexion de systèmes distants au local system avec l'ID de connexion UUCP nuit gravement à la sécurité de votre système. En effet, les systèmes distants ainsi connectés peuvent afficher voire modifier (selon les droits définis dans l'entrée LOGNAME du fichier **Permissions**) les fichiers locaux **Systems** et **Permissions**. Il est donc vivement recommandé d'attribuer d'autres ID de connexion BNU aux systèmes distants et de réserver les ID uucp aux administrateurs BNU du système local. Pour une sécurité optimale, chaque système distant appelé à communiquer avec le local system doit posséder un ID de connexion unique avec un

UID unique. Ces ID de connexion doivent avoir un ID groupe (GID) de 5. Par défaut, le système d'exploitation comprend l'ID de connexion nuucp pour le transfert de fichiers.

a. Vous avez la possibilité d'utiliser, pour l'ensemble des connexions, un ou plusieurs ID de connexion. La première option est conseillée si vous souhaitez contrôler parfaitement l'accès à chaque machine. Dans ce cas, créez les différents ID et associez les entrées MACHINE et LOGNAME dans le fichier **Permissions**. Voici quelques exemples d'entrées **/etc/passwd** :

```
Umicrtk:!:105:5:micrtk
uucp:/usr/spool/uucppublic:/usr/sbin/uucp/uucico
 Ufloydl:!:106:5:floydl
uucp:/usr/spool/uucppublic:/usr/sbin/uucp/uucico
 Uicus:!:107:5:icus uucp:/usr/spool/uucppublic:/usr/sbin/uucp/uucico
 Urisctkr:!:108:5::/usr/spool/uucppublic:/usr/sbin/uucp/uucico
```

b. Si vous souhaitez disposer d'un seul jeu de droits d'accès au lieu de contrôler séparément chacune de vos connexions UUCP, définissez un seul ID de connexion pour l'ensemble des machines :

```
nuucp:!:6:5::/usr/spool/uucppublic:/usr/sbin/uucp/uucico
```

c. L'ID utilisateur (champ de la troisième colonne) doit être unique pour garantir la sécurité de système. L'ID de groupe (quatrième champ) doit être égal à 5, le même groupe qu'uucp. Vous pouvez définir comme répertoire personnel (sixième champ) n'importe quel répertoire valide, mais le shell de connexion (septième champ) doit être `/usr/sbin/uucp/uucico`.

d. Vérifiez que le fichier **/etc/group** contient les nouveaux utilisateurs. Voici un exemple d'entrée de ce type :

```
uucp:!:5:uucp,uucpadm,nuucp,Umicrtk,Uicus,Urisctakr
```

e. Vous pouvez ajouter des utilisateurs à un groupe uucp appelé à utiliser des modems avec d'autres programmes que **cu.**

f. Editez ces fichiers en tant qu'utilisateur racine et attribuez un mot de passe aux nouveaux utilisateurs à l'aide de la commande `passwd nom_utilisateur`.

g. Il peut arriver que la strophe par défaut herald avec tous ses Ctrl-J interrompe le processus de connexion uucico. (Il se peut que le message `Enough already s'affiche`.) Pour l'éviter, placez la strophe par défaut en commentaire (avec des astérisques) et définissez une strophe pour votre tty, de la forme :

```
/dev/tty0:
      herald = "\nrisc001 login:"
```

h. Si vous modifiez un mot de passe à partir d'une connexion racine, les entrées d'indicateurs de la strophe utilisateur dans `/etc/security/passwd` doivent se présenter comme suit :

```
flags = ADMCHG
```

Indiquez à la place :

```
flags =
```

Faute de quoi, à la connexion du uucico distant, un nouveau mot de passe est demandé, qu'il ne peut fournir. Et, la connexion échoue.

i. Modifiez le fichier **Poll** à l'aide d'un éditeur de texte ASCII ou de la commande **uucpadm**. Ajoutez une entrée pour chaque système à solliciter.

Remarque : Les systèmes répertoriés dans le fichier **Poll** doivent également figurer dans le fichier **/etc/uucp/Systems**.

j. Modifiez le fichier **/var/spool/cron/crontabs/uucp** à l'aide d'un éditeur de texte ASCII. Enlevez les marques de commentaire (#) des lignes qui exécutent les commandes **uudemon.hour** et **uudemon.poll**. Vous pouvez modifier le moment

d'exécution de ces commandes. Veillez toutefois à programmer la commande **uudemon.poll** environ cinq minutes *avant* la commande **uudemon.hour**.

k. Vérifiez que les modifications sont prises en compte, via la commande :

```
crontab -l
```

l. Configurez les fichiers de données de BNU : Systems, Permissions, Devices, Dialers et Sysfiles. Pour une définition initiale, utilisez la commande **/usr/sbin/uucp/uucpadm** et modifiez-les ensuite selon vos besoins. Notez que Sysfiles permet de spécifier des fichiers de configuration BNU autres que **/etc/uucp/Systems**, **/etc/uucp/Devices** et **/etc/uucp/Dialers**. Pour en savoir plus, reportez-vous à **Sysfiles**.

3. Décidez si vous souhaitez des abréviations pour les numéros de téléphone (voir le format du fichier **Dialcodes**). Si vous décidez d'utiliser des abréviations dans les fichiers **Systems**, définissez l'entrée **Dialcodes** pour chaque abréviation. Pour en savoir plus, reportez–vous à Dialcodes File Format for BNU dans le manuel *AIX Files Reference*.

   Si vous utilisez TCP/IP pour vos connexions BNU, vérifiez à l'aide de la commande **netstat** que le démon **uucpd** est exécutable :

   ```
   netstat -a
   ```

   Le démon **uucpd** est lancé par **inetd**. Si le démon **uucpd** ne peut être exécuté, reconfigurez le démon **inetd** pour lancer le démon **uucpd** (voir Configuration du démon inetd).

4. A l'aide de votre liste d'unités récupérée avant de commencer cette procédure, modifiez le fichier **Devices** sur votre système. Créez une entrée pour chaque modem et chaque connexion directe (voir Informations à collecter avant de configurer BNU). Si vous utilisez TCP/IP, vérifiez que l'entrée TCP/IP n'est plus en commentaire dans le fichier **Devices**. Vous pouvez configurer le fichier **/etc/uucp/Sysfiles** pour spécifier d'autres fichiers de configuration de Devices. Pour en savoir plus sur le fichier Devices, reportez–vous à Devices File Format for BNU dans le manuel *AIX 5L Version 5.2 Files Reference*. Reportez–vous à Sysfiles File Format for BNU pour plus d'informations sur le fichier Sysfiles dans le manuel *AIX 5L Version 5.2 Files Reference*.

   De plus, si vous utilisez TCP/IP, vérifiez que le fichier **/etc/services** contient la ligne :

   ```
   uucp        540/tcp          uucpd
   ```

   Sinon, ajoutez–la.

5. A partir des informations relatives à chacun des systèmes distants récupérées avant de commencer cette procédure, modifiez le fichier **Systems** sur votre système (voir Informations à collecter avant de configurer BNU). Aidez–vous des exemples commentés dans le fichier **Systems** pour effectuer votre configuration. Pour plus d'informations, reportez–vous à "BNU Systems File Format" dans le manuel *AIX 5L Version 5.2 Files Reference*. Si vous utilisez TCP/IP, assurez–vous que la table des noms d'hôte dans le fichier **/etc/hosts** comprend le nom de la machine distante à laquelle vous souhaitez vous connecter. Vous pouvez configurer le fichier **/etc/uucp/Sysfiles** pour spécifier d'autres fichiers de configuration de Systems. Pour plus d'informations, reportez–vous à Sysfiles File Format for BNU dans le manuel *AIX 5L Version 5.2 Files Reference*.

6. A partir des informations relatives aux unités et modems récupérées avant de commencer cette procédure, vérifiez que le fichier **Dialers** de votre système contient une entrée pour chaque modem (voir Informations à collecter avant de configurer BNU). Si vous utilisez des connexions TCP/IP et directes, vérifiez que les entrées correspondantes figurent dans le fichier. Pour plus d'informations, reportez–vous à Dialers File Format for BNU dans le manuel *AIX 5L Version 5.2 Files Reference*. Vous pouvez configurer le fichier **/etc/uucp/Sysfiles** pour spécifier d'autres fichiers de configuration de Dialers. Pour plus d'informations, reportez–vous à Sysfiles File Format for BNU dans le manuel *AIX 5L Version 5.2 Files Reference*.

7. Déterminez l'accessibilité de votre système pour chaque système distant appelé à communiquer avec vous. Définissez les entrées correspondantes pour chaque système et nom de connexion dans le fichier **Permissions**. Pour plus d'informations, reportez–vous à Permissions File Format for BNU dans le manuel *AIX 5L Version 5.2 Files Reference*.
8. Exécutez la commande **uucheck** pour vérifier que tout est en place :

   ```
   /usr/sbin/uucp/uucheck -v
   ```

   La commande **uucheck** vérifie que les répertoires, les programmes et les fichiers de support sont correctement configurés et que les entrées du fichier **Permissions** sont cohérentes. Si la commande **uucheck** signale une erreur, corrigez–la.
9. Si vous le souhaitez, définissez un contrôle automatique des opérations BNU et un sondage automatique des systèmes distants (voir Configuration du contrôle automatique de BNU et Configuration du sondage BNU des systèmes distants).

Configuration du contrôle automatique de BNU

Préalables

- Effectuez les étapes indiquées dans Configuration de BNU.
- Vous devez être utilisateur racine pour éditer le fichier **/var/spool/cron/crontabs/uucp**.

Procédure

BNU fait appel au démon **cron** pour lancer les démons BNU et contrôler l'activité BNU. Le démon **cron** lit les instructions du fichier **/var/spool/cron/crontabs/uucp** sur le lancement des procédures BNU.

1. Connectez-vous en tant qu'utilisateur racine.
2. Modifiez le fichier **/var/spool/cron/crontabs/uucp** à l'aide d'un éditeur de texte ASCII.
3. Enlevez les marques de commentaire des lignes sur les procédures de maintenance BNU, **uudemon.admin** et **uudemon.cleanup**. Vous pouvez modifier à votre guise la fréquence d'exécution de ces procédures sur votre système. Toutefois, il est conseillé d'exécuter **uudemon.admin** au moins une fois par jour et **uudemon.cleanup** au moins une fois par semaine.
4. Vous disposez du fichier **crontabs/uucp** pour programmer d'autres commandes de maintenance BNU, telles que **uulog**, `uuclean` ou **uucleanup**. Le fichier **crontabs/uucp** vous permet également d'indiquer au démon **cron** de planifier le lancement des démons **uucico**, **uuxqt** ou **uusched** par `cron`.

Appel automatique BNU des systèmes distants

Prérequis

1. Exécutez les étapes décrites à la section "Configuration de BNU", page9-11.
2. Vous devez être utilisateur racine pour éditer les fichiers **/var/spool/cron/crontabs/uucp** et **/etc/uucp/Poll**.

Procédure

Pour permettre à BNU de solliciter des systèmes distants pour des travaux, dressez la liste de ces systèmes dans le fichier **/etc/uucp/Poll**. Exécutez en outre périodiquement les commandes **uudemon.hour** et **uudemon.poll**.

1. Décidez des systèmes distants à solliciter automatiquement. Décidez de la fréquence de leur sollicitation. Indiquez chaque fréquence (au moins une fois par jour) dans le fichier **Poll**.
2. Connectez-vous en tant qu'utilisateur racine.

3. Modifiez le fichier **Poll** à l'aide d'un éditeur de texte ASCII ou de la commande **uucpadm**. Ajoutez une entrée pour chaque système à solliciter.

   Remarque : Les systèmes répertoriés dans le fichier **Poll** doivent également figurer dans le fichier **/etc/uucp/Systems**.

4. Modifiez le fichier **/var/spool/cron/crontabs/uucp** à l'aide d'un éditeur de texte ASCII. Enlevez les marques de commentaire (#) des lignes qui exécutent les commandes **uudemon.hour** et **uudemon.poll**. Vous pouvez modifier le moment d'exécution de ces commandes. Veillez toutefois à programmer la commande **uudemon.poll** environ cinq minutes *avant* la commande **uudemon.hour**.

BNU sollicitera automatiquement les systèmes répertoriés dans le fichier **Poll** aux moments spécifiés.

