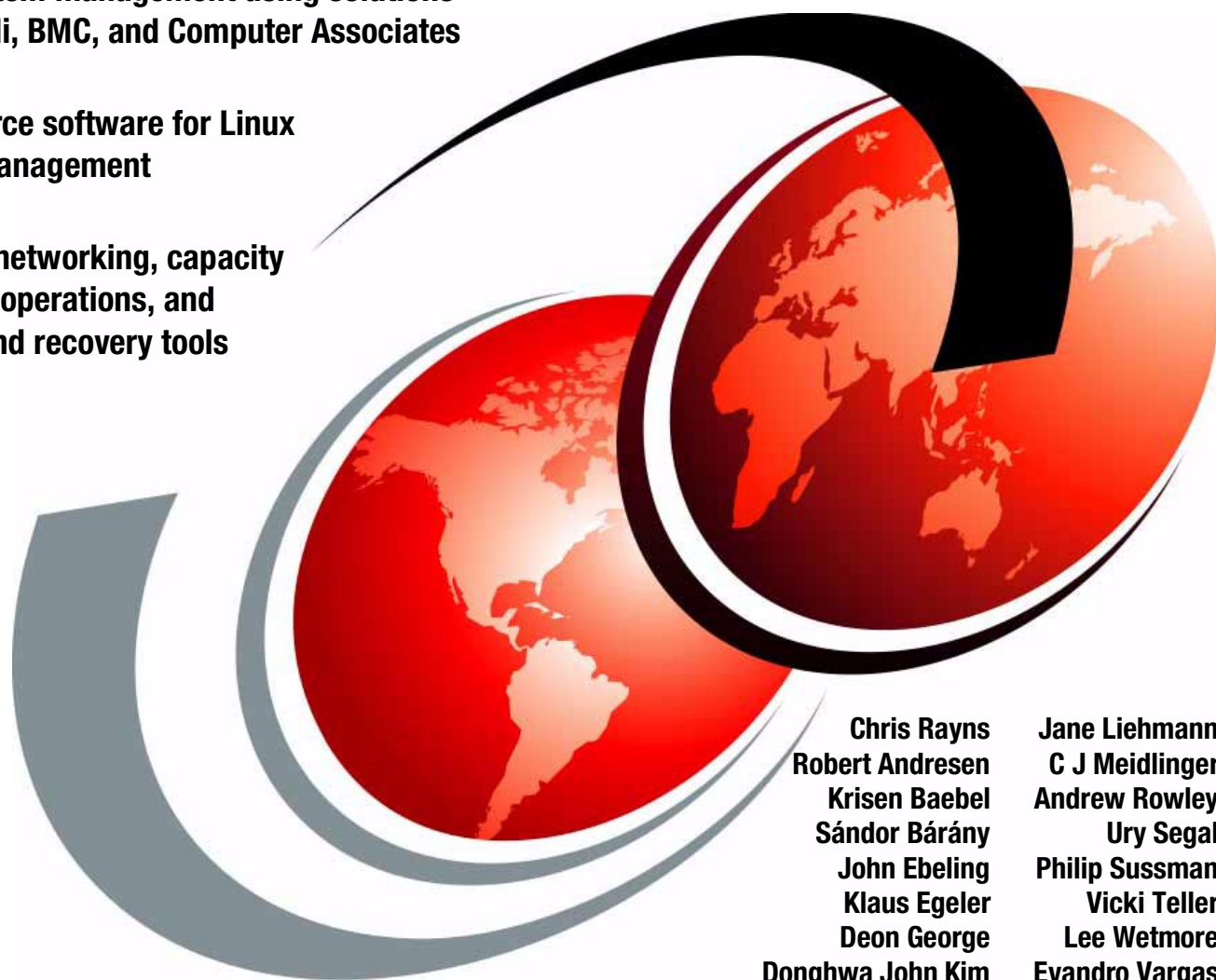


# Linux on IBM server<sup>™</sup> zSeries and S/390: System Management

Linux system management using solutions from Tivoli, BMC, and Computer Associates

Open source software for Linux system management

Security, networking, capacity planning, operations, and backup and recovery tools



Chris Rayns	Jane Liehmann
Robert Andresen	C J Meidlinger
Krisen Baebel	Andrew Rowley
Sándor Bárány	Ury Segal
John Ebeling	Philip Sussman
Klaus Egeler	Vicki Teller
Deon George	Lee Wetmore
Donghwa John Kim	Evandro Vargas

# Redbooks





International Technical Support Organization

**Linux on IBM @server zSeries and S/390:  
System Management**

November 2002

**Take Note!** Before using this information and the product it supports, be sure to read the general information in “Notices” on page ix.

**First Edition (November 2002)**

This edition applies to z/VM 4.3 and many different Linux distributions. RedHat 7.2 for zSeries and SuSE 7.0 were used for examples in this book.

Comments may be addressed to:  
IBM Corporation, International Technical Support Organization  
Dept. HYJ Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

**© Copyright International Business Machines Corporation 2002. All rights reserved.**

Note to U.S Government Users - Documentation related to restricted rights - Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>Notices</b> .....	ix
Trademarks .....	x
<b>Preface</b> .....	xi
The team that wrote this redbook .....	xi
Become a published author .....	xiii
Comments welcome .....	xiii

<b>Part 1. Overview of Linux on zSeries and S/390</b> .....	1
-------------------------------------------------------------	---

<b>Chapter 1. System overview</b> .....	3
1.1 System layout .....	4
1.2 z/VM configuration .....	5
1.2.1 Format and label DASD .....	6
1.2.2 Create user definitions .....	6
1.2.3 Logging on for the first time .....	9
1.2.4 Building the latest kernel .....	13
1.3 Cloning systems .....	15
1.4 Resource management .....	18
1.4.1 Network configuration .....	18
1.4.2 Memory configuration .....	29
1.4.3 CPU configuration .....	30
1.5 Monitoring .....	31
1.5.1 The indicate command .....	31
1.5.2 The RealTime Monitor (RTM) .....	32
1.5.3 FCON/ESA .....	34
<b>Chapter 2. RAID tools, LVM, and EVMS</b> .....	37
2.1 RAID tools .....	38
2.1.1 RAID levels .....	38
2.1.2 Setting up Level 0 RAID using the Red Hat installation tool .....	38
2.1.3 Setting up Level 1 RAID manually .....	41
2.2 EVMS .....	43
2.2.1 How to get and install EVMS .....	43
2.2.2 EVMS interface options .....	43
2.2.3 Terminology .....	44
2.2.4 How to start with EVMS .....	45
2.2.5 How to use EVMS .....	48
<b>Chapter 3. RMF PMS</b> .....	55
3.1 RMF PMS overview .....	56
3.1.1 How RMF PMS works .....	56
3.2 Planning for RMF PMS .....	57
3.3 Generating RMF PMS .....	60
3.4 Testing RMF PMS .....	64
3.4.1 Creating data views and a new machine RMF PM client .....	64
3.4.2 Creating a graph in RMF PM client .....	68
3.4.3 Testing the HTTP/XML interface in an RMF PMS .....	71
3.4.4 Testing archive and unarchive of performance data .....	73

3.5 RMF mix environment . . . . .	74
3.5.1 FCON/ESA . . . . .	74
3.5.2 z/OS RMF PM Java client . . . . .	75
3.5.3 RMF LDAP interface . . . . .	76
<b>Part 2. Open source software . . . . .</b>	<b>79</b>
<b>Chapter 4. Tripwire . . . . .</b>	<b>81</b>
4.1 Tripwire overview . . . . .	82
4.1.1 How Tripwire works . . . . .	82
4.2 Using Tripwire . . . . .	83
4.2.1 Planning . . . . .	83
4.2.2 Generating Tripwire . . . . .	83
4.2.3 Creating initial configuration file . . . . .	83
4.2.4 Creating and updating a database . . . . .	84
4.2.5 Testing Tripwire . . . . .	86
4.2.6 Tips . . . . .	93
<b>Chapter 5. Moodss . . . . .</b>	<b>95</b>
5.1 Prerequisites for moodss installation . . . . .	96
5.2 Installing moodss . . . . .	98
5.2.1 Download the source code . . . . .	98
5.2.2 Build RPM packages . . . . .	98
5.2.3 Install . . . . .	99
5.3 Using moodss . . . . .	99
5.3.1 Starting moodss . . . . .	99
5.3.2 Drag and drop . . . . .	107
5.3.3 Thresholds . . . . .	108
5.3.4 Poll time . . . . .	110
5.3.5 Creating and saving configurations files . . . . .	110
5.3.6 Developing your own moodss modules . . . . .	112
5.3.7 Where to find more information . . . . .	112
<b>Chapter 6. Amanda . . . . .</b>	<b>113</b>
6.1 Amanda overview . . . . .	114
6.2 Backup process . . . . .	114
6.2.1 Limitations . . . . .	114
6.2.2 Setting up the server . . . . .	115
6.2.3 Configuration details . . . . .	115
6.2.4 Setting up the clients . . . . .	120
6.3 Recovering data . . . . .	121
6.3.1 Recovering a client system . . . . .	124
6.3.2 Recovering the server . . . . .	126
<b>Chapter 7. OpenLDAP . . . . .</b>	<b>129</b>
7.1 Managing users using OpenLDAP and pam_ldap . . . . .	130
7.2 OpenLDAP overview . . . . .	130
7.2.1 PAM and pam_ldap . . . . .	130
7.2.2 Configuring OpenLDAP . . . . .	131
7.2.3 Configuring Name Service Switch . . . . .	137
7.2.4 Configuring PAM . . . . .	137
7.2.5 Securing the connection and verifying the server . . . . .	140
7.2.6 Generating the keys and certificates . . . . .	140
7.2.7 User administration . . . . .	142

7.2.8 Replicating servers using slurpd . . . . .	144
<b>Chapter 8. System Installation Suite . . . . .</b>	<b>147</b>
8.1 Features of SIS . . . . .	148
8.2 SIS components . . . . .	149
8.3 SIS on s390 . . . . .	150
8.4 Obtaining SIS . . . . .	151
<b>Part 3. Tivoli . . . . .</b>	<b>153</b>
<b>Chapter 9. Setting up the IBM Tivoli environment . . . . .</b>	<b>155</b>
9.1 Tivoli Storage Manager . . . . .	157
9.1.1 TSM Backup/Archive client . . . . .	158
9.1.2 Installation and configuration of the TSM Web client . . . . .	160
9.2 IBM Directory Server . . . . .	160
9.2.1 Installation . . . . .	161
9.2.2 Configuration . . . . .	162
9.3 IBM Tivoli Access Manager for e-business . . . . .	169
9.4 IBM Tivoli Identity Manager . . . . .	174
9.4.1 Component layout . . . . .	175
9.4.2 Installation . . . . .	175
9.5 Tivoli Management Framework . . . . .	177
9.5.1 Installation of the Tivoli Management Framework Server . . . . .	178
9.5.2 Installation of the Tivoli Management Framework Managed Node . . . . .	180
9.5.3 Installation of other Tivoli Framework products . . . . .	183
9.5.4 Installation of the Tivoli patches . . . . .	184
9.5.5 Installation of a Tivoli Endpoint gateway . . . . .	185
9.5.6 Installation of the Tivoli Endpoints . . . . .	186
9.5.7 The view from the Tivoli Desktop . . . . .	187
9.6 Tivoli Software Distribution . . . . .	188
9.6.1 Preparing to install Tivoli Software Distribution . . . . .	189
9.6.2 Installation of Tivoli Software Distribution . . . . .	190
9.6.3 Prepare the MDIST2 database . . . . .	192
9.7 Tivoli Distributed Monitoring (Advanced Edition) . . . . .	194
9.7.1 Understanding Tivoli Distributed Monitoring (Advanced Edition) . . . . .	195
9.7.2 Preparing for Tivoli Distributed Monitoring (Advanced Edition) . . . . .	196
9.7.3 Installation of Tivoli Distributed Monitoring (Advanced Edition) . . . . .	197
9.7.4 Preparing the Linux/390 Endpoints . . . . .	198
9.7.5 Installation of the Health Console . . . . .	200
9.8 IBM Tivoli Enterprise Console . . . . .	201
9.8.1 Component layout . . . . .	202
9.8.2 Installation . . . . .	203
<b>Chapter 10. System management using IBM Tivoli Software . . . . .</b>	<b>207</b>
10.1 Operations . . . . .	208
10.1.1 Availability management with Tivoli Distributed Monitoring (Advanced Edition) . . . . .	208
10.1.2 Software deployment with Tivoli Software Distribution . . . . .	218
10.2 Data Management . . . . .	228
10.2.1 File backup and restore . . . . .	228
10.3 Security . . . . .	233
10.3.1 User definition and administration . . . . .	233
10.3.2 Access control . . . . .	239
10.3.3 Firewalls . . . . .	240
10.3.4 Managing for audit . . . . .	244

<b>Part 4. BMC</b> .....	249
<b>Chapter 11. BMC products</b> .....	251
11.1 MAINVIEW for Linux Servers .....	252
11.1.1 Configuration/layout .....	252
11.1.2 Installation .....	253
11.1.3 Customization .....	254
11.2 PATROL for Linux Enterprise Server .....	262
11.2.1 Configuration/layout .....	263
11.2.2 Installation of PATROL for Linux Enterprise Manager .....	263
11.2.3 Customization .....	271
11.3 PATROL Internet Server Manager .....	271
11.3.1 Broad Range of Server Support .....	272
11.3.2 Configuration/layout .....	273
11.3.3 Installation .....	273
11.3.4 Customization .....	278
<b>Chapter 12. System management using MAINVIEW for Linux Servers</b> .....	283
12.1 Operations for MAINVIEW for Linux Servers .....	284
12.1.1 Availability Management .....	291
12.1.2 Health monitoring: Heartbeat data .....	293
12.1.3 Health monitoring: Automation .....	308
<b>Chapter 13. System management using PATROL</b> .....	317
13.1 Operations using PATROL for Linux Enterprise Server .....	318
13.1.1 Availability management .....	318
13.1.2 Health monitoring with PATROL for Linux Enterprise Server .....	321
13.1.3 Automation .....	323
13.2 Data management using PATROL for Linux Enterprise Server .....	329
13.2.1 Monitoring logs .....	329
13.3 Security monitoring using PATROL for Linux Enterprise Server .....	331
13.3.1 Monitoring files .....	332
13.3.2 Monitoring user activity .....	334
13.4 Operations using PATROL Internet Server Manager .....	337
13.4.1 Availability management .....	337
13.4.2 Health monitoring .....	347
13.5 Security monitoring using PATROL Internet Server Manager .....	347
13.5.1 SSL certificate monitoring .....	347
<b>Part 5. Computer Associates</b> .....	349
<b>Chapter 14. Computer Associates Linux solutions</b> .....	351
14.1 Computer Associates solutions for Linux for zSeries and S/390 .....	352
14.1.1 Unicenter: Enterprise management for Linux for zSeries and S/390 .....	352
14.1.2 eTrust: Security for Linux for zSeries and S/390 .....	352
14.1.3 BrightStor: Storage for Linux for zSeries and S/390 .....	353
14.1.4 Advantage: Data management and application development for Linux for zSeries and S/390 .....	353
14.1.5 z/VM Solutions for Linux for zSeries and S/390 .....	353
14.1.6 CleverPath: Portal solutions for Linux for zSeries and S/390 .....	354
14.2 Unicenter Network and Systems Management .....	354
14.2.1 Installing the Manager .....	354
14.2.2 Installing Agent Technology and Agents .....	355



14.3	Unicenter NSM performance management . . . . .	356
14.3.1	The System Agent . . . . .	357
14.3.2	The Performance Agents . . . . .	357
14.3.3	The DB2 Agent . . . . .	358
14.3.4	The Ingres Agent . . . . .	358
14.3.5	The Process Agent . . . . .	358
14.3.6	The Log Agent . . . . .	359
14.3.7	The Informix Database Agent . . . . .	360
14.4	Unicenter Management for Web Servers (Apache Agent) . . . . .	360
14.4.1	Installing the Apache Agent . . . . .	361
14.5	Unicenter Software Delivery Agent . . . . .	361
14.5.1	Installing Software Delivery . . . . .	362
14.6	Unicenter Universal Job Management Agent . . . . .	363
14.6.1	Installing the Unicenter Universal Job Management Agent . . . . .	363
14.7	eTrust Access Control . . . . .	366
14.7.1	Installing eTrust Access Control . . . . .	366
14.7.2	Invoking eTrust Access Control . . . . .	367
14.8	eTrust Directory . . . . .	368
14.8.1	Installing eTrust Directory . . . . .	369
14.9	eTrust CA-ACF2 Security (interface to Linux for zSeries and S/390) . . . . .	369
14.10	eTrust CA-Top Secret Security (interface to Linux for zSeries and S/390) . . . . .	370
14.11	PAM support for Computer Associates External Security Managers . . . . .	370
14.11.1	Installing a source distribution . . . . .	370
14.11.2	Installing a binary distribution . . . . .	371
14.12	eTrust Admin . . . . .	371
14.12.1	Using eTrust Admin in the native Linux 390 environment . . . . .	372
14.13	eTrust Audit . . . . .	372
14.13.1	Routing audit information from Linux . . . . .	372
14.14	BrightStor Enterprise Backup . . . . .	373
14.14.1	Installing BrightStor Enterprise Backup . . . . .	373
14.15	Advantage Ingres Enterprise Relational Database . . . . .	374
14.15.1	Installing Advantage Ingres Enterprise Relational Database . . . . .	374
14.16	Advantage CA-XCOM Data Transport . . . . .	377
14.16.1	Installing Advantage CA-XCOM . . . . .	378
14.17	Advantage Data Transport Agent . . . . .	379
14.17.1	Installation considerations . . . . .	379
14.17.2	Installing Advantage Data Transport . . . . .	379
14.18	VM:Manager VM Management Suite for Mainframe Linux . . . . .	380
14.18.1	Installing VM:Manager VM Management Suite for Mainframe Linux . . . . .	380
14.19	Unicenter VM:Account . . . . .	382
14.20	Unicenter VM:Operator . . . . .	382
14.21	Unicenter VM:Schedule . . . . .	383
14.22	Unicenter VM:Spool . . . . .	383
14.23	BrightStor VM:Backup . . . . .	383
14.24	BrightStor VM:Tape . . . . .	383
14.25	eTrust VM:Director . . . . .	384
14.26	Unicenter CA-Explore Performance Management for VM . . . . .	384
14.26.1	Installing Unicenter CA-Explore Performance Management for VM . . . . .	385
14.27	CleverPath Portal . . . . .	386
14.27.1	Installing CleverPath Portal . . . . .	387
<b>Chapter 15.</b>	<b>System management with Computer Associates software . . . . .</b>	<b>389</b>
15.1	Operations . . . . .	390

15.1.1	Availability management . . . . .	390
15.1.2	Health Monitoring . . . . .	399
15.1.3	Automation . . . . .	402
15.2	Data management . . . . .	407
15.2.1	File backup and restore . . . . .	407
15.2.2	Managing shared file systems . . . . .	408
15.3	Security . . . . .	414
15.3.1	User definition and administration . . . . .	414
15.3.2	Access control . . . . .	422
15.3.3	Audit . . . . .	427
15.4	Managing VM . . . . .	429
15.4.1	VM:Manager VM Management for Mainframe Linux . . . . .	429
15.5	Portals . . . . .	446
15.5.1	Portal management . . . . .	446
	<b>Related publications</b> . . . . .	451
	IBM Redbooks . . . . .	451
	Other resources . . . . .	451
	Referenced Web sites . . . . .	452
	How to get IBM Redbooks . . . . .	452
	IBM Redbooks collections . . . . .	452
	<b>Index</b> . . . . .	453

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:  
*IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.


This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

## Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

IBM eServer™	MORE™	SAA®	Console®
Redbooks(logo)™ 	MQSeries®	S/390®	TME®
AIX®	Multiprise®	SecureWay®	VM/ESA®
CICS®	MVS™	Sequent®	VTAM®
DB2®	OfficeVision®	SP™	WebSphere®
ECKD™	OfficeVision/VM™	SP1®	z/OS™
ESCON®	OS/2®	SP2®	z/VM™
Hummingbird®	OS/390®	SQL/DS™	zSeries™
IBM®	Perform™	Tivoli®	
IMS™	Redbooks™	Tivoli Enterprise™	
Informix®	RMF™	Tivoli Enterprise	

The following terms are trademarks of International Business Machines Corporation and Lotus Development Corporation in the United States, other countries, or both:

Domino™	Notes®	Lotus®	Word Pro®
---------	--------	--------	-----------

The following terms are trademarks of other companies:

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Unicenter, eTrust, BrightStor, AllFusion, Advantage, and CleverPath are trademarks or registered trademarks of Computer Associates International, Inc.

Other company, product, and service names may be trademarks or service marks of others.

# Preface

This IBM Redbook describes system management for Linux on IBM @server zSeries and S/390. It will help you understand the primary management processes for a set of Linux images on zSeries and S/390, and how to identify appropriate tools for your environment.

First we describe the setup of an S/390 environment with multiple Linux images. We discuss how to install and configure several of the most commonly used Linux distributions and the system management tools that are included with those distributions.

Next we discuss the installation, configuration, and use of open source software for both general and special purpose system management, highlighting the offerings from Tripwire, Moodss, Amanda, OpenLDAP, and System Installation Suite.

We then discuss the proprietary Linux system management solutions provided by IBM Tivoli, BMC, and Computer Associates. For each company's offerings, we present an overview of the products, details for installing and configuring them, and information on how to use the solutions to manage your Linux systems.

Many additional tools are available for Linux system management, both from open source projects and from commercial vendors. Because of the rapid growth in this area, it is impossible to provide comprehensive coverage of all the available products. However, the overview of system management issues for Linux presented in this book, along with details for a broad cross-section of products in the marketplace, is useful even for those considering solutions not specifically covered here.

## The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center.

**Chris Rayns** is an IT Specialist and project leader at the International Technical Support Organization, Poughkeepsie Center. He writes extensively and teaches IBM classes worldwide on all areas of S/390 Security. Before joining the ITSO, Chris worked in IBM Global Services in the UK as an IT Specialist.

**Robert Andresen** is a Principal Software Consultant with BMC Software. He has been with BMC for five years and has worked with Linux since 1995. He holds a degree in Mathematics from the Illinois Institute of Technology. His area of expertise at BMC includes the MAINVIEW and PATROL product lines. In addition, he has significant experience in MQSeries, z/OS, CICS, UNIX, DB2, and networking.

**Krisen Baebel** is an Information Developer with BMC Software. She has been with BMC for two years and has worked with Linux for the last year. Krisen holds a degree in English from Texas Tech University and has 6 years of experience in the technical communications field. Prior to joining BMC, she was a technical writing consultant in the computer hardware, software, and oil and gas industries.

**Sándor Bárány** is a Systems Architect with IBM Austria. Prior to joining IBM, he worked for the consulting firm M&M Investitionsberatungs GmbH. In addition to an MS degree in Mathematics from the ELTE University of Budapest, Hungary, Sándor has more than 25 years of IT experience. His areas of expertise include portability, connectivity, system architectures, and security; particularly as they apply to the banking industry.

**John Ebeling** is a Product Developer with BMC Software. He has been with BMC for 11 years and has worked with Linux since 2000. He holds a degree in Business Management from Tulane University. Before joining BMC, John worked for IBM for 18 years, on networking software for VM, MVS, and OS/2 systems. His areas of expertise include networking, configuration/systems management, GUIs, Linux, and technical writing.

**Klaus Egeler** is a IT Systems Management Specialist with IBM Global Services Germany. He has more than ten years of experience as a VSE and VM systems programmer. He has worked with Linux for S/390 for two years.

**Deon George** is a senior IT Specialist with IBM Software Group Australia. He has worked with Linux systems for ten years and with IBM for the last four years. He holds a degree in Business Information Systems. At IBM, Deon has focused on the IBM Tivoli portfolio of products, where he has provided architecture, design, installation, and consulting services for Tivoli customers and the IBM Tivoli sales team. His areas of expertise include Internet technologies, Linux and UNIX, networking and systems management.

**Donghwa John Kim** is a Staff Software Engineer at the Linux Technology Center in Poughkeepsie, NY. Prior to joining the Linux Technology Center in March of 2001, he had two years of experience in z/OS operating system function testing and development. He is actively involved in open source projects pertaining to systems management on Linux on zSeries and currently is in charge of porting of System Installation Suite (SIS) on zSeries. He holds a B.S. degree in Computer Science from Rutgers College.

**Jane Liehmann** is a Senior Technical Writer at Computer Associates. In her 18 years at Computer Associates, she has also worked in quality assurance and documentation support. She holds a BS in Computer Science from New York Institute of Technology.

**C. J. Meidlinger** works for IBM Tivoli Worldwide Education in the USA.

**Andrew Rowley** is a Senior OS/390 Technical Specialist with IBM Global Services Australia. He has 11 years of experience as an MVS and OS/390 systems programmer, as well as 3 years of Solaris system administration experience. At IBM GSA he works with both OS/390 and Linux on S/390.

**Ury Segal** is CTO and Co-founder of Aduva Inc. He has worked on the Mosix Kernel, a multi computer operating system. He has several years experience with GUI work, including Motif and Java, and has expertise in kernel programming in Solaris, HP/UX, AIX, SYSV.

**Philip Sussman** is a Senior Technical Writer at Computer Associates. He acted as lead writer and coordinator for the Computer Associates contribution to this redbook. He has 14 years of experience as a technical writer, including 7 years at Computer Associates. He holds a BA in English from State University of New York at Binghamton.

**Vicki Teller** was a Senior Technical Writer at Computer Associates for 4 years. Before joining Computer Associates, she worked as a Technical Writer for several software companies in Israel. She holds a BA degree in the History and Literature of Religions from Northwestern University.

**Lee Wetmore** is a Senior Software Quality Assurance Engineer for BMC. He has been with BMC for four years and has worked with Linux for the last year. His areas of expertise at BMC include the PATROL and MAINVIEW product lines. Prior to joining BMC, he spent 25 years as a systems programmer in mainframe and UNIX environments.

**Evandro Vargas** is a Senior IT Specialist with IBM Global Services Brazil. He has worked on mainframe systems VM and VSE since 1980, and with Linux for S/390 since for three years. His areas of expertise include management and support at outsourcing clients.

Thanks to the following people for their contributions to this project:

Bill White, Roy Costa, Dave Bennin, Gregory Geiselhart  
International Technical Support Organization, Poughkeepsie Center

Mike MacIsaac, Linux on zSeries Technical Marketing Support, Poughkeepsie

Carlos Ordonez, Linux Technical Support, Poughkeepsie

Luc Fontaine, for his help with moodss

Ben e Aldo

Axel Buecker, Tivoli Security Specialist, ITSO Austin

IBM Tivoli Worldwide Education

## Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- ▶ Use the online **Contact us** review redbook form found at:

[ibm.com/redbooks](http://ibm.com/redbooks)

- ▶ Send your comments in an Internet note to:

[redbook@us.ibm.com](mailto:redbook@us.ibm.com)

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization  
Dept. HYJ Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400







## Part 1

# Overview of Linux on zSeries and S/390

This part provides an overview of Linux on zSeries and S/390. It includes a description of the environment we established to test system management software for this redbook, a discussion of the most common Linux distributions, and explanations of some of the basic system management features incorporated into those distributions.





## **System overview**

In this chapter we describe the setup and configuration of a simple virtual network inside a System 390. This virtual network is connected to a real network resembling our company intranet network with real servers on the company network. The virtual network is also connected to the Internet in a configuration with 2 firewalls and a DMZ.

Our goal in this chapter is to show a generic company network with Linux/390 virtual machines. We do not discuss the best ways of setting up a network, nor the reason we chose to set it up the way we did. We simply want to use this generic network for demonstration purposes, to show how a network can be managed with proprietary tools like Tivoli, CA, and BMC, and also with Open Source tools.

After reading this chapter, you should be able to build and grow your own virtual network of Linux/390 systems—your own “Penguin Farm.”

# 1.1 System layout

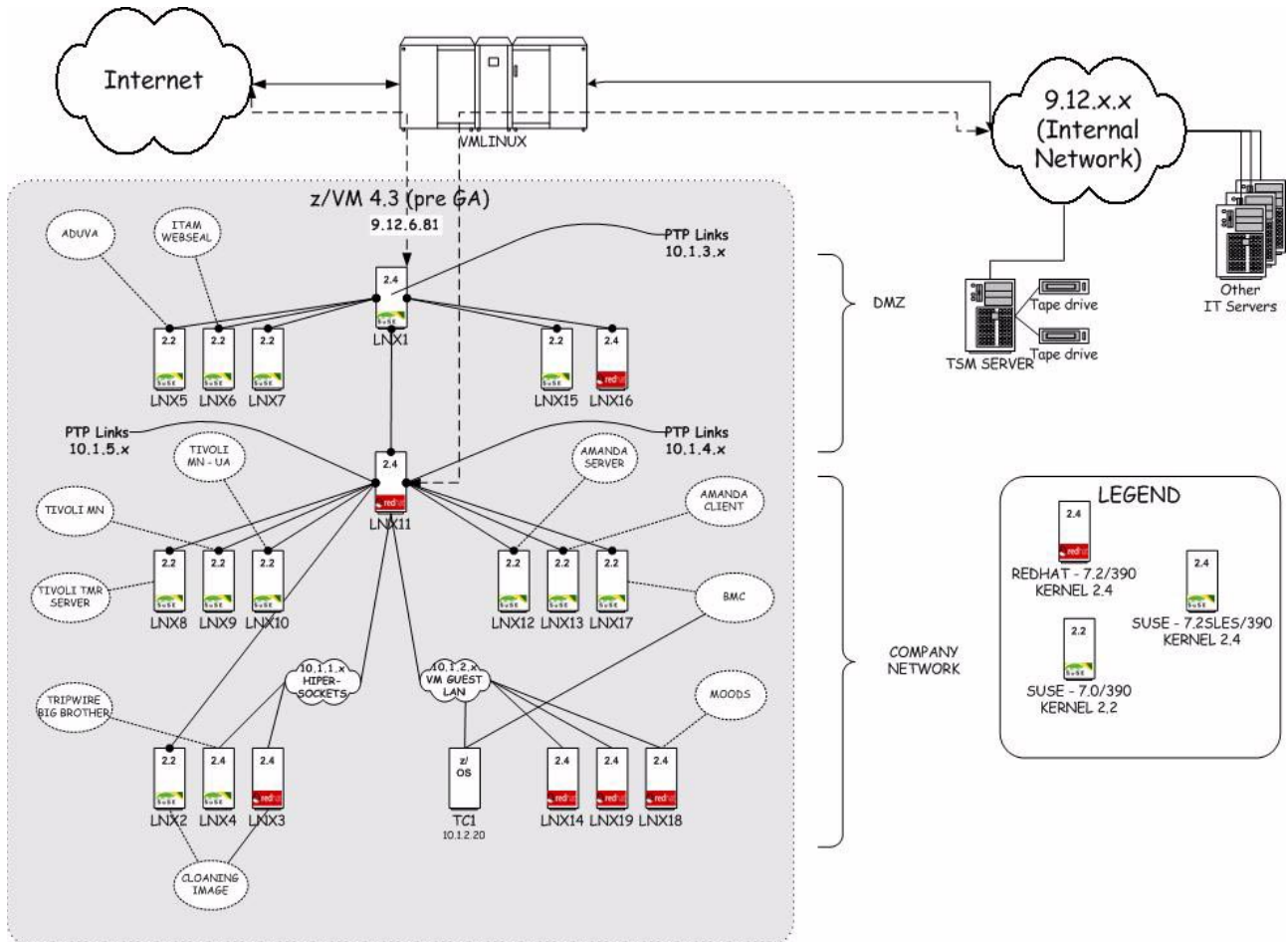


Figure 1-1 Our model virtual network.

Our final virtual network of Linux/390 systems is shown in Figure 1-1. Our mainframe z900/390 system (VMLINUX) is hosting the virtual Linux/390 servers and network shown in the shaded box.

Physically, VMLINUX is connecting us to the Internet (on the left), and our own network (9.12.x.x on the right).

Logically, VMLINUX is providing us with five networks (10.1.1 - 10.1.5) with 20 servers, each connected to one of the five logical networks. LNK1, in the logical network, has several network connections, joining our logical network (in the shaded box) and our real network (9.12.x.x) with the Internet on the left. LNK1 is our primary firewall and router.

LNK1 has point-to-point links to Linux hosts LNK5, 6, 7, 15, and 16, which are hosts that we have placed in our (virtual) "DMZ." We could have chosen to use a virtual LAN here instead of point-to-point links (as we have done in our virtual intranet). Choosing between virtual point-to-point links or a LAN for hosts in a DMZ would be a good discussion topic with your IT security department.

LNK1 has a point-to-point link to LNK11, providing us with a path to the Internet (for e-mail and Web access) for hosts underneath LNK11. It is also providing our Internal virtual LAN, together with an interface to our real network and other real hosts on our real LAN.

**Note:** This is not exactly true. We didn't have the interface on LNX11 to our internal network, but that's where we would have it in a real scenario. We just routed via LNX1 to see our hosts on the 9.12.x.x network.

Under LNX11, we have other hosts that are hosting our various IT servers and systems management applications. In fact, these could be our farm of File/Print servers, Web Application back-end servers, mail servers, and so forth.

While our example shows a simple corporate setup, there is no reason why it could not be scaled to include many more DMZs, virtual networks, and virtual hosts.

We used the following distributions of Linux to build our virtual penguin farm:

- ▶ RedHat 7.2 ([www.redhat.com](http://www.redhat.com)) based on the 2.4 kernel
- ▶ SuSE 7.0 ([www.suse.com](http://www.suse.com)) based on the 2.2 kernel
- ▶ SuSE 7.0SLES (also known as SuSE 7.2) based on the 2.4 kernel

Our VMLINUX host is using a pre-GA version of z/VM v4.30.

In an environment like ours, there are only two roles for our virtual Linux/390 systems to perform:

- ▶ Firewall/router systems

Two of our systems are configured as firewalls/routers (LNX1 and LNX11 in Figure 1-1). Normally for systems that are only being used for this purpose, the kernel is optimized and recompiled to include only networking functions, and very few software applications are installed (probably only monitoring and/or management ones, at that).

Since these virtual machines will really only be authorizing the "passing of packets" from one interface to another, they should be as light as possible for security, performance, and overall resource reasons.

We won't be describing how to optimize systems with this role in this book, so we didn't optimize our LNX1 and LNX11 virtual machines. (They were configured the same as our application systems).

- ▶ Application systems

Our other virtual Linux/390 systems are configured the same as a real system would be: to host applications. We configured our machines as per the default Linux install provided by the distributions that we use.

Our systems management applications are installed, so we can either manage or be managed by the other (application) systems.

## 1.2 z/VM configuration

In this section, we briefly describe how to set up the Linux z/VM guests. We assume that:

- ▶ You have z/VM already installed and running.
- ▶ You have assigned some DASD so that z/VM can use it for minidisks.
- ▶ You know how to define minidisks.
- ▶ You have a general understanding of how to use and administer z/VM.

In order to successfully build Linux systems as z/VM guests, you must perform the following steps:

- ▶ Format and label DASD.
- ▶ Create z/VM user definitions.
- ▶ Set up TCP/IP definitions (covered in Section 1.4.1, “Network configuration” on page 18).
- ▶ Log on for the first time and install Linux.
- ▶ Customize the kernel if required.
- ▶ Clone the Linux guest and personalize.

## 1.2.1 Format and label DASD

**Note:** It is assumed that the DASD has been previously initialized using `ICKDSF INSTALL`.

z/VM has the ability to take a physical DASD (Direct Access Storage Device or hard disk drive) and split it into smaller sizes, called minidisks. You can then assign minidisks to individual z/VM guest systems.

**Tip:** To be on the safe side, we recommend initializing and labelling all the Linux DASD with volume serial names that makes it obvious they are Linux disks. If one of these disks is inadvertently varied online to another LPAR, it should be obvious that the disk already belongs to a Linux environment.

One example of a volume serial naming convention is: LInnnn

LI	Identifies this disk as a Linux disk
nnnn	The device number of the volume

To format the DASD so that it is ready for z/VM, you need to:

- ▶ Attach the physical DASD device to a virtual device.
- ▶ Invoke the `CP FORMAT` utility.
- ▶ Format cylinder 0 so that z/VM will recognize it. (This is much faster than formatting the whole disk.)
- ▶ Enter an appropriate volume label when prompted.

Once the physical packs have been initialized, you should define the z/VM guest users. Part of the z/VM guest user definition is to define minidisks, subareas of the physical disks you've just initialized.

## 1.2.2 Create user definitions

The z/VM user directory describes the configuration and operating characteristics of each virtual machine that can be created by z/VM.

**Explanation:** For those not familiar with z/VM and mainframe terminology, each of our Linux Virtual Machines is a *user* of z/VM, and thus has a user login and password to start the session. Once a session is *started*, you need to *disconnect* to leave it running (using #CP DISC). If you log off a user, you are effectively “pulling the power plug out,” (so make sure you shut it down first).

Edit the user directory file with **xedit** as follows:

```
X USER DIRECT
```

**Note:** You must be logged on as MAINT to do this.

If certain directory control statements are repeated for several users, you can make use of directory profiles to save space in the directory. Since you can potentially create many hundreds of Linux z/VM guest systems on a single System 390, you can create a profile for Linux z/VM guests to define the things they have in common. Example 1-1 shows the IBMDFLT default profile.

*Example 1-1 User Profile for common definitions for each user.*

---

```
PROFILE IBMDFLT
  SPOOL 000C 2540 READER *
  SPOOL 000D 2540 PUNCH A
  SPOOL 000E 1403 A
  CONSOLE 009 3215 T
  LINK MAINT 0190 0190 RR
  LINK MAINT 019D 019D RR
  LINK MAINT 019E 019E RR
  LINK MAINT 0402 0402 RR
  LINK MAINT 0401 0401 RR
  LINK MAINT 0405 0405 RR
```

---

This profile comprises the following information:

- ▶ The name of the profile is IBMDFLT.
- ▶ The SPOOL statement defines virtual unit record devices. Virtual reader, punch, and printer devices are defined in this profile.
- ▶ The CONSOLE statement defines the virtual machine console.
- ▶ The LINK statements define links to the MAINT user’s minidisks. These minidisks contain common commands and utilities that most z/VM guests require.

Also in the USER DIRECT file, you define your z/VM guest users. A sample Linux user definition is shown in Example 1-2. Use this definition as a template for the creation of your Linux/390 users.

*Example 1-2 User direct definition for our first Linux user.*

---

```
USER LNX1    LNX1    128M 512M G
  INCLUDE IBMDFLT
  MACHINE XA
  IPL 190 PARM AUTOCHR
  MDISK 191 3390 3000 0025 430W01 MR READ    WRITE    MULTIPLE
  MDISK 201 3390 0001 0200 LX3752 MR READ    WRITE    MULTIPLE
  MDISK 202 3390 0201 3138 LX3752 MR READ    WRITE    MULTIPLE
*
```

---

The entries that make up the z/VM guest definition in USER DIRECT have the following meanings:

- ▶ The USER line sets the user ID of this virtual machine, which is LNX1, with the password also being LNX1.

This machine is defined to have a default memory storage of 128 MB when it logs in. If the user was to use the DEFINE STORAGE command, they would be allowed to increase the memory allocation to a maximum of 512 MB space. (This is a disruptive change).

Before increasing the storage size above 128 MB, it is important to analyze your z/VM paging environment and the characteristics of your Linux system's memory use. (This analysis is outside the scope of this document.)

- ▶ INCLUDE IBMDFLT will include the IBMDFLT profile, described previously.
- ▶ IPL 190 PARM AUTOOCR will make CMS boot automatically at logon (required for installation of Linux).
- ▶ MACHINE XA Sets the virtual machine to run in ESA mode.
- ▶ Each MDISK statement creates a minidisk for a z/VM guest user. The minidisk usage is defined in Table 1-1.

Table 1-1 Minidisk configuration

Minidisk	DASD Type	Size in Cylinders	Real DASD	Usage
191	3390	25	430W01	Guest's "home" disk (A-DISK). Used for the PROFILE EXEC and other CMS files.
201	3390	199	LX3752	Linux SWAP disk
202	3390	2938	LX3752	Linux root "/" filesystem.

This configuration gives each Linux approximately 1 GB of disk storage on the root file system. You may consider altering the disk layout of your Linux virtual machines so that you have more file systems (for example, /usr, /home, etc), and even have some file systems shared read only between Linux z/VM guests (for example /usr).

Use the requirements of your virtual machines to help you define your DASD storage allocations for each z/VM guest.

**Tip:** It is best to analyze your Linux virtual machine requirements and come up with a standard Linux image. Thus, when you create new Linux virtual machines, you may be able to automate the cloning of Linux user IDs if your USER DIRECT definitions are similar for each user.

**Note:** The maximum space on a single 3390 Model 3 is 3339 cylinders. This equates to roughly 2.8 GB. However, when the DASD is formatted to use the ext2 file system under Linux, you get around 2.3 GB of usable space.

Once you have your first Linux user definition, you can use that definition to define all the Linux z/VM guests that you will have. We defined Linux z/VM guests LNX1 to LNX19.

After all the definitions have been made, execute the following CMS command after you have saved the USER DIRECT file:

```
DISKMAP USER
```



The DISKMAP command summarizes the MDISK statements in the user directory and produces an output showing gaps and overlaps between minidisk assignments.

To search for and correct any conflicting minidisk assignments, use the following steps.

1. Edit the new USER DISKMAP file by running the command:

```
X USER DISKMAP
```

2. Search for the string *overlap* by typing:

```
/overlap <F8>
```

Repeat these steps until you have no more conflicts.

When the allocations are correct, run the command:

```
DIRECTXA USER (EDIT
```

If necessary, correct any syntax errors and issue this command again.

When all minidisk allocations and user definitions are correct, run the following command to update the z/VM user directory, so that the users are created and can log in:

```
DIRECTXA USER
```

**Note:** You can use DIRMAINT for user administration if it is available on your z/VM installation!

### 1.2.3 Logging on for the first time

You are now ready to set up the first Linux z/VM guest. Use the following steps for each of your new Linux virtual machines:

1. Set up the z/VM guest's A disk.
2. Get the Linux kernel and parameter files into the A disk.
3. Use these files to IPL Linux for the Linux install.

#### Format the A disk

Start your 3270 emulation program and log in to your Linux z/VM guest.

Because your A disk is not yet formatted, you may receive some error messages from CMS regarding it when you log in. At the CMS prompt, format your minidisk using the following command:

```
FORMAT 191 A
```

Example 1-3 shows the output from this command.

*Example 1-3 Output from formatting a minidisk*

---

```
DMSFOR603R FORMAT will erase all files on disk B(191). Do you wish to continue? Enter 1 (YES) or 0 (NO).
```

```
1
```

```
DMSFOR605R Enter disk label:
```

```
191LNX1
```

```
DMSFOR733I Formatting disk A
```

```
DMSFOR732I 50 cylinders formatted on A(191)
```

---

Once it is formatted, you can create a PROFILE EXEC (think of it as an AUTOEXEC.BAT for your z/VM guest user). Example 1-4 shows a PROFILE EXEC for a system with CTC links, which are covered in Section 1.4.1, “Network configuration” on page 18.

*Example 1-4 Sample PROFILE EXEC for a z/VM Linux (with CTC connections)*

---

```
/* */
'CP SET PF12 RET'
'CP DEF CTC 800 '
'CP DEF CTC 801 '
'COUPLE 800 LNX1 809'
'COUPLE 801 LNX1 808'
'TERM MORE 2 2'
'TERM HOLD OFF'
'SET RETR MAX'
'IPL 202 CLEAR'
```

---

## Setting up for Linux

To install the first Linux instance, you must go through the following steps:

1. Upload a copy of the Linux kernel, parmfile, and initial RAM disk to the z/VM user minidisk.
2. Send the kernel, parmfile, and RAM disk to the z/VM reader and IPL from it.
3. Build the first Linux instance (using YaST for SuSE or anaconda for RedHat).
4. Reboot using the newly installed Linux system (i.e. IPL from the minidisk instead of the z/VM Reader).

### **Upload Linux IPL material to z/VM**

Locate the Linux for S/390 boot files (on the SuSE 7.0 or RedHat 7.2 for S/390 CD 1) and transfer them to minidisk A (device 191). Example 1-5 shows you how to retrieve the SuSE 7.0 boot files from the SuSE CD attached to another server using FTP.

*Example 1-5 Using FTP to get the Linux files from the SuSE 7.0 CD*

---

```
VMLINK TCPMAINT 592
DMSVML2060I TCPMAINT 592 linked as 0120 file mode Z
Ready; T=0.01/0.01 10:14:22
FTP 9.12.6.53
VM TCP/IP FTP Level 430
Connecting to 9.12.6.53, port 21
220 linux390 FTP server (Version wu-2.4.2-VR17(1) Thu May 18 03:18:13 EDT 2000)
ready.
USER (identify yourself to the host):
anonymous
>>>USER anonymous
331 Anonymous login ok, send your complete email address as your password.
Password:

>>>PASS *****
230 Anonymous access granted, restrictions apply.
Command:
BIN
>>>TYPE i
200 Type set to I.
LOCSITE FIX 80
Command:
CD /SuSE/CD1/suse/images
>>>CWD /SuSE/CD1/suse/images
250 CWD command successful.
```

```

Command:
GET vmrdr.ikr suse.image
>>>PORT 9,12,6,66,4,6
200 PORT command successful.
>>>RETR vmrdr.ikr
150 Opening BINARY mode data connection for vmrdr.ikr (2696704 bytes).
1296560 bytes transferred.
2069440 bytes transferred.
226 Transfer complete.
2696720 bytes transferred in 39.756 seconds. Transfer rate 67.83 Kbytes/sec.
Command:
GET initrd suse.initrd
>>>PORT 9,12,6,66,4,11
200 PORT command successful.
>>>RETR initrd
150 Opening BINARY mode data connection for initrd (9776384 bytes).
50160 bytes transferred.
1384400 bytes transferred.
3235840 bytes transferred.
5141440 bytes transferred.
5653440 bytes transferred.
6222800 bytes transferred.
7192320 bytes transferred.
8756160 bytes transferred.
226 Transfer complete.
9776400 bytes transferred in 133.998 seconds. Transfer rate 72.96 Kbytes/sec.
Command:
ASC
>>>TYPE a
200 Type set to A.
Command:
GET parmfile suse.parm
>>>PORT 9,12,6,66,4,12
200 PORT command successful.
>>>RETR parmfile
150 Opening ASCII mode data connection for parmfile (38 bytes).
226 Transfer complete.
40 bytes transferred in 0.149 seconds. Transfer rate 0.27 Kbytes/sec.
Command:
QUIT

```

---

An explanation of the process:

- ▶ First, link to TCPMAINT's 592 minidisk. This provides you with the FTP command.
- ▶ Use FTP to connect to the server where the files are located.
- ▶ Log on with your user ID and password (or anonymous if it is available).
- ▶ Set binary file transfer and the record length of the transferred files to 80 bytes and the record format to fixed.
- ▶ Change to the directory for CD1, then into the suse/images directory.
- ▶ Get the kernel file vmrdr.ikr to SUSE IMAGE A.
- ▶ Get the initial RAM disk file initrd to SUSE INITRD A.
- ▶ Set ASCII file transfer and get the parmfile (/suse/images/parmfile) to SUSE PARM A.

**Note:** You might need to modify the values in the parameter line file to suit your environment.

## Creating LINIPL EXEC

To easily boot the Linux system using the z/VM Reader, use a REXX program to automate the process a little. (Keep this REXX program, you'll probably use it more than once).

Create file LINIPL EXEC on minidisk A.

```
X LINIPL EXEC
```

Example 1-6 shows a REXX program that we used to load the kernel into the z/VM reader; it also boots the kernel from the reader.

*Example 1-6 REXX code to IPL Linux for the first time*

---

```
/**/  
'CLOSE RDR'  
'PURGE RDR ALL'  
'SPOOL PUNCH * RDR'  
'PUNCH SUSE IMAGE A (NOH'  
'PUNCH SUSE PARM A (NOH'  
'PUNCH SUSE INITRD A (NOH'  
'CHANGE RDR ALL KEEP NOHOLD'  
'IPL 00C CLEAR'
```

---

The commands have the following meanings:

/**/	Informs the system that the file is a REXX executable.
'CLOSE RDR'	Closes all open files in the reader so that they can be purged.
'PURGE RDR ALL'	Empties the z/VM reader. You should ensure that any important reader files have been moved to another location before issuing this command!
'SPOOL PUNCH * RDR'	Directs the output of the punch device to the reader.
'PUNCH SUSE IMAGE A (NOH'	Moves the Linux boot file to the reader.
'PUNCH SUSE PARM A (NOH'	Moves the Linux parameters file to the reader.
'PUNCH SUSE INITRD A (NOH'	Moves the initial RAM disk file (initial root file system) to the reader.
'CHANGE RDR ALL KEEP NOHOLD'	Makes sure the content of the reader is not changed or deleted after the process is finished.
'IPL 00C CLEAR'	Sends the reader an Initial Program Load (IPL) command. This boots Linux for S/390.

## Boot Linux using the z/VM READER

You are now ready to boot the initial Linux system using the command:

```
LINIPL
```

You will see a number of messages fly past on the console from the Linux kernel as it is initializing and discovering the resources assigned to the Linux system.

If the IPL was successful, you should be now able to set up Linux according to the normal SuSE or RedHat setup instructions.

## 1.2.4 Building the latest kernel

We decided that before we created all our systems, we would update the kernel to the latest version available for our SuSE 7.0 systems. This involved downloading the kernel source, applying patches, and compiling the kernel.

Kernel patches for Linux for S/390 and zSeries can be found at:

<http://oss.software.ibm.com/developerworks/opensource/linux390/index.shtml>

From there, we found the patches to build the latest 2.2.16 kernel at:

[http://oss.software.ibm.com/developerworks/opensource/linux390/current2\\_2.shtml](http://oss.software.ibm.com/developerworks/opensource/linux390/current2_2.shtml)

These pages contain only the Linux for S/390 and zSeries patches - they do not have the complete kernel source. The patches must be applied to the original 2.2.16 source downloaded from one of the normal sources, such as [www.kernel.org](http://www.kernel.org). This is the same source code as for any other version of Linux; the differences required for S/390 are included in the patches.

The patches must be applied in the correct sequence to build the source correctly. To create the complete kernel sources, the following patches need to be applied in this sequence:

```
linux-2.2.16.tar.gz (original kernel source)
+ linux-2.2.16.2-s390.diff (IBM)
+ linux-2.2.16-dasd_erp.diff (IBM)
+ linux-2.2.16-dasd.diff (IBM)
+ linux-2.2.16-xchg.diff (IBM)
+ linux-2.2.16-iucv.diff (IBM)
+ linux-2.2.16-pgtable.diff (IBM)
+ linux-2.2.16-sense.diff (IBM)
+ linux-2.2.16-signal.diff (IBM)
+ linux-2.2.16-pagecache.diff (IBM)
+ linux-2.2.16-iucv2.diff (IBM)
+ linux-2.2.16-io.diff (IBM)
+ linux-2.2.16-dasd_performance.diff (IBM)
+ linux-2.2.16-tape.diff (IBM)
+ linux-2.2.16-s390-dump.diff (IBM)
+ linux-2.2.16-fixes.diff (IBM)
+ linux-2.2.16-iucv3.diff (IBM)
+ linux-2.2.16-dasd_fba.diff (IBM)
+ linux-2.2.16-rva_perf.diff (IBM)
+ linux-2.2.16-kernel_symbols.diff (IBM)
+ linux-2.2.16-lcs.diff (IBM)
```

Read the description of each patch carefully—there can be new versions of OCO (Object Code Only) modules that also need to be downloaded to go with a patch. The files from the Web site will most likely be compressed tar files, so they need to be extracted.

Normally the kernel source is installed in the directory `/usr/src`. First extract the original kernel source:

```
cd /usr/src
tar -zxf linux-2.2.16.tar.gz
```

Then extract each patch and apply it:

```
tar -zxf linux-2.2.16.2-s390.tar.gz
patch -p1 -d linux < linux-2.2.16.2-s390.diff
tar -zxf linux-2.2.16-dasd_erp.tar.gz
patch -p1 -d linux < linux-2.2.16-dasd_erp.diff
...
```

The patch may not have the normal directory structure, and needs different `-p` and `-d` options. This was the case with the `linux-2.2.16-iucv3.diff` patch.

```
patch -d linux/drivers/s390/net < linux-2.2.16-iucv3.diff
```

After all the patches are applied, change to the directory containing the source and run:

```
make mrproper
make menuconfig
```

After a few messages, the kernel configuration menu is displayed.

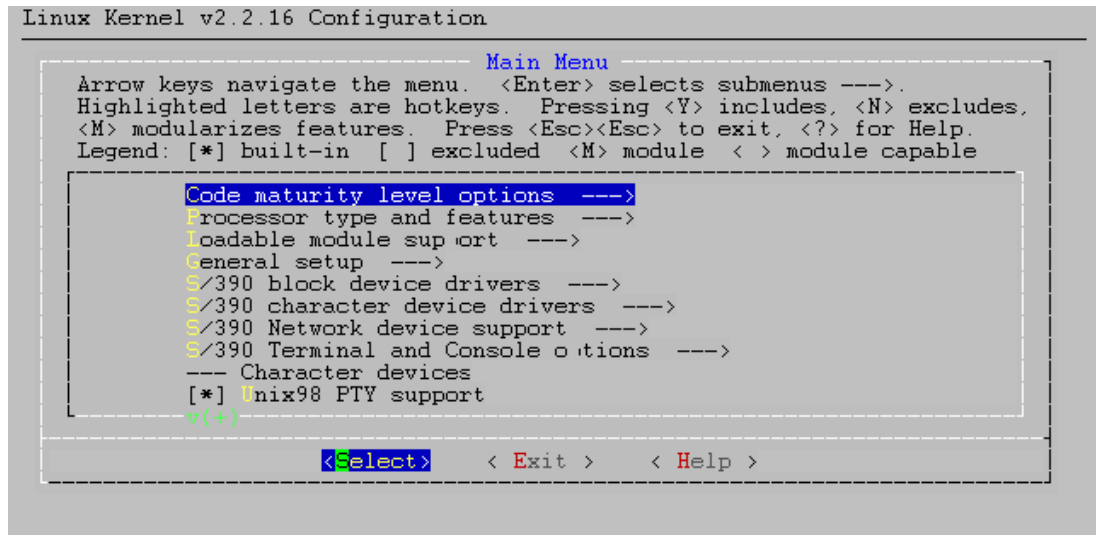


Figure 1-2 Linux kernel configuration menu

Most of the options can be left as they are. The changes we made were:

- ▶ Enable tape support.  
S/390 character device drivers -> S/390 tape device support
- ▶ Enable ISO 9660 filesystem support. While we can't access a CDROM drive, we can mount an ISO9660 image on a loopback device.  
Filesystems -> ISO 9660 CDROM filesystem support
- ▶ Enable quota support.  
Filesystems -> Quota support

Then make dependencies:

```
make dep
```

Compile the kernel itself:

```
make image
```

Compile and install any kernel modules:

```
make modules
make modules-install
```

The newly created Linux kernel will be found in the subdirectory `./arch/s390/boot`.

Back up the existing kernel in `/boot` and copy the new kernel there.

```
cp /boot/image /boot/image.ok
```

```
cp ./arch/s390/boot/image /boot
```

Run `si1o` to make the new kernel bootable. Ensure that you specify the correct boot device.

```
si1o -f image -d /dev/dasdb -t2
```

Reboot the system to bring in the new kernel.

## 1.3 Cloning systems

After we manually installed our first systems, we wanted to quickly create a number of extra systems. We could have run through the install as described in the preceding section and created the new machines, or we could actually make copies of the minidisk the first machine was created on.

z/VM provides the `DDR` command, which can be used to copy minidisks. This means the new machines can be created much more quickly; requiring only the time to copy the minidisks. In addition to making the process simple, it means that we can build a minidisk and define it as a *Gold Image*, then use it to clone from. We will then have a standard image with the same base software installed on each system.

Before cloning the systems, it is worthwhile to spend some time configuring the system to be cloned so it has the generic configuration that you want. We did this by building a new kernel as described in “Building the latest kernel” on page 13, and removing a number of packages that the SuSE install process installed, but that we didn’t actually need.

To clone the systems effectively, the hardware configurations defined in z/VM should be the same. The systems we cloned all had the same DASD definitions, and were connected through CTCs to another Linux system which provided a connection to the outside world. (See systems LNX8, LNX9, LNX10, LNX12, LNX13 and LNX17 in Figure 1-1 on page 4.) We set up the CTC definitions so that each cloned system used the same local device addresses, even though the remote addresses were different.

### *Example 1-7 LNX12 CTC definitions*

---

```
'CP DEFINE CTC 900'  
'CP DEFINE CTC 901'  
'COUPLE 900 LNX11 917'  
'COUPLE 901 LNX11 916'
```

---

### *Example 1-8 LNX13 CTC definitions*

---

```
'CP DEFINE CTC 900'  
'CP DEFINE CTC 901'  
'COUPLE 900 LNX11 919'  
'COUPLE 901 LNX11 918'
```

---

The CTCs at the other end of the link (LNX11) are already defined, so after we clone the system all we need to change is the IP address of our new system and we will be able to connect to it.

**Important:** Ensure that the system you are copying from is shut down. If the system is active, there will probably be changes to the file system occurring while the copy is being made. Since the copy takes time, the new file system is not consistent to a single point in time. The result is inconsistencies in the file system much worse than those resulting from a system crash.

Log on to the z/VM guest for the new system to be created. Link the root disk from the original system to the new system:

```
link link13 202 1202
ENTER READ PASSWORD:
```

Use DDR to copy the disk from the source system to the target:

---

```
DDR
z/VM DASD DUMP/RESTORE PROGRAM
ENTER:
sys cons
ENTER:
in 1202 3390
ENTER:
out 202 3390
ENTER:
copy all
HCPDDR711D VOLID READ IS 0X0202
DO YOU WISH TO CONTINUE? RESPOND YES, NO OR REREAD:
yes
HCPDDR711D VOLID READ IS x8040
DO YOU WISH TO CONTINUE? RESPOND YES, NO OR REREAD:
yes
COPYING 0X0202
COPYING DATA 05/16/02 AT 17.52.14 GMT FROM 0X0202 TO x8040
INPUT CYLINDER EXTENTS OUTPUT CYLINDER EXTENTS
START STOP START STOP
END OF COPY
ENTER:

END OF JOB
```

---

Copying the data takes some time—about 5 to 10 minutes in our case. This obviously is dependent on the size of your disks and the speed of your system. The DDR process copies the whole disk, including the IPL information, so it is not necessary to run `si1o` to enable it to be IPLed.

The swap partition also must be created. There is no useful data in it, so it can be created using the normal procedure, or you can repeat the disk copy process for the swap partition. We copied the swap partition using DDR. (Since the swap partition is much smaller, it only took a few minutes.)

IPL the system. It will come up with all the attributes of the source system, including the IP address and hostname. This means you probably can't get to it via the network, so the initial configuration changes have to be done on the z/VM console using command line tools.

**HiperSockets differences:** This process may need some tailoring for a system using a HiperSockets LAN. We did no testing in this area; however, the initial IPL may cause problems because of two systems on the LAN with the same IP address. One way around this could be to leave the source system down, and reserve the IP address for newly created systems. This may also allow you to connect via the network immediately.

Log in as root on the console. The login prompt and root password are, of course, the same as the source system. Under SuSE the system configuration is defined in the file `/etc/rc.config`. The `SuSEconfig` utility is used to read that file and update the appropriate



system files. Since you cannot use full screen editors on the console, the goal at this point is simply to enable network access to the system.

Search for the IP network of the source system in `/etc/rc.config`.

```
grep 10.1.4 /etc/rc.config
IPADDR_1="10.1.4.13"
IFCONFIG_1="10.1.4.13 pointopoint 10.1.4.11 mtu 1500 up"
```

Use `sed` to change the address. Write the changes to a new file so you can check them before replacing the existing file.

```
sed 's/10.1.4.13/10.1.4.12/' < /etc/rc.config > /tmp/rc.config
```

If the IP address of the other end of the link is changing too, repeat the process using `/tmp/rc.config` as the input file and creating another output file.

Check the changes that were made:

```
diff /etc/rc.config /tmp/rc.config
156c156
< IPADDR_1="10.1.4.13"
---
> IPADDR_1="10.1.4.12"
175c175
< IFCONFIG_1="10.1.4.13 pointopoint 10.1.4.11 mtu 1500 up"
---
> IFCONFIG_1="10.1.4.12 pointopoint 10.1.4.11 mtu 1500 up"
```

You can see that the changes you want have been made and you haven't created any other errors. Copy the file into `/etc` and run **SuSEconfig**:

```
cp /tmp/rc.config /etc
SuSEconfig
```

Reboot the system so it uses the new IP address.