Fichier /etc/uucp/Systems

Les systèmes distants figurent dans les fichiers **/etc/uucp/Systems**. Le fichier **/etc/uucp/Systems** est le fichier **Systems** par défaut. L'administrateur système peut en définir d'autres dans le fichier **/etc/uucp/Sysfiles**.

Chaque entrée de fichier **Systems** comprend :

- le nom du système distant,
- les créneaux horaires pendant lesquels le système distant est accessible,
- le type de liaison (directe ou par modem),
- la vitesse de transmission par liaison,
- les informations requises pour la connexion au système distant.

Chaque entrée d'un fichier **Systems** représente un système distant. Pour établir la communication, le système distant doit être répertorié dans le fichier **Systems** local. Un fichier **Systems** doit être installé sur chaque système exploitant BNU. Normalement, seul l'utilisateur racine est habilité à lire les fichiers **Systems**. Tout utilisateur peut toutefois afficher la liste des systèmes BNU distants, via la commande **uuname**.

Édition des fichiers Devices pour connexion câblée

Prérequis

Vous devez être utilisateur racine pour éditer le fichier **/etc/uucp/Devices** ou tout autre fichier déclaré comme fichier **Devices** dans **/etc/uucp/Sysfiles**.

Création d'une entrée de nom de système

Pour définir une connexion câblée, avec un port et un système distant, créez une entrée comme suit :

1. Spécifiez, dans le champ *Type* (seconde ligne), le nom du système distant auquel connecter la machine locale via la ligne câblée.
2. Spécifiez, dans le champ *Line* des deux lignes de l'entrée, le nom de l'unité pour la connexion câblée utilisée sur votre site.
3. Insérez un tiret (`-`) de réserve dans le champ *Line2* sur les deux lignes de l'entrée.
4. Indiquez dans le champ *Speed,* sur les deux lignes de l'entrée, le débit de transmission pour la connexion câblée utilisée sur votre site.
5. Entrez `direct` (en minuscules) dans le champ *Dialer-Token Pairs* sur les deux lignes de saisie.

Par exemple :

```
type device - speed direct
```

Complétez ainsi le fichier **Devices** jusqu'à ce que toutes les unités câblées reliant le système local à un système distant soient répertoriées.

Création d'une entrée directe

Pour définir une connexion câblée entre deux systèmes utilisant une connexion série asynchrone permanente, créez une entrée d'une ligne comme suit :

1. Entrez le nom du système distant dans le premier champ (*Type*).
2. Entrez le nom de l'unité tty dans le second champ (*Line*).
3. Insérez un tiret (`-`) de réserve dans le troisième champ (*Line2*).
4. Indiquez dans le quatrième champ (*Class*), le débit de transmission pour la connexion câblée utilisée sur votre site.
5. Entrez `direct` (en minuscules) dans le cinquième champ (*Dialer–Token Pairs*).

Par exemple :

```
type device - speed direct
```

Complétez ainsi le fichier **Devices** jusqu'à ce que toutes les unités câblées reliant le système local à un système distant soient répertoriées.

Édition du fichier Devices pour connexion automatique

Prérequis

Vous devez être utilisateur racine pour éditer le fichier **/etc/uucp/Devices** ou tout autre fichier déclaré comme fichier **Devices** dans **/etc/uucp/Sysfiles**.

Procédure

Dans les entrées de connexion téléphonique, le champ *Type* est spécifié comme une unité ACU (automatic calling unit). Indiquez ACU dans le champ *Type* dans toutes les connexions à distance établies via une ligne téléphonique. Pour définir les entrées du fichier **Device** pour connexions automatiques, créez une entrée d'une ligne pour chaque modem, comme suit :

1. Entrez `ACU` dans le premier champ (*Type*).
2. Le deuxième champ (*Line*) contient le nom de l'unité raccordée au modem. Entrez le nom qui convient pour votre site.
3. Entrez un `-` (tiret) de réserve dans le troisième champ (*Line2*), sauf si le numéroteur automatique est un numéroteur 801 standard. Si le numéroteur est un 801 standard, entrez `801`.
4. Indiquez dans le quatrième champ (*Speed*), le débit en bauds correspondant à votre modem et votre ligne (300, 1200, 2400 ou plus, selon le modem) ou la classe du modem utilisé (par exemple, D2400).

   Remarque : Si le modem est exploitable à plusieurs débits, créez une entrée distincte pour chaque débit dans le fichier **Devices**. Si le modem peut être utilisé à n'importe quel débit, indiquez `Any` dans le champ *Speed.*
5. Entrez le nom du modem dans la partie *Dialer* du cinquième champ (*Dialer Token Pair*). Si vous envisagez d'inclure des numéros d'appel complets dans le fichier **/etc/uucp/Systems** ou un fichier Systems spécifié dans **/etc/uucpSysfiles**, laissez la partie *Token* en blanc. (Un blanc indique au programme BNU d'utiliser l'option par défaut `\D`.) Si vous souhaitez utiliser les codes d'accès directs spécifiés dans le fichier **/etc/uucp/Dialcodes**, entrez l'option `\T`.

Par exemple :

```
type line - speed dialer - token pair
```

Complétez ainsi le fichier **Devices** jusqu'à répertorier toutes les unités câblées reliant le système local à un système distant via une ligne téléphonique ou un modem.

Édition du fichier Devices pour TCP/IP

Prérequis

Vous devez être utilisateur racine pour éditer le fichier **/etc/uucp/Devices** ou tout autre fichier déclaré comme fichier **Devices** dans **/etc/uucp/Sysfiles**.

Procédure

Si vous utilisez TCP/IP, insérez l'entrée TCP/IP correspondante dans le fichier **Devices**. Pour configurer le fichier pour l'utiliser avec le système TCP/IP, insérez la ligne suivante dans le fichier **Devices** :

```
TCP - - - TCP
```

Maintenance de BNU

Pour que BNU fonctionne correctement, vous devez effectuer un certain nombre de tâches de maintenance. Pour maintenir BNU :

- Consultez et supprimez régulièrement les fichiers journaux.
- Utilisez les commandes **uuq** et **uustat** pour contrôler les files d'attente BNU et vous assurer que les travaux sont correctement transférés aux systèmes distants.
- Planifiez les commandes automatiques qui sollicitent les systèmes distants pour les travaux, qui réexpédient les fichiers non envoyés aux utilisateurs et qui vous informent périodiquement de l'état de BNU.
- Effectuez régulièrement une mise à jour des fichiers de configuration pour intégrer les modifications apportées à votre système.

Par ailleurs, tenez-vous informé, auprès des administrateurs de systèmes distants, des modifications apportées à leurs systèmes, susceptibles d'influer sur votre configuration. Par exemple, si le superviseur de `venus` modifie votre mot de passe système, vous devez le déclarer dans le fichier **/etc/uucp/Systems** (ou dans le fichier Systems correspondant spécifié par **/etc/uucp/Sysfiles**) avant de vous connecter à `venus`.

Reportez–vous à Fichiers, commandes et répertoires BNUpage 9-40, pour connaître les commandes de maintenance de BNU.

Fichiers journaux BNU

BNU crée des fichiers journaux et des fichiers d'erreurs pour le suivi de vos activités. Veillez à consulter et supprimer ces fichiers régulièrement afin d'éviter tout encombrement de l'espace système. BNU offre plusieurs commandes pour le vidage des journaux :

- **uulog**
- **uuclean**
- **uucleanup**
- **uudemon.cleanu**

Exécutez ces commandes manuellement ou déclarez-les dans le fichier **/var/spool/cron/crontabs/uucp** pour que le démon **cron** les exécute automatiquement.

Fichiers journaux des répertoires .Log et .Old

BNU crée des fichiers journaux individuels dans le répertoire **/var/spool/uucp/.Log**. BNU crée ces journaux pour chaque système distant accessible, via la commande **uucp**, **uux** ou **uuxqt**. BNU consigne dans le fichier journal correspondant l'état de chaque transaction, chaque fois qu'un utilisateur du système fait appel à BNU. Lorsque plusieurs processus BNU sont exécutés, le système ne peut accéder au fichier journal. L'état est alors consigné dans un fichier séparé et suffixé par **.LOG**.

La commande **uulog** fournit un récapitulatif, par utilisateur ou par système, des requêtes **uucp** ou **uux**. La commande **uulog** affiche les fichiers. Mais vous pouvez également demander à BNU de fusionner automatiquement les fichiers journaux dans un fichier principal. L'opération, appelée *compactage*, exécutée par la commande **uudemon.cleanu**, est généralement lancée par le démon **cron**.

Le démon **cron** exécute **uudemon.cleanu**. La commande **uudemon.cleanu** regroupe les fichiers journaux **uucico** et **uuxqt** sur le système local et les stocke dans le répertoire **/var/spool/uucp/.Old**. Simultanément, la commande supprime les anciens fichiers journaux précédemment stockés dans le répertoire **.Old**. Par défaut, la commande **uudemon.cleanu** sauvegarde les fichiers journaux datant de deux jours.

En cas d'encombrement de l'espace de stockage, réduisez la durée de conservation (nombre de jours) des fichiers journaux. Pour pister les transactions BNU sur une plus longue période, augmentez cette durée. Il est possible de changer le délai de sauvegarde par défaut des fichiers journaux : pour ce faire, modifiez la procédure shell pour la commande **uudemon.cleanu**. Ce script, stocké dans le répertoire **/usr/sbin/uucp,** peut être modifié avec des droits d'utilisateur racine.

Autres fichiers journaux BNU

BNU recueille également des informations qu'il stocke dans le répertoire **/var/spool/uucp/.Admin**. Ce répertoire contient les fichiers **errors**, **xferstats**, **Foreign** et **audit**. Consultez et supprimez régulièrement ces fichiers pour ne pas occuper inutilement de l'espace de stockage. BNU crée chaque fichier quand nécessaire.

Lorsqu'un système contacte le vôtre avec la mise au point **uucico** activée, il fait appel au démon **uucico** sur votre système avec mise au point activée. Les messages de mise au point générés par le démon sur le système local sont stockés dans le fichier **audit**. Ce fichier peut être relativement volumineux. Consultez souvent le fichier **audit** et supprimez–le.

Le fichier **errors** enregistre les erreurs détectées par le démon **uucico**. Consultez-le pour corriger les incidents relevés (droits d'accès erronés sur les fichiers de travail BNU par exemple).

Le fichier **xferstats** contient les informations d'état sur tout transfert de fichiers. Consultez-le régulièrement et supprimez-le.

Le fichier **Foreign** joue un rôle déterminant dans la sécurité du système. Chaque fois qu'un système inconnu tente de se connecter au système local, BNU appelle la procédure shell **remote.unknown**. Cette procédure shell consigne la tentative dans le fichier **Foreign**. Le fichier **Foreign** contient le nom de tous les systèmes qui ont tenté en vain de se connecter au système local. Si un système a effectué plusieurs tentatives d'accès, vous pouvez envisager d'autoriser son accès.

Fichiers journaux au niveau système utilisés par BNU

Nombre de processus BNU requièrent des droits racine et génèrent de multiples entrées dans le fichier journal **/var/spool/sulog**. De même, la planification des tâches BNU via le démon **cron** génère diverses entrées dans le fichier **/var/spool/cron/log**. Consultez et videz régulièrement ces fichiers.

Commandes de maintenance BNU

BNU regroupe diverses commandes destinées à surveiller les activités BNU et à nettoyer les fichiers et répertoires BNU.

Commandes de nettoyage

BNU propose trois commandes pour nettoyer les répertoires et supprimer les fichiers non envoyés :

uuclean	Supprime, dans les répertoires administratifs BNU, les fichiers datant de plus d'un certain nombre d'heures. Spécifiez, via la commande **uuclean** le répertoire à vider ou le type de fichier à supprimer. Vous pouvez également demander sur la commande prévienne les propriétaires des fichiers supprimés. La commande **uuclean** est l'équivalent Berkeley de la commande **uucleanup**.
uucleanup	Exécute des fonctions semblables à celles de la commande **uuclean.** Une différence : la commande **uucleanup** vérifie l'âge des fichiers en *jours* et non en heures. Utilisez la commande **uucleanup** pour avertir les utilisateurs dont les fichiers n'ont pas été transférés et sont maintenus en file d'attente. La commande **uucleanup** permet également de supprimer des fichiers relatifs à un système distant particulier.
uudemon.cleanu	Procédure shell qui émet les commandes **uulog** et **`uucleanup`** pour compresser les fichiers journaux et fichiers de travail BNU datant de plus de trois jours. La commande **uudemon.cleanu** est exécutée par le démon **cron**.