After the reboot it may be necessary to set the default gateway for the system before you can connect to it. Log on to the console and check:

```
route -n
```

The default gateway is the route for destination 0.0.0.0.

```
0.0.0.0          10.1.4.11      0.0.0.0        UG  0        0        0 ctc1
```

If the default gateway is incorrect, fix it:

```
route del default
route add default gw 10.1.4.11
```

You should now be able to log on to the system over the network. Edit `/etc/rc.config` and make any other changes that are required, e.g. the hostname. Check `/etc/hosts` and `/etc/route.conf` to see if changes are necessary.

Run **SuSEconfig** again to install the changes to `rc.config`, and boot the system once more.

## 1.4 Resource management

### 1.4.1 Network configuration

Within our simulated environment, we are using four types of network connectivity:

- ▶ Direct connection to an Open Systems Adapter (OSA) physically connects us to the real network. It is used on LNX1 (and would be used on LNX11).
- ▶ VCTC (Virtual Channel-to-Channel communications) provides virtual point-to-point links between virtual systems. VCTCs are provided by z/VM.
- ▶ VHSI (Virtual HiperSockets) provides a virtual LAN that systems can connect with using their “virtual” network adapter. VSHI is provided by z/VM.
- ▶ HSI (HiperSockets) also provides a virtual LAN, except that HiperSockets is provided by microcode in the z900 system, not by z/VM.

We could have also used:

- ▶ Inter-User Communications Vehicle (IUCV) connections
- ▶ Real CTCs (defined outside of z/VM)

The version of the Linux kernel used determines how all the drivers are configured.

- ▶ For 2.2 kernels, the device details are supplied as parameters on the command line when loading the modules, or defined in `/etc/modules.conf`. These details are required so that the module can find the networking device.
- ▶ For 2.4 kernels, the device details are stored in `/etc/chandev.conf` and are used when the modules are loaded.

For more information on networking under z/VM, refer to *Linux on IBM eServer zSeries and S/390: ISP/ASP Solutions*, SG 24-6299.

#### OSA connections

An OSA connection provides the ability to connect us to a real network, as would be provided by plugging a network cable into a network card, with the other end of the cable going into a switch or hub (or router). An OSA card has a limitation in that it can only support four IP addresses, and thus would only enable us to connect four systems (real or virtual) to the real network with a single card.

Four systems may be adequate to provide some sort of clustering, load balancing, failover configurations, or just sharing the adapter with four systems; however, it would not be a viable solution if you wanted to connect many virtual systems (hundreds or even thousands of them) to a real network. HiperSockets or Guest LAN would be a better and more cost-effective solution. (See “z/VM Guest LAN (Emulated HiperSockets)” on page 25.)

In our configuration, we are using the OSA card to provide us with physical connectivity to the Internet (via LNX1).

#### ***z/VM configuration***

An OSA configuration is defined outside of z/VM, so you will need to talk to your system programmer to get details of which OSA cards are available to you (if any).

Once an OSA card has been defined and allocated to you, you should be able to see the OSA device assigned for your use by running the following command (where 2320 is the OSA device number):

```
CP Q 2320
OSA 2320 ON OSA 2320 SUBCHANNEL = 0000
2320 QDIO-ELIGIBLE
```

### **Linux configuration**

In our setup, we are using an OSA-Express adapter in QDIO mode and thus need to use the `qeth.o` and `qdio.o` Linux drivers. Linux treats this adapter as a normal Ethernet (`ethN`) adapter. All the drivers should be supplied with your Linux/390 distribution.

**Note:** If you are using the OSA-Express adapter in non-QDIO mode, or you are using an OSA2 adapter, use the `1cs.o` module. If you have a Token Ring OSA card, you may need other drivers as well.

On LNX1, the OSA adapter will be our first ethernet adapter, so it will have the device name of `eth0`. Regardless of what version of kernel we use, it is best to update `/etc/modules.conf` so that the kernel will understand that our `eth0` device is made available via the `qeth.o` module:

```
alias eth0 qeth
```

The next step depends on which version of the kernel is used:

► 2.2.x kernel

Add the following additional line to `/etc/modules.conf`:

```
options qeth qeth_options=auto,0x2320,0x2321,0x2322,primary_router,portname:OSA2320
```

► 2.4.x kernel

Add the device details to the `/etc/chandev.conf` configuration file:

```
add_parms,0x10,0x2320,0x2322,primary_router,portname:OSA2320
qeth0,0x2320,0x2321,0x2322
```

Inform the kernel of the `chandev` configuration, by:

```
echo 'reset_conf;read_conf' > /proc/chandev.conf
```

(Our OSA device had devices `0x2320`, `0x2321`, `0x2322` and portname `OSA2320`; your OSA details may vary.)

Finally, issue the following command:

```
depmod -a
```

Then when you assign an IP address to your Ethernet interface, the module will be loaded automatically and `ifconfig` should show that you have an Ethernet interface:

---

```
#ifconfig eth0 9.12.6.81 netmask 255.255.254.0 network 9.12.6.0 broadcast 9.12.7.255
#ls mod
Module                Size  Used by
qeth                   135488  1 (autoclean)
qdio                   39504   1 (autoclean) [qeth]
#ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:06:29:6C:CB:BA
          inet addr:9.12.6.81  Mask:255.255.254.0
          inet6 addr: fe80::206:29ff:fe6c:cbba/10 Scope:Link
          UP RUNNING NOARP MULTICAST  MTU:1492  Metric:1
          RX packets:1425 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1223 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:54040 (52.7 Kb)  TX bytes:52766 (51.5 Kb)
```

---

## Virtual Channel-to-Channel connections

Virtual Channel-to-Channel (VCTC) connections are provided by z/VM and have the same characteristics as normal Channel-to-Channel (CTC) connections. The only limitation with VCTCs is that they are only available to join between two guests of the same z/VM host.

VCTCs provide point-to-point links between 2 systems, which could be between:

- ▶ Two virtual Linux systems
- ▶ A Linux system and the s/390 TCPIP stack
- ▶ A Linux system and another mainframe OS (for example MVS)

CTCs look and act like point-to-point links, somewhat like starting a SLIP or PPP link between two systems. There are advantages and disadvantages to using CTCs and VCTCs; they are as follows:

### **Advantages**

- ▶ They limit and control the traffic between two systems, which is ideal for hosts in a DMZ. In our environment, any traffic between LNX5 and LNX6 must go through LNX1 (which is our firewall). There is no way that LNX5 can communicate with LNX6 directly, unless we define a CTC connecting those two hosts together.
- ▶ You may require CTC connectivity to become visible on the network if you choose not to share or assign an OSA card to a z/VM guest, for example, if you are using an s/390 TCPIP stack that gave you access outside of the s/390 system.
- ▶ They can be used to join two virtual networks together that are on two physical 390 systems. This applies only to CTCs.

### **Disadvantages**

- ▶ There could be instances where if one end of the CTCs is brought down (for example, that system is rebooted), the other end may require rebooting as well before the link can be re-established. (We found a couple of situations when we booted LNX5 or LNX6 that we also had to boot LNX1 because the CTC would not come back up until we did so.)
- ▶ CTCs are defined on each host that has an end in the CTC link. Before a host recognizes a new CTC that has been defined to it, the `ctc.o` module needs to be unload and reloaded (which means if that is your only method of connecting to that virtual machine, you will lose network connectivity when you unload the `ctc` module).
- ▶ The unloading and reloading of the `ctc` module in point 2 can affect the other end of other CTC links, and thus the other ends may require rebooting as well. (This can be very cumbersome where one or both of the systems has many CTC links to other systems).
- ▶ To enable two machines to talk directly to each other, without routing through a third, you need to define a CTC link to each machine, reboot and configure, which can be time consuming.

**Tip:** Plan your CTC connections in advance! It may be difficult to add each end of a CTC link to an existing running host, since it will most likely require a re-IPL for it to be successful. If you will be cloning systems, see “Tips for a penguin farm” on page 29.

In our environment, we used VCTCs that were configured as follows:

### **z/VM configuration**

For each host that will have an end of a CTC link, include in the PROFILE EXEC the definitions to create the VCTC before Linux is IPLed:

```
CP DEF CTC xxx
```

```
CP DEF CTC xxx+1
COUPLE xxx nnn yyy+1
COUPLE xxx+1 nnn yyy
```

#### Where

- ▶ nnn is the name of the other end of the link that this CTC joins to.
- ▶ xxx is the CTC device address for this host.
- ▶ yyy is the CTC device address for the host at the other end of the link.

**Tip:** In fact, you only need the COUPLE statement in one of the host's PROFILE EXECs since the couple will occur when that system is logged in. However, by putting in the COUPLE in the PROFILE EXEC of each system, then regardless of which system is booted last, the couple will always occur.

Also, be sure that you cross the device numbers when you couple. That is, couple the first device on one machine to the last device on the other and visa versa.

In our configuration, for all hosts that had only 1 CTC link (all hosts except LNX1 and LNX11), we defined their end of the CTC link as device 0x800 and 0x801.

Our LNX1 PROFILE EXEC is shown in Example 1-9.

#### *Example 1-9 LNX1 PROFILE EXEC*

---

```
'CP SET PF12 RET'
'CP DEF CTC 900'
'CP DEF CTC 901'
'COUPLE 900 LNX1 815'
'COUPLE 901 LNX1 814'
'CP DEF NIC 504 HIPER DEVICES 3'
'COUPLE 504 TO SYSTEM PUBLAN2'
'TERM MORE 2 2'
'TERM HOLD OFF'
'SET RETR MAX'
'CP DEF CTC 902'
'CP DEF CTC 903'
'CP DEF CTC 904'
'CP DEF CTC 905'
'CP DEF CTC 906'
'CP DEF CTC 907'
'CP DEF CTC 908'
'CP DEF CTC 909'
'CP DEF CTC 90A'
'CP DEF CTC 90B'
'CP DEF CTC 90C'
'CP DEF CTC 90D'
'CP DEF CTC 90E'
'CP DEF CTC 90F'
'CP DEF CTC 910'
'CP DEF CTC 911'
'CP DEF CTC 912'
'CP DEF CTC 913'
'CP DEF CTC 914'
'CP DEF CTC 915'
'CP DEF CTC 916'
'CP DEF CTC 917'
'CP DEF CTC 918'
'CP DEF CTC 919'
'CP DEF CTC 91A'
```

```

'CP DEF CTC 91B'
'CP DEF CTC 91C'
'CP DEF CTC 91D'
'CP DEF CTC 91E'
'CP DEF CTC 91F'
'CP DEF CTC 920'
'CP DEF CTC 921'
'CP DEF CTC 922'
'CP DEF CTC 923'
'CP DEF CTC 924'
'CP DEF CTC 925'
'COUPLE 902 LNX2 901'
'COUPLE 903 LNX2 900'
'COUPLE 904 LNX3 901'
'COUPLE 905 LNX3 900'
'COUPLE 906 LNX4 901'
'COUPLE 907 LNX4 900'
'COUPLE 908 LNX5 901'
'COUPLE 909 LNX5 900'
'COUPLE 90A LNX6 901'
'COUPLE 90B LNX6 900'
'COUPLE 90C LNX7 901'
'COUPLE 90D LNX7 900'
'COUPLE 90E LNX8 901'
'COUPLE 90F LNX8 900'
'COUPLE 910 LNX9 901'
'COUPLE 911 LNX9 900'
'COUPLE 912 LNX10 90'
'COUPLE 913 LNX10 90'
'COUPLE 914 LNX11 90'
'COUPLE 915 LNX11 90'
'COUPLE 916 LNX12 90'
'COUPLE 917 LNX12 90'
'COUPLE 918 LNX13 90'
'COUPLE 919 LNX13 90'
'COUPLE 91A LNX14 90'
'COUPLE 91B LNX14 90'
'COUPLE 91C LNX15 90'
'COUPLE 91D LNX15 90'
'COUPLE 91E LNX16 90'
'COUPLE 91F LNX16 90'
'COUPLE 91E LNX16 901'
'COUPLE 91F LNX16 900'
'COUPLE 920 LNX17 901'
'COUPLE 921 LNX17 900'
'COUPLE 922 LNX18 901'
'COUPLE 923 LNX18 900'
'COUPLE 924 LNX19 901'
'COUPLE 925 LNX19 900'
'IPL 202 CLEAR'
* * * End of File * * *

```

---

Example 1-10 is our PROFILE EXEC for LNX9, one of the hosts at the other end of LNX11's CTC links.

*Example 1-10 LNX9 PROFILE EXEC*

---

```

/* */
'CP SET PF12 RET'
'CP DEF CTC 900'

```

```
'CP DEF CTC 901'  
'COUPLE 900 LNX11 911'  
'COUPLE 901 LNX11 910'  
'TERM MORE 2 2'  
'TERM HOLD OFF'  
'SET RETR MAX'
```

---

### **Linux configuration**

In Linux, VCTCs (and CTCs) are configured with the `ctc.o` driver, providing a `ctcN` device to configure with the IP address.

To assist with managing the many CTC links on LNX1 and LNX11, we decided that the `ctcN` device names (on LNX1 and LNX11) should reflect the hosts' names. For example, the LNX5 device is known as `ctc5` on LNX1 even though it is `ctc0` on LNX5.

So that the kernel knows that a `ctcN` device is provided by the `ctc.o` driver, add the following to your `/etc/modules.conf`:

---

```
alias ctc0 ctc  
alias ctc1 ctc  
alias ctc2 ctc  
...  
alias ctc19 ctc
```

---

The kernel-dependent portion of the configuration is as follows:

► **2.2.x kernel**

No other configuration is required. When you load the module, it automatically probes and discovers all defined VCTCs.

► **2.4.x kernel**

Add the following information to your `/etc/chandev.conf` configuration file:

```
ctc0,0x800,0x801
```

After changing the `chandev.conf` file, run the following command to inform the kernel:

```
echo 'reset_conf;read_conf' > /proc/chandev.conf
```

Finally, perform a `depmod -a`, then when you assign an IP address to your `ctc` interface, the module will be loaded automatically. Example 1-11 shows what this looks like on LNX1.

*Example 1-11 LNX1 CTC definition*

---

```
#ifconfig ctc0 10.1.5.11 pointopoint 10.1.5.8 mtu 1500
```

**#lsmod**

Module	Size	Used by
ctc	52588	9 (autoclean)

**#ifconfig ctc9**

```
ctc8 Link encap:Serial Line IP  
inet addr:10.1.5.11 P-t-P:10.1.5.9 Mask:255.255.255.255  
UP POINTOPOINT RUNNING NOARP MTU:1500 Metric:1  
RX packets:64402 errors:0 dropped:1 overruns:0 frame:1  
TX packets:6878 errors:230 dropped:230 overruns:0 carrier:230  
collisions:0 txqueuelen:100  
RX bytes:4104945 (3.9 Mb) TX bytes:1651094 (1.5 Mb)
```

---

Example 1-12 shows what it looks like on LNX8 (one of the other ends).

*Example 1-12 LNX8 CTC definition*

---

```
#ifconfig ctc0 10.1.5.9 pointopoint 10.1.5.11 mtu 1500

#lsmod ctc0
Module                Size Used by
ctc                   21640  1 (autoclean)

#ifconfig ctc0
ctc0  Link encap:Serial Line IP
      inet addr:10.1.5.9 P-t-P:10.1.5.11  Mask:255.255.255.255
      UP POINTOPOINT RUNNING NOARP  MTU:1500  Metric:1
      RX packets:242 errors:1 dropped:0 overruns:0 frame:0
      TX packets:58008 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:100
```

---

**Some debugging**

For kernel 2.2, if you want to check that ctc found all your available ctc devices, you can look at your `/var/log/messages` for some information. It should look similar to the following:

---

```
Apr 26 14:20:28 lnx9 kernel: CTC driver Version: 1.25.2.5  initialized
Apr 26 14:20:28 lnx9 kernel: channel: 4 Parallel channel found - 0 ESCON channel found
Apr 26 14:20:28 lnx9 kernel: ctc0: read dev: 0800 irq: 000e - write dev: 0801 irq: 000f
Apr 26 14:20:28 lnx9 kernel: ctc1: read dev: 0900 irq: 0010 - write dev: 0901 irq: 0011
Apr 26 14:20:28 lnx9 kernel: ctc0: connected with remote side
Apr 26 14:20:28 lnx9 kernel: ctc1: connected with remote sid
```

---

For kernel 2.4, you can also look in your `/var/log/messages`:

---

```
Apr 24 08:21:04 lnx11 kernel: CTC driver Version: 1.51  initialized
Apr 24 08:21:04 lnx11 kernel: ctc17: read: ch 0920 (irq 0035), write: ch 0921 (irq 0036) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc13: read: ch 0918 (irq 002d), write: ch 0919 (irq 002e) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc12: read: ch 0916 (irq 002b), write: ch 0917 (irq 002c) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc10: read: ch 0912 (irq 0027), write: ch 0913 (irq 0028) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc9: read: ch 0910 (irq 0025), write: ch 0911 (irq 0026) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc8: read: ch 090e (irq 0023), write: ch 090f (irq 0024) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc4: read: ch 0906 (irq 001b), write: ch 0907 (irq 001c) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc2: read: ch 0902 (irq 0017), write: ch 0903 (irq 0018) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc0: read: ch 0800 (irq 000d), write: ch 0801 (irq 000e) proto: 0
Apr 24 08:21:04 lnx11 kernel: ctc0: connected with remote side
```

---

You can look at `/proc/chandev` as well:

---

channels detected										
	chan	cu	cu	dev	dev				in	chandev
	irq	devno	type	type	model	type	model	pim	chpids	use reg.
0x000d	0x0800	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x000e	0x0801	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0015	0x0900	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0016	0x0901	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0017	0x0902	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0018	0x0903	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0019	0x0904	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x001a	0x0905	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x001b	0x0906	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes



0x001c	0x0907	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x001d	0x0908	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x001e	0x0909	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x001f	0x090a	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0020	0x090b	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0021	0x090c	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0022	0x090d	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0023	0x090e	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0024	0x090f	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0025	0x0910	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0026	0x0911	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0027	0x0912	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0028	0x0913	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0029	0x0914	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x002a	0x0915	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x002b	0x0916	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x002c	0x0917	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x002d	0x0918	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x002e	0x0919	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x002f	0x091a	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0030	0x091b	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0031	0x091c	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0032	0x091d	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0033	0x091e	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0034	0x091f	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0035	0x0920	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0036	0x0921	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	yes	yes
0x0037	0x0922	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0038	0x0923	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x0039	0x0924	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no
0x003a	0x0925	0x05	0x3088	0x08	0x0000	0x00	0x80	0x0400000000000000	no	no

## z/VM Guest LAN (Emulated HiperSockets)

Emulated HiperSockets became available in z/VM 4.2. They enable s/390 operating systems (z/VM and z/VM guests) to communicate with each other as if they were connected to a real LAN network.

The main difference between HiperSockets and Guest LAN is Guest LAN is an “emulated” version of HiperSockets provided by z/VM. It doesn’t have the resource limitations; however, it is only available to z/VM guests on the same z/VM system.

**Restriction:** HiperSockets are not available to Linux systems for kernels before 2.4.7.

### z/VM configuration

In our environment, LNX18 is connected to a Guest LAN named PUBLAN2. To create this virtual LAN, as the system user, issue the following command:

```
CP define lan PUBLAN2 ownerid SYSTEM
```

**Tip:** Although you are able to do the `define lan PUBLAN2` as the logged in user (that is, as LNX18), it would be better to have the LAN defined and owned by the system account (and even started automatically at system IPL). Otherwise, if this user is logged off, the virtual LAN evaporates, leaving any other virtual systems connected to the LAN isolated!

For each host that is to be a part of the virtual LAN PUBLAN2, put into the PROFILE EXEC the following details:

```
CP define nic 500 hiper devices 3
CP couple 500 to system PUBLAN2
```

To verify that it all worked, run the query commands shown in Example 1-13. You should get some information about your new virtual LAN.

*Example 1-13*

---

```
q lan lanname det
LAN LNX16 LANNAME      Type: HIPERS   Active: 1     MAXCONN: INFINITE
  TRANSIENT UNRESTRICTED MFS: 16384  ACCOUNTING: OFF
  Adapter Owner: USER  NIC: 0500   Name: UNASSIGNED
Ready; T=0.01/0.01 14:52:10

q nic 500 det
Adapter 0500 Type: HIPERS   Name: UNASSIGNED  Devices: 3
Port 0 MAC: 00-04-AC-00-00-0A LAN: USER  LANNAME      MFS: 16384
RX Packets: 0          Discarded: 0      Errors: 0
TX Packets: 0          Discarded: 0      Errors: 0
RX Bytes: 0           TX Bytes: 0
Unassigned Devices:
  Device: 0500 Unit: 000  Role: Unassigned
  Device: 0501 Unit: 001  Role: Unassigned
  Device: 0502 Unit: 002  Role: Unassigned
Ready; T=0.01/0.01 14:53:19
```

---

### **Linux configuration**

HiperSockets are made available in Linux via new versions of the `qdio.o` and `qeth.o` modules provided by IBM. You'll find them at the following URL:

<http://oss.software.ibm.com/developerworks/opensource/linux390/index.shtml>

After you download and install the modules, make the following Linux configuration changes to see the new HiperSockets Guest LAN.

Update `/etc/modules.conf` so that the kernel will understand that your `hsi0` device is made available via the `qeth.o` module:

```
alias hsi0 qeth
```

Add the following information to your `/etc/chandev.conf` configuration file:

```
qeth0,0x500,0x501,0x502,0,0,0
```

After changing the `chandev.conf` file, run the following command to inform the kernel:

```
echo 'reset_conf;read_conf' > /proc/chandev.conf
```

Finally, perform a `depmod -a`. Then, when you assign an IP address to your `hsi` interface, the module will be loaded automatically.

---

```
#ifconfig hsi0 10.1.2.18 netmask 255.255.255.0 broadcast 10.1.2.255

#ls mod
Module          Size  Used by
qeth            135864  1
qdio            38548  1 [qeth]

#ifconfig hsi0
```

```

hsi0      Link encap:Ethernet  HWaddr 00:00:00:00:00:00
          inet addr:10.1.2.18  Mask:255.255.255.0
          UP RUNNING NOARP MULTICAST  MTU:8192  Metric:1
RX packets:140197 errors:0 dropped:0 overruns:0 frame:0
          TX packets:282729 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:7379081 (7.0 Mb)  TX bytes:405583452 (386.7 Mb)
          Interrupt:17

```

---

### **Some debugging**

For kernel 2.4, you can look in your /var/log/messages:

---

```

Apr 25 13:52:44 lnx18 kernel: qdio: loading QDIO base support version 2 ($Revision: 1.78.2.4
$/Revision: 1.44.2.1 $)
Apr 25 13:52:44 lnx18 kernel: debug: reserved 1 areas of 4 pages for debugging qdio_setup
Apr 25 13:52:44 lnx18 kernel: debug: qdio_setup: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 2 areas of 4 pages for debugging qdio_labs
Apr 25 13:52:44 lnx18 kernel: debug: qdio_labs: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 1 areas of 2 pages for debugging qdio_sense
Apr 25 13:52:44 lnx18 kernel: debug: qdio_sense: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 2 areas of 4 pages for debugging qdio_trace
Apr 25 13:52:44 lnx18 kernel: debug: qdio_trace: new level 2
Apr 25 13:52:44 lnx18 kernel: qeth: loading qeth S/390 OSA-Express driver ($Revision:1.136.2.3
$/Revision: 1.53.2.2 $/$Revision: 1.18 $)
Apr 25 13:52:44 lnx18 kernel: qeth: allocated 0 spare buffers
Apr 25 13:52:44 lnx18 kernel: debug: reserved 1 areas of 8 pages for debugging qeth_setup
Apr 25 13:52:44 lnx18 kernel: debug: qeth_setup: new level 3
Apr 25 13:52:44 lnx18 kernel: debug: reserved 1 areas of 2 pages for debugging qeth_misc
Apr 25 13:52:44 lnx18 kernel: debug: qeth_misc: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 1 areas of 8 pages for debugging qeth_data
Apr 25 13:52:44 lnx18 kernel: debug: qeth_data: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 2 areas of 8 pages for debugging qeth_control
Apr 25 13:52:44 lnx18 kernel: debug: qeth_control: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 1 areas of 2 pages for debugging qeth_sense
Apr 25 13:52:44 lnx18 kernel: debug: qeth_sense: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 2 areas of 2 pages for debugging qeth_qerr
Apr 25 13:52:44 lnx18 kernel: debug: qeth_qerr: new level 2
Apr 25 13:52:44 lnx18 kernel: debug: reserved 2 areas of 4 pages for debugging qeth_trace
Apr 25 13:52:44 lnx18 kernel: debug: qeth_trace: new level 2
Apr 25 13:52:44 lnx18 kernel: qeth: Trying to use card with devnos 0x500/0x501/0x502
Apr 25 13:52:44 lnx18 kernel: qdio : Adapter interruption facility not installed.
Apr 25 13:52:44 lnx18 kernel: qdio : Not all CHSCs supported. Continuing.
Apr 25 13:52:44 lnx18 kernel: qeth: Device 0x500/0x501/0x502 is a HiperSockets card
Apr 25 13:52:44 lnx18 kernel: with link type magic (portname:      )
Apr 25 13:52:44 lnx18 kernel: qeth: IPv6 not supported on hsi0

```

---

Also, looking at /proc/chandev, you should see the devices:

---

```

channels detected
          chan  cu  cu  dev  dev          in chandev
          irq devno type  type model type  model pim    chpids    use reg.
          =====
0x000f 0x0500 0x10 0x1731 0x05 0x1732 0x05 0x80 0x0500000000000000 yes yes
0x0010 0x0501 0x10 0x1731 0x05 0x1732 0x05 0x80 0x0500000000000000 yes yes
0x0011 0x0502 0x10 0x1731 0x05 0x1732 0x05 0x80 0x0500000000000000 yes yes

```

---

And, from z/VM:

---

```
#cp q nic det
Adapter 0500 Type: HIPERS Name: UNASSIGNED Devices: 3
Port 0 MAC: 00-04-AC-00-00-03 LAN: SYSTEM PUBLAN2 MFS: 16384
RX Packets: 94979 Discarded: 0 Errors: 0
TX Packets: 168032 Discarded: 3 Errors: 0
RX Bytes: 7117654 TX Bytes: 236964082
Connection Name: HALLOLE State: Session Established
Device: 0500 Unit: 000 Role: CTL-READ
Device: 0501 Unit: 001 Role: CTL-WRITE
Device: 0502 Unit: 002 Role: DATA
Unicast IP Addresses:
 10.1.2.18
Multicast IP Addresses:
 224.0.0.1 01-00-5E-00-00-01
 224.0.0.9 01-00-5E-00-00-09
```

---

Then when you have a few systems connected to the LAN, it will look something like this:

---

```
#cp q lan det
LAN SYSTEM PUBLAN2 Type: HIPERS Active: 5 MAXCONN: INFINITE
PERSISTENT UNRESTRICTED MFS: 16384 ACCOUNTING: OFF
Adapter Owner: LNX11 NIC: 0504 Name: UNASSIGNED
 10.1.2.11 224.0.0.1 224.0.0.9
Adapter Owner: LNX14 NIC: 0500 Name: UNASSIGNED
 10.1.2.14 224.0.0.1 224.0.0.9
Adapter Owner: LNX18 NIC: 0500 Name: UNASSIGNED
 10.1.2.18 224.0.0.1 224.0.0.9
Adapter Owner: LNX19 NIC: 0500 Name: UNASSIGNED
 10.1.2.19 224.0.0.1 224.0.0.9
Adapter Owner: TC1 NIC: 0500 Name: IUTIQDDO
 9.12.3.66 10.1.2.20 224.0.0.1
```

---

## HiperSockets

HiperSockets for Linux is very similar to an OSA connection. In fact, Linux doesn't even know the difference between the real OSA device and the HiperSockets device (which looks like an OSA card even to z/VM).

### *z/VM configuration*

The creation of HiperSocket LANs is provided by microcode in your z800 or z900 system, and thus is dependent on the type of hardware you have. If you are not using a z800 or z900, we recommend that you use the Guest LAN feature described previously.

The z/VM configuration for HiperSockets is beyond the scope of this document since it is hardware-related and requires some configuration in your HCD. Have a look at the zSeries HiperSockets redbook for details on how to enable your hardware and configure z/VM for HiperSockets.

### *Linux configuration*

The Linux configuration for HiperSockets is exactly like the z/VM configuration described previously, so you can follow the configuration steps in "z/VM Guest LAN (Emulated HiperSockets)" on page 25. The only changes you'll need to make are the device addresses.

**Important:** We discovered some serious Guest LAN performance problems on LNX11 when we defined a real HiperSockets interface (hsi0) and a virtual Guest LAN interface (hsi1). We didn't get to research the cause of the problem, but noted that when we removed the definitions of the real HiperSockets from the LNX11 user (out of z/VM), the Guest LAN operated normally again.

(While we had the real HiperSockets defined, it also operated normally at the expense of the Guest LAN. Since in a normal environment, you wouldn't run both types of connectivity from one virtual box, we didn't spend a lot of time exploring the problem.)

## Tips for a penguin farm<sup>1</sup>

- ▶ For systems that you want to clone, cloning will be simpler and require fewer local changes if you make the configurations as generic as possible.

For example, if you use CTC, Guest LAN, or HiperSocket connections, then make the device number of the CTC or Guest LAN connection the same for each cloned machine. This will save you from having to update your `/etc/modules.conf` or `/etc/chandev.conf`.

(For CTC links, the other end of the CTC link will have a unique device number, which is unavoidable. Thus your `PROFILE EXEC` will be different for each machine that is cloned; however, you won't need to make any further Linux device configurations.)

- ▶ If you want the flexibility to add virtual machines and have them be part of a virtual network, then if the virtual machine can be Linux 2.4 kernel, use HiperSockets or Guest LANs for the network.

If the HiperSocket or Guest LAN is created with the system account, then you can have nodes attached and removed from the virtual LAN without affecting any other nodes connected to the LAN. You don't have any dependency on any other node, either.

If CTC links are your only option, create more CTC devices than you need, in advance, so that you can add systems as required. Remember, CTC links have 2 ends, and you need to reload the `ctc.o` module in order to see the new CTC devices. (Unloading and reloading the module involves stopping all CTC links on that system, which may not be desirable.)

Also bear in mind, CTC links are not as flexible as HiperSockets Links. If the link won't start, you may need to IPL the other end of the link.

- ▶ If you create a complex virtual network inside your mainframes (like we have), then you may want to consider running a routing daemon to help with your network discovery. We would recommend that you only consider a routing daemon for those nodes that are actually acting as a router, and run the routing daemon on those nodes only. (In our environment, we are running `zebra`<sup>2</sup> on LNX1 and LNX11 only).

Routing daemons take up processor cycles, which in a large penguin farm could add up to a significant number. Running routing daemons only on those systems that really require them and using static default routes on other systems will help give you back those processing cycles.

## 1.4.2 Memory configuration

Defining memory to a virtual server is done by CMS before you IPL the Linux systems. It is best to have it placed in your `USER DIRECT` file so that when the Linux user logs on it is already configured.

<sup>1</sup> As referred to in *Linux on IBM eServer zSeries and S/390: ISP/ASP Solutions*

<sup>2</sup> **Zebra** is one of the many routing daemons available for Linux. We used it because it was considerably easier to set up than some of the other ones.

To set the storage manually before IPL time, issue the following command:

```
CP DEFINE STORAGE 256M
STORAGE = 256M
Storage cleared - system reset.
```

To check what storage a z/VM guest has, issue this command to CP:

```
CP Q STORAGE
STORAGE = 256M
```

**Attention:** Don't issue the command `CP DEFINE STORAGE 256M` while a z/VM guest is running: CP will reset the running virtual machine! There isn't any way of increasing memory available to a virtual machine.

### 1.4.3 CPU configuration

The number of CPUs available to a virtual machine requires two definitions:

- ▶ In the `USER DIRECT` include the statement:

```
SET MACHINE XA ##
```

Where `##` is the number of CPUs you would like to allocate to the virtual machine.

- ▶ In your `PROFILE EXEC` include the statement:

```
CP DEF CPU 1 2 3 .. n
```

(You define each CPU number, separated by a space, that you want z/VM to create.)

Then, when your system boots (assuming you have a multi-processor kernel), you can query `/proc/cpuinfo` to confirm that Linux sees all available CPUs:

---

```
[root@lnx3 root]# cat /proc/cpuinfo
vendor_id       : IBM/S390
# processors    : 3
bogomips per cpu: 784.79
processor 0: version = FF,  identification = 0C0ECB,  machine = 2064
processor 1: version = FF,  identification = 0C0ECB,  machine = 2064
processor 2: version = FF,  identification = 0C0ECB,  machine = 2064
```

---

While you could enter the command `#CP DEF CPU 4` on the console to add an additional CPU to a running machine (CPU number 4 in this example), there is no way that Linux will recognize the extra CPU until it has been rebooted.

## 1.5 Monitoring

When you are running many guest operating systems under z/VM (whether they are Linux or not), it is useful to get an understanding of the performance of the environment from z/VM's point of view.

In this section, we describe very briefly three ways to get information about the system performance of z/VM.

### 1.5.1 The `indicate` command

The CP command `indicate` allows you to get some information about the status of the z/VM guest system as well as the status of the system resources of z/VM.

If you are a class G user, you can use `indicate` to display:

- ▶ Recent contention for system resources. This can be helpful to predict system throughput and response time characteristics that your virtual machine may experience now and in the near future.
- ▶ Environment characteristics of your virtual machine. This includes machine type, the origin of the system IPLed (loaded) in your virtual machine, and the presence or quantity of system resources available to your virtual machine.
- ▶ Measurements of resources used by your virtual machine. Such measurements are *accumulators*, which means they are always increasing after the logon of your virtual machine.

If you are a class E user, the `indicate` command provides all class G functions and the following:

- ▶ Detailed information on use of, and contention for, system resources. User IDs of virtual machines currently using certain resources can be displayed.
- ▶ The status of current active virtual machines as determined by the system scheduler and dispatcher.
- ▶ Environment characteristics of, and measurements of resources used by, any virtual machine logged on.

There are some important variations of the `indicate` command. For more details see the appropriate chapter in the *z/VM V4Rx.x CP Command and Utility Reference* manual.

Example 1-14 shows the use of the `indicate user` command.

*Example 1-14 indicate user lnx18 command issued from user maint*

---

```
ind user lnx18
USERID=LNX18   MACH=XA   STOR=512M VIRT=V XSTORE=NONE
IPLSYS=DEV 0202 DEVNUM=00020
PAGES: RES=00011217 WS=00011023 LOCK=00000019 RESVD=00000000
NPREF=00122103 PREF=00000000 READS=00744057 WRITES=00620062
XSTORE=001399 READS=113730 WRITES=558770 MIGRATES=442429
CPU 00: CTIME=74:40 VTIME=124:29 TTIME=156:43 IO=996390
          RDR=000000 PRT=000000 PCH=000000
Ready; T=0.01/0.01 15:52:26
```

---

This response gives all data from the user's VMDBK relevant to the user's virtual machine paging activity, resource occupancy, processor usage, and accumulated I/O activity counts since logon. Time and count values are ever-increasing accumulators.

Example 1-15 shows the use of the **indicate load** command.

*Example 1-15 indicate load command issued from user maint*

---

```
ind load
AVGPROC-006% 02
XSTORE-000023/SEC MIGRATE-0008/SEC
MDC READS-000004/SEC WRITES-000001/SEC HIT RATIO-048%
STORAGE-101% PAGING-0033/SEC STEAL-000%
Q0-00001(00000)                                DORMANT-00015
Q1-00001(00000)                                E1-00000(00000)
Q2-00000(00000) EXPAN-002 E2-00000(00000)
Q3-00018(00000) EXPAN-002 E3-00000(00000)

PROC 0000-006%                                PROC 0001-007%

LIMITED-00000
Ready; T=0.01/0.01 16:08:17
```

---

The response to this command shows you such important information as the percentage of usage for each processor in your system, the usage of real storage, the paging rate, the number of users in the dispatch, eligible and dormant list, and so forth.

Example 1-16 shows the use of the **indicate active** command.

*Example 1-16 indicate active command issued from user maint*

---

```
ind active
0023 USERS, 0021 DISP, 0000 ELIG, 0002 DORM
Ready; T=0.01/0.01 16:29:02
```

---

This command shows the total number of users active in a specified time interval (the default is 60 seconds), and the number of users in the dispatch, eligible, and dormant list that were active in the specified time interval.

## 1.5.2 The RealTime Monitor (RTM)

In this section we discuss the z/VM RealTime Monitor Function Level 410 for z/VM Version 4. RTM is a pre-installed priced feature of z/VM Version 4.

With RTM VM/ESA you can obtain an immediate view of current VM system performance. It is intended for short-term monitoring, analysis, and problem solving. RTM VM/ESA was designed as a real-time monitor and diagnostic tool for monitoring, analysis, and problem solving. It is also recommended that RTM/ESA be used for installations of hardware or software to assist in validating the system components and establishing requirements for additional hardware or software.

Before using RTM there are some tailoring tasks to do. For example, the configuration file has to be customized. For detailed information on how to setup and customize RTM see *z/VM RTM Program Description/Operations Manual*, SC24-6028-02.

To start RTM, just type **rtm**. After the initial RTM screen is displayed, you can issue different RTM commands.

Following are some examples how RTM will appear on your 3270 VM screen.



Example 1-17 is an example of what the **DISPLAY** command will show (without any parameters).

*Example 1-17 example of the RTM GENERAL screen*

---

```

z/VM  CPU2064 SERIAL OCOECB 1280M DATE 05/14/02 START 11:10:25 END 11:10:55
*
<USERID> %CPU %CP %EM ISEC PAG  WSS  RES   UR PGES SHARE VMSIZE TYP,CHR,STAT
LNX17    1.9 .22 1.6  23 51 16K 16K  .0 20K  100  126M VUX,DSC,DISP
LNX5     1.4 .09 1.3  2.6 .00 20K 16K  .0 13K  100  126M VUX,DSC,DISP
SYSTEM   1.1 1.1 .00  .00 .00   0  39  .0 4G  .....  2G SYS,

<--- DEVICE ---> <----- DEVICE RDEV DATA -----> <-- MEASUREMENT FACILITY -->
*
DEV TYPE VOLSER IOREQST SEC  %Q %ER R %LK  LNK PA %UT  ACC  FPT  DCT  CN %CN
3753 3390 430PAG    610 20 .00 .00  .00  0 4 14   7  0  3  3 6.3
3B44 3390 430PG2    544 18 .00 .00  .00  0 4 11   6  0  1  4 7.8
3773 3390 LX3773    283  9 .00 .00  .00  4 4 25  26  0 16 10 9.8
3755 3390 LX3755     64  2 .00 .00  .00  4 4 .82   3  0  1  2 .49

<----- CPU STATISTICS -----> <-- VECTOR ---> <STORAGE><XSTORE>
NC %CPU %US %EM %WT %SY %SP  XSI %SC NV %VT %OT RSTR %ST PSEC %XS XSEC  TTM
-> 2  11 2.2 7.4 189 1.2 .01 3365 96 0  0  0  0 114 142 98  76 6.003
<-.. 13 1.7 9.9 187 .99 .02 2657 95 ..  0  0  0 115  13 94  20 7.590
-----<-- 07 LOG ACTIONS INDICATED -->-----

```

---

Example 1-18 is an example of what the **display dasd** command will show.

*Example 1-18 example of the RTM DASD screen*

---

```

<>z/VM  CPU2064 SERIAL OCOECB 1280M DATE 05/14/02 START 08:00:22 END 11:37:25<>
<--- DEVICE ---> <----- DEVICE RDEV DATA -----> <-- MEASUREMENT FACILITY -->
*
  DEV TYPE VOLSER IOREQST SEC  %Q %ER R %LK  LNK PA %UT  ACC  FPT  DCT  CN %CN
3753 3390 430PAG    38671  2 .00 .00  .00  0 4 1.9   6  0  4  2 .75
3B44 3390 430PG2    33719  2 .00 .00  .00  0 4 1.8   7  0  4  2 .73
3755 3390 LX3755    29974  2 .00 .00  .00  4 4 .94   4  0  1  2 .50
3773 3390 LX3773    13398  1 .00 .00  .00  4 4 1.2  12  0  5  6 .69
3730 3390 LX3730    12021  0 .00 .00  .00  4 4 .33   3  0  1  2 .22
3771 3390 LX3771    10979  0 .00 .00  .00  4 4 1.0  12  0  4  8 .75
3757 3390 LX3757     9498  0 .00 .00  .00  4 4 1.0  14  0  4  9 .72
3A44 3390 LX3A44     6183  0 .00 .00  .00  1 4 .16   3  0  1  1 .08
3750 3390 430RES     3333  0 .00 .00  .00 273 4 .10   3  0  2  1 .04
5090 CTCA          3253  0 .00 .00  .05  0 1 49 2.0S  0 2.0S  0 .00
3770 3390 LX3770     2990  0 .00 .00  .00  4 4 .10   4  0  2  2 .05
3754 3390 LX3754     2700  0 .00 .00  .00  4 4 .09   4  0  2  2 .04
3A43 3390 LX3A43     2562  0 .00 .00  .00  2 4 .10   5  0  2  2 .05
3772 3390 LX3772     2461  0 .00 .00  .00  4 4 .09   4  0  2  2 .05
3756 3390 LX3756     1146  0 .00 .00  .00  6 4 .07   8  0  5  3 .03
3752 3390 LX3752      288  0 .00 .00  .00  2 4 .01   6  0  4  2 .00
ISSUE THE COMMAND "NEXT" TO VIEW THE NEXT SCREEN OF DATA

```

---

Example 1-19 is an example of what the **display user** command will show.

*Example 1-19 Example of the RTM USER screen*

---

```
z/VM  CPU2064 SERIAL 0COECB 1280M DATE 05/14/02 START 12:00:19 END 12:00:49
      *
<USERID> %CPU %CP %EM ISEC PAG  WSS  RES   UR PGES SHARE  VMSIZE TYP,CHR,STAT

LNX4      3.9 .03 3.9  6.0 .10 9443 8467  .0 26K  100  126M VUX,DSC,DISP
LNX12     2.5 .07 2.4  1.3 .00 27K  22K  .0 9723  100  126M VUX,DSC,DISP
LNX17     1.8 .07 1.7  1.3 .00 28K  23K  .0 8224  100  126M VUX,DSC,DISP
SYSTEM    .96 .96 .00 .00 .00  0  673  .0  4G  . . . . .  2G SYS,
LNX7      .82 .06 .76  .43 .00 6272 6267  .0 128K  100  512M VUX,DSC,DISP
LNX1      .80 .46 .34  26 .00 2442 5705  .0 24K  100  126M VUX,DSC,DISP
LNX9      .79 .06 .72  1.6 .00 47K  39K  .0 95K  100  512M VUX,DSC,DISP
LNX18     .66 .10 .56  7.5 .53 17K  16K  .0 120K  100  512M VUX,---,DISP
LNX5      .56 .08 .47  2.4 .00 18K  16K  .0 17K  100  126M VUX,DSC,DISP
LNX8      .49 .10 .39  12 .00 27K  22K  .0 11K  100  126M VUX,DSC,DISP
LNX11     .42 .15 .26  30 .00 5197 5138  .0 27K  100  126M VUX,DSC,DISP
LNX19     .25 .06 .18  .83 .03 40K  25K  .0 106K  100  512M VUX,DSC,DISP
VMRTM     .23 .18 .05 .03 .00 278  299  .0  0  3%A  32M VUS,QDS,SIMW
LNX14     .23 .08 .15  1.4 .10 16K  13K  .0 24K  100  126M VUX,DSC,DISP
LNX3      .20 .07 .12  .56 .03 16K  13K  .0 24K  100  126M VUX,DSC,DISP
LNX15     .19 .06 .13  .36 .00 23K  19K  .0 22K  100  126M VUX,DSC,DISP
ISSUE THE COMMAND "NEXT" TO VIEW THE NEXT SCREEN OF DATA
```

---

The interval for RTM screen refreshes is control by the **interval** command. The default is 30 seconds, but you can set the interval to any value from 1 second up to 3600 seconds (1 hour).

### 1.5.3 FCON/ESA

This section describes the FCON/ESA performance monitor for z/VM.

FCON/ESA (VM/ESA Full Screen Operator **CON**sole and Graphical Realtime Performance Monitor) was developed by Eginhard Jaeger for IBM Switzerland. FCON/ESA is a very powerful and user-friendly z/VM performance monitor that is used in many IBM z/VM installations. The most recent version is FCON/ESA V.3.2.03.

The latest version enables you to monitor Linux/390 systems running under a guest of z/VM. The Linux/390 guests don't necessarily need to be under the same z/VM. Monitoring data is gathered use TCP/IP.

The retrieval of monitoring data is based on the RMF DDS interface, which was originally developed for use with RMF PM. There is permanent data collection in Linux and the history data is saved on the Linux system. The IP addresses of the Linux systems that need to be monitored must be defined in the file FCONX LINUXUSR fm.

The following examples show how FCON/ESA will appear on your screen:

**Note:** The following screen shots of FCON/ESA were made on another system, not on the VMLINUX system we used for this project. They are general samples only.

Example 1-20 Sample main menu of FCON/ESA

---

FCX124	Performance Screen Selection	EMEAVM4
General System Data	I/O Data	History Data (by Time)
1. CPU load and trans.	11. Channel load	31. Graphics selection
2. Storage utilization	12. Control units	32. History data files*
3. Storage subpools	13. I/O device load*	33. Benchmark displays*
4. Priv. operations	14. CP owned disks*	34. Correlation coeff.
5. System counters	15. Cache extend. func.*	35. System summary*
6. CP IUCV services	16. DASD I/O assist	36. Auxiliary storage
7. SPOOL file display*	17. DASD seek distance*	37. CP communications*
8. LPAR data	18. I/O prior. queueing*	38. DASD load
9. Shared segments	19. I/O configuration	39. Minidisk cache*
A. Shared data spaces	1A. I/O config. changes	3A. Paging activity
B. Virt. disks in stor.		3B. Proc. load & config*
C. Transact. statistics	User Data	3C. Logical part. load
	21. User resource usage*	3D. Response time (all)*
D. Monitor data	22. User paging load*	3E. RSK data menu*
E. Monitor settings	23. User wait states*	3F. Scheduler queues
F. System settings	24. User response time*	3G. Scheduler data
G. System configuration	25. Resources/transact.*	3H. SFS/BFS logs menu*
	26. User communication*	3I. System log
H. Exceptions	27. Multitasking users*	3K. TCP/IP data menu*
	28. User configuration*	3L. User communication
I. User defined data*	29. Linux systems*	3M. User wait states

Pointers to related or more detailed performance data  
can be found on displays marked with an asterisk (\*).

Command ==>

---

*Example 1-21 Example of the CPU screen of FCON/ESA*

---

```

FCX100      CPU 9672  SER 42073  Interval 15:14:04 - 15:15:04      EMEAVM4

CPU Load
PROC  %CPU  %CP  %EMU  %WT  %SYS  %SP  %SIC  %LOGLD  %VTOT  %VEMU  REST  Status or
P00   10   6   4   90   4   1   83   11   not installed  Master
P01   9    3   5   91   2   1   97   9    not installed  Alternate
P02   8    3   5   92   2   1   98   8    not installed  Alternate
P03   7    3   4   93   1   1   98   7    not installed  Alternate
P04   8    2   5   92   1   1   98   8    not installed  Alternate
P05   6    2   4   94   1   1   98   6    not installed  Alternate

Total SSCH/RSCH      57/s      Page rate      19.8/s      Priv. instruct.  24/s
Virtual I/O rate     6/s      XSTORE paging  60.4/s      Diagnose instr. 13/s
Total rel. SHARE     14200    Tot. abs SHARE  0%

Queue Statistics:    Q0      Q1      Q2      Q3      User Status:
VMDBKs in queue     1       0       0      42      # of logged on users  67
VMDBKs loading      0       0       0       0      # of dialled users    0
Eligible VMDBKs          0       0       0       0      # of active users     57
El. VMDBKs loading    0       0       0       0      # of in-queue users   43
Tot. WS (pages)      300     0       0  355554  % in-Q users in PGWAIT  0
Expansion factor      2       2       2       2      % in-Q users in IOWAIT 95
85% elapsed time     .520    .065    .520    3.120  % elig. (resource wait) 0

Transactions      Q-Disp  trivial  non-trv  User Extremes:
Average users      .4       .0       .0      Max. CPU % LNXAT005  3.7
Trans. per sec.    .3       .4       .1      Max. VECT % ..... ..
Av. time (sec)     1.394   .020    .125   Max. IO/sec TCPIP    1.1
UP trans. time     .020    .125   Max. PGS/s LINUX008  3.7

Command ==>
F1=Help F4=Top F5=Bot F7=Bkwd F8=Fwd F12=Return

```

---

**Tip:** We recommend using FCON/ESA for problem determination and performance monitoring on a z/VM system if it is available.

The following link will provide you with more information about FCON/ESA.

<http://www.vm.ibm.com/perf/perfprod.html>



## RAID tools, LVM, and EVMS

System administration of storage devices is hard work, especially if you have a large number of devices to deal with. This is especially true for Linux on zSeries and S/390 because of the nature of ECKD DASD. Typically, the largest DASD are emulated 3390-3s which are only 2.3 GB when formatted. These are very small by today's standards. To get a useful volume of DASD, some type of pooling or volume management solution must be used. There are two common solutions used, and a newer one that looks promising. They are:

- ▶ RAID tools (also known as multiple disk or md-tools) are commonly used on Red Hat Linux.
- ▶ The Logical Volume Manager (LVM) is commonly used on SuSE Linux.
- ▶ Enterprise Volume Management System (EVMS) is the newest solution.

LVM is discussed thoroughly in the redbook *Linux IBM eServer zSeries and S/390: Distributions*, SG24-6264, so that solution is not described in this chapter. See the redbook on the Web at:

<http://www.redbooks.ibm.com/abstracts/sg246264.html>

RAID tools and EVMS are described in this chapter. Both of these tools require code in the kernel as well as user commands, concentrating on disk management. Regardless of the volume management tool being used, careful planning and preparation is required, including:

- ▶ Planning the file system sizes and partitioning or pooling DASD accordingly
- ▶ Monitoring and managing the file system space usage
- ▶ Protecting the file systems against corruption through well-planned backup schemes
- ▶ Configuring new file systems dynamically

## 2.1 RAID tools

Implementation of Redundant Array of Independent Disks (RAID) can be done in hardware or software; usually hardware RAID gives the best performance. Use of RAID tools is often how RAID is implemented in software on Linux.

From an application's point of view, RAID is just a conventional disk drive. The application can initiate an I/O to and from the RAID just the way it does with other block devices in the system.

### 2.1.1 RAID levels

There are many levels of RAID possible to satisfy different requirements. Following is an overview of the most commonly used RAID levels: 0, 1, 4, 5 and linear mode (though RAID 0 and linear mode are not truly RAID).

- ▶ Level 0 (striping): This level copies data across disks. However, there is no redundancy of the data, so it is not truly RAID. Because there is no redundancy, a failure of any disk will result in the loss of all the data. The devices should have approximately the same size. Since all access is done in parallel, the devices will be filled up equally. If one device is much larger than the other devices, that extra space is still utilized in the RAID device. However, there is no I/O performance increase in this high end of the RAID device.
- ▶ Level 1 (mirroring): Mirroring requires two or more disk sets of equal storage capacity. This level of RAID makes recovery of data during disk failure simple since all that has to be done is to recover the copy from the undamaged disk. This level of RAID is used mainly to achieve high availability.
- ▶ Level 4: This level uses large stripes, which means you can read from a single drive, allowing you to take advantage of overlapped I/O for read operations. Because write operations have to update the parity drive, no I/O overlapping is possible. Because of this and other technical difficulties, RAID 4 is rarely used.
- ▶ Level 5: This level works basically like level 4, except the parity information is spread out across multiple drives. You must have at least three drives and one drive's worth of space is lost to parity information. If one drive fails, the data is immediately reconstructed. Spreading out of parity information eliminates the bottleneck associated with write operations; however, this also means the recovery of data during disk failure can be very complex. Since this level of RAID offers flexibility, high performance, and redundancy, it is used most often.
- ▶ Level -1 (linear mode): This level creates a large logical drive out of a number of small drives. The disks are appended one after the other and are filled up in order. The drives do not need to be of the same size. In this level, the data is not accessed in parallel, so there is no I/O performance gain. Like RAID-0 there is no redundancy, so it is not truly RAID.

### 2.1.2 Setting up Level 0 RAID using the Red Hat installation tool

For this exercise we have 5 DASD minidisk devices as described in Table 2-1.

Table 2-1 DASD minidisk devices and sizes.

DASD	Size in Cylinders	Use	Comment
201	200	SWAP	Swap space
202	1469	/	Root file system
203	734	RAID-0	/home

DASD	Size in Cylinders	Use	Comment
204	735	RAID-0	/home
205	1469	RAID-1	Mirror of /home

Because Red Hat 7.2 for s390 comes with RAID support, we used its installation tool for setting up the /home file system using RAID-0. We set up RAID-1 (mirroring) on the /home file system manually after completing the installation.

First, we brought up the Red Hat installation initial kernel, RAMdisk, and parameter file via an EXEC, and set up the network device. Then we got this message:

```
Please connect now to <your.IP.address> and start 'loader' from this shell.
```

We telnet into the RAMdisk system, but before invoking **loader**, we checked the DASD:

```
# cat /proc/dasd/devices
0201(ECKD) at ( 94: 0) is dasda      : active at blocksize: 4096, 36000 blocks, 140 MB
0202(ECKD) at ( 94: 4) is dasdb      : active n/f
...
```

If the minidisk is already formatted, as was the case with 201, loader proceeds fine. If the minidisk has not been formatted for Linux yet, as was the case with 202, proceeding with loader produces the following warning:

```
Unable to determine geometry of file/device/tmp/dasdb. You should not use Parted unless
you REALLY know what you're doing!
```

We could proceed by choosing **ignore** and the drive would be formatted for us, or we could invoke **dasdfmt** for each minidisk to be formatted before invoking loader.

We followed the installation procedure until we were offered the **Disk Setup** option. We highlighted **dasdb** (vdev 202) and selected **Edit Partitions** as shown in Figure 2-1. This took us into the **fdasd** tool, from which we could partition the disk.



Figure 2-1 Disk setup panel

We only needed to have one partition on the disk, so we assigned the whole DASD space to the first partition on which we will host the root (/) partition. After editing the partition information for **dasdb**, we did the same for **dasda**, **dasdc**, and **dasdd**.

**Note:** The partition table must be created on all DASDs using fdasd in order to change the partition ID (file system type) later in the installation process.

After editing partition information for all DASDs, we selected **OK**. Here we were given a chance to specify the mount point and change the partition ID. We edited /dev/dasdb1 to be ext2 type and have / as the mount point.

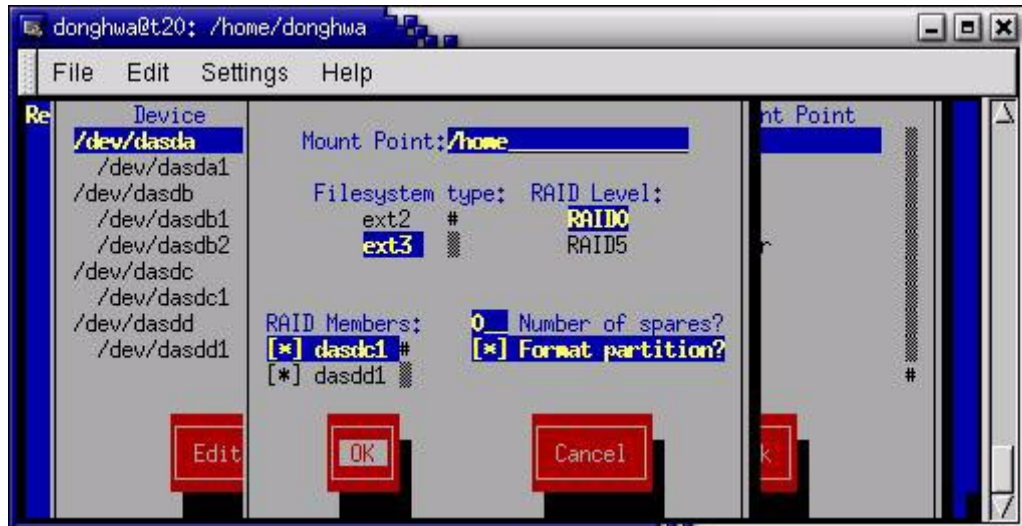


Figure 2-2 Raid setup panel

For /dev/dasdc1 and /dev/dasdd1, we specified the file system type to be software RAID. Then, we selected RAID and the panel shown in Figure 2-2 was presented. Here we could specify the mount point, file system type, RAID Level, RAID Members and number of spares. We specified the Mount Point to be /home and the file system type to be ext3. We selected RAID0 for the RAID level.



Figure 2-3 Raid Setup panel



After specifying all disk setup options, we continued with the rest of the installation process.

We will not go over the rest of installation process here. Consult the Red Hat installation manual. After the installation was complete, we logged onto the system (using `ssh`), then issued the command `df`. The output is shown in Example 2-1.

*Example 2-1 Output of `df` after the installation*

---

```
# df
filesystem      1k-blocks    Used   Available  Use%   Mounted on
/dev/dasdb1     1040912     200384   787652    21%    /
/dev/md0        1040708     32844   955000    4%     /home
```

---

Notice that the `/home` file system is hosted on `/dev/md0` device and its size is about the same as that of `/dev/dasdb1`.

Look at the `/etc/raidtab` file. These are the entries the Red Hat installation tool has specified. There are two entries defined for device option: `/dev/dasdc1` and `/dev/dasdd1`. This is exactly what we want.

### 2.1.3 Setting up Level 1 RAID manually

In this example we describe how to set up RAID-1 (mirroring) over the `/home` file system.

First, dynamically configure the extra DASD (vdev 205) online. Notice that DASD 205 has 1469 cylinders, which is equal to the total size of DASD 203 and DASD 204, the first RAID. Issue the following commands.

```
# echo "add device range=205" >> /proc/dasd/devices
# echo "set device range=205 on" >> /proc/dasd/devices
```

Verify that the DASD is configured online by looking into `/proc/dasd/devices`. Take note of the device name assigned to it. In our case it is `dasde`. Format the DASD using the `dasdfmt` command:

```
# /sbin/dasdfmt -b 4096 -l 1nx16b -d cd1 -f /dev/dasde
```

After the DASD has been formatted, create a partition table on it using the `fdasd` tool. For more information on `fdasd`, see the man page.

Example 2-2 shows our `/etc/raidtab` file.

*Example 2-2 `/etc/raidtab`*

---

```
# cat /etc/raidtab
raiddev          /dev/md0
raid-level       0
nr-raid-disks    2
chunk-size       64k
persistent-superblock 1
nr-spare-disks   0
  device         /dev/dasdc1
  raid-disk      0
  device         /dev/dasdd1
  raid-disk      1

raiddev          /dev/md1
raid-level       1
nr-raid-disks    2
chunk-size       64k
```

```

persistent-superblock      1
nr-spare-disks             0
  device                   /dev/md0
raid-disk                  0
  device                   /dev/dasde1
raid-disk                  1

```

---

Set the `raiddev` (RAID device) to be `/dev/md1`. It will be created with `raid-level 1`. Devices involved will be `/dev/md0` (the first RAID device we have created with `/dev/dasdc1` and `/dev/dasdc2` using RAID-0) and `/dev/dasde1`. The value for the `nr-raid-disks` option must equal the number of devices to be used.

Set the `persistent-superblock` option on: it specifies whether a special superblock is written at the end of all disks participating in the array. This superblock allows the kernel to read the configuration of RAID devices directly from the disks involved, rather than obtaining that information from the `/etc/raidtab` file during the boot time.