Commandes de contrôle d'état

BNU propose également des commandes pour contrôler l'état des transferts et des fichiers journaux :

uuq	Affiche les travaux actuellement en file attente BNU. Lancez la commande **uuq** pour afficher l'état d'un travail particulier ou de tous les travaux. En session racine, vous pouvez utiliser la commande **uuq** pour supprimer un travail de la file d'attente.
uustat	Remplit des fonctions similaires à la commande **uuq** mais présente les informations sous un format différent. Utilisez la commande **uustat** pour contrôler l'état des travaux et supprimer vos propres travaux. En session racine, vous pouvez également l'utiliser pour supprimer des travaux appartenant à d'autres utilisateurs.
uulog	Fournit un récapitulatif, par utilisateur ou par système, des requêtes **uucp** et **uux**. La commande **uulog** affiche le nom des fichiers. Reportez-vous à "Fichiers journaux BNU", page 9-19.
uupoll	Force une interrogation d'un système distant. Cette opération est utile lorsqu'un travail destiné à ce système est en attente et doit être transféré, alors que l'appel automatique du système n'a pas encore été programmé.
uusnap	Affiche un récapitulatif succinct de l'état de BNU. Pour chaque système distant, cette commande affiche le nombre de fichiers en attente de transfert. Elle n'indique toutefois pas la durée de l'attente. La commande **uusnap** est l'équivalent Berkeley de la commande **uustat**.

Procédures shell

BNU est livré avec deux procédures shell dédiées à la maintenance :

uudemon.cleanu	Reportez–vous à Commandes de nettoyage, page 9-21.
uudemon.admin	Lance la commande **uustat**. La commande **uustat** fournit l'état des travaux BNU. Elle transmet le résultat sous forme de courrier à l'ID de connexion uucp. Vous pouvez modifier la procédure shell **uudemon.admin** pour réacheminer le courrier, ou utiliser un programme de messagerie pour réacheminer le courrier destiné à un ID de connexion uucp vers l'administrateur BNU.

Ces procédures shell sont stockées dans le répertoire **/usr/sbin/uucp**. Si vous souhaitez modifier les procédures, copiez-les et modifiez la copie. Lancez les procédures à partir de la ligne de commande ou programmez leur exécution par le démon **cron**.

Pour exécuter automatiquement les commandes **uudemon.cleanu** et **uudemon.admin**, supprimez les marques de commentaire (#) au début des lignes correspondantes, dans le fichier **/var/spool/cron/crontabs/uucp**.

Contrôle d'une connexion distante BNU

Prérequis

- Le programme BNU doit être installé sur votre système.
- Une liaison (par câble, modem ou TCP/IP) doit relier votre système au système distant.
- Les fichiers de configuration BNU **Systems**, **Permissions**, **Devices, Dialers** (et **Sysfiles**, le cas échéant) doivent autoriser la communication entre votre système et le système distant.

  Remarque : Vous devez être utilisateur racine pour modifier les fichiers de configuration BNU.

Procédure

La commande **Uutry** aide à contrôler le processus démon **uucico** en cas d'incident de transfert de fichier signalé par les utilisateurs de votre site.

1. Lancez la commande **uustat** pour déterminer l'état de tous les travaux de transfert actuellement en file d'attente :

   ```
   uustat -q
   ```

   Le système affiche un compte rendu d'état comme suit :

   ```
   venus 3C (2) 05/09-11:02 CAN'T ACCESS DEVICE
   hera 1C 05/09-11:12 SUCCESSFUL
   merlin 2C 5/09-10:54 NO DEVICES AVAILABLE
   ```

   Ce compte rendu indique que trois fichiers de commande (**C**.*) destinés au système distant `venus` sont en file d'attente depuis deux jours. Ce délai peut être dû à différentes causes. Par exemple, le système `venus` a été arrêté pour maintenance, le modem est éteint, etc.

2. Avant de poursuivre plus avant les procédures de résolution d'incident, lancez la commande **Uutry** pour vérifier que votre système local est en mesure de contacter le système `venus` :

   ```
   /usr/sbin/uucp/Uutry -r venus
   ```

Si la connexion est établie, l'incident de transfert vers le fichier temporaire est supposé résolu. Relancez la commande **uustat** pour vous assurer que les fichiers stockés dans le répertoire de spoulage sont effectivement transférés vers le système distant. Sinon, reportez vous à Contrôle de transfert de fichiers BNU.

3. Si votre système local parvient à établir la connexion avec le système `venus`, la sortie de mise au point contient de nombreuses informations. C'est la dernière ligne du script qui présente le plus d'intérêt :

   ```
   Conversation Complete: Status SUCCEEDED
   ```

   Si la connexion est établie, l'incident de transfert vers le fichier temporaire est supposé résolu. Relancez la commande **uustat** pour vous assurer que les fichiers stockés dans le répertoire de spoulage sont effectivement transférés vers le système distant. Sinon, exécutez les étapes décrites à la section "Contrôle du transfert de fichier BNU", page 9-23 , pour tester le transfert entre votre système et le système distant.

4. Si le système local ne parvient pas à contacter le système distant, le résultat de mise au point généré par la commande **Uutry** fournit des informations sous la forme ci-dessous (la présentation peut varier) :

   ```
   mchFind called (venus)
   conn (venus)
   getto ret -1
   Call Failed: CAN'T ACCESS DEVICE
   exit code 101
   Conversation Complete: Status FAILED
   ```

   Vérifiez les connexions physiques entre le système local et le système distant. Vérifiez la mise sous tension de la machine distante, le câblage, l'activation adéquate des ports sur les deux systèmes et le fonctionnement du modem.

   Si les connexions physiques sont correctes et fiables, examinez les fichiers de configuration des deux systèmes :

   – Contrôlez les entrées des fichiers **Devices**, **Systems**,**Permissions** (et, le cas échéant, **Sysfiles**) dans le répertoire **/etc/uucp.**

   – Dans le cas d'une liaison par modem, vérifiez que le fichier **/etc/uucp/Dialers** (ou tout autre fichier déclaré dans **/etc/uucp/Sysfiles**) contient l'entrée adéquate. Si vous utilisez des codes d'accès directs, vérifiez-les dans le fichier **/etc/uucp/Dialcodes**.

   – Dans le cas d'une liaison TCP/IP, vérifiez que le démon **uucpd** peut être exécuté sur le système distant et que les fichiers de configuration contiennent les entrées TCP correctes.

5. Une fois les connexions physiques et les fichiers de configuration contrôlés, relancez la commande **Uutry.** Si le résultat de mise au point signale toujours l'échec de la connexion, contactez le support technique. Sauvegardez le résultat de la commande **Uutry**. Il pourra être utile pour diagnostiquer l'incident.

Contrôle du transfert de fichier BNU

Prérequis

1. Le programme BNU doit être installé et configuré sur votre système.
2. Etablissez une connexion au système distant comme indiqué à Contrôle d'une connexion distante BNU.

Contrôle du transfert de fichier

Cette procédure permet de contrôler un transfert de fichier vers un système distant. Ce contrôle est utile lorsque les transferts de fichiers vers un système distant n'aboutissent pas pour une raison inconnue. Les informations de mise au point générées par le démon **uucico** (appelé par la commande **Uutry**) peuvent servir à identifier l'incident.

Pour effectuer le contrôle, utilisez la commande **Uutry** comme suit :

1. Préparez un fichier au transfert en exécutant la commande **uucp** assortie de l'indicateur **–r** :

   ```
   uucp -r test1 venus!~/test2
   ```

   –r demande au programme UUCP de créer et de placer les fichiers de transfert nécessaires dans la file d'attente *sans* lancer le démon **uucico**.

2. Exécutez la commande **Uutry** assortie de l'indicateur **–r** pour lancer le démon **uucico** avec la mise au point activée :

   ```
   /usr/sbin/uucp/Uutry -r venus
   ```

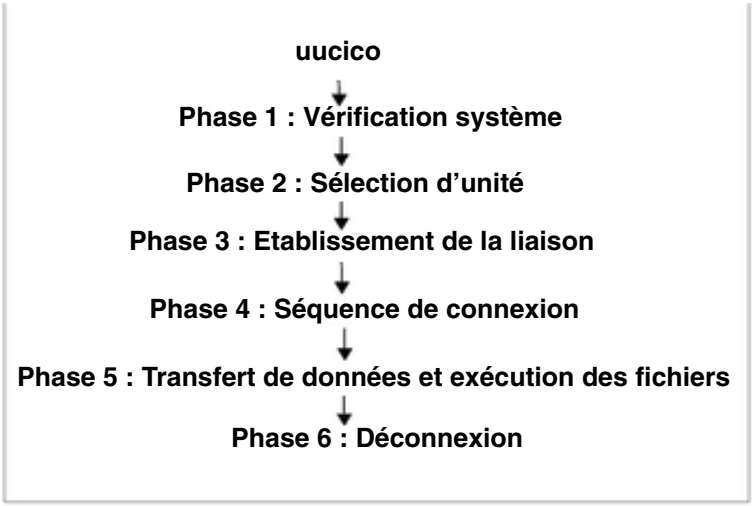
   Cette commande demande au démon **uucico** de contacter le système distant `venus` sans tenir compte du délai imparti par défaut pour les tentatives. Le démon contacte alors le système `venus`, établit la connexion et transfère le fichier. Dans le même temps, la commande **Uutry** génère une sortie de mise au point qui vous permettra de contrôler le processus **uucico**. Appuyez sur la combinaison de touches d'interruption pour arrêter la sortie de mise au point et revenir à l'invite de commande.

   La commande **Uutry** enregistre également la sortie de mise au point dans le fichier **/tmp/** *SystemName*. Si vous interrompez la sortie de mise au point avant l'établissement de la connexion, vous pourrez parcourir les pages du fichier de sortie pour visualiser l'aboutissement de la connexion.

Résolution des incidents BNU

Des messages d'erreur BNU peuvent être liés à une phase particulière des échanges. Utilisez le schéma de flux BNU et le descriptif des erreurs ci-après pour diagnostiquer les incidents. Certains messages ne sont pas émis par BNU mais sont néanmoins cités en cas d'utilisation d'une autre version d'UUCP.

Figure 36. Schéma de flux BNU Ce schéma montre le flux et les différentes phases de la conversion BNU. En partant de uucico au sommet, les données sont transférées à la phase 1 – Vérification système, puis à la phase 2 – Sélection d'unité, et à la phase 3 – Etablissement de la liaison, puis à la phase 4 – Séquence de connexion, puis à la phase 5 – Transfert des données et exécution des fichiers, et enfin à la phase 6 – Déconnexion.



Messages d'état de la phase 1

`Assert Error` (Erreur interne)	Problèmes au niveau de l'unité du système local. Recherchez les causes possibles dans le compte rendu d'erreur, à l'aide de la commande *errpt –a*	*pg*.
`System not in Systems`	Le nom de système distant indiqué ne figure pas dans les fichiers **Systems**. BNU s'arrête. Lancez la commande **uuname** pour vérifier ce nom.
`Wrong time to call`	Des restrictions définies sur le système **Systems** limitent les plages horaires où les appels sortants sont autorisés. BNU renouvelle la tentative jusqu'à ce que l'heure admise soit atteinte. Vérifiez le fichier **Systems**.
`Callback required`	L'utilisation du réseau est limitée pour des questions de sécurité ou d'économie. L'accès est refusé à ce moment précis.
`Cannot call` `No Call`	BNU a récemment tenté sans succès d'appeler le système distant. Il n'effectue pas immédiatement de nouvelle tentative. Ce message peut également être dû à un ancien fichier d'état du système qui empêche **uucico** d'effectuer une nouvelle tentative.

Messages d'état de la phase 2

`Dialer Script Failed`	Votre script du fichier **Dialers** n'a pas abouti.
`No Device Available` `Can't Access Device`	Le modem ou la ligne téléphonique sortante de votre système est occupé. Vérifiez l'entrée relative à l'unité dans le fichier **Systems**. Examinez les fichiers **Devices** et **Dialers** : les unités logiques doivent avoir des unités physiques associées. Le fichier */etc/uucp/Sysfiles* peut spécifier un autre fichier **Systems**, **Devices** ou **Dialers** qui n'a pas été correctement configuré. L'unité est–elle utilisée par d'autres programmes ? Vérifiez le verrouillage des ports dans le répertoire **/var/locks**. Si un fichier de verrouillage existe (par exemple, **LCK..TTY0**), vérifiez que le processus identifié par le numéro dans le fichier de verrouillage est toujours actif. Dans la négative, supprimez-le (par exemple, `rm /var/locks/LCK..TTY0`). Vérifiez également les droits d'accès au port.

`Dial Failed` `Failed (call to system)`	Votre système est parvenu à contacter un autre système mais ce dernier ne répond pas. Il se peut également que les fichiers **Devices** comportent une anomalie. Entrez la commande `uucico -r1 -x6 -s` *`SystemName`*. Il est possible que BNU attende une chaîne qu'il ne reçoit pas. Effectuez la connexion manuellement pour déterminer ce qui doit être ajouté à l'entrée de fichiers **Systems** pour satisfaire la requête. Veillez à respecter les éventuels délais (par exemple, si la séquence d'appel d'un modem est associée à un délai). D'autres causes peuvent être à l'origine du message : port occupé, numéro composé erroné, BNU non propriétaire du port, etc.
`OK` `Auto Dial`	Messages d'information ne signalant aucune erreur.

Messages d'état de la phase 3

`Handshake Failed (LCK)`	L'unité est en cours d'utilisation par un autre utilisateur. Le processus n'a pas pu créer de fichier **LCK**. Parfois, les fichiers **LCK** doivent être supprimés manuellement par l'administrateur. Après un certain nombre de tentatives, faites appel à l'administrateur système. Vérifiez si un autre processus mobilise le port (par exemple, une autre instance **uucico**).
`Login Failed`	Échec d'établissement de liaison du fait d'une connexion défaillante ou d'une machine trop lente.
`Timeout`	Le système distant n'a pas répondu dans les délais impartis. Il se peut également qu'il y ait un problème avec le script chat.
`Succeeded (Call to System)`	L'appel a abouti.
`BNU (continued)`	Messages d'information ne signalant aucune erreur.

Messages d'état de la phase 4

`Startup Failed` `Remote reject after login`	Après connexion, **uucico** est lancé sur le système distant. Ces messages s'affichent lorsqu'un incident se produit en début de conversation entre les deux systèmes. Il se peut également que la connexion n'ait pas été établie sur le compte BNU adéquat ou que la mise en liaison initiale (”handshaking”) ait échoué.
`Wrong machine name`	Vous avez mal appelé une machine ou son nom a changé.

`Bad login/machine combination`	La connexion au système distant a échoué. Plusieurs causes possibles : numéro de téléphone incorrect, nom de connexion ou mot de passe incorrect, erreur dans le script chat, etc.
`Remote has a LCK file for me`	Les deux systèmes ont tenté simultanément de s'appeler. La requête locale échoue temporairement.
`OK` `Talking`	Messages d'information ne signalant aucune erreur.
`LOGIN:` `PASSWORD:`	Si l'invite de connexion ou de mot de passe figure entièrement en lettres majuscules, le modem fonctionne en mode écho (E1 sur compatibles Hayes). Dans ce cas, le modem envoie (ou renvoie) un code RING à votre système, à la réception d'un appel entrant. La commande **getty** reçoit la chaîne et passe l'invite de connexion `login:` ou le mot de passe `password:` en lettres majuscules. **Désactivez (off**) le mode écho sur le modem (via `ATE0` pour les compatibles Hayes). **Remarque :** Après avoir effectué cette modification, vous devez utiliser `ATE1` dans le script chat de vos fichiers **Dialers**, faute de quoi vous n'obtiendrez pas la réponse `OK` de la part de votre modem. Si le port distant est configuré pour `delay` ou `getty -r` et que le script chat attend que vous appuyiez sur une touche, les ports configurés pour `delay` attendent l'entrée d'un ou de plusieurs retours chariot avant de poursuivre la procédure de connexion. Insérez en début du script chat, côté système appelant, la ligne : `"" \r\d\r\d\r\d\r in:--in:  ...` Cette ligne signifie : aucune touche attendue, envoyer les codes retour (return), délai (delay), retour, délai, retour, délai, retour.

Messages d'état de la phase 5

`Alarm`	*uucico* rencontre des difficultés de connexion. La connexion est défaillante ou "xon/xoff" est activé sur le modem.
`Remote access to path/file denied` `copy (failed)`	Messages signalant une anomalie relative aux droits d'accès. Vérifiez les droits associés aux fichiers et aux chemins.

`Bad read`	Espace insuffisant sur le système distant, probablement dans la zone de spoulage, ou lecture ou écriture sur l'unité impossible pour *uucico.*
`Conversation failed`	La détection de porteuse du modem a été perdue. Le modem a été mis hors tension, le câble est desserré ou déconnecté, le système distant est arrêté ou bloqué. Ou alors la liaison téléphonique est interrompue.
`Requested Copy (succeeded)`	Messages d'information ne signalant aucune erreur.

Messages d'état de la phase 6

`OK (Conversation Complete)`	Le système distant refuse la demande de raccrochage et inverse les rôles (il a un travail à soumettre au système local). Dès que les deux **uucico** n'ont plus de travaux à soumettre, ils raccrochent.
`Conversation succeeded`	Message d'information ne signalant aucune erreur.

Résolution des incidents de connexion BNU via le démon uucico

Prérequis

- BNU doit être installé sur votre système.
- Une liaison (par câble, modem ou TCP/IP) doit relier votre système au système distant.
- Les fichiers de configuration BNU **Systems**, **Permissions**, **Devices, Dialers** (et **Sysfiles**, le cas échéant) doivent autoriser la communication entre votre système et le système distant.

  Remarque : Vous devez être utilisateur racine pour modifier les fichiers de configuration BNU.

- Vous devez être utilisateur racine pour appeler le démon **uucico** en mode mise au point.

Procédure

1. Pour obtenir des informations de mise au point sur une connexion défaillante entre systèmes local et distant, lancez le démon **uucico** assorti de l'indicateur **–x** :

   ```
   /usr/sbin/uucp/uucico –r 1 –s venus –x 9
   ```

   `–r 1` indique le mode maître ou appelant ; `-s venus`, le nom du système distant et `-x 9`, le niveau de mise au point maximal (informations de mise au point les plus détaillées).

2. Si l'entrée expect-send du fichier Systems au format **/etc/uucp/Systems** est :

   ```
   venus Any venus 1200 – "" \n in:--in: uucp1 word:
     mirror
   ```

   Le démon **uucico** connecte le système local au système distant `venus`. La mise au point renvoie un résultat du type :

   ```
   expect: ""
   got  it
   sendthem (^J^M)
   expect (in:)^
   M^Jlogin:got it
   sendthem (uucp1^M)
   ```

```
expect (word:)^
M^JPassword:got it
sendthem (mirror^M)
imsg >^M^J^PShere^@Login Successful: System=venus
```

où :

`expect: ""`

Indique que le système local n'attend pas d'informations de la part du système distant.

`got it`

Accusé de réception du message.

`sendthem (^J^M)`

Indique que le local system va envoyer au système distant un retour chariot et un caractère de ligne suivante.

`expect (in:)`

Indique que le système local attend du système distant une invite de connexion, terminée par la chaîne `in:`.

`^M^Jlogin:got it`

Confirme que le système local va recevoir l'invite de connexion distante.

`sendthem (uucp1^M)`

Indique que le système va envoyer l'ID de connexion `uucp1` au système distant.

`expect (word:)`

Indique que le système local attend du système distant une invite de connexion, terminée par la chaîne `word:`.

`^M^JPassword:got it`

Confirme que le système local a reçu l'invite de connexion distante.

`sendthem (mirror^M)`

Indique que le système local va envoyer le mot de passe pour l'ID de connexion `uucp1` au système distant.

`imsg >^M^J^PShere^@Login Successful: System=venus`

Confirme la connexion du système local au système distant `venus`.

Remarques :

1. La sortie de mise au point "expect-send" générée par la commande **uucico** peut provenir d'informations issues du fichier **/etc/uucp/Dialers** ou du fichier **/etc/uucp/Systems**. Les informations de communication sur le modem sont extraites du fichier **Dialers**, et celles sur le système distant, du fichier **Systems**. (Notez que **/etc/uucp/Systems** et **/etc/uucp/Dialers** sont les fichiers de configuration par défaut de BNU. D'autres fichiers peuvent être spécifiés dans **/etc/uucp/Sysfiles** pour jouer le même rôle.)
2. Pour établir une connexion avec un système distant, vous devez connaître la séquence de connexion à ce système.

Communication avec des systèmes UNIX via la commande tip

Utilisez la commande **tip** pour contacter un système connecté et exploité sous UNIX. La commande **tip**, installée avec BNU, peut utiliser les mêmes connexions asynchrones que celles de BNU.

La commande **tip** a recours à des variables, séquences d'échappement et indicateurs. Les indicateurs peuvent être entrés à partir de la ligne de commande. Les séquences d'échappement peuvent être utilisées sur une connexion à distance pour lancer, réacheminer et arrêter des transferts de fichier et se brancher sur un sous-shell.

Variables de tip

Les variables de la commande **tip** définissent des paramètres tels que le caractère de fin de ligne, le signal d'interruption et le mode de transfert de fichier. Les variables peuvent être initialisées au moment de l'exécution via un fichier **.tiprc**. Les variables peuvent également être modifiées pendant l'exécution via le signal d'échappement **~s**. Certaines variables, tel le caractère de fin de ligne, peuvent être définies pour un système particulier, via l'entrée propre à ce système dans le fichier **remote**.

La commande **tip** lit trois fichiers (**phones**, **remote** et **.tiprc**) pour déterminer la configuration initiale de ses variables. Le fichier **.tiprc** doit toujours résider dans le répertoire personnel de l'utilisateur. Le nom et l'implantation des fichiers **remote** et **phones** peuvent varier. Pour déterminer le nom des fichiers **remote** et **phones**, vous disposez des variables d'environnement :

PHONES Fichier téléphonique de l'utilisateur. Ce fichier peut avoir n'importe quel nom valide et doit respecter le format du fichier **/usr/lib/phones–file**. Le fichier par défaut est **etc/phones**. Le cas échéant, le fichier spécifié avec la variable **PHONES** remplace le fichier .

REMOTE Fichier de définition du système distant de l'utilisateur. Ce fichier peut avoir n'importe quel nom valide et doit respecter le format du fichier **/usr/lib/remote–file**. Le fichier par défaut est **/etc/remote**. Le cas échéant, le fichier spécifié avec la variable **REMOTE** remplace le fichier **/etc/remote**.

Une variable d'environnement n'est applicable que définie avant le lancement de la commande **tip**. Vous pouvez également redéfinir le nom des fichiers **phones** et **remote** à l'aide de la commande **tip** et des variables **phones** et **remote** dans le fichier **.tiprc**.

Remarque : La commande **tip** ne lit que le *dernier* fichier **remote** ou **phones** spécifié. Ainsi, si vous spécifiez un fichier **remote** ou **phones** avec une variable, le nouveau fichier remplace ceux spécifiés précédemment.

La commande **tip** lit les variables dans l'ordre suivant :

1. La commande vérifie la valeur des variables d'environnement **PHONES** et **REMOTE** pour les fichiers à utiliser en tant que **phones** et **remote**.
2. La commande lit le fichier **.tiprc** et attribue ensuite une valeur aux variables. Si la variable **phones** ou **remote** est définie dans le fichier **.tiprc**, cette valeur prime sur celle de la variable d'environnement.
3. Lorsqu'une connexion à un système distant est lancée, la commande lit l'entrée du fichier **remote** pour ce système. La valeur de l'entrée du fichier **remote** prime sur celle définie dans le fichier **.tiprc**.
4. Si l'indicateur **-***débit_baud* est associé à la commande **tip**, le taux spécifié prime sur ceux définis précédemment.
5. La valeur attribuée à une variable via la séquence d'échappement **~s** prime sur toute valeur précédemment affectée à cette variable.

   Remarque : Tout utilisateur de **tip** peut créer un fichier **.tiprc** et l'utiliser pour spécifier la valeur initiale de ces variables **tip**. Le fichier **.tiprc** doit être placé dans le répertoire **$HOME** de l'utilisateur.

Fichiers de configuration de tip

La commande **tip** ne permet la connexion à un système distant qu'une fois les fichiers **/etc/remote** et **/etc/phones** constitués.

/etc/remote Définit les attributs des systèmes distants, tels que le port et le type d'unité à utiliser pour atteindre le système, de même que les signaux signifiant le début et la fin des transmissions.