The `mkraid` command will read the `/etc/raidtab` file and create the software RAID devices specified. Before issuing the `mkraid` command, unmount devices involved in the construction of the RAID. In our case, we only need to unmount `/dev/md0`. Issue the `mkraid` command, which is used to create a single RAID array out of a set of block devices.

```

# /sbin/mkraid /dev/md1
handling MD device /dev/md1
analyzing super-block
disk 0: /dev/md0, 1057344kB, raid superblock at 1057280kB
disk 1: /dev/dasde1, 1057584kB, raid superblock at 1057472kB

```

Check the `/proc/mdstat` file. It should look something like the following:

```

# cat /proc/mdstat
Personalities : [raid0] [raid1]
read_ahead 1024 sectors
md1: active raid1 dasde1[1] dasdbm[0]
      143808 blocks [2/2] [UU]
      [>.....] resync = 1.7% (6124/287616) finish=43.9min speed=103K/sec
md0: active raid0 dasdc1[0] dasdd1[1]
      1057344 blocks 64k chunks

```

Now, you can remount `/dev/md1` and have it go back to work.

At this point, reconstruction will start on the mirror device. After a few minutes, check the `/proc/mdstat` file again. You will notice that some of the entries for `md1` have changed. Pay special attention to the position of the arrow and field after that, such as `resync`, `finish`, and `speed`. These fields have the following meanings:

```

resync      The percentage of the synchronization that is done.
finish      Time left until completion.
speed       The I/O bandwidth used for reconstruction.

```

**Important:** After setting up the RAID, make sure to specify the virtual device range of the new DASD added dynamically to the system in the kernel boot parameter section of the `/etc/zipl.conf` file. Then, run `zipl` again.

This concludes all the steps you need to set up the RAID. For more information on RAID, refer to the following Web site:

<http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/Software-RAID-HOWTO.pdf>

## 2.2 EVMS

There is a new, integrated package for storage system administration, which extends the domain of the low-level disk administration with other storage management functions, including file system tasks. The Enterprise Volume Management System (EVMS) project has the goal of providing unparalleled flexibility and extensibility in managing storage. It represents a new approach to logical volume management, as the architecture introduces a plug-in model that allows for easy expansion or customization of various levels of volume management.

**Important:** Because EVMS is new, it should be used in a production environment cautiously.

You can find the EVMS project home page at:

<http://evms.sourceforge.net/>

The current stable version of EVMS is 1.1.0. It was released on July 31, 2002.

### 2.2.1 How to get and install EVMS

The EVMS package is not included in the official Linux kernel source code yet. Most of the distributions also do not have the EVMS utilities packages: only Debian has included EVMS into the standard distribution, both source and precompiled binary packages. However, it is expected that EVMS will be included with new Linux distributions, specifically, UnitedLinux. If you are using a distribution that does not include EVMS, you probably have to compile your own kernel and also the EVMS utilities in order to use EVMS. For the purpose of this discussion, we assume that you have already managed to download the kernel sources and patched them with all the necessary S390 patches, and also that you have installed the whole kernel development tool chain.

**Important:** If you have support from a Linux distributor, a rebuilt kernel will probably not be supported. Therefore, such a system should be for experimental/pilot project purposes only.

Download the file `evms-1.1.0-1.src.rpm` from the Web site:

[http://sourceforge.net/project/showfiles.php?group\\_id=25076](http://sourceforge.net/project/showfiles.php?group_id=25076)

Most File System Interface Module (FSIM) plug-ins, such as the JFS FSIM, require their respective file system utilities to be installed in order to work. A warning will be displayed if the appropriate file system utilities are not found.

Unpack the source, and follow the README. There is a really easy way to patch and recompile the kernel. Do not compile the EVMS features as modules; currently there is no built-in mechanism to load EVMS modules. After rebooting the kernel, you can start using EVMS.

### 2.2.2 EVMS interface options

There are three user interfaces possible to communicate with the EVMS backend:

- ▶ A command line tool, `evms`, similar to LVM and RAID-tools
- ▶ An ncurses tool, `evmsn`, similar to `yast`
- ▶ An X-Window application (`evmsgui`)

- ▶ Utilities - replacement utilities for users of LVM and md-tools that allow them to work with EVMS using their old familiar command line utilities. See 2.2.4, “How to start with EVMS” on page 45.

The `evms` command line tool is significantly different than LVM and RAID tools. It is quite extensive and not addressed in this chapter. It is an excellent tool for scripting, or writing your own, higher level, volume management tools.

If you have an X server on your desktop, `evmsgui` is recommended when scripting is not necessary. A nice feature is fly-over tips (short descriptions), which are available for many input fields, to get some details about each of the input fields.

Be sure to set the `DISPLAY` environment variable and invoke the `evmsgui` command. It should bring up a window similar to that shown in Figure 2-4 on page 46. If you do not have an X server, `evmsn` can be used.

Many small, valuable features of EVMS, especially `evmsgui`, make the day-to-day volume management work much easier and less error-prone. EVMS capabilities include:

- ▶ Automatic discovery of storage objects, realized with the help of persistent metadata (superblocks and so forth) in various objects. This helps with the initial setup and change maintenance.
- ▶ A view via a GUI of every level of the storage hierarchy. The listings help the inventory with much less administration effort.
- ▶ The sequence of views suggests the proper sequence to define higher level objects. Because this makes the storage hierarchy more intuitive, there is less hesitation.
- ▶ The content of the pop-up menus depends on the status of the system: if the preconditions of an operation are not satisfactory, that operation is not shown in the menu. Again, this makes operations more intuitive because only valid operations are offered.
- ▶ Creation of new objects can be initiated at any time through the global **Actions** menu.

### 2.2.3 Terminology

EVMS has a complex design to handle various hardware platforms and operating systems. The terminology it uses is therefore also a bit more complex than usual. To get acquainted with the special EVMS terms, take a look at:

<http://evms.sourceforge.net/terminology.html>

Table 2-2 is a quick review of some EVMS terms and their Linux counterparts.

*Table 2-2 EVMS and corresponding Linux terms*

EVMS term	Linux term
Logical disk	390 DASD
Segment	Partition
Container	Group
Region	Logical volume
Logical volume	Linux volume
Feature	Nothing comparable in Linux

EVMS builds storage stacks, layering objects on top of one another. If you are familiar with LVM layering on top of MD and partitions, the EVMS concept of stacks is analogous.

However, EVMS can include other region managers (for example, AIX and OS/2) and other features (such as bad-block-relocation) in its stacks.

EVMS volumes are created from partitions, disks, regions, or any top-most object in the feature stack. For example, EVMS can export a disk that hasn't been partitioned as a Linux volume. If a disk has been partitioned, individual partitions can be exported as volumes. If disks are linked together with MD-1 or EVMS drive linking, these linear RAID collections can be exported as volumes. Basically, whatever is sitting at the top of the stack can become an EVMS volume.

Every plug-in in EVMS is responsible for its own metadata. The MS-DOS partition manager uses the standard MBR/EBR scheme for its metadata. The EVMS S/390 segment manager uses the label track as its metadata. The GPT segment manager follows the EFI specification, which describes the metadata layout for this partitioning scheme. The EVMS drive-linking feature consumes several sectors at the end of a storage object to hold its metadata. File systems lay down their own metadata. A logical volume is how you present the storage stack to the user. Something simple like `/dev/evms/hda1` is a logical volume. Something more complex, like mirrored LVM regions, can also be exported as a volume. Logical volumes are how the stack is presented. In fact, something like `/dev/evms/dasdb1` would not have any EVMS-specific metadata on it at all if it was exported as a compatibility volume.

## 2.2.4 How to start with EVMS

If you are familiar with LVM command line tools, EVMS provides versions of all the utilities. If you would rather use these, you can use the EVMS versions:

- ▶ `evms_lvcreate`
- ▶ `evms_lvdisplay`
- ▶ `evms_lvextend`
- ▶ `evms_lvreduce`
- ▶ `evms_lvremove`
- ▶ `evms_lvscan`
- ▶ `evms_pvdisplay`
- ▶ `evms_pvscan`
- ▶ `evms_vgcreate`
- ▶ `evms_vgdisplay`
- ▶ `evms_vgextend`
- ▶ `evms_vgreduce`
- ▶ `evms_vgremove`
- ▶ `evms_vgscan`

However, if you are moving to a new tool, you will probably want to use the tool's native interface.

In the example that follows, Linux is installed on a single file system and the device is named `/dev/evms/dasda1` (at address 515b). This file system we want to leave as is. We want to experiment with the devices `/dev/evms/dasdb1 - dasdf1` (at addresses 0192-0195), which are presently unused.

We will create the equivalent of an LVM volume of `dasdb1` and `dasdc1`, and also the equivalent of multi-disk (MD) linear array of `dasdd1` and `dasde1`. Then we will create some containers and EVMS logical volumes on top of these volumes, and create a couple of different file systems. Finally we will expand one of these file systems.

When `emvsgui` is invoked, all the available DASD devices are recognized (look for a list of them in the Disks view), and all of these devices are also defined as logical volumes.

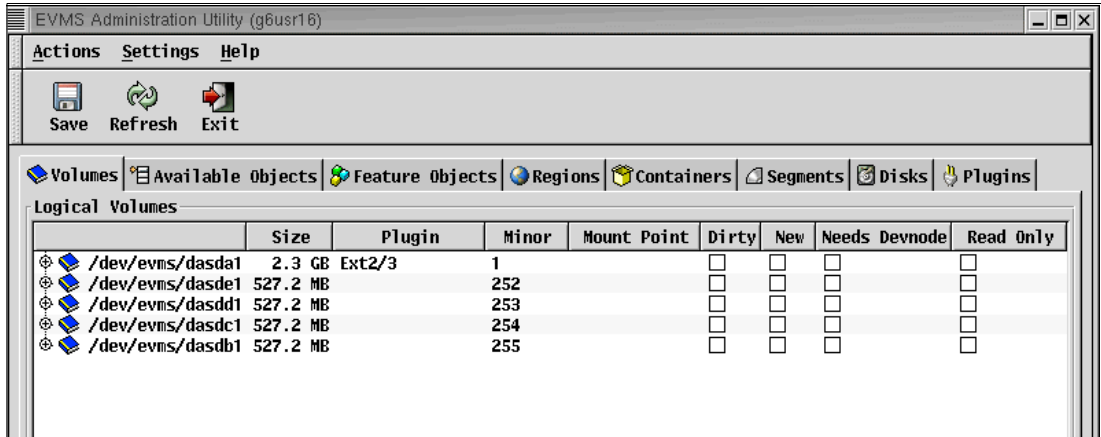


Figure 2-4 EVMS Volumes view after evmsgui first started

We do not want to see them as EVMS volumes because we want to pool them, so the first step is to remove them from this list. This operation is easy and intuitive: one click with the right mouse button on the /dev/evms/dasdb1 volume and a pop-up context menu is shown with **Remove** as one of the operations (see Figure 2-5 on page 46).

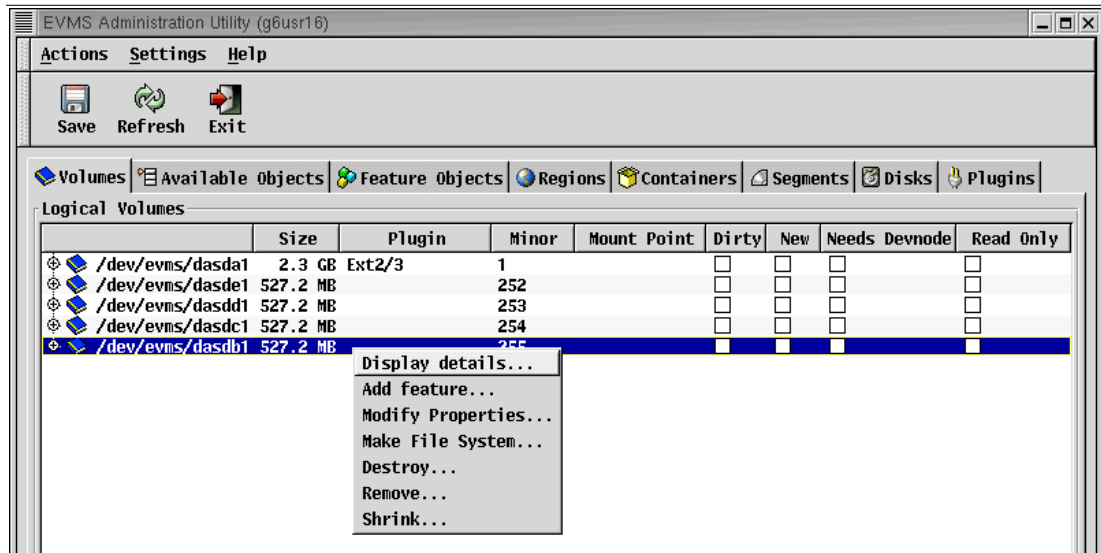


Figure 2-5 Remove unwanted EVMS volumes

Select **Remove** and the volume is no longer displayed. Repeat this for all unused devices so they no longer appear in the Volumes view. Now there are four new objects in the Available Objects view. It is recommended that you remove the previous contents of those DASD by formatting them before proceeding. Again right-click and a context menu is shown. Select **Format disk dasde**.

**Important:** Formatting the DASD will destroy data, so be sure you are working with the correct DASD!

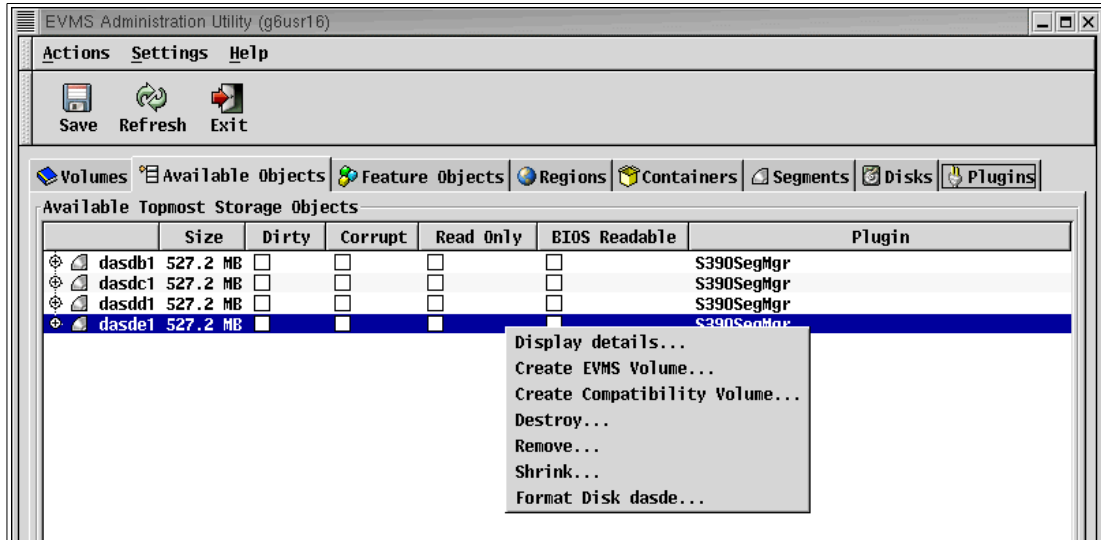


Figure 2-6 Format disk dasde

All DASD devices must be partitioned before use. This is typically done via the `fdasd` command line tool. However, EVMS is a well integrated system: it has the ability to check certain preconditions and knows the proper sequence of operations. The `evmsgui` tool makes it easy to perform the DASD format as well as many other operations. Therefore it is recommended that you let `evmsgui` do this for you.

The next screen presented is shown in Figure 2-7. Both of the defaults are recommended. The Compatible Disk Layout (**CDL**) is recommended for the Disk Layout so that z/OS can recognize Linux DASD if necessary. 4 KB (**4096**) is the maximum block size and is recommended, unless you have a good reason to use a smaller one.

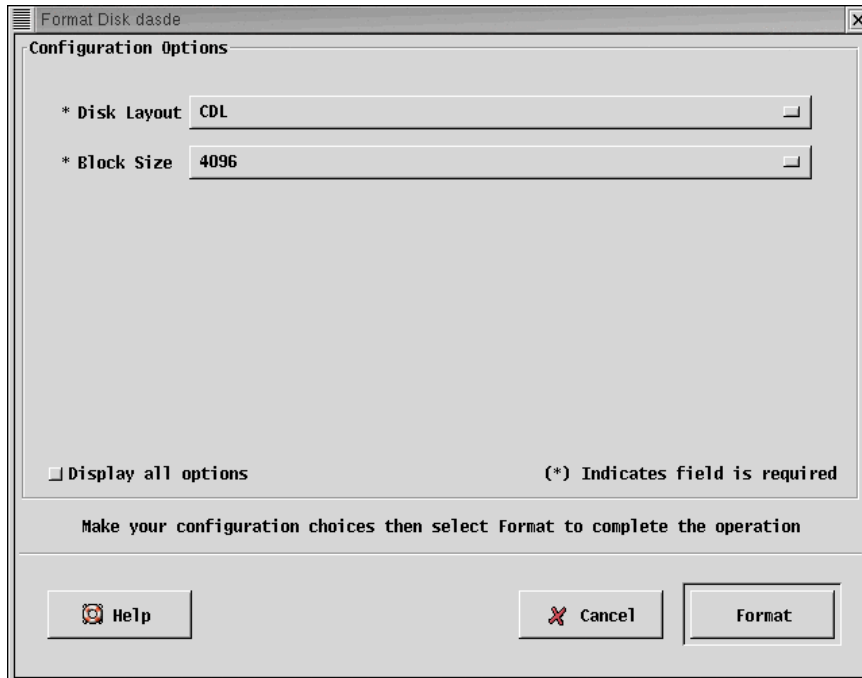


Figure 2-7 Specify dasdfmt parameters

## 2.2.5 How to use EVMS

From the existing segments we have to build the next levels in the storage hierarchy. First we have to define containers. The Linux LVM region manager plug-in uses containers, so this is what will be used. Later we should create regions on top of containers. Regions could be converted to EVMS volumes, which, in turn, could be formatted to file systems, then mounted and used.

In continuing with the example, refer to Figure 2-8 on page 48. The available segments are now identified so a new container can be defined. The **Actions/Create** menu is selected, then the **LVM Region Manager**. You can specify which segments (DASDs) should belong to the new container. The segments `dasdb1` and `dasdc1` are selected.

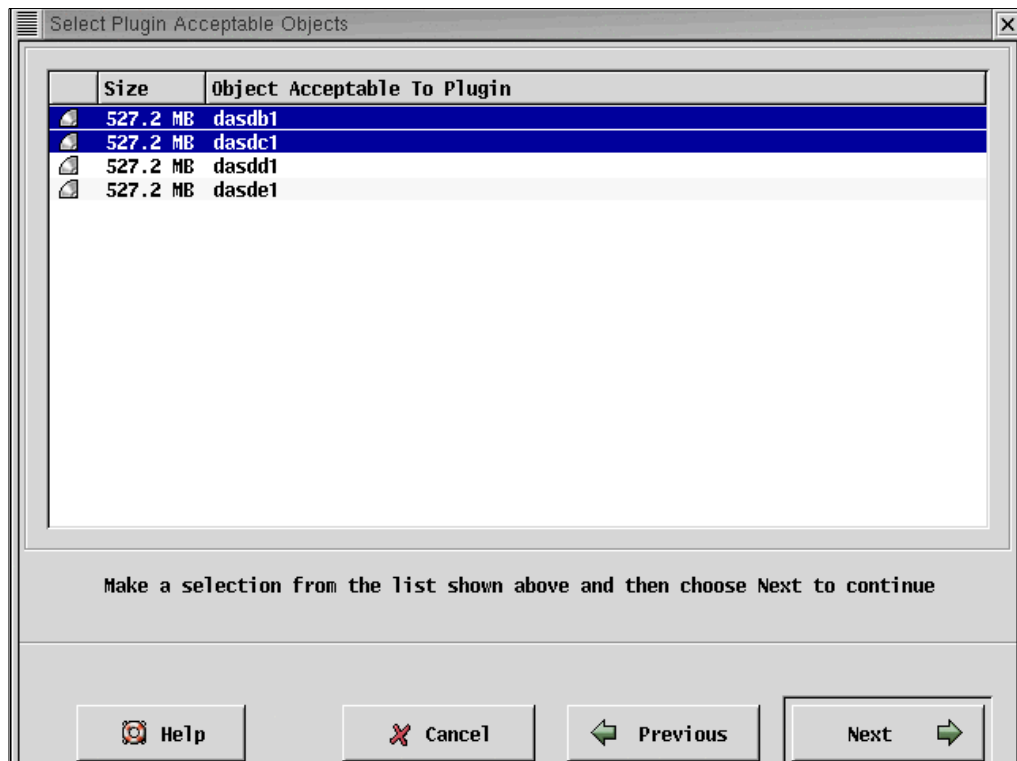


Figure 2-8 Select container segments

You should also specify some necessary parameters on the next screen, shown in Figure 2-9 on page 49. The name of the storage container and the PE size are set here (the term physical extent is a sign again for containers being basically the same thing as volume groups in LVM).



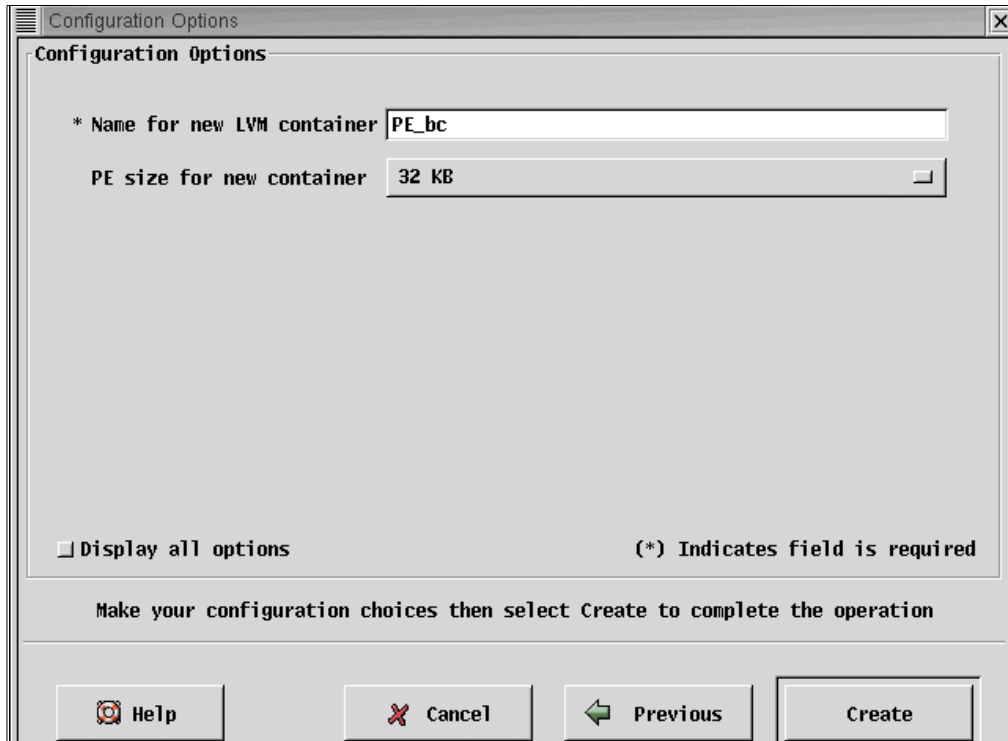


Figure 2-9 LVM container parameters

Regions can be defined as LVM logical volumes or as RAID devices. We have no real region yet; the container PE\_bc just defined has only free space. The Region Manager has full control of the situation, and shows this fact, naming the region as Freespace. This is shown in Figure 2-10 on page 50.

To show a different example we define another type of a region, a real RAID region, by selecting the menu choice **Actions -> Create -> Region**. We use the remaining two DASD partitions for a linear RAID array by selecting the corresponding option and marking the objects dasdd1 and dasde. The results of the create region operation are also shown in Figure 2-10.

**Note:** The concepts of LVM and RAID are different - there is no Freespace on the object md/md0 like on lvm/PE\_bc/Freespace. The RAID volume is not dividable and is ready for use; however, the structure of the LVM object should be further refined.

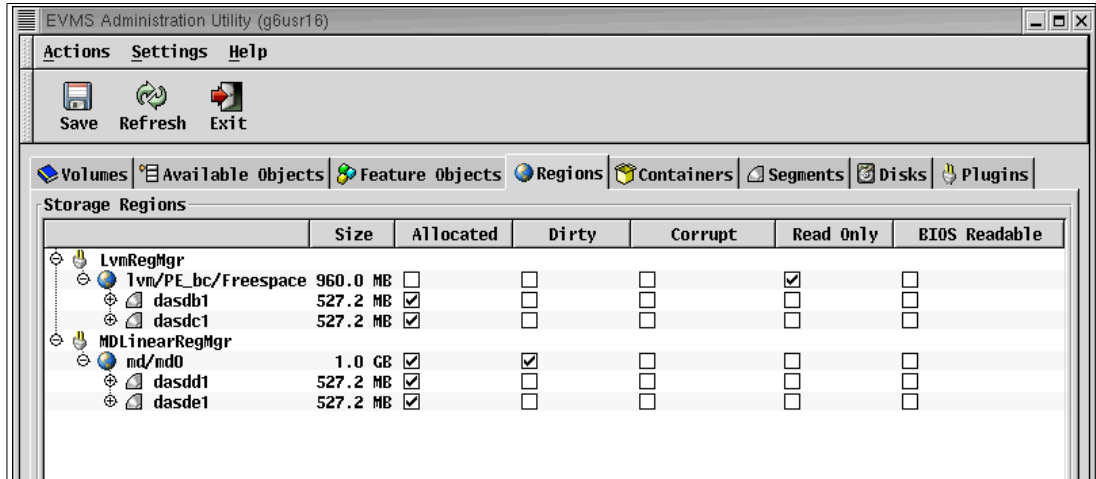


Figure 2-10 Allocated regions

Now we define a real LVM region. Again select the menu choice **Actions -> Create -> Region** and choose the **LVM Region** manager. It will show the Freespace container as the only acceptable object. Click **Next** to bring up configuration options screens. Many parameters can be set. The most important ones are the name for the new LVM Region (Logical Volume name), the number of the logical extents, the size of the new region, the number of stripes, and the stripe size.

The next logical step is to convert this new region (named lvm/PE\_bc/LV\_b) into an EVMS volume, and to give it a name, for example, *my\_space*. It then appears as a new volume (device) under the name `/dev/evms/my_space`. This is the name to be used for file system creation, or for the `mount` command if you have to use the command line interface. But, because EVMS is an integrated system, you can create the file system within EVMS. Figure 2-11 shows you how to do this.

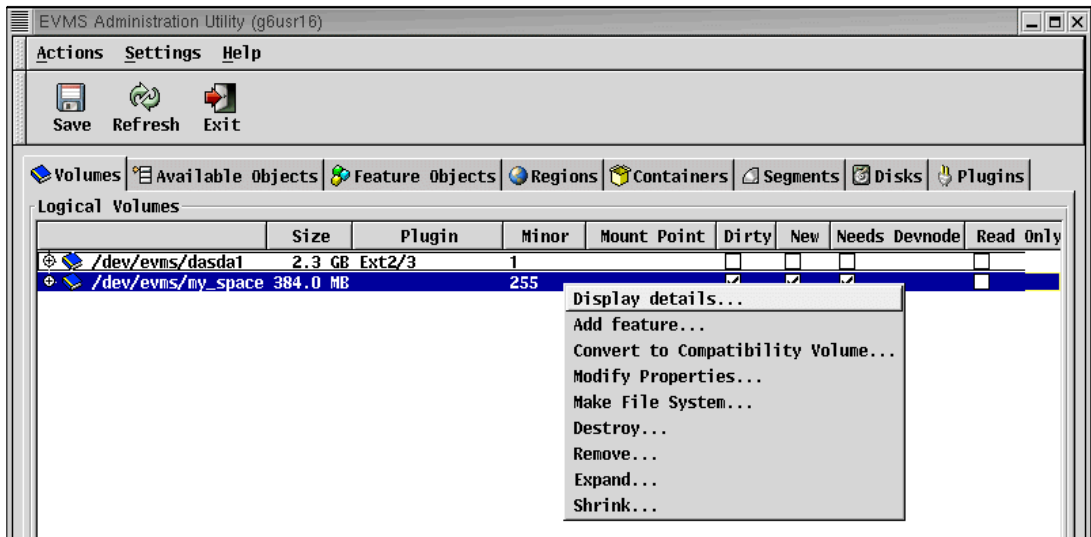


Figure 2-11 Make file system

You must specify the file system type and parameters on further input screens. Currently ext2/ext3, reiserfs and JFS are supported; these are the available plug-ins. Work is being done on an Xfs plug-in. Given the plug-in structure of EVMS, you can also support your own file system relatively easily (in the best open source manner).

The file system creation will not happen immediately. EVMS works in such a way that you can undo most of the actions in a session since the operations work mostly only on internal metadata. There are, of course, operations working with actual disk data, just like the file system formatting. These are deferred till you ask EVMS to save the session as shown in Figure 2-12.

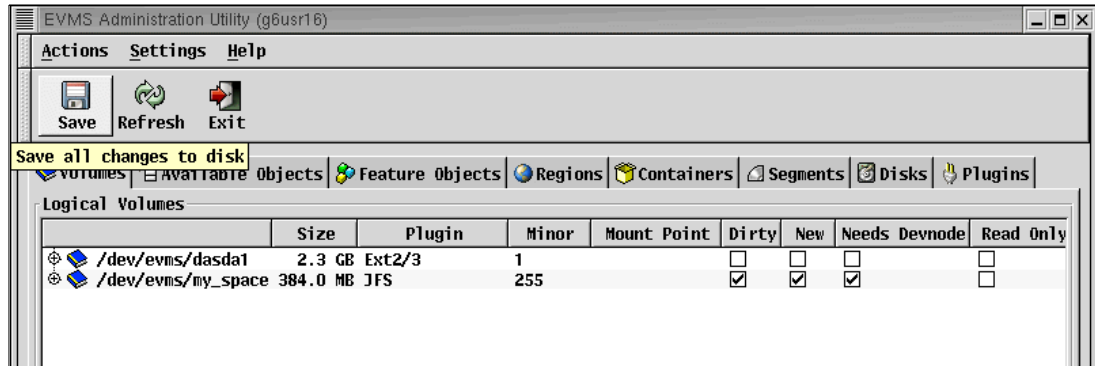


Figure 2-12 File system creation is deferred until 'Save changes'

You can see the JFS plug-in is associated with /dev/evms/my\_space, and it has the dirty bit set (which signifies an action is pending). Clicking the **Save** button triggers the deferred format action, and it succeeds. The mount operation is not yet available in the current EVMS version; it is one of the few functions not yet implemented. Therefore, go to a command line, define a mount point and mount the file system, and it is discovered by EVMS immediately. You will see it mounted under /jfs on a later figure.