/etc/phones Répertorie les numéros d'appel pour contacter des systèmes distants via un modem.

Pour constituer l'un de ces fichiers, copiez un fichier-type, changez son nom et modifiez-le pour l'adapter à votre site. Des exemples de fichiers **remote** et **phones** sont fournis dans le module bos.net.uucp. Le fichier **phones** exemple est appelé **/usr/lib/remote–file**. Le fichier **phones** exemple est appelé **/usr/lib/phones–file**. Copiez **/usr/lib/remote–file** dans **/etc/remote** and modify **/etc/remote**.

Un utilisateur de **tip** peut également créer des fichiers **remote** et **phones** personnalisés. Un fichier **remote** doit respecter le format du fichier **/usr/lib/remote-file** et être spécifié avec la variable **remote** ou la variable d'environnement **REMOTE**. Un fichier **phones** doit respecter le format du fichier **/usr/lib/phones-file** et être spécifié avec la variable **phones** ou la variable d'environnement **PHONES**. Associer une de ces variables au fichier **phones** ou **remote** permet de le lire *à la place* (et non en plus) du fichier **/etc/phones** ou **/etc/remote**.

Les utilisateurs de **tip** peuvent combiner les fichiers **phones** et **remote**. Par exemple, un utilisateur peut utiliser le fichier **remote** par défaut, **/etc/remote**, et utiliser un fichier **phones** personnel appelé avec la variable **phones**.

Fichiers de configuration BNU

BNU (Basic Network Utilities) utilise les fichiers de configuration suivants :

/etc/uucp	Contient tous les fichiers de configuration pour BNU.
/var/spool/uucppublic	Contient les fichiers qui ont été transférés.
/etc/uucp/Systems	Contient la liste des systèmes auxquels le programme **uucico** peut se connecter.
/etc/uucp/Devices	Définit le type, l'emplacement, la vitesse et autres paramètres de communication des unités pour plusieurs programmes d'appel de système. Seules les connexions sortantes utilisent ce fichier.
/etc/uucp/Permissions	Etablit un contrôle de sécurité, avec restrictions d'accès, pour les machines qui tentent de communiquer avec la vôtre.
/etc/uucp/Dialers	Spécifie les types de numéroteurs. Chaque numéroteur utilise un jeu de commandes spécifique pour appeler le modem. Les types de numéroteurs les plus courants sont `hayes`, `direct` et `TCP` (Transmission Control Protocol).
/etc/uucp/Dialcodes	Crée des noms normalisés pour remplacer certaines portions d'un numéro d'appel. Par exemple, si vous utilisez fréquemment le code d'une zone de San Francisco, vous pouvez créer l'entrée suivante : `SF09,1415`.
/etc/uucp/Sysfiles	Permet à l'administrateur BNU de spécifier des fichiers remplaçant les fichiers de configuration BNU **/etc/uucp/Systems**, **/etc/uucp/Devices** et **/etc/uucp/Dialers.** La distinction est possible entre les fichiers destinés aux activités **uucico** et **cu** (**cu**, **ct**, **slattach**).
/usr/sbin/uucp/remote.unknown	Définit un script shell. Il est exécuté par BNU lorsqu'une machine distante non répertoriée dans la liste des machines autorisées tente de communiquer avec le système local.
/etc/uucp/Poll	Planifie l'interrogation des systèmes passants. Son format est similaire à celui du fichier **crontab**. La séquence d'appel observe le format : *nom_site*, tabulation et heures d'appel (0-23), séparés par des espaces.

Corrélation de fichiers

Fichier Systems :	`SystemName Any v32ibm 9600 555-1111`
Fichier Devices :	`v32ibm tty0 - Any ibm \D`
Fichier Dialers :	`ibm =, -, #" \d ATSFI\r\c#OK#AFE1SD3L2MIC0SCI\r\c#OK...`

Exemple de configuration BNU pour connexion TCP/IP

Les fichiers suivants sont configurés pour une connexion TCP/IP (Transmission Control Protocol/Internet Protocol) entre les systèmes `zeus` et `hera`, où `zeus` est supposé être le système local et `hera` le système distant.

Entrées dans les fichiers du système local

Les fichiers contenant des entrées de connexion téléphonique sur le système local `venus` sont les suivants :

Fichier Systems

Pour que `zeus` puisse contacter `hera`, le fichier **Systems** sur `zeus` doit comporter la ligne :

```
hera Any TCP,t - - in:--in: uzeus word: birthday
```

Cette ligne indique que le système `zeus` peut appeler `hera` à tout moment via le protocole **t** pour communiquer avec le système `hera`. Le système `zeus` se connecte au système `hera` comme `uzeus` avec le mot de passe `birthday`.

Remarque : Le protocole **t** prend en charge le protocole **tcp**. Par conséquent, utilisez toujours le protocole **t** pour les communications BNU via des connexions TCP/IP. En revanche, le protocole **t** n'est pas admis avec une connexion par modem ou dont le champ *Type* est `ACU` (automatic calling unit, unité d'appel automatique).

BNU se fonde sur les champs *Type* et *Class* du fichier **Systems** pour déterminer l'unité adaptée à la connexion. Sur cette base, il recherche une entrée de type `TCP` dans le fichier **Devices**.

Fichier Devices

Un fichier **Devices** utilisé par **uucico** sur le système `zeus` doit comporter pour les connexions TCP/IP l'entrée :

```
TCP  -  -  -  TCP
```

Le type d'unité étant `TCP`, il n'y a pas d'entrées *Class*, *Line* ou *Line2*. De même, `TCP` est également spécifié pour **Dialer**. BNU recherche alors une entrée `TCP` pour les fichiers **Dialers**.

Fichiers Dialers

Le fichier **Dialers** utilisé par **uucico** sur `zeus` doit comporter l'entrée TCP/IP :

```
TCP
```

Cette entrée indique qu'aucune configuration de numéroteur n'est requise.

Remarque : La configuration du numéroteur n'est jamais requise sur une connexion TCP/IP.

Fichier Permissions

Pour donner à `hera` accès à `zeus`, le fichier **Permissions** du système `zeus` contient l'entrée :

```
LOGNAME=uhera SENDFILES=yes REQUEST=yes \
 MACHINE=zeus:hera VALIDATE=uhera /
 READ=/var/spool/uucppublic:/home/hera \
 WRITE=/var/spool/uucppublic:/home/hera COMMANDS=ALL
```

La combinaison des entrées LOGNAME et MACHINE fournit au système `hera` les droits d'accès au système `zeus` :

- `hera` peut demander et envoyer des fichiers quel que soit l'émetteur de l'appel.
- `hera` peut lire et écrire sur le répertoire public et le répertoire **/home/hera** du système `zeus`.
- `hera` peut exécuter toutes les commandes sur le système `zeus`.
- `hera` doit se connecter à `zeus` sous le nom d'utilisateur `uhera` et ne peut pas utiliser d'autre ID de connexion pour des transactions BNU.

Remarque : Les droits d'accès restent inchangés quel que soit le système émetteur de l'appel, c'est pourquoi les entrées LOGNAME et MACHINE sont combinées. Spécifiées séparément, elles se présentent comme suit :

```
LOGNAME=uhera VALIDATE=hera SENDFILES=yes REQUEST=yes& \
READ=/var/spool/uucppublic:/home/hera \
WRITE=/var/spool/uucppublic:/home/hera

MACHINE=zeus:hera REQUEST=yes COMMANDS=ALL\
READ=/var/spool/uucppublic:/home/hera \
WRITE=/var/spool/uucppublic:/home/hera
```

Entrées dans les fichiers du système distant

Les fichiers contenant des entrées de connexion téléphonique sur le système distant `merlin` sont les suivants :

Fichier Systems

Pour permettre à `hera` de contacter `zeus`, un fichier **Systems** sur `hera` doit comporter l'entrée suivante :

```
zeus Any TCP,t - - ogin:--ogin: uhera ord: lightning
```

Cette ligne indique que le système `hera` peut appeler `zeus` à tout moment via le protocole **t** pour communiquer avec le système `zeus`. Le système `hera` se connecte à `zeus` sous le nom d'utilisateur `uhera` et le mot de passe `lightning`. De nouveau, BNU recherche une entrée de type `TCP` dans les fichiers **Devices**.

Remarque : Le protocole **t** prend en charge le protocole **tcp**. Par conséquent, utilisez toujours le protocole **t** pour les communications BNU via des connexions TCP/IP. Le protocole **t** ne peut toutefois être utilisé si le champ *Type* a la valeur `ACU` ou lorsque la connexion est établie via un modem.

Fichier Devices

Le fichier **Devices** utilisé par **uucico** sur `hera` doit comporter pour les connexions TCP/IP l'entrée suivante :

```
TCP  -  -  -  TCP
```

Le type d'unité étant `TCP`, il n'y a pas d'entrées *Class*, *Line* ou *Line2*. De même, `TCP` est également spécifié pour `Dialer`. BNU recherche alors une entrée `TCP` pour les fichiers **Dialers**.

Fichiers Dialers

Le fichier **Dialers** utilisé par **uucico** sur le système `hera` doit comporter une entrée TCP/IP comme suit :

```
TCP
```

Cette entrée indique qu'aucune configuration de numéroteur n'est requise.

Remarque : La configuration du numéroteur n'est jamais requise sur une connexion TCP/IP.

Fichier Permissions

Pour donner à `zeus` accès à `hera`, le fichier **Permissions** du système `hera` contient l'entrée :

```
LOGNAME=uzeus SENDFILES=yes REQUEST=yes \
 MACHINE=hera:zeus VALIDATE=zeus COMMANDS=rmail:who:uucp
```

La combinaison des entrées LOGNAME et MACHINE fournit au système `zeus` les droits d'accès au système `hera` :

- `zeus` peut demander et envoyer des fichiers quel que soit l'émetteur de l'appel.

- zeus peut lire et écrire uniquement sur le répertoire public (par défaut).
- zeus ne peut exécuter que les commandes **rmail**, **who** et **uucp.**
- zeus doit se connecter à hera sous le nom d'utilisateur uzeus et ne peut pas utiliser d'autre ID de connexion pour les transactions BNU.

  Remarque : Séparément, les entrées LOGNAME et MACHINE se présentent comme suit :

```
LOGNAME=uzeus VALIDATE=zeus SENDFILES=yes REQUEST=yes
MACHINE=hera:zeus COMMANDS=rmail:who:uucp REQUEST=yes
```

Exemple de configuration BNU pour connexion téléphonique

Les fichiers exemples suivants sont configurés pour connecter les systèmes venus et merlin par le biais d'une ligne téléphonique et de modems. venus est supposé être le système local et merlin le système distant.

Sur les deux systèmes, l'unité tty1 est raccordée à un modem Hayes à 1200 bauds. L'ID de connexion utilisé par venus pour se connecter à merlin est uvenus et le mot de passe associé est mirror. L'ID de connexion utilisé par merlin pour se connecter à venus est umerlin et le mot de passe associé est oaktree. Le numéro d'appel du modem raccordé à venus est 9=3251436, celui du modem raccordé à merlin est 9=4458784. Les deux machines comportent des numéros d'appel partiels dans leurs fichiers **Systems** et des codes d'accès dans leurs fichiers **Dialcodes**.

Entrées sur le système local

Les fichiers contenant des entrées de connexion téléphonique sur le système local venus sont les suivants :

Fichier Systems

Le fichier **Systems** sur venus doit comporter une entrée pour merlin incluant un numéro et un préfixe d'appel, comme suit :

```
merlin Any ACU 1200 local8784 "" in:--in: uvenus word: mirror
```

venus peut appeler merlin à tout moment avec une unité ACU à 1200 bauds, sous le nom uvenus et le mot de passe mirror. Le numéro d'appel est développé avec le code local dans le fichier **Dialcodes** et l'unité à utiliser est déterminée en fonction des entrées *Type* et *Class*. Sur cette base, BNU recherche une unité de type ACU et de classe 1200 dans les fichiers **Devices**.

Fichier Dialcodes

Le fichier **Dialcodes** sur venus comporte le préfixe d'appel à associer au numéro figurant dans le fichier **Systems** :

```
local 9=445
```

Ainsi, le développé du numéro d'appel pour le système merlin dans **Systems** est 9=4458784.