The EVMS system shows its real strength on the following example. We defined another region/volume (named 'small') and formatted it as an ext3 file system. Later we decided to expand the whole region, and also the file system on it, of course. We selected the volume and right-clicked it. The pop-up menu has an **Expand** entry (what you have in the context menu really depends on the file system and the corresponding plug-in – if these implement a corresponding feature, you can also shrink the region and the file system, too). This is shown in Figure 2-13 on page 52.

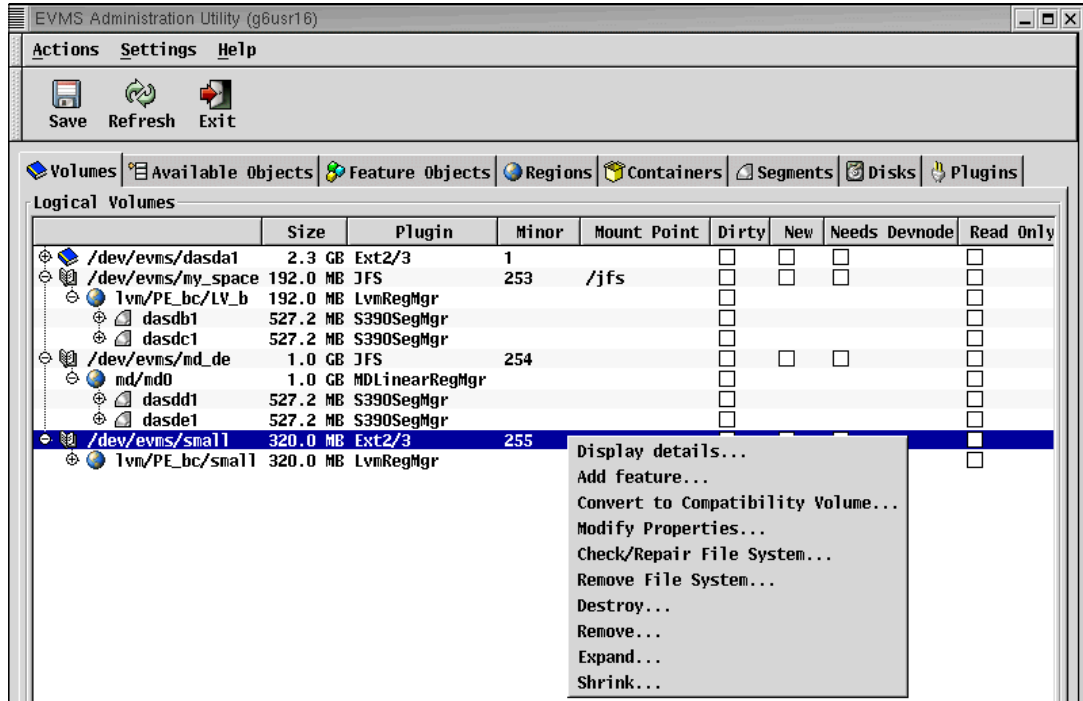


Figure 2-13 Expand

After specifying the new size, the free space, the number of additional extents, and so forth, EVMS schedules all the necessary steps, in the right order. Again, the execution is deferred until the next save.

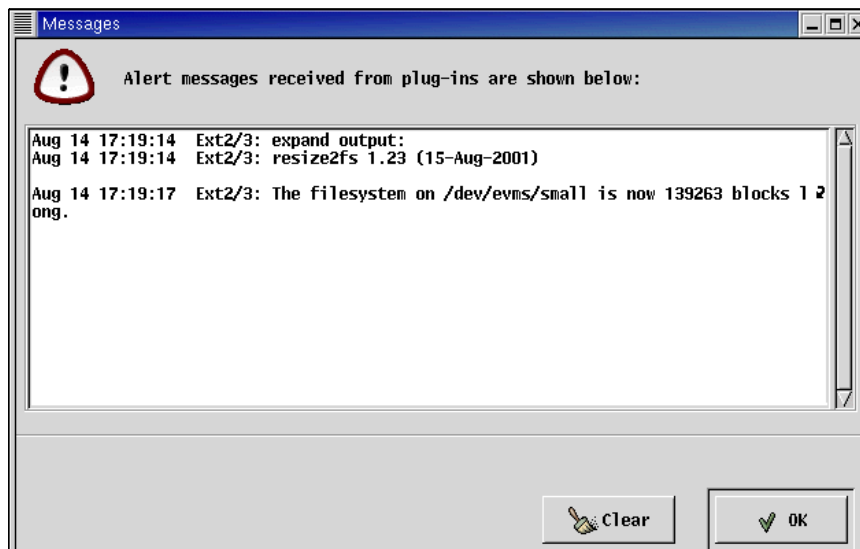


Figure 2-14 The file system is expanded now

This was a short tour displaying most of the integrated features of `evmsgui`.

In addition to the tools `evms`, `evmsn` and `evmsgui` described previously, there are some other EVMS tools that complete the suite. They are described in the remainder of this chapter.

► **evms\_gather\_info**

This script will generate several pages of information written to the console. Analyzing this script and its output you can see that the **evms** command is called with some commands:

```
echo "q:p      /* List all engine plug-ins. */" >> evms_gather_info.tmp.$$
echo ":q:D,lo /* List all disks.          */" >> evms_gather_info.tmp.$$
echo ":q:S,lo /* List all segments.       */" >> evms_gather_info.tmp.$$
echo ":q:C,lo /* List all containers     */" >> evms_gather_info.tmp.$$
echo ":q:R,lo /* List all regions        */" >> evms_gather_info.tmp.$$
echo ":q:O,U,lo /* List all feature objects */" >> evms_gather_info.tmp.$$
echo ":q:V,lo /* List all volumes        */" >>
```

You can see what we see if **evms** is called from the command line with the **q:V** command.

► **evms\_devnode\_fixup**

The EVMS HOWTO describes this command well. Find the details at:

<http://evms.sourceforge.net/howto/usingevms.html>

“On systems that are not running devfs, the device nodes in the `/dev/evms` directory may become out of sync with the volumes that are exported by the EVMS kernel.

Using a user interface could alleviate many of the problems listed above. If you use the user interfaces and commit changes, the EVMS Engine will update the device nodes in the `/dev/evms` so that the nodes agree with the volumes that are exported by the EVMS kernel.

The `evms_devnode_fixup` program provides an alternative means of fixing the device nodes in the `/dev/evms` directory by updating the device nodes without the overhead of starting a user interface.

`evms_devnode_fixup` can be run as a daemon by specifying the `-d` option. In daemon mode, `evms_devnode_fixup` will first fix the device nodes in the `/dev/evms` directory. It then loops, waiting for notifications of volume changes from the EVMS Runtime. On each notification, `evms_devnode_fixup` fixes the device nodes in the `/dev/evms` directory.”

Following is an example:

```
# evms_devnode_fixup
Devfs is running on this system. Devfs will keep the EVMS device nodes up to date.
evms_devnode_fixup has nothing to do.
```

► **evms\_info\_level**

This command determines how much kernel logging is done. The same string is also a kernel command-line boot parameter. If you want to get more boot-up messages pertaining to EVMS, try adding the parameter `evms_info_level=8`.

Following is an example of the help and of using the command:

```
# evms_info_level
Usage: evms_info_level [new info level(range: 0 - 10)]
       If no parameter, reports current evms info level.
       Optional parameter, sets current evms info level.
Current evms info level : 6
# evms_info_level 2
Current evms info level : 6
New evms info level : 2
```

► **evms\_rediscover**

The HOWTO says the following with regard to building and installing an EVMS kernel:

“You also have the option of building EVMS as kernel modules instead of compiling directly into the kernel. However, if you choose this method, EVMS will not

automatically perform discovery at kernel boot time. You will need to load the necessary kernel modules from user-space using **insmod** or **modprobe**. Then, use the **evms\_rediscover** utility to tell EVMS to discover the logical volumes. “

Following is an example:

```
# evms_rediscover
Rediscover successful.
After: Volume(s) info:
(major,minor): volume-name
(117,1): "/dev/evms/dasda1"
(117,2): "/dev/evms/dasde1"
(117,3): "/dev/evms/dasde2"
(117,4): "/dev/evms/dasde3"
(117,5): "/dev/evms/dasde4"
(117,6): "/dev/evms/dasde5"
```



## RMF PMS

In this chapter we discuss the Resource Measurement Facility Performance Monitoring Server (RMF PMS), a product used for performance and capacity planning purposes.

The capabilities of RMF PMS include the following:

- ▶ Data collection and management of that data
- ▶ Data visualization (online and reports)
- ▶ Allowing the performance analyst to make real time operational changes to a running image
- ▶ Workload management by scripts and policies
- ▶ Capacity planning and modeling (transforming Linux data into the formats needed by various tools)

RMF PMS is not available in source format; for more information about RMF PMS usage, refer to the RMF Web page at:

<http://www-1.ibm.com/servers/eserver/zseries/zos/rmf/>

## 3.1 RMF PMS overview

The RMF PMS is a modular data gatherer for Linux. It captures data in a Linux environment, available either online or through history log files, for analysis using the RMF PM client application. This is called the Distributed Data Server (DDS).

A Linux environment can be analyzed and measured through the data captured using RMF PMS.

**Note:** This is a beta version!

The following reminder is posted on the RMF PM Web site:

“Note that this offer is not a part of any IBM product, so we can't promise you any service for it. We need your feedback about this offer; please send a note (contact RMF team).

<http://www-1.ibm.com/servers/eserver/zseries/zos/rmf/rmfhtmls/contrmf.htm>

### 3.1.1 How RMF PMS works

RMF PMS collects data in a Linux environment; this data is then available for analysis. The default interval collection time is 60 seconds, but this can be changed if required.

All gatherer modules work as daemons. They do not need to be synchronized; this means they can either all be started, or just selected ones can be started.

RMF PMS can do the following:

- ▶ Generate graphical trend reports
- ▶ Filter performance data
- ▶ Store the data in spreadsheet format (for example, MS Excel, Lotus 123)
- ▶ Gather historical performance data
- ▶ Configure screens in RMF PM client
- ▶ Mix z/OS and Linux performance data on one screen

Figure 3-1 presents a schematic overview of RMF PMS.



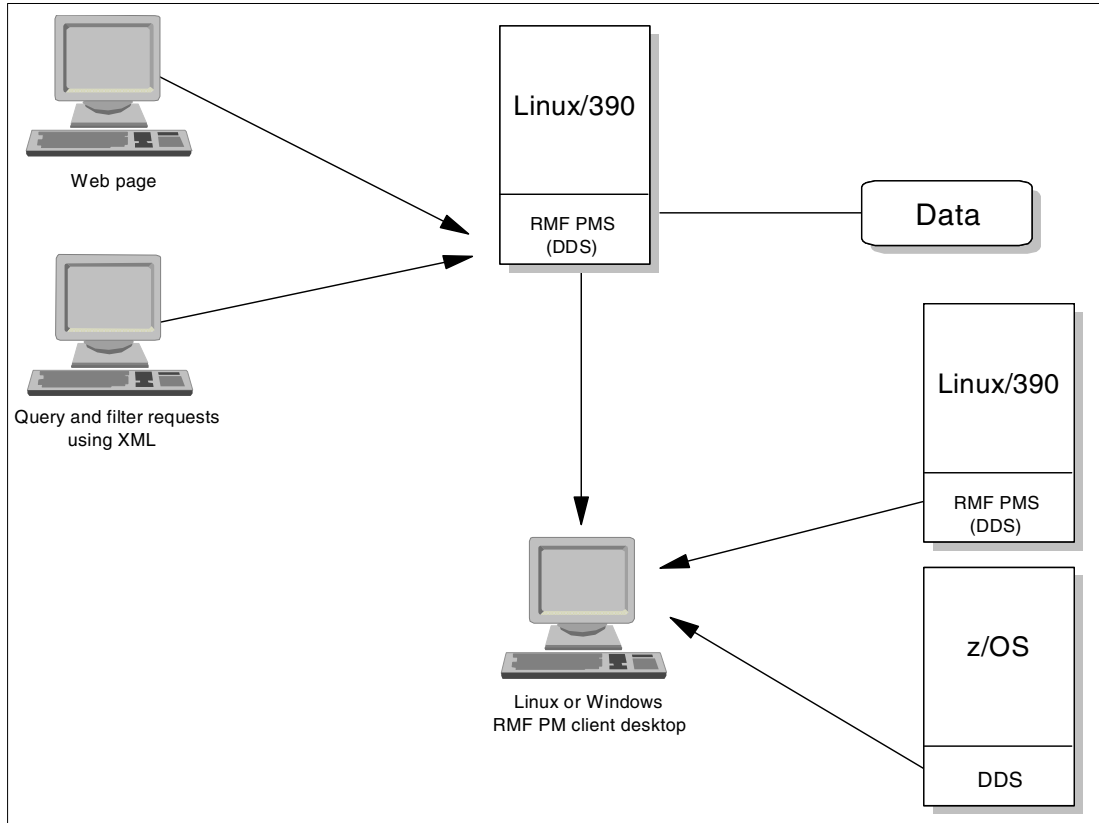


Figure 3-1 RMF PMS overview

RMF PMS data can be accessed by a Linux or Windows client using the desktop, via a Web page, and through the local program using XML queries and filters.

## 3.2 Planning for RMF PMS

Before you install RMF PMS, consider the following points:

- ▶ Allocate sufficient DASD space for RMF PMS.
- ▶ Define the schedule to archive the data.
- ▶ Define which machines will be monitored.
- ▶ Define either a Web interface, or an RMF PM client or gatherer server only.
- ▶ Define which processes to monitor.
- ▶ Define the use of proper programs to manipulate the data.

In addition, make sure that your system meets the following prerequisites:

- ▶ For S/390 server:
  - Linux version 2.2 or 2.4
  - IEEE CPU support
  - glibc 2.1
  - libpthread
- ▶ For client:
  - Windows NT, Windows 2000, Windows 9x or Linux with Java 1.3

## Alert about browser

Use the Internet Explorer Web browser; Netscape 5.5 and Mozilla have problems with the use of XML.

## Directories

The RMF PMS has three directories in its home directory: bin, doc and .rmfpms. The bin directory contains the RMF PMS programs, scripts, and system configuration files; the doc directory contains the document files.

The .rmfpms directory holds data about PID owner process and the following subdirectories:

- deltas        Deltas files (Interval)
- logs         RMF PMS logs files
- directory of data for each collected day  
              In the format: YYYYMMDD, where YYYY is the year, MM is the month and DD is the day.

## Main programs

Table 3-1 identifies and describes the main RMF PMS programs.

Table 3-1 Main RMF PMS programs

Name of program	Function	Parameters
rmfpms	Start and stop the gatherer modules and the Web server	<i>start</i> <i>stop</i> <i>status</i> <i>restart</i>
rmfpms_archive	Archive all performance data not created on the same day	none
rmfpms_unarchive	Restore the performance data of one specified date	YYYYMMDD

## Differences between Linux kernel 2.2 and 2.4 version

Metrics for the DASD option were added to Linux kernel version 2.4.

## Configuration files

RMF PMS has a number of configuration files. The main RMF PMS configuration file is called .rmfpms\_config and is in the RMF PMS home directory. The other configuration files are system configurations, except gpmsrv00.ini, which is where you can modify the default http port number. This file is located in the bin directory.

The default .rmfpms\_config is shown in Example 3-1.

Example 3-1 RMF PMS default .rmfpms\_config file

```
# rmfpms_config - included in rmfpms bash shell script
#
# 11/14/2000, 07/26/2001 Oliver Benke
# (c) IBM Deutschland Entwicklung GmbH, IBM Corp.
#
# configuration parameters
export IBM_PERFORMANCE_REPOSITORY=$HOME/rmfpms/.rmfpms
export IBM_PERFORMANCE_HOME=$HOME/rmfpms/bin/
export IBM_PERFORMANCE_MINTIME=60
```

```
export LD_LIBRARY_PATH=$IBM_PERFORMANCE_HOME:$LD_LIBRARY_PATH
export APACHE_ACCESS_LOG=/var/log/httpd/access_log
export APACHE_SERVER=localhost
export APACHE_SERVER_PORT=80
```

---

In the `rmfpms_config` file, enter the home of RMF PMS, along with repository and interval times for collection of data in `IBM_PERFORMANCE_MINTIME` in seconds.

In the Apache parameters section, enter the location of the Apache log file in `APACHE_ACCESS_LOG`; the hostname in `APACHE_SERVER`; and the port address of the server in `APACHE_SERVER_PORT`.

To incorporate data from Apache into RMF PMS, verify the following parameters in your `http.conf` file:

- ▶ `LoadModule status_module`
- ▶ `AddModule mod_status.c`
- ▶ `ExtendedStatus on`
- ▶ In Location `/server-status` directive: `SetHandler server-status`

**Note:** In the `httpd.conf` file of some distributions of the Apache Server, there is a module list with `AddModule` directives. The module list, found just below the sample `LoadModule` directive, reconstructs the module execution order.

If you don't want to monitor an Apache Web server, ignore the Apache parameters and remove or rename the programs `apachegat` and `apachecomp` in the RMF PMS bin directory.

**Note:** It is not possible to monitor more than one Apache Web server per RMF PMS data gatherer.

## RMF PMS logging

RMF PMS uses the subdirectory logs in the `.rmfpms` as a the system log file. When you start RMF PMS with the default startup, it starts all gatherer modules. These modules collect the data shown in Table 3-2.

*Table 3-2 Description of gatherer modules*

Module name	Purpose
<code>apachegat</code>	Collect Apache data
<code>dasdgat</code>	Collect DASD data
<code>filegat</code>	Collect file system data
<code>gengat</code>	Collect system data
<code>netgat</code>	Collect network data
<code>procgat</code>	Collect process data

You can start as many modules as you like, depending on what information is required. This lets you use your benchmark with minimal interference. When your benchmark finishes, you can access the data using RMF PM client, Web server.

## RMF PMS metrics

The available metrics are grouped by resources, and there are two classes of counters: a list-valued counter and a single counter. The single counter is an exact value; whereas the list-valued counter, is made up of a list of value names. The available resources are:

- ▶ System
- ▶ Network
- ▶ CPU
- ▶ File system
- ▶ Memory

**Note:** DASD information, available only in the Linux kernel 2.4 version, is included in the file system resource.

The following lists present a few examples of the available metrics. For more information see the online documents in the `rmfpms` directory.

- ▶ System resource metrics
  - Rate of process created
  - Apache HTTP server: rate of 404 errors
  - Apache HTTP server: rate of requests
  - Apache HTTP server: bytes per requests
- ▶ Network resource metrics
  - Bytes received/transmitted
  - Packets received/transmitted by network device
  - Receive/transmit error
- ▶ CPU resource metrics
  - Load average
  - Percent of CPU total active by processor
  - Percent of CPU idle time
  - Percent of CPU time in kernel mode by process
  - Accumulated CPU time in user mode by process
- ▶ File system resource metrics
  - Space available
  - Size of all file systems
  - DASD I/O requests per second
  - Percent of space used
  - DASD I/O average response time per request
- ▶ Memory resource metrics
  - Memory used
  - Swap space used
  - Cache memory
  - Number of pages swapped in/out
  - Shared memory
  - Total memory size

## 3.3 Generating RMF PMS

The option we used for gathering information was RMF PM client. The RMF PM client is available for Linux and the Windows desktop. In the lab, we used the Windows desktop.

RMF PM client and gatherer code was downloaded from the home page:

```
http://www-1.ibm.com/servers/eserver/zseries/zos/rmf/rmfhtmls/pmweb/pmlin.htm
```

We created a user *rmfpms* with its proper user group *rmf* and a home of */datacoll/rmfpm*s. We used a directory other than home because this is a software data collection, so we allocated a specific DASD device for it.

After downloading the client, we installed it, specifying Linux in the “Specify type of system to monitor” field and the IP address of the Linux machine in “Enter TCP/IP hostname of server;” in our case it was 10.1.3.4. We have not shown this procedure in detail because the process is commonly understood by Windows users.

Once the gatherer code was available in our Linux for zSeries and S/390 environment, we logged on as *rmfpms* user and extracted the tar ball file to the *rmfpms* directory.

```
tar -xzf rmfpms_390.tgz
ls
rmfpms rmfpms_s390.tgz
```

We changed to the directory *rmfpms* and started the initial gatherer without modifying any parameters. The initial gatherer startup is shown in Example 3-2.

*Example 3-2 Initial gatherer startup*

---

```
rmfpms@lnx4:~ > cd rmfpms
rmfpms@lnx4:~/rmfpms > ./bin/rmfpm start
Creating /datacoll/rmfpm/rmfpm/.rmfpms ...
Starting performance gatherer backends ...
DDSRV: RMF-DDS-Server/Linux-Beta (Mar 1 2002) started.
DDSRV: Functionality Level=1.813
DDSRV: Reading exceptions from gpmexsys.ini and gpmexusr.ini.
DDSRV: Server will now run as a daemon process.
done!
```

---

We started the RMF PM client in Windows desktop by selecting:

**Start -> Programs -> IBM RMF Performance Management -> RMF PM**

The initial RMF PM client window is shown in Figure 3-2 on page 62.

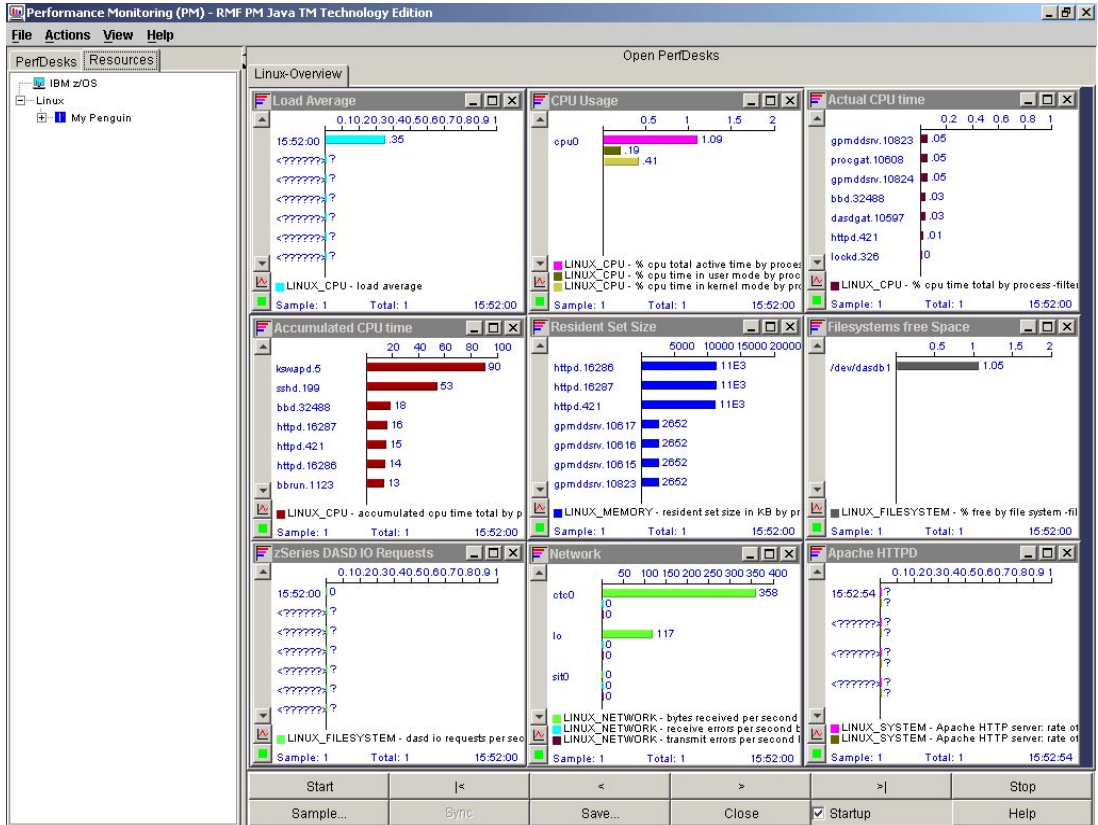


Figure 3-2 Initial RMF PM client window in Windows desktop

Figure 3-3 on page 63 shows the RMF PMS Web page <http://10.1.3.4:8803>

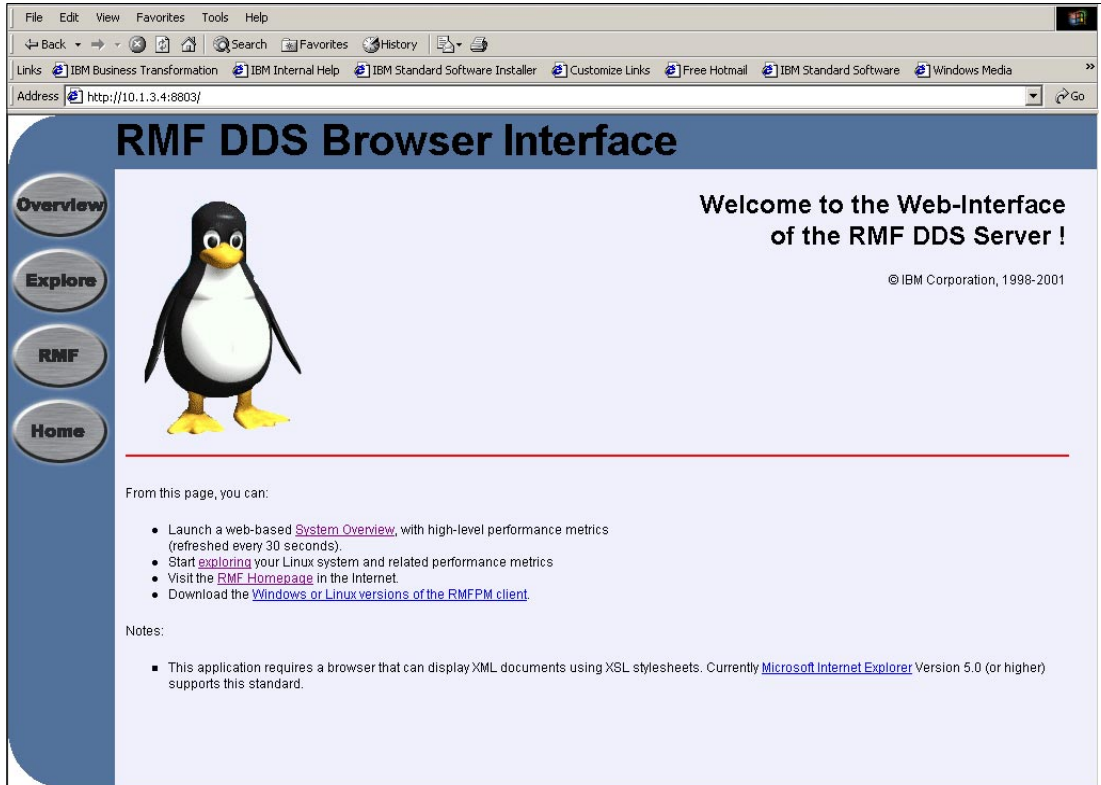


Figure 3-3 First RMF PMS Web page

Figure 3-4 shows the RMF PMS Web page that is comparable to the RMF PM client window in the Windows desktop.

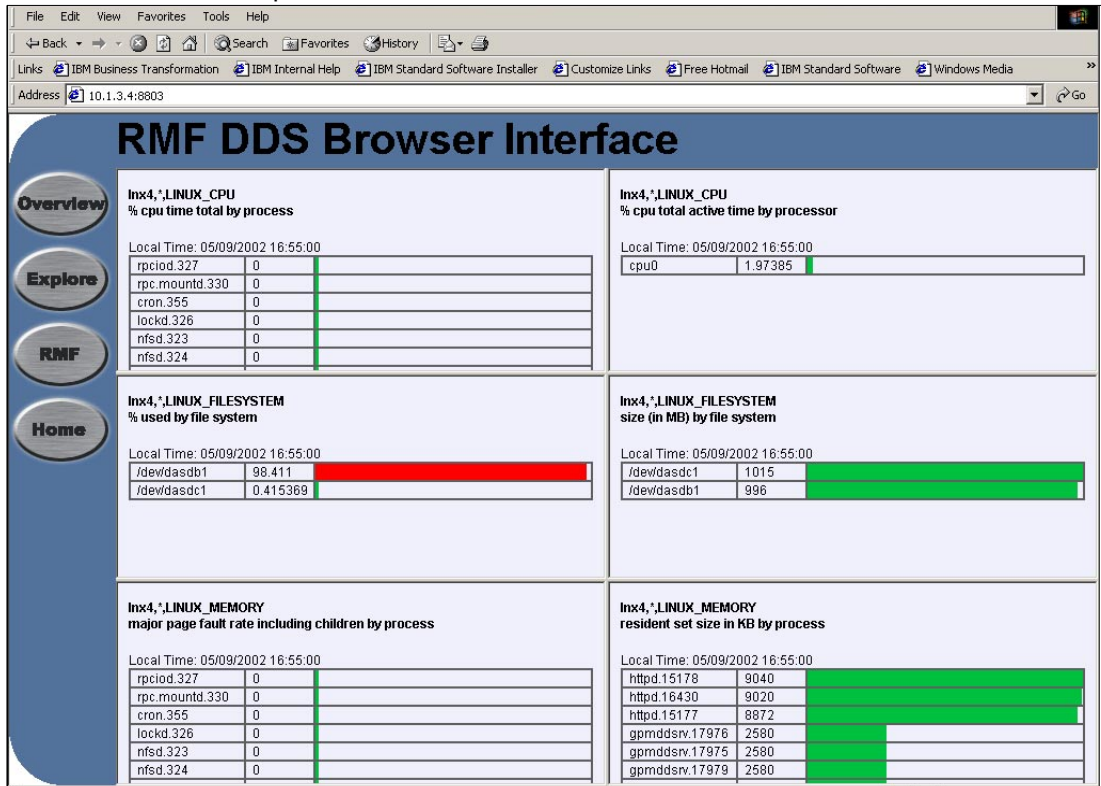


Figure 3-4 RMF PMS browser interface Web page

## 3.4 Testing RMF PMS

For the purpose of this writing, we concentrated on testing only the RMF PM client.

### 3.4.1 Creating data views and a new machine RMF PM client

Beyond creating a new machine, we also tested a mix of information on three machines – one using SuSE 2.4, another using Red Hat 2.4, and a third running SuSE 2.2.

As part of this exercise, we demonstrate the creation of data views and the creation of a new Linux image in the RMF PM client. We did not explore all the possible configurations of the RMF PM client.