Fichier Devices

Le fichier **Devices** côté venus doit comporter, pour la connexion à merlin, l'entrée suivante :

```
ACU tty1  -  1200  hayes \T
```

Le port à utiliser est tty1 et la valeur associée à l'entrée *Dialer* dans le champ *Dialer-Token Pairs* est hayes. Pour l'entrée *Token*, \T indique que le numéro d'appel est développé à l'aide d'un code issu du fichier **Dialcodes**. BNU recherche le numéroteur hayes dans les fichiers **Dialers**.

Fichiers Dialers

Un fichier **Dialers** utilisé par **uucico** sur `venus` doit comporter pour le modem `hayes` l'entrée suivante :

```
hayes =,-, "" \dAT\r\c OK \pATDT\T\r\c CONNECT
```

Remarque : La séquence expect-send est définie dans le format de fichier **Dialers.**

Fichier Permissions

Pour spécifier au système `merlin` le mode d'exécution des transactions **uucico** et **uuxqt** avec le système `venus`, le fichier **Permissions** sur `venus` doit contenir les entrées :

```
LOGNAME=umerlin REQUEST=yes SENDFILES=yes \
READ=/var/spool/uucppublic:/home/merlin \
WRITE=/var/spool/uucppublic:/home/merlin
MACHINE=venus:merlin VALIDATE=umerlin REQUEST=yes SENDFILES=yes
 \
COMMANDS=ALL \
READ=/var/spool/uucppublic:/home/merlin \
WRITE=/var/spool/uucppublic:/home/merlin
```

Le système `merlin` se connecte à `venus` sous le nom de `umerlin`, qui est un nom de connexion unique pour le système `merlin`. Il peut demander et envoyer des fichiers quel que soit l'émetteur de l'appel. Le système `merlin` peut en outre lire et écrire dans le répertoire **/var/spool/uucppublic** et dans le répertoire **/home/merlin** sur le système `venus`. Il peut lancer toutes les commandes du jeu de commandes par défaut sur le système `venus`.

Entrées sur le système distant

Les fichiers contenant des entrées de connexion téléphonique sur le système distant `merlin` sont les suivants.

Fichier Systems

Un fichier **Systems** sur `merlin` doit contenir pour `venus` une entrée incluant un numéro et un préfixe d'appel, comme suit :

```
venus Any ACU 1200 intown4362 "" in:--in: umerlin word: oaktree
```

`merlin` peut appeler `venus` à tout moment, avec une unité `ACU` à `1200` bauds, sous le nom `umerlin` et le mot de passe `oaktree`. Le numéro d'appel est développé avec le code `intown` dans le fichier **Dialcodes**, et l'unité à utiliser est déterminée en fonction des entrées *Type* et *Class*. Sur cette base, BNU recherche une unité de type `ACU` et de classe `1200` dans les fichiers **Devices.**

Fichier Dialcodes

Le fichier **Dialcodes** sur `merlin` comporte le préfixe d'appel à associer au numéro figurant dans le fichier **Systems** :

```
intown 9=325
```

Ainsi, le numéro d'appel développé pour accéder au système `venus` est `9=3254362`.

Fichier Devices

Pour la connexion à `venus`, un fichier **Devices** côté `merlin` doit comporter la ligne :

```
ACU  tty1  -  1200  hayes \T
```

L'unité ACU est raccordée au port `tty1` et le numéroteur est `hayes`. Le numéro d'appel est développé à l'aide des informations extraites du fichier **Dialcodes**. BNU recherche une entrée pour modem `hayes` dans les fichiers **Dialers**.

Fichiers Dialers

Un fichier **Dialers** utilisé par **uucico** sur `merlin` doit comporter pour son modem l'entrée :

```
hayes  =,-,  ""  \dAT\r\c OK \pATDT\T\r\c CONNECT
```

Fichier Permissions

Pour que `venus` puisse accéder à `merlin`, le fichier **Permissions** sur `merlin` doit comporter les entrées suivantes :

```
LOGNAME=uvenus SENDFILES=call REQUEST=no \
WRITE=/var/spool/uucppublic:/home/venus \
READ=/var/spool/uucppublic:/home/venus
MACHINE=merlin:venus VALIDATE=uvenus  \
READ=/ WRITE=/ COMMANDS=ALL REQUEST=yes \
NOREAD=/etc/uucp:/usr/etc/secure \
NOWRITE=/etc/uucp:/usr/etc/secure
```

Exemple de configuration BNU pour connexion directe

Les fichiers suivants sont configurés pour une connexion câblée entre les systèmes `zeus` et `hera`, où `zeus` est supposé être le système local et `hera` le système distant. L'unité câblée est `tty5` sur `zeus` et `tty1` côté `hera`. La vitesse de connexion est `1200` bps. L'ID de connexion à `zeus` sur `hera` est `uzeus` avec le mot de passe associé `thunder`. L'ID de connexion à `hera` sur `zeus` est `uhera` avec le mot de passe `portent`.

Entrées dans les fichiers du système local

Les fichiers contenant des entrées de connexion téléphonique sur le système local `venus` sont les suivants :

Fichier Systems

Un fichier **Systems** sur `zeus` doit contenir pour le système distant `hera` l'entrée suivante :

```
hera Any hera 1200 - "" \r\d\r\d\r in:--in: uzeus word:  thunder
```

Cette entrée spécifie que `hera` peut se connecter à `zeus` à tout moment, via une connexion directe spécifiée dans les fichiers **Devices**. Pour trouver cette entrée dans les fichiers **Devices**, BNU utilise le troisième et le quatrième champs de l'entrée **Systems**. BNU recherche dans les fichiers **Devices** une entrée dont le champ *Type* a la valeur `hera` et la *classe* `1200`. Le système `zeus` se connecte au système `hera` sous le nom `uzeus` et le mot de passe `thunder`.

Fichier Devices

Pour la connexion au système distant `hera`, le fichier **Devices** sur `zeus` doit comporter l'entrée :

```
hera    tty5  -  1200  direct
```

Cette entrée indique que le système `zeus` utilise l'unité `tty5` à `1200` bps pour communiquer avec `hera`. Notez que *Dialer* dans les deux champs *Dialer-Token Pairs* a la valeur `direct`. Lors de la connexion à `hera`, BNU recherche une entrée `direct` dans le fichier **Dialers**.

Fichiers Dialers

Pour les connexions directes, un fichier **Dialers** sur `zeus` doit comporter l'entrée :

```
direct
```

Cette entrée spécifie qu'aucune mise en liaison ("handshaking") n'est requise pour la connexion directe.

Fichier Permissions

Pour spécifier au système `hera` le mode d'exécution des transactions **uucico** et **uuxqt** avec `zeus`, le fichier **Permissions** du système `zeus` doit contenir les entrées suivantes :

```
LOGNAME=uhera MACHINE=hera VALIDATE=uhera REQUEST=yes \
 SENDFILES=yes MACHINE=zeus READ=/ WRITE=/ COMMANDS=ALL
```

Cette entrée indique que le système `hera` se connecte sous le nom de `uhera`. L'option `VALIDATE=uhera` étant incluse, le système `hera` ne peut pas se connecter à `zeus` avec un autre ID de connexion, et aucun autre système distant ne peut utiliser l'ID `uhera`. Le système `hera` peut lire et écrire sur n'importe quel répertoire du système `zeus`, et demander et envoyer des fichiers quel que soit l'émetteur de l'appel. `hera` peut également lancer toutes les commandes sur le système `zeus`.

Remarque : Les droits d'accès accordés sont les mêmes quel que soit l'émetteur de la connexion. C'est pourquoi les entrées LOGNAME et MACHINE ont été combinées. Spécifiées séparément, elles se présentent comme suit :

```
LOGNAME=uhera REQUEST=yes SENDFILES=yes READ=/ WRITE=/
MACHINE=zeus:hera VALIDATE=uhera READ=/ WRITE=/ REQUEST=yes \
COMMANDS=ALL
```

Attention : Attribuer des droits d'accès comme dans l'exemple précédent équivaut à doter tout utilisateur résidant sur le système distant d'un ID de connexion sur le système local. Ces droits d'accès souples peuvent nuire à la sécurité de votre système et il est conseillé de ne les accorder qu'aux systèmes distants sécurisés résidant sur le même site.

Entrées dans les fichiers du système distant

Les fichiers contenant des entrées de connexion téléphonique sur le système distant `merlin` sont les suivants.

Fichier Systems

Un fichier **Systems** sur le système `hera` doit contenir l'entrée suivante pour le système `zeus`:

```
zeus Any zeus 1200 - "" \r\d\r\d\r in:--in: uhera word: portent
```

Cette entrée spécifie que `hera` peut se connecter à `zeus` à tout moment, via une connexion directe spécifiée dans les fichiers **Devices**. Pour trouver cette entrée dans les fichiers **Devices**, BNU utilise le troisième et le quatrième champs de l'entrée **Systems**. BNU recherche une entrée dans les fichiers **Devices** avec pour *Type* la valeur `zeus` et pour *Class* la valeur `1200`. Le système `hera` se connecte à `zeus` sous le nom d'utilisateur `uhera` et le mot de passe `portent`.

Fichier Devices

Un fichier **Devices** sur le système `hera` doit comporter pour la communication avec `zeus` l'entrée suivante :

```
zeus    tty1  -  1200  direct
```

Cette entrée indique que le système `hera` utilise l'unité `tty1` à `1200` bps pour communiquer avec le système `zeus`. **Dialer** étant positionné sur `direct`, BNU recherche dans les fichiers **Dialers** une entrée `direct`.

Fichiers Dialers

Un fichier **Dialers** sur `hera` doit comporter pour les connexions directes l'entrée :

```
direct
```

Cette entrée indique qu'aucune configuration de numéroteur n'est requise sur la connexion directe.

Fichier Permissions

Le fichier **Permissions** sur `hera` doit comporter les entrées ci-dessous pour spécifier le mode d'exécution des transactions **uucico** et **uuxqt** par `zeus` sur `hera` :

```
LOGNAME=uzeus REQUEST=yes SENDFILES=yes READ=/ WRITE=/
MACHINE=hera:zeus VALIDATE=uzeus REQUEST=yes COMMANDS=ALL READ=/\
WRITE=/
```

Ces entrées indiquent que le système `zeus` se connecte à `hera` sous le nom de `uzeus`. L'option `VALIDATE=uzeus` étant incluse, le système `zeus` ne peut pas se connecter à `hera` avec un autre ID de connexion et aucun autre système distant ne peut utiliser l'ID `uzeus`. Le système `zeus` peut lire et écrire sur n'importe quel répertoire du système `hera`, et demander et envoyer des fichiers quel que soit l'émetteur de l'appel. `zeus` peut également lancer toutes les commandes sur le système `hera`.

Ces entrées indiquent que le système `zeus` se connecte à `hera` sous le nom de `uzeus`. L'option `VALIDATE=uzeus` étant incluse, le système `zeus` ne peut pas se connecter à `hera` avec un autre ID de connexion et aucun autre système distant ne peut utiliser l'ID `uzeus`. Le système `zeus` peut lire et écrire dans n'importe quel répertorie sur le système `hera`, et peut envoyer et demander des fichiers quel que soit l'auteur de l'appel. Le système `zeus` peut également lancer toutes les commandes sur le système `hera`.

Attention : Attribuer des droits d'accès comme dans l'exemple précédent équivaut à doter tout utilisateur résidant sur le système distant d'un ID de connexion sur le système local. Ces droits d'accès souples peuvent nuire à la sécurité de votre système et il est conseillé de ne les accorder qu'aux systèmes distants résidant sur le même site.

Référence des fichiers, commandes et répertoires BNU

Répertoires BNU

/etc/uucp	Contient tous les fichiers de configuration BNU (Basic Network Utilities).
/var/locks	Contient les fichiers de verrouillage pour les unités système. Utilisé par les autres sous-systèmes en complément de BNU.
/var/spool/uucppublic	Contient les fichiers qui ont été transférés par BNU.
/var/spool/uucp	Contient les fichiers administratifs BNU.
/var/spool/uucp/.Workspace	Contient les fichiers temporaires utilisés en interne par les programmes de transport de fichier.
/var/spool/uucp/.Xqtdir	Contient les fichiers exécutables avec les listes des commandes exécutables par les systèmes distants.
/var/spool/uucp/ *SystemName*	Contient les fichiers utilisés par les programmes de transport de fichier.

Fichiers BNU

/etc/uucp/Systems	Liste de systèmes auxquels **uucico** peut se connecter.
/etc/uucp/Devices	Définit les paramètres de communication de base pour les connexions sortantes.
/etc/uucp/Permissions	Définit les autorisations pour les machines distantes contactant la machine locale via BNU.
Maxuuscheds	Limite les travaux programmés simultanément.
Maxuuxqts	Limite les exécutions simultanées de commandes à distance.
/etc/uucp/Dialers	Spécifie le type du modem et du numéroteur.
/etc/uucp/Dialcodes	Contient les premiers chiffres des numéros de téléphone utilisés pour établir les connexions à distance via une ligne téléphonique.
/usr/sbin/uucp/remote.unknown	Script shell exécuté lorsqu'un ordinateur distant inconnu tente d'établir une communication.
/usr/sbin/uucp/Sysfiles	Affecte des fichiers système, unité et numéroteur, secondaires ou supplémentaires.
/etc/uucp/Poll	Détermine le moment d'appel d'un système distant.
uudemon.admin	Envoie un rapport d'état BNU à l'ID de connexion spécifié.
uudemon.cleanu	Nettoie les répertoires de spoulage BNU à des moments programmés.
uudemon.hour	Lance les appels de transport de fichier vers les systèmes distants.
uudemon.poll	Interroge les systèmes distants répertoriés dans le fichier **/etc/uucp/Poll**.
/var/spool/uucp/audit	Contient les informations d'audit relatives aux activités BNU.

/var/spool/uucp/Foreign	Contient les informations relatives aux erreurs intervenues au cours des activités BNU.
/var/spool/uucp/errors	Contient les informations relatives aux erreurs intervenues au cours des activités BNU.
/var/spool/uucp/xferstats	Contient les informations relatives aux statistiques sur les activités BNU.
/var/spool/uucp/Corrupt	Contient la copie des fichiers que le programme BNU ne peut pas traiter.
/var/spool/uucp/.Log	Contient les fichiers journaux issus des transactions BNU courantes.
/var/spool/uucp/.Old	Contient les fichiers journaux issus des anciennes transactions BNU.
/var/spool/uucp/.Status	Prend date de la dernière tentative du démon **uucico** de communiquer avec les systèmes distants.
/var/spool/uucp/ *SystemName***/ C.***	Ces fichiers sont les commandes autorisées lors d'une connexion avec *SystemName*.
/var/spool/uucp/ *SystemName***/ D.***	Ces fichiers sont des fichiers de données associés à *SystemName*.
/var/spool/uucp/ *SystemName***/ X.***	Fichiers exécutables sur *SystemName*.
/var/spool/uucp/ *SystemName***/ TM.***	Fichiers temporaires utilisés pendant la connexion à *SystemName*.

Commandes BNU

ct	Établit la connexion à un autre système via une ligne téléphonique.
cu	Établit la connexion à un autre système.
tip	Variante de **cu** qui nécessite une configuration particulière.
uucp	Copie les fichiers d'un système vers un autre système exécutant BNU ou une version d'UUCP (UNIX–to–UNIX Copy Program).
uudecode	Reconstitue un fichier binaire codé avec **uuencode**.
uuencode	Code un fichier binaire dans un format ASCII pour la transmission via BNU.
uuname	Fournit des informations sur les systèmes accessibles.
uupoll	Force un appel à un système distant.
uuq	Affiche la file d'attente des travaux BNU.
uusend	Envoie un fichier à un hôte distant exécutant BNU ou UUCP.
uusnap	Affiche un récapitulatif succinct de l'état de BNU.
uustat	Rend compte de l'état des opérations BNU.
uuto	Copie des fichiers vers un autre système exécutant BNU ou UUCP.
uux	Exécute une commande sur un système distant.
uucheck	Recherche dans le fichier **/etc/uucp/Permissions** la configuration correcte.
uuname	Affiche les noms de tous les systèmes accessibles via BNU.
uuclean	Nettoie les répertoires de spoulage BNU.
uucleanup	Nettoie les répertoires de spoulage BNU.
uukick	Contacte un système distant avec la mise au point activée.
uulog	Affiche les fichiers journaux BNU.

uutry Contacte un système distant avec la mise au point activée. Ne tient pas compte des délais impartis pour les tentatives.

uucpadm Administre le système BNU.

uupick Permet de récupérer des fichiers dans le répertoire **/var/spool/uucppublic.**

uucp ID de connexion avec droits d'accès administratifs complets sur le sous-système BNU.

Uutry Contacte un système distant avec la mise au point activée et sauvegarde la sortie de mise au point dans un fichier.

Démons BNU

uucico Contacte les systèmes distants et transfère les fichiers.

uucpd Permet l'exécution de BNU sur le dessus de TCP/IP (Transmission Control Protocol/Internet Protocol).

uusched Planifie les travaux BNU.

uuxqt Exécute des requêtes de commande à partir de systèmes distants.

Annexe A. Cartes PCI

Cette section contient des informations sur l'installation et la configuration des cartes PCI. Les sujets abordés sont le support et la configuration des cartes PCI WAN (Wide Area Network) (Pilote d'unité multiprotocole HDLC 2 ports : Généralités page A-1 et Carte ARTIC960Hx PCI : Généralités page A-2).

Cartes PCI WAN (Wide Area Network)

Cette section décrit les conditions requises pour l'installation et la configuration de la carte multiprotocoles 2 ports (PCI), et de la carte ARTIC960Hx PCI.

Pilote d'unité multiprotocole HDLC 2 ports : généralités

Le pilote de carte multiprotocole HDLC (High Level Data Link Control) 2 ports est un composant du sous–système d'E/S de communication. Ce pilote d'unité prend en charge les opérations HDLC sur la carte multiprotocole 2 ports, à la vitesse de 1544 Mbps au maximum.

Les options ci–dessous permettent l'accès au pilote de carte multiprotocole HDLC 2 ports :

- Architecture unifiée de réseau (SNA)
- Version SDLC (synchronous data link control) de l'interface de programmation GDLC
- Applications utilisateur compatibles avec l'API (Application Programming Interface) SDLC MPQP (Multiprotocol Quad Port).

Remarque : Ces options requièrent l'utilisation du fichier spécial **mpc** *n*, qui donne accès à la carte multiprotocole 2 ports via le sous–système d'émulation de pilote d'unité SDLC COMIO. Ce sous–système doit être installé et configuré pour chaque unité HDLC du réseau.

- Applications utilisateur compatibles avec l'API HDLC CDLI

Le pilote d'unité de la carte multiprotocole 2 ports permet la connectivité à des systèmes hôte distants via la carte multiprotocole 2 ports, soit directement via une ligne spécialisée, soit via des circuits commutés. Le pilote d'unité peut fournir une passerelle entre les environnements de groupe de travail (work group) et les fonctions informatiques à distance.

Configuration de la carte multiprotocole 2 ports

Le tableau ci–dessous expliquent comment configurer la carte multiprotocole 2 ports.

Tâche	**Raccourci SMIT**	Web-based System Manager Management Environment [1]
Ajout d'un pilote d'unité	**smit mkhdlcdpmpdd**	
Reconfiguration du pilote d'unité	**smit chhdlcdpmpdd**	
Retrait d'un pilote d'unité	**smit rmhdlcdpmpdd**	
Déclaration d'un pilote d'unité disponible	**smit cfghdlcdpmpdd**	
Retrait d'un émulateur SDLC COMIO	**smit mksdlcsciedd**	
Reconfiguration de l'émulateur SDLC COMIO	**smit chsdlcsciedd**	
Retrait d'un émulateur SDLC COMIO	**smit rmsdlcsciedd**	
Passage d'un émulateur SDLC COMIO défini à l'état disponible	**smit cfgsdlcsciedd**	

Remarque : Ces tâches ne sont pas disponibles dans l'environnement de gestion Web-based System Manager.

Carte ARTIC960Hx PCI : Généralités

Le pilote d'unité MPQP COMIO de la carte ARTIC960Hx PCI est un élément du sous–système d'E/S de communication. Ce pilote assure la prise en charge de la carte ARTIC960Hx PCI à la vitesse maximale de 2 Mbps. La synchronisation incombe aux modems, car seule une synchronisation externe est prise en charge.

Les options suivantes donnent accès au pilote d'unité MPQP COMIO de la carte ARTIC960Hx PCI :

- Architecture unifiée de réseau (SNA)
- Interface de programmation générique GDLC (Generic Data Link Control)
- Applications utilisateur compatibles avec l'API (Application Programming Interface) MPQP (Multiprotocol Quad Port), par exemple les applications SDLC et bisynchrones.

Ces options requièrent l'utilisation du fichier spécial **mpq** *x*, qui donne accès à la carte ARTIC960HX PCI via le pilote d'unité d'émulation MPQP COMIO. Ce pilote d'unité doit être installé et configuré pour chaque port de la carte ARTIC960Hx PCI. Le fichier **mpq** *x* réside dans le répertoire **/dev**.

Remarque : Le *x* de **mpq** *x* spécifie l'instance du pilote d'unité, par exemple `mpq0`.

Le pilote d'unité d'émulation MPQP COMIO permet de connecter des systèmes hôtes distants par le biais d'une carte ARTIC960Hx PCI, soit directement, soit par l'intermédiaire d'une ligne spécialisée. Le pilote d'unité peut fournir une passerelle entre les environnements de groupe de travail (work group) et les fonctions informatiques à distance.

Configuration du pilote d'émulation MPQP COMIO sur la carte ARTIC960Hx PCI

Le tableau suivant explique comment configurer le pilote d'émulation MPQP COMIO sur la carte ARTIC960Hx PCI.

Tâche	**Raccourci SMIT**	Web-based System Manager Management Environment [1]
Ajout d'un pilote d'unité	**smit mktsdd**	
Reconfiguration du pilote d'émulation MPQP COMIO	**smit chtsdd**	
Suppression d'un pilote d'unité	**smit rmtsdd**	
Configuration d'un pilote d'unité défini	**smit cfgtsdd**	
Ajout d'un port	**smit mktsdports**	
Reconfiguration d'un port d'émulation MPQP COMIO	**smit chtsdports**	
Suppression d'un port	**smit rmtsdports**	
Configuration d'un port défini	**smit cfgtsdports**	
Suivi d'un pilote d'émulation MPQP COMIO	**smit trace_link**	

Remarque : Ces tâches ne sont pas disponibles dans l'environnement de gestion Web-based System Manager.

Annexe B. Configuration de la sauvegarde de l'interface réseau dans les versions précédentes d'AIX

Ces instructions vous guident dans les étapes de configuration d'un EtherChannel dans AIX 4.3.3 et AIX 5.1. Pour plus d'informations sur EtherChannel, ou pour savoir comment effectuer cette tâche dans AIX 5.2 et les versions supérieures, reportez–vous à Agrégation de liaison Channel et IEEE 802.3ad page 4-186.

1. Avec les droits root, tapez `smit etherchannel` sur la ligne de commande.
2. Sélectionnez **Add an Etherchannel** et appuyez sur Entrée.
3. Sélectionnez les cartes (principales et secondaires) que vous voulez inclure dans l'EtherChannel.

   Remarque : **Cartes réseau disponibles** affiche toutes les cartes Ethernet. Si vous sélectionnez une carte Ethernet déjà utilisée, vous obtenez un message d'erreur. Vous devez d'abord détacher cette interface à l'aide de la commande **ifconfig**.
4. Entrez les informations dans les zones en respectant les consignes suivantes :
   - **Cartes Etherchannel**: Vous devez voir s'afficher les cartes sélectionnées dans l'étape précédente.
   - **Activation adresse ETHERCHANNEL secondaire**: Ce champ est facultatif. Le choix de la valeur `yes` vous permet de préciser une adresse MAC que l'EtherChannel doit utiliser. Si vous définissez la valeur `no` pour cette option, l'EtherChannel utilisera l'adresse MAC de la première carte indiquée.
   - **ALTERNATE ETHERCHANNEL address**: Si vous attribuez à **Enable ALTERNATE ETHERCHANNEL address** la valeur `yes`, indiquez l'adresse MAC à utiliser par l'EtherChannel. L'adresse indiquée doit commencer par `0x` et être une valeur hexadécimale à 12 chiffres.
   - **Mode**: Sélectionnez `netif_backup`.
   - **Enable GIGABIT ETHERNET JUMBO frames**: Ce champ est facultatif. Pour l'utiliser, voter commutateur doit prend en charge les trames jumbo. Ceci fonctionne uniquement avec Ethernet standard et non avec IEEE 802.3. Affectez la valeur `yes` si vous voulez l'activer.
   - **Internet Address to Ping**: Ce champ est facultatif. L'EtherChannel lance une commande ping sur l'adresse IP indiquée. Si l'EtherChannel ne parvient pas à lancer une commande ping au terme du nombre de tentatives précisé dans **Number of Retries** dans les intervalles **Retry Timeout**, l'EtherChannel effectue un basculement des cartes.
   - **Number of Retries**: Entrez le nombre d'échecs de réponses ping autorisés avant que l'EtherChannel ne change de cartes. La valeur par défaut est 3. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.
   - **Retry Timeout**: Entrez le nombre de secondes entre les envois de commande ping de l'EtherChannel à **Internet Address to Ping**. La valeur par défaut est une seconde. Ce champ est facultatif et valide uniquement si vous définissez **Internet Address to Ping**.