When we started the RMF PM client for the first time, it showed various data view screens; we closed these screens.

Figure 3-5 illustrates the creation of a new data view. We used the following steps to create this view:

1. Click the + sign next to the My Penguin icon to expand the registry.
2. Highlight the Linux\_Filesystem resource and right-click it.
3. Enter a title in the New DataView properties box Title field; we entered SuSE 2.4 - Size filesystem. Specify a bar orientation (we chose horizontal), and click **OK**.

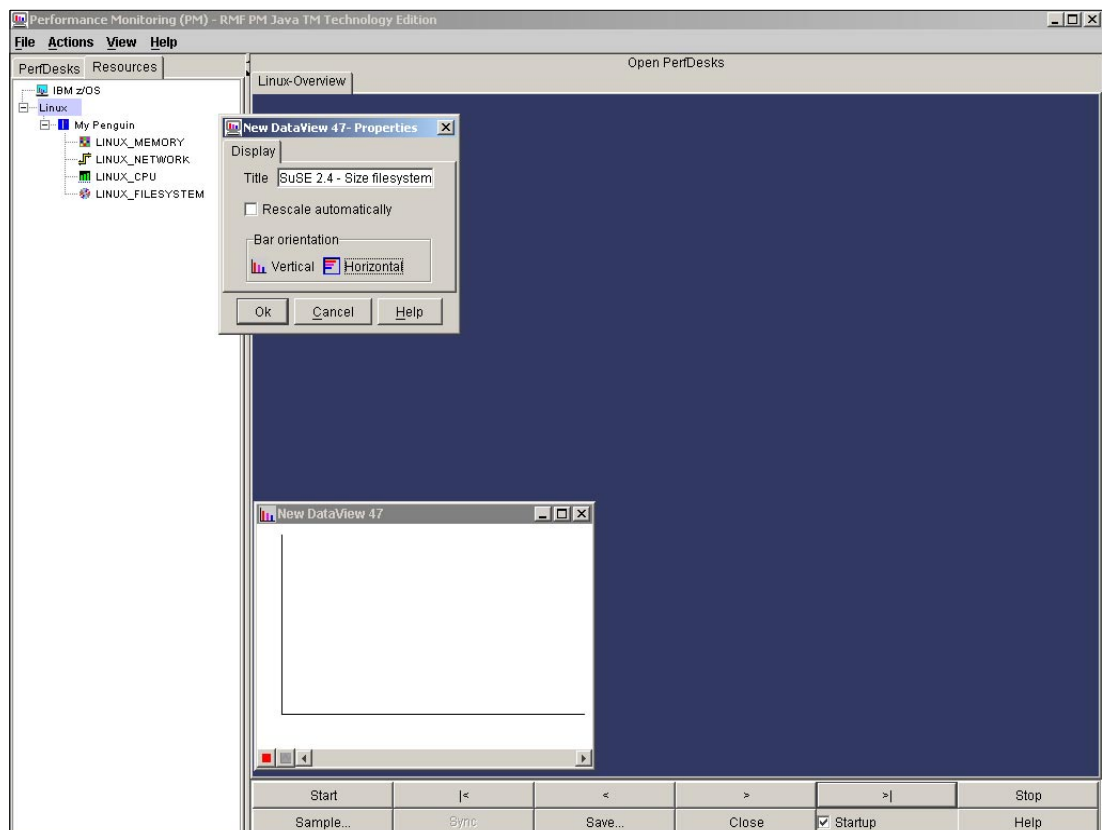


Figure 3-5 Creating a data view

4. In the Series Definition dialog box, specify the desired metric by clicking it, then click **Add**. (As shown in Figure 3-6 on page 65, we chose the “Size (in MB) by filesystem” metric.) When a new screen is presented, click **Close**.



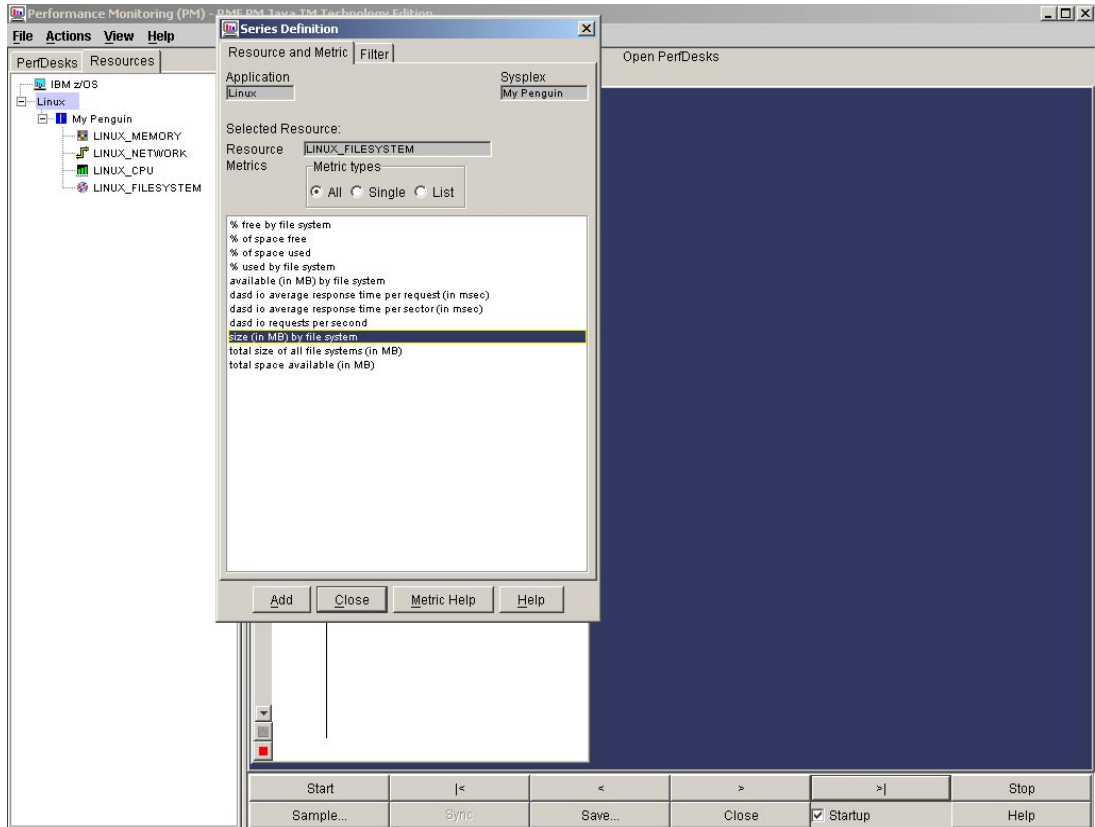


Figure 3-6 Creating a metric

5. Click the red button located in the lower left of the data view to resize the data view window.

The newly constructed data view is shown in Figure 3-7 on page 65.

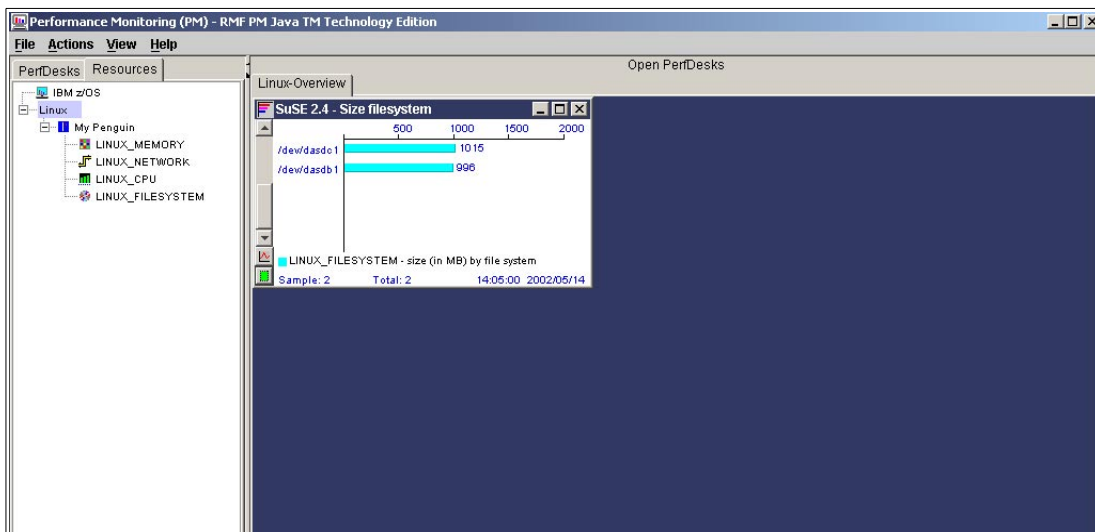


Figure 3-7 New data view completed

**Note:** The new data view was created and executed on our initial Linux machine.

Before a new Linux image was created, we created a user called rmpfms on a remote machine, and the RMF PMS was generated and started as described in 3.3, “Generating RMF PMS” on page 60.

After installation and error-free startup of the RMF PMS on the remote machine, we create a new Linux image in the RMF PM client, as shown in Figure 3-8. We did this by selecting **File -> New -> Linux image**.

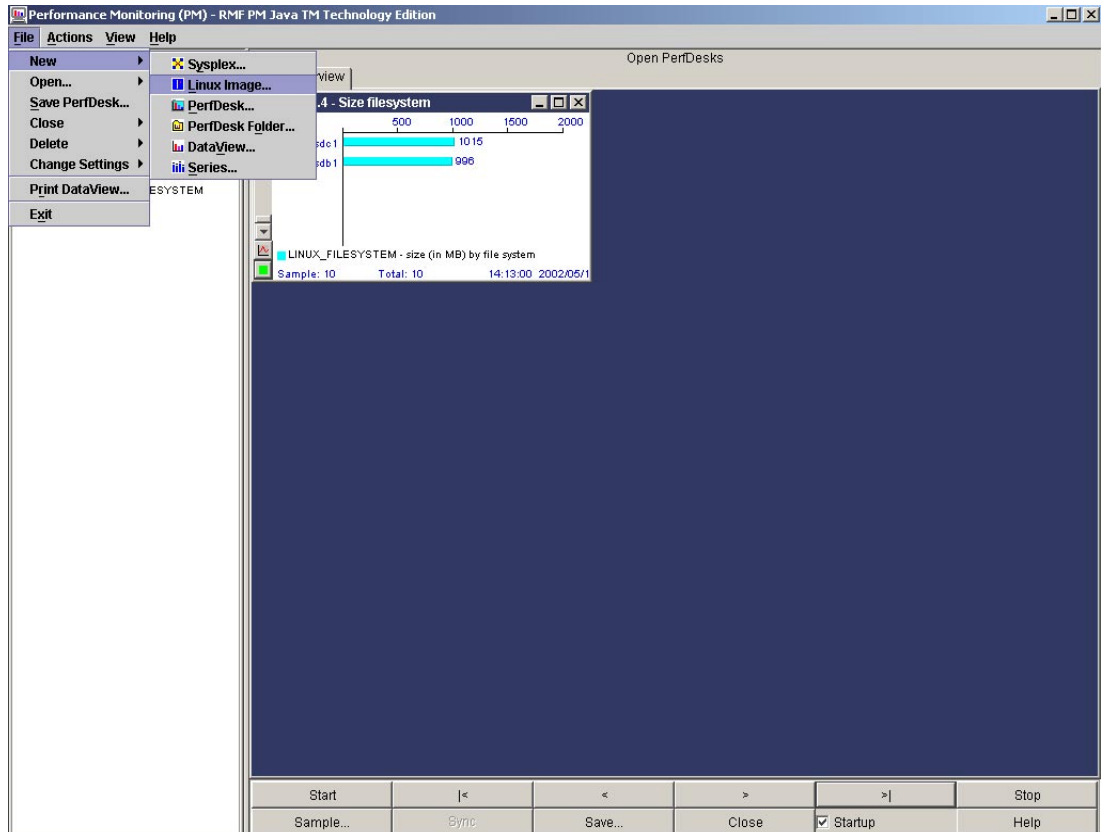


Figure 3-8 Creating a new Linux image

The Linux image was selected, and a dialog box was returned. For the Red Hat 2.4 distribution we entered information about the new Linux image, the name, IP address, and the user ID of our second machine; we then clicked **OK**. This is shown in Figure 3-9 on page 67.

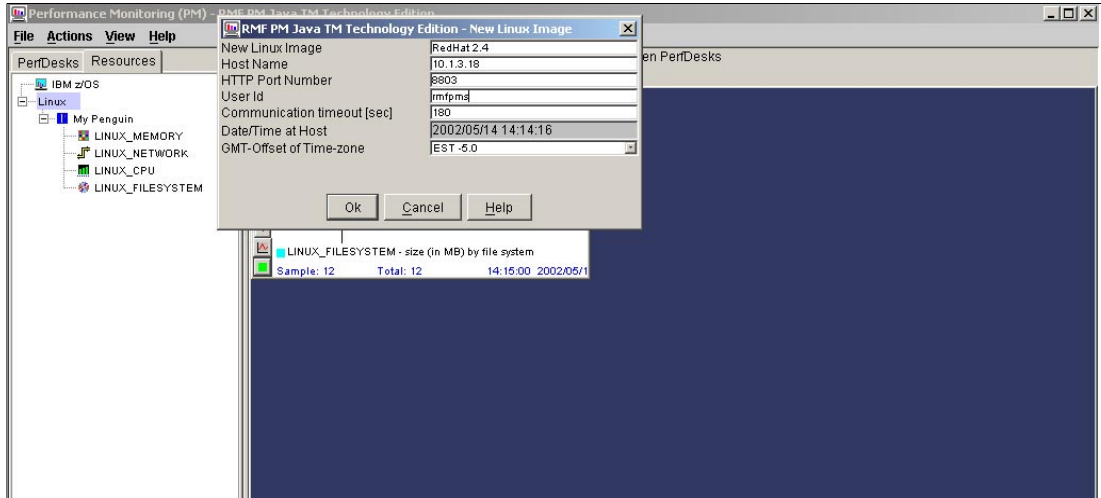


Figure 3-9 Entering data for a new Linux image

In the Red Hat 2.4 Linux image, right-click to open the Choice option.

Using the same steps we described previously, we created a new data view by clicking the RedHat 2.4 icon, right-clicking Linux\_Memory, and choosing the New DataView option.

Before we started to create the next new Linux image, we create a user called rmfpms in the remote machine and we generate and start the RMF PMS as described in 3.3, “Generating RMF PMS” on page 60. This is our last Linux image, a SuSE 2.2 distribution.

After installation and error-free startup of RMF PMS in the remote machine, a metric for the last image was added by right-clicking the SuSE 2.2 icon, right-clicking the Linux\_Cpu resource, choosing the New DataView option, and selecting the “Accumulated cpu time total by process” metric.

The complete and final example screen with all data views is shown in Figure 3-10 on page 68.

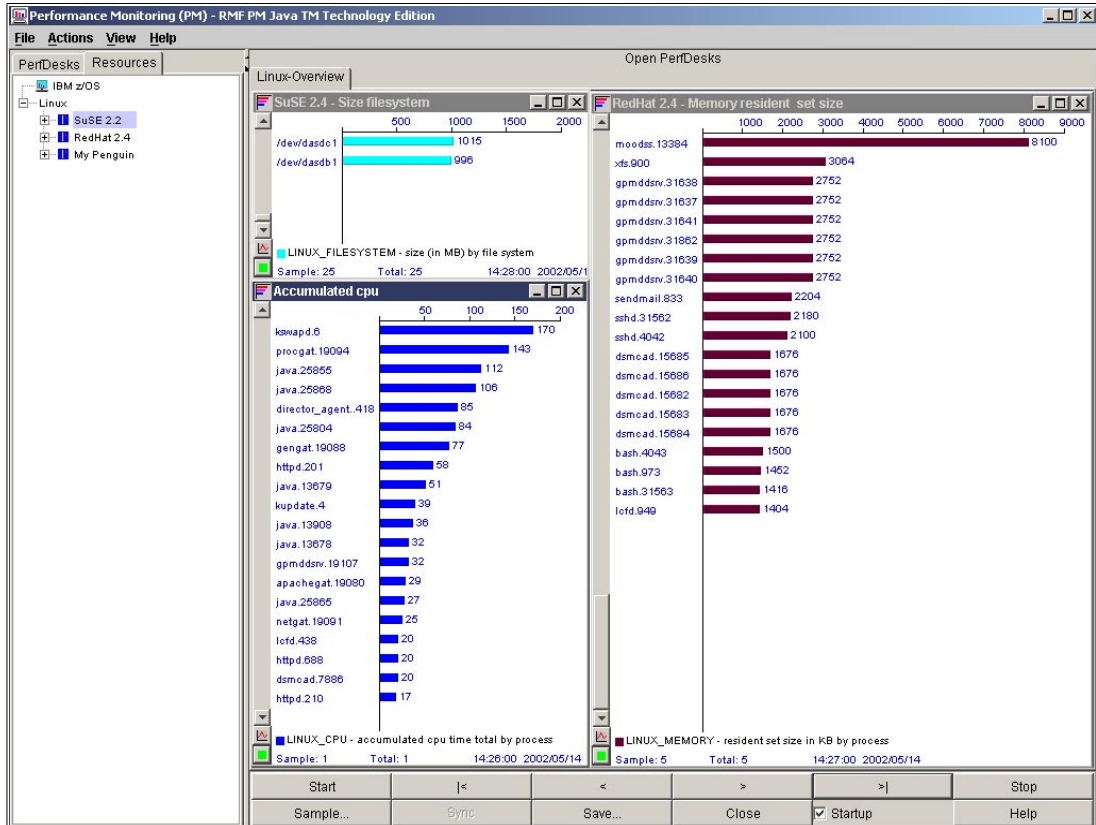


Figure 3-10 Example complete with all data views

### 3.4.2 Creating a graph in RMF PM client

The graph creation process can be achieved in two ways – either by copying an existing data view or by creating a new one. For the purposes of this writing we created a new data view.

We used the following steps to create the new data view:

1. Click the + sign next to the My Penguin icon to expand the registry.
2. Highlight the Linux\_Network resource and right-click it.
3. Enter a title in the New DataView properties box Title field; we entered My Penguin - Bytes received/transmitted per second. Select a bar orientation; we chose Vertical.
4. Click **OK**. See Figure 3-11 on page 69.

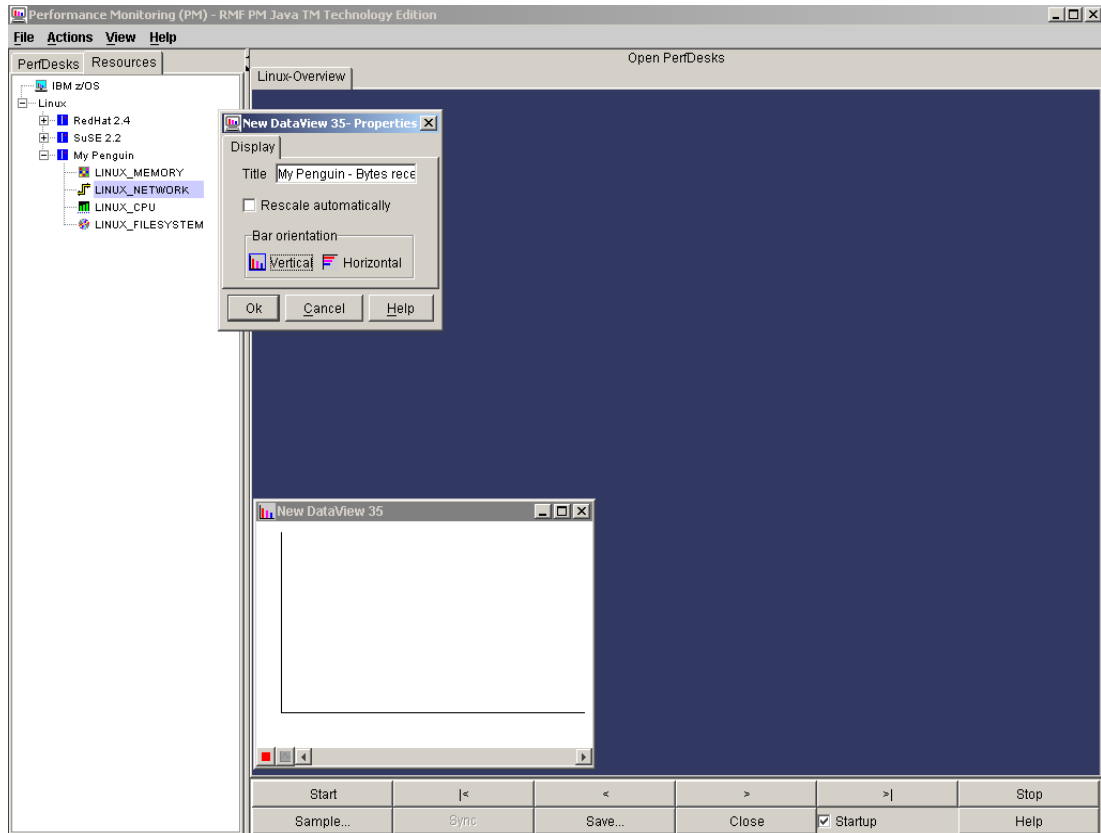


Figure 3-11 Creating a graph

5. In the Series Definition dialog box shown in Figure 3-12 on page 70, specify the metric of interest by highlighting “Bytes received per second,” and click **Add**.
6. On the new screen displayed, highlight “Bytes transmitted per second,” click **Add**, then click **Close**.

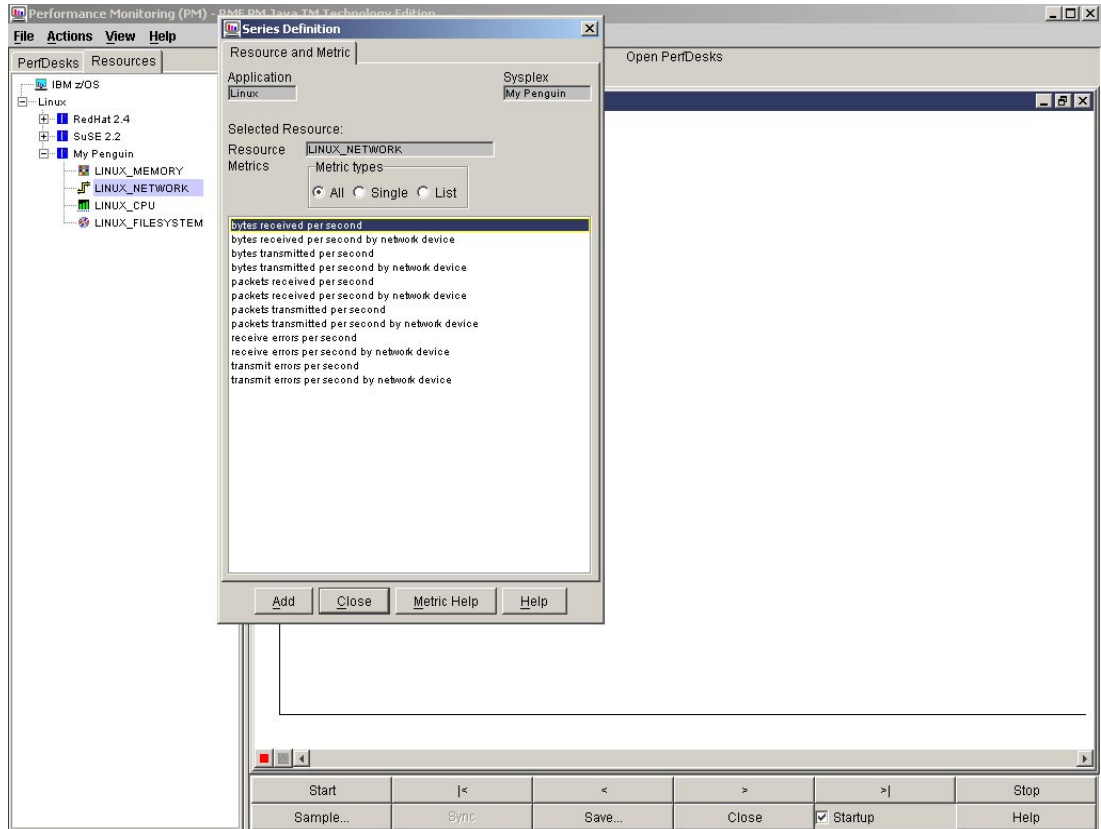


Figure 3-12 Choosing the metrics

7. On the new data view graph click the red button at the lower left of the window. After several minutes the screen in Figure 3-13 on page 71 is shown.

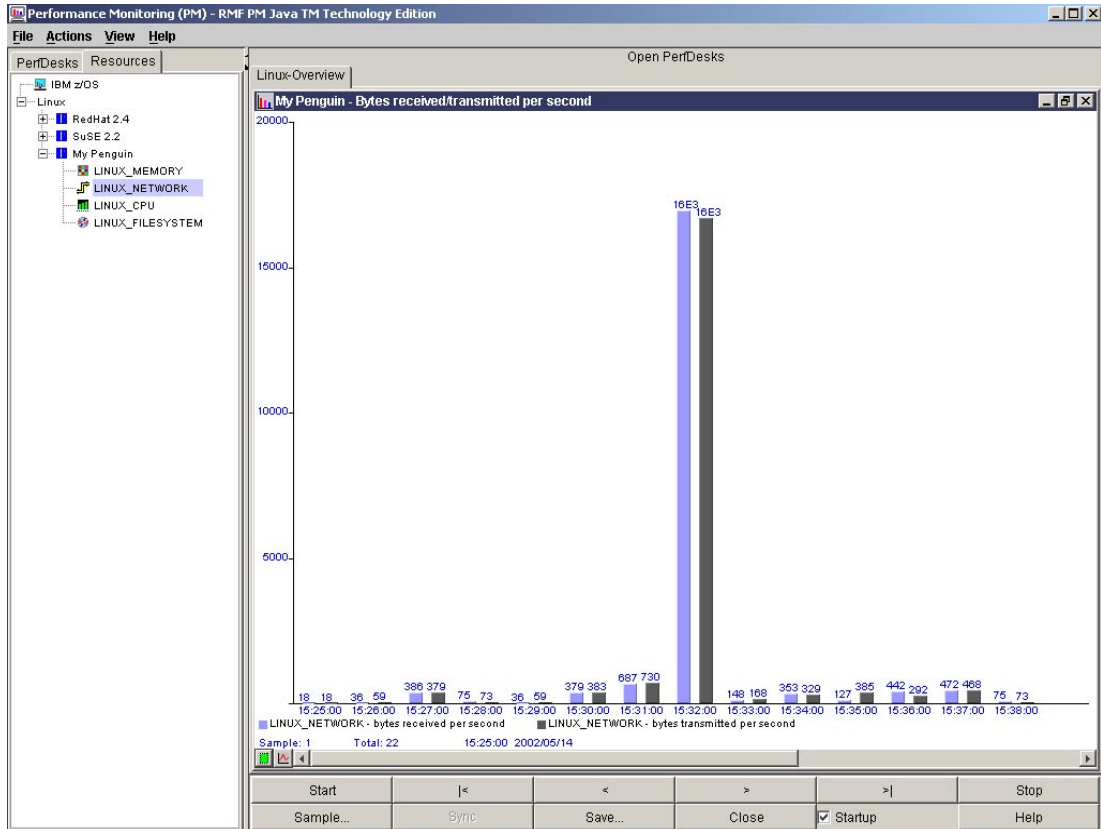


Figure 3-13 Example complete with graph

### 3.4.3 Testing the HTTP/XML interface in an RMF PMS

This section describes how to exploit the performance data in your applications. This is only pertinent if you intend to write programs.

**Note:** This is a simple HTTP URL-based API. The API is *not* an official part of any IBM product.

To explore further possibilities for using XML in your own applications, see the online material in subdirectory docs in the .rmfpm directory.

The format is of the URL:

```
http://host.domain:8803
```

To get a listing of all the metrics supported by the DDS server, we entered the following URL in our Microsoft Internet Explorer browser:

```
http://10.1.3.4:8803/gpm/config/index.xml
```

The result is shown in Figure 3-14 on page 72.

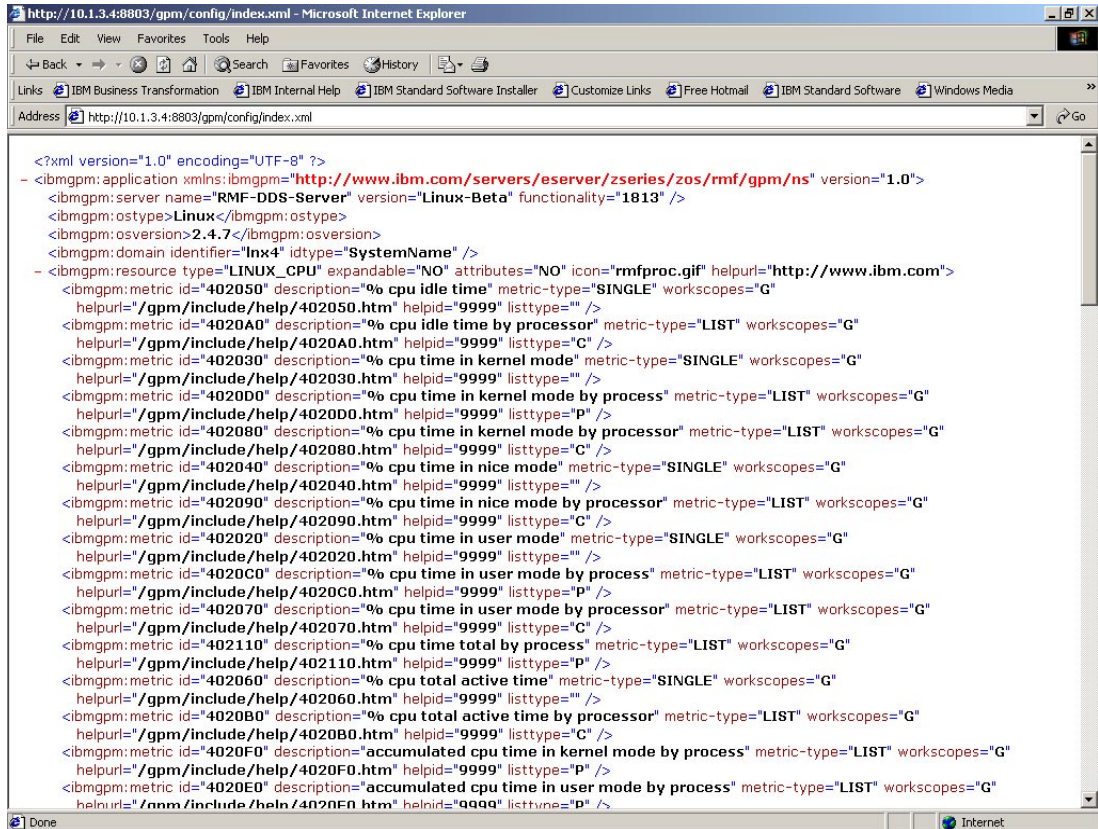


Figure 3-14 File with all metrics supported by the DDS server

In this example, we sent HTTP URL requests to the RMF PMS (DDS) to get performance data about the CPU resource, with the single counter request. The command we used was:

```
http://10.1.3.4:8803/gpm/perform/perform.xml?resource="lnx4,*,LINUX_CPU"&id=4020A0
```

The result is shown in Figure 3-15.