5. Appuyez sur Entrée après avoir modifié les champs voulus pour créer EtherChannel.
6. Configurez IP via la nouvelle interface en tapant `smit chinet` sur la ligne de commande.
7. Sélectionnez votre nouvelle interface EtherChannel dans la liste.
8. Remplissez tous les champs requis et appuyez sur Entrée.

Annexe C. Table de conversion

ASCII	Décimal	Hexadécimal	Octal	Binaire
nul	0	0	0	0
début d'en–tête	1	1	1	1
début du texte	2	2	2	10
fin du texte	3	3	3	11
fin de transmission	4	4	4	100
interroger	5	5	5	101
accuser réception	6	6	6	110
sonnerie	7	7	7	111
espacement arrière	8	8	10	1000
tabulation horizontale	9	9	11	1001
saut de ligne	10	A	12	1010
tabulation verticale	11	B	13	1011
présentation du formulaire	12	C	14	1100
retour chariot	13	D	15	1101
décalage externe	14	E	16	1110
décalage interne	15	F	17	1111
échappement liaison de données	16	10	20	10000
contrôle d'unité 1/Xon	17	11	21	10001
contrôle d'unité 2	18	12	22	10010
contrôle d'unité 3/Xoff	19	13	23	10011
contrôle d'unité 4	20	14	24	10100
accusé de réception négatif	21	15	25	10101
mode inactif synchrone	22	16	26	10110
bloc de fin de transmission	23	17	27	10111
annuler	24	18	30	11000
fin de support	25	19	31	11001
fin de fichier/substitut	26	1A	32	11010
échappement	27	1B	33	11011
séparateur de fichier	28	1C	34	11100

séparateur de groupe	29	1D	35	11101
séparateur d'enregistrement	30	1E	36	11110
séparateur d'unité	31	1F	37	11111
espace	32	20	40	100000
!	33	21	41	100001
”	34	22	42	100010
#	35	23	43	100011
$	36	24	44	100100
%	37	25	45	100101
&	38	26	46	100110
’	39	27	47	100111
(	40	28	50	101000
)	41	29	51	101001
*	42	2A	52	101010
+	43	2B	53	101011
,	44	2C	54	101100
–	45	2D	55	101101
.	46	2E	56	101110
/	47	2F	57	101111
0	48	30	60	110000
1	49	31	61	110001
2	50	32	62	110010
3	51	33	63	110011
4	52	34	64	110100
5	53	35	65	110101
6	54	36	66	110110
7	55	37	67	110111
8	56	38	70	111000
9	57	39	71	111001
:	58	3A	72	111010
;	59	3B	73	111011
<	60	3C	74	111100
=	61	3D	75	111101
>	62	3E	76	111110
?	63	3F	77	111111
@	64	40	100	1000000
A	65	41	101	1000001
B	66	42	102	1000010
C	67	43	103	1000011
D	68	44	104	1000100
E	69	45	105	1000101

F	70	46	106	1000110
G	71	47	107	1000111
H	72	48	110	1001000
I	73	49	111	1001001
J	74	4A	112	1001010
K	75	4B	113	1001011
L	76	4C	114	1001100
M	77	4D	115	1001101
N	78	4E	116	1001110
O	79	4F	117	1001111
P	80	50	120	1010000
Q	81	51	121	1010001
R	82	52	122	1010010
S	83	53	123	1010011
T	84	54	124	1010100
U	85	55	125	1010101
V	86	56	126	1010110
W	87	57	127	1010111
X	88	58	130	1011000
Y	89	59	131	1011001
Z	90	5A	132	1011010
[	91	5B	133	1011011
\	92	5C	134	1011100
]	93	5D	135	1011101
^	94	5E	136	1011110
_	95	5F	137	1011111
‘	96	60	140	1100000
a	97	61	141	1100001
b	98	62	142	1100010
c	99	63	143	1100011
d	100	64	144	1100100
e	101	65	145	1100101
f	102	66	146	1100110
g	103	67	147	1100111
h	104	68	150	1101000
i	105	69	151	1101001
j	106	6A	152	1101010
k	107	6B	153	1101011
l	108	6C	154	1101100
m	109	6D	155	1101101
n	110	6E	156	1101110
o	111	6F	157	1101111
p	112	70	160	1110000

q	113	71	161	1110001
r	114	72	162	1110010
s	115	73	163	1110011
t	116	74	164	1110100
u	117	75	165	1110101
v	118	76	166	1110110
w	119	77	167	1110111
x	120	78	170	1111000
y	121	79	171	1111001
z	122	7A	172	1111010
{	123	7B	173	1111011
|	124	7C	174	1111100
}	125	7D	175	1111101
~	126	7E	176	1111110
SUPPR	127	7F	177	1111111
	128	80	200	10000000
	129	81	201	10000001
	130	82	202	10000010
	131	83	203	10000011
	132	84	204	10000100
	133	85	205	10000101
	134	86	206	10000110
	135	87	207	10000111
	136	88	210	10001000
	137	89	211	10001001
	138	8A	212	10001010
	139	8B	213	10001011
	140	8C	214	10001100
	141	8D	215	10001101
	142	8E	216	10001110
	143	8F	217	10001111
	144	90	220	10010000
	145	91	221	10010001
	146	92	222	10010010
	147	93	223	10010011
	148	94	224	10010100
	149	95	225	10010101
	150	96	226	10010110
	151	97	227	10010111
	152	98	230	10011000
	153	99	231	10011001
	154	9A	232	10011010
	155	9B	233	10011011

	156	9C	234	10011100
	157	9D	235	10011101
	158	9E	236	10011110
	159	9F	237	10011111
	160	A0	240	10100000
	161	A1	241	10100001
	162	A2	242	10100010
	163	A3	243	10100011
	164	A4	244	10100100
	165	A5	245	10100101
	166	A6	246	10100110
	167	A7	247	10100111
	168	A8	250	10101000
	169	A9	251	10101001
	170	AA	252	10101010
	171	AB	253	10101011
	172	AC	254	10101100
	173	AD	255	10101101
	174	AE	256	10101110
	175	AF	257	10101111
	176	B0	260	10110000
	177	B1	261	10110001
	178	B2	262	10110010
	179	B3	263	10110011
	180	B4	264	10110100
	181	B5	265	10110101
	182	B6	266	10110110
	183	B7	267	10110111
	184	B8	270	10111000
	185	B9	271	10111001
	186	BA	272	10111010
	187	BB	273	10111011
	188	BC	274	10111100
	189	BD	275	10111101
	190	BE	276	10111110
	191	BF	277	10111111
	192	C0	300	11000000
	193	C1	301	11000001
	194	C2	302	11000010
	195	C3	303	11000011
	196	C4	304	11000100
	197	C5	305	11000101
	198	C6	306	11000110

	199	C7	307	11000111
	200	C8	310	11001000
	201	C9	311	11001001
	202	CA	312	11001010
	203	CB	313	11001011
	204	CC	314	11001100
	205	CD	315	11001101
	206	CE	316	11001110
	207	CF	317	11001111
	208	D0	320	11010000
	209	D1	321	11010001
	210	D2	322	11010010
	211	D3	323	11010011
	212	D4	324	11010100
	213	D5	325	11010101
	214	D6	326	11010110
	215	D7	327	11010111
	216	D8	330	11011000
	217	D9	331	11011001
	218	DA	332	11011010
	219	DB	333	11011011
	220	DC	334	11011100
	221	DD	335	11011101
	222	DE	336	11011110
	223	DF	337	11011111
	224	E0	340	11100000
	225	E1	341	11100001
	226	E2	342	11100010
	227	E3	343	11100011
	228	E4	344	11100100
	229	E5	345	11100101
	230	E6	346	11100110
	231	E7	347	11100111
	232	E8	350	11101000
	233	E9	351	11101001
	234	EA	352	11101010
	235	EB	353	11101011
	236	EC	354	11101100
	237	ED	355	11101101
	238	EE	356	11101110
	239	EF	357	11101111
	240	F0	360	11110000
	241	F1	361	11110001

	242	F2	362	11110010
	243	F3	363	11110011
	244	F4	364	11110100
	245	F5	365	11110101
	246	F6	366	11110110
	247	F7	367	11110111
	248	F8	370	11111000
	249	F9	371	11111001
	250	FA	372	11111010
	251	FB	373	11111011
	252	FC	374	11111100
	253	FD	375	11111101
	254	FE	376	11111110
	255	FF	377	11111111

INDEX

Symboles

Numbers

A

B

C

D

E

F

G

I

L

M

N

R

S

T

U

V

W

X

Vos remarques sur ce document / Technical publication remark form

Titre / Title : Bull Guide de gestion du système : Communications et réseaux

Nº Reférence / Reference Nº : 86 F2 27EF 03

Daté / Dated : Juin 2003

ERREURS DETECTEES / ERRORS IN PUBLICATION

AMELIORATIONS SUGGEREES / SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Vos remarques et suggestions seront examinées attentivement.
Si vous désirez une réponse écrite, veuillez indiquer ci-après votre adresse postale complète.

Your comments will be promptly investigated by qualified technical personnel and action will be taken as required.
If you require a written reply, please furnish your complete mailing address below.

NOM / NAME : _____ Date : _____

SOCIETE / COMPANY : _____

ADRESSE / ADDRESS : _____

_____

Remettez cet imprimé à un responsable BULL ou envoyez-le directement à :

Please give this technical publication remark form to your BULL representative or mail to:

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

Technical Publications Ordering Form

Bon de Commande de Documents Techniques

To order additional publications, please fill up a copy of this form and send it via mail to:
Pour commander des documents techniques, remplissez une copie de ce formulaire et envoyez-la à :

BULL CEDOC
ATTN / MME DUMOULIN
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

Managers / Gestionnaires :
Mrs. / Mme : **C. DUMOULIN** +33 (0) 2 41 73 76 65
Mr. / M : **L. CHERUBIN** +33 (0) 2 41 73 63 96

FAX : +33 (0) 2 41 73 60 19
E–Mail / Courrier Electronique : srv.Cedoc@franp.bull.fr

Or visit our web sites at: / Ou visitez nos sites web à:
http://www.logistics.bull.net/cedoc
http://www-frec.bull.com **http://www.bull.com**

CEDOC Reference # Nº Référence CEDOC	**Qty** Qté	**CEDOC Reference #** Nº Référence CEDOC	**Qty** Qté	**CEDOC Reference #** Nº Référence CEDOC	**Qty** Qté
_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]	
_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]	
_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]	
_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]	
_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]	
_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]	
_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]		_ _ _ _ _ _ _ _ _ _ _ _ _ [_ _]	
[_ _] : **no revision number means latest revision** / pas de numéro de révision signifie révision la plus récente					

NOM / NAME : _____ Date : _____

SOCIETE / COMPANY : _____

ADRESSE / ADDRESS : _____

_____

PHONE / TELEPHONE : _____ FAX : _____

E–MAIL : _____

For Bull Subsidiaries / Pour les Filiales Bull :

Identification: _____

For Bull Affiliated Customers / Pour les Clients Affiliés Bull :

Customer Code / Code Client : _____

For Bull Internal Customers / Pour les Clients Internes Bull :

Budgetary Section / Section Budgétaire : _____

For Others / Pour les Autres :

Please ask your Bull representative. / Merci de demander à votre contact Bull.

PLACE BAR CODE IN LOWER LEFT CORNER

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
86 F2 27EF 03

Bull

Utiliser les marques de découpe pour obtenir les étiquettes.
Use the cut marks to get the labels.

AIX

Guide de gestion
du système :
Communications et
réseaux

86 F2 27EF 03

AIX

Guide de gestion
du système :
Communications et
réseaux

86 F2 27EF 03

AIX

Guide de gestion
du système :
Communications et
réseaux

86 F2 27EF 03