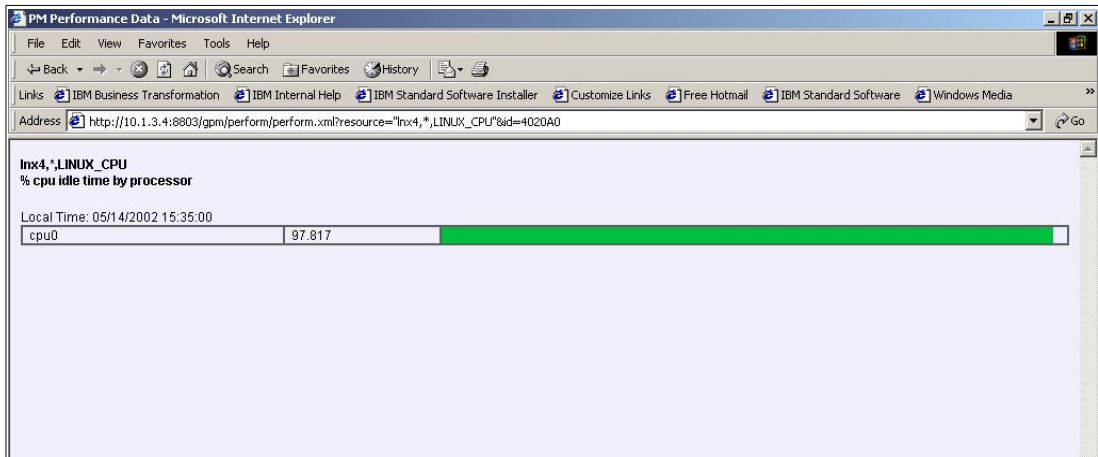


Figure 3-15 Display of single counter



In the next example, HTTP URL requests were sent to the RMF PMS (DDS) to get performance data about the Memory resource, with the list-valued counter request. The command used was:

```
http://10.1.3.4:8803/gpm/perform/perform.xml?resource="lnx4,*,LINUX_MEMORY"&id=405100
```

The result is shown in Figure 3-16.

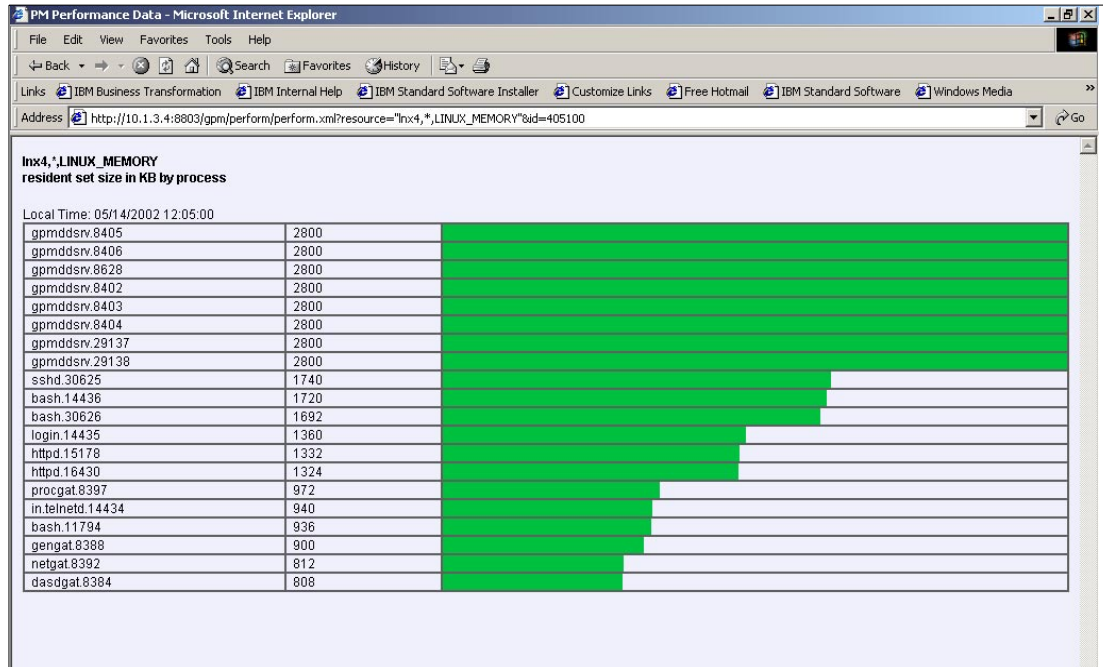


Figure 3-16 Display of list valued counter

### 3.4.4 Testing archive and unarchive of performance data

We left RMF PMS to generate logs. The archive of the logs was not placed in cron; however, it is generally recommended that you put the archive in cron to be executed automatically. Example 3-3 shows a sample archive script.

*Example 3-3 Using the rmfpm\_archive script*

```
rmfpms@lnx4:~/rmfpms > ls .rmfpms
20020513 20020514 deltas logs
rmfpms@lnx4:~/rmfpms > ./bin/rmfpm_archive
rmfpms@lnx4:~/rmfpms > ls .rmfpms
20020514 archive deltas logs
rmfpms@lnx4:~/rmfpms > ls .rmfpms/archive
20020513.tgz
```

The use of unarchive is shown in Example 3-4.

*Example 3-4 Using the rmfpm\_unarchive*

```
rmfpms@lnx4:~/rmfpms > ./bin/rmfpm_unarchive 20020513
rmfpms@lnx4:~/rmfpms > ls .rmfpms
20020513 20020514 archive deltas logs
rmfpms@lnx4:~/rmfpms > ls .rmfpms/archive
20020513.tgz
```

Example 3-5 simulates an execution of cron after an unarchive. The result is that unarchive generates a new archive that is overlapped by the old.

*Example 3-5 Test a new archive after unarchive*

---

```
rmfpms@lnx4:~/rmfpms > ls .rmfpms/archive -la
total 792
drwxr-xr-x  2 rmfpms  rmf          4096 May 14 16:26 .
drwxr-xr-x  7 rmfpms  rmf          4096 May 14 16:29 ..
-rw-r--r--  1 rmfpms  rmf       798389 May 14 12:11 20020513.tgz
rmfpms@lnx4:~/rmfpms > ./bin/rmfpms_archive
rmfpms@lnx4:~/rmfpms > ls .rmfpms/archive -la
total 792
drwxr-xr-x  2 rmfpms  rmf          4096 May 14 16:26 .
drwxr-xr-x  6 rmfpms  rmf          4096 May 14 16:32 ..
-rw-r--r--  1 rmfpms  rmf       798389 May 14 16:32 20020513.tgz
```

---

## 3.5 RMF mix environment

In this section we discuss how RMF interfaces with the following products:

- ▶ FCON/ESA (VM/ESA Full Screen Operator Console and Graphical Real-time Performance Monitor)
- ▶ z/OS RMF PM Java client
- ▶ RMF LDAP (Lightweight Directory Access Protocol) interface

### 3.5.1 FCON/ESA

FCON/ESA enables system operation in full screen mode, with real-time performance monitors that can use thresholds, exceptions, benchmarks, and more. It can provide an immediate view of system performance, or post process its own history files or VM monitor data selectively. This view is shown in Figure 3-17.

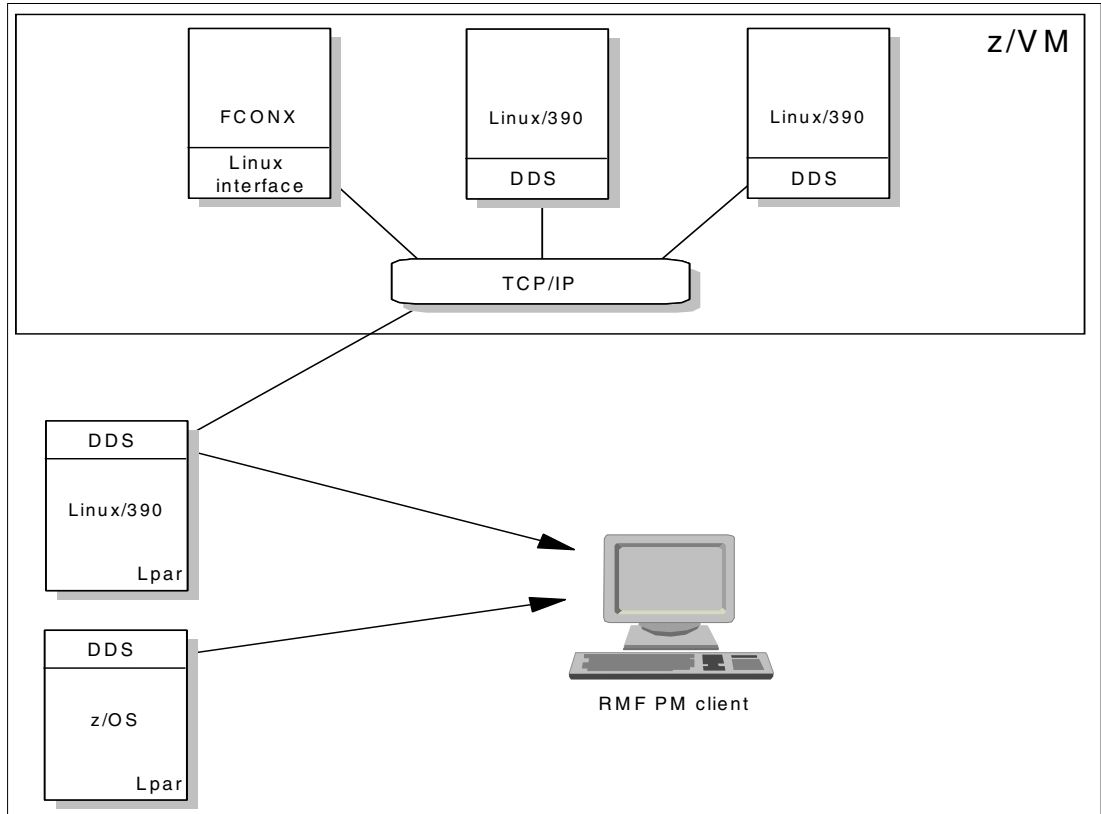


Figure 3-17 View of FCON/ESA

### 3.5.2 z/OS RMF PM Java client

The z/OS RMF PM Java client is the same as described in previous sections. Here we show evidence of the integration with z/OS; see .

For more information about z/OS RMF PM client, refer to the RMF Web page:

<http://www-1.ibm.com/servers/eserver/zseries/zos/rmf/rmfhtmls/pmweb/pmweb.htm>

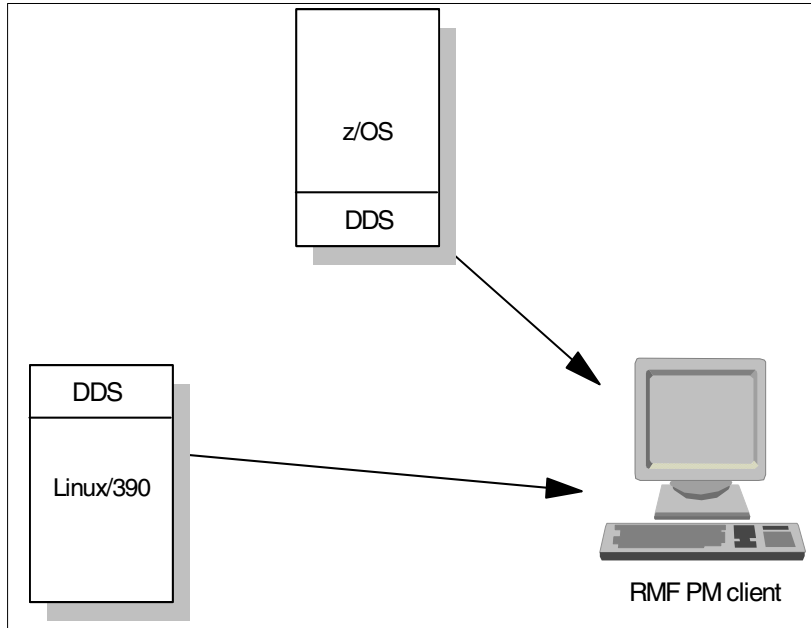


Figure 3-18 View of z/OS RMF PM Java client

### 3.5.3 RMF LDAP interface

The RMF LDAP backend works on top of distributed DDS. The RMF LDAP backend routes the incoming requests to the DDS, and through the RMF LDAP backend here it has access to the performance data. This view is shown in Figure 3-19.

For more information about the RMF LDAP interface, refer to the RMF Web page.

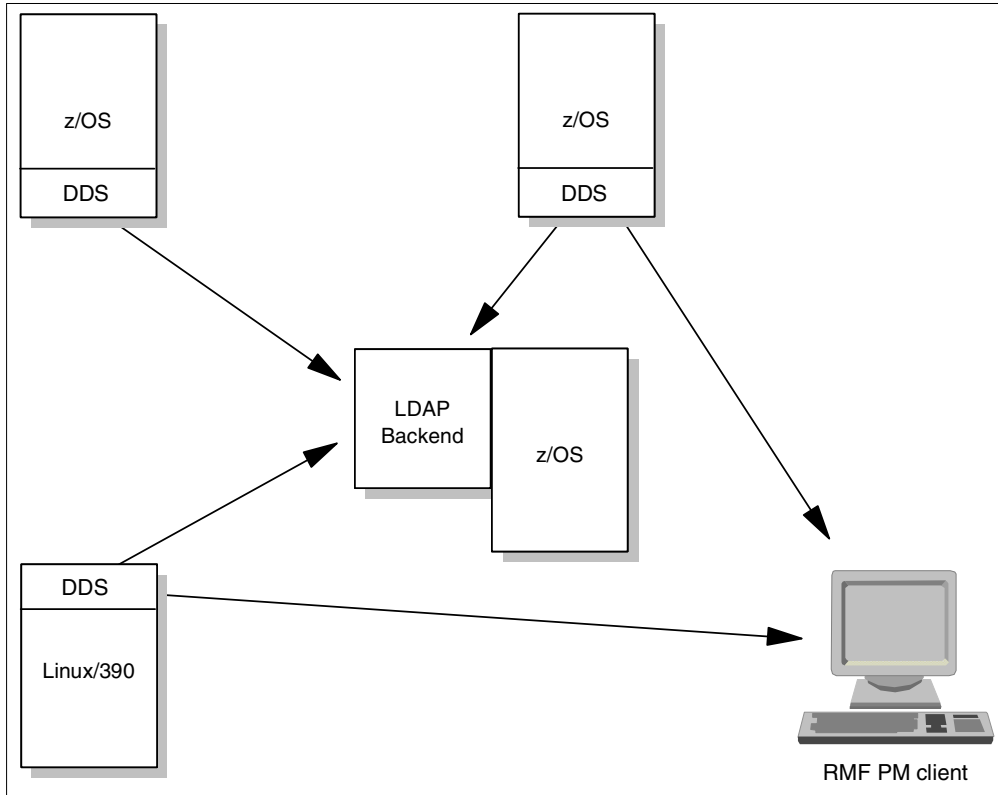


Figure 3-19 View of RMF LDAP interface





## Part 2

# Open source software

The following open source packages are described in this part:

- ▶ Tripwire
- ▶ Moodss
- ▶ Amanda
- ▶ OpenLDAP
- ▶ System Installation Suite







# Tripwire

Tripwire is an open source and commercial security, intrusion detection, damage assessment, and recovery package created by Dr. Eugene Spafford and Gene Kim in 1992 at Purdue University.

In this chapter we describe its operation and use in a Linux for zSeries environment.

The Tripwire package is available in the SuSE distribution. The version of Tripwire that we used was OpenSource Version 1.

For more information about general Tripwire usage, refer to the Tripwire Web pages:

- ▶ For the commercial version, see:  
<http://www.tripwire.com>
- ▶ For OpenSource version 2, see:  
<http://sourceforge.net/projects/tripwire/>

## 4.1 Tripwire overview

Tripwire is a policy-based program that monitors file system changes as specified in a configuration file. A database is used to help keep track of modifications that have occurred in a system.

**Note:** Like many other open source software products, Open Source Tripwire comes with no warranty.

### 4.1.1 How Tripwire works

The goal of Tripwire is to detect and notify system administrators of changed, added, and deleted files in a monitored environment.

A high-level view of Tripwire is shown in the Figure 4-1. Tripwire compares your current file system (item 1 in the diagram) with your current database (item 2) and creates change reports (item 3).

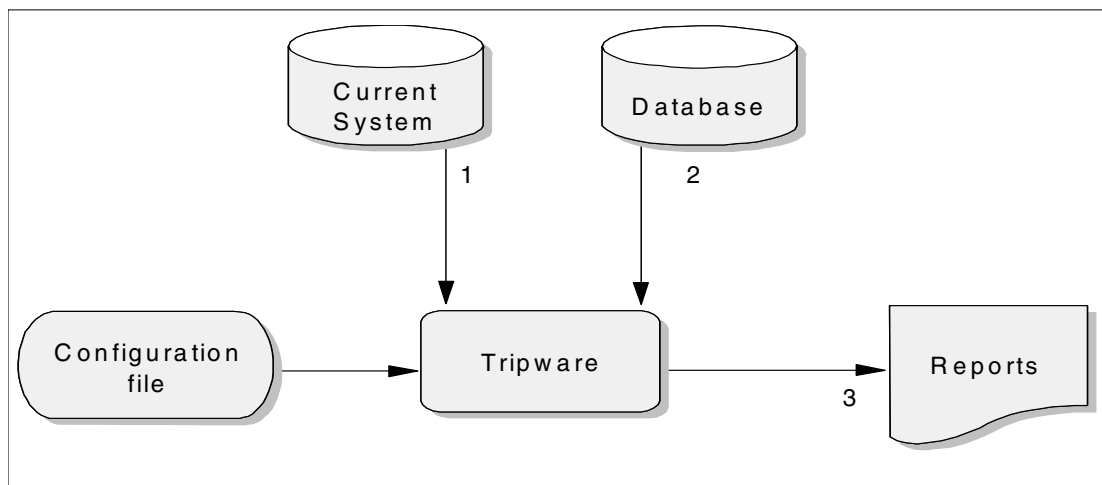


Figure 4-1 Tripwire high-level view

Tripwire uses two inputs: the configuration file and one database file that was previously created by the Tripwire program using the initialize option.

The configuration file contains a list of directories and files that must be monitored using one selection mask. The database file is generated by Tripwire, and contains a list of entries with filenames, inode attribute values, and selection masks.

Tripwire controls the following types of changes:

- ▶ Permissions
- ▶ Inode number
- ▶ Number of links (inode reference count)
- ▶ User ID of owner
- ▶ Group ID of owner
- ▶ File size
- ▶ Access timestamp
- ▶ Modification timestamp
- ▶ Inode creation/modification timestamp
- ▶ Signatures

The four modes of Tripwire operation are:

- ▶ Database initialization mode
- ▶ Integrity check mode
- ▶ Interactive database update mode
- ▶ Database update mode

## 4.2 Using Tripwire

In the following section we review Tripwire installation and use.

### 4.2.1 Planning

Before Tripwire installation, there are a few decisions that need to be made. First, decide how many times you wish to execute the checking (crontab), and which files and directories you are going to monitor. Make sure that there is enough DASD space available, and finally, decide which machines you are going to monitor.

### 4.2.2 Generating Tripwire

The Tripwire rpm file is available in SuSE SLES7 CD1 /suse/sec2/tripwire.rpm.

Once this rpm file is available in your Linux for zSeries and S/390 environment, issue the following command:

```
rpm -ivh tripwire.rpm
```

After rpm installation, the documentation files are in /usr/share/doc/packages/tripwire, and binaries files in var/adm/tripwire/bin directories.

### 4.2.3 Creating initial configuration file

In this section we look at the syntax of the configuration file.

#### Syntax of configuration file

Two types of codification exist in Tripwire: an entry format and preprocessing directives (like C preprocessor and M4 macro processor).

The configuration file is named tw.config.

The syntax of the two types is shown in Example 4-1.

*Example 4-1 Partial syntax of tw.config*

---

```
[! | =] entry [select-flags | template]
```

entry	An entry is the absolute pathname of a file or a directory.
!	Inclusive prune.
=	Exclusive prune.
select-flags	select-flags describe inode and file attributes
template	Template are predefined sets of select-flags that are commonly used by system administrators.

Possible values for the select-flags are the following:

- Ignore the following attributes

**+** Record and check the following attributes  
**p** Permission and file mode bits  
**i** Inode number  
**n** Number of links  
**u** User ID of owner  
**g** Group ID of owner  
**s** Size of file  
**a** Access timestamp  
**m** Modification timestamp  
**c** Inode creation/modification timestamp  
**0-9** Signatures

Possible values for the template are the following:

**R** [R]ead-only (+pinugsm12-ac3456789) (default)  
**L** [L]og file (+pinug-sacm123456789)  
**N** Ignore [N]othing (+pinusgsamc123456789)  
**E** Ignore [E]verything (-pinusgsamc123456789)  
**>** monotonically growing file (+pinug>-samc1233456789)

Some of the possible preprocessor values are:

@@ifhost HOSTNAME  
 @@ifnhost HOSTNAME  
 @@else  
 @@define VAR STRING  
 @@endif  
 @@ifdef VAR  
 @@include "PATHNAME"

---

In the /usr/share/doc/packages/tripwire directory is a tw.config sample called tw.conf.example.linux. This file is shown in Figure 4-2.

*Example 4-2 tw.config sample*

---

```

#
# Tripwire config-file
#
/ R
!/proc
!/var
!/root
/root/bin
!/dev
!/tmp
!/etc/mstab
!/etc/ntp.drift
!/etc/ld.so.cache
!/etc/snmpd.agentinfo
!/etc/ssh_random_seed
!/etc/mail/sendmail.st
  
```

---

## 4.2.4 Creating and updating a database

To create the initial database, use the following command:

```
tripwire -init
```

## Integrity check

An integrity check is initiated by the following command:

```
tripware
```

## Running an interactive database update

For interactive database update, use the following command:

```
tripwire -interactive
```

Once in interactive database update mode, for each deleted, added, or changed file, the user is prompted whether the entry corresponding to the file or directory should be updated.

The possible responses and their meanings are as follows:

- y** Update the specified file.
- n** Don't update the file.
- Y** Update the specified file or directory, *and* all other files or directories that share the same entry in the `tw.config` file.
- N** Ignores all files and directories corresponding to the specified entry.
- h** An inode information summary is shown.
- ?** Help information summary is shown.

## Database update example

Database updates are allowed through the following commands:

```
tripwire -update /new.file (An specific file)
```

```
tripwire -update /directory (An specific directory and your contents)
```

## Databases update considerations

In database update mode, Tripwire updates the specified files, directories, or entries in the database. The old database is saved in the `./databases` directory with the `.old` suffix. The new, updated database is also written to the `./databases` directory. The new database must be manually moved to the Tripwire database directory (`/var/adm/tripwire/db`) in initial creation or any database update.

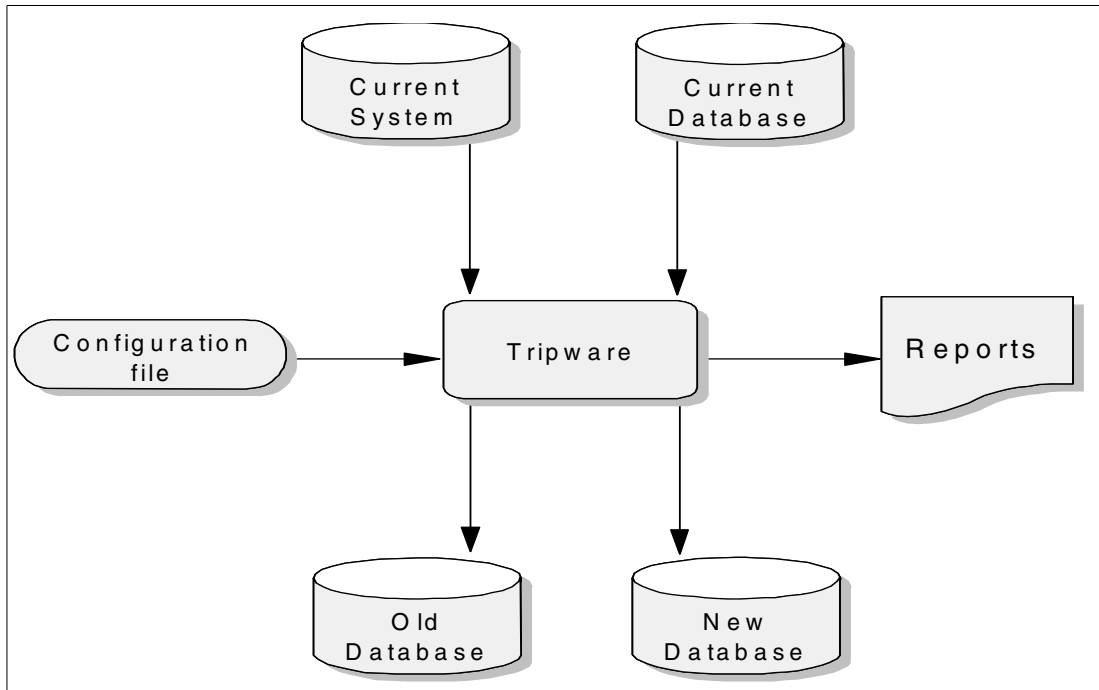


Figure 4-2 Database update

A script called `twdb_check.pl` was added as an interim mechanism to ensure database consistency. When new entries are added to the `tw.config` file, database entries may no longer be associated with the proper entry number. The `twdb_check.pl` script analyzes the database, and remaps each database entry with its proper `tw.config` entry, and the old database is saved in the `.databases` file with the `.BAK` suffix.

## 4.2.5 Testing Tripwire

Before the creation of the database file, you have to modify the configuration file because it is used as input in the Tripwire program database creation.

We will initially use the file shown in Example 4-2 on page 84 for database creation.

The file is copied to `/var/adm/tripwire` and its name is modified to `tw.config`.

```
cp /usr/share/doc/packages/tripwire/tw.conf.example.linux /var/adm/tripwire/tw.config
```

Now we can create the initial database, which will be created in the following format:

```
tw.db_hostname
```

where `hostname` is replaced with your machine hostname. In our test it was `lnx4`.

```
/var/adm/tripwire/bin/tripwire -init
```

Tripwire waits one minute to complete the initial creation of database.

The result of the execution is shown in Example 4-3 on page 87.

### Example 4-3 Initial creation of database

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
### Phase 3:  Creating file information database
###
### Warning:  Database file placed in ./databases/tw.db_lnx4.
###
###          Make sure to move this file and the configuration
###          to secure media!
###
###          (Tripwire expects to find it in '/var/adm/tripwire/db'.)
```

---

The database was created in `./databases/tw.db_lnx4` and we copied it to the specified directory:

```
cp databases/tw.db_lnx4 /var/adm/tripwire/db
```

## Running an integrity check

We modified the `/etc/crontab` file, placing a `#` in it, and included a file named `test_tw` in the `root/bin` directory, and:

```
/var/adm/tripwire/bin/tripwire
```

The result of the execution is shown in Example 4-4.

### Example 4-4 Running an integrity check

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
### Phase 3:  Creating file information database
### Phase 4:  Searching for inconsistencies
###
###          Total files scanned:          52886
###          Files added:                  1
###          Files deleted:                0
###          Files changed:                52885
###
###          After applying rules:
###          Changes discarded:            52883
###          Changes remaining:            4
###
added:  -rw-r--r-- root          14 Apr 26 10:54:06 2002 /root/bin/test_tw
changed: drwxr-xr-x root        4096 Apr 26 10:53:31 2002 /etc
changed: -rw-r--r-- root          502 Apr 26 10:53:31 2002 /etc/crontab
changed: drwxr-xr-x root        4096 Apr 26 10:54:06 2002 /root/bin
### Phase 5:  Generating observed/expected pairs for changed files
###
### Attr      Observed (what it is)      Expected (what it should be)
### =====
/etc
      st_mtime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002
      st_ctime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002

/etc/crontab
      st_size: 502                              500
      st_mtime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002
      st_ctime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002
      md5 (sig1): 0YGosX80ZEtdcXlEyFTZAO      3NKCCyVBbJ0863BV70I1QU
      snefru (sig2): 0z8pXkUuywnZLoRNSZbYQ8      1i9fjV72Y4.j1TMXFtFUST
```

```

/root/bin
      st_mtime: Fri Apr 26 10:54:06 2002      Fri Apr 26 09:51:39 2002
      st_ctime: Fri Apr 26 10:54:06 2002      Fri Apr 26 09:51:39 2002

```

---

**Note:** This integrity check does not update the database, it only reports divergences.

## Updating the database after an integrity check

There are two ways to modify a database: interactive mode and update mode.

### Interactive mode

```
/var/adm/tripwire/bin/tripwire -interactive
```

The result of the execution is shown in Example 4-5.

#### Example 4-5 Updating the database in interactive mode

---

```

### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
### Phase 3:  Creating file information database
### Phase 4:  Searching for inconsistencies
###
###                Total files scanned:          52886
###                Files added:                  1
###                Files deleted:                0
###                Files changed:               52885
###
###                After applying rules:
###                Changes discarded:           52883
###                Changes remaining:           4
###
added:  -rw-r--r-- root          14 Apr 26 10:54:06 2002 /root/bin/test_tw
----> File: '/root/bin/test_tw'
----> Update entry? [YN(y)nh?] y
changed: drwxr-xr-x root          4096 Apr 26 10:53:31 2002 /etc
changed: -rw-r--r-- root          502 Apr 26 10:53:31 2002 /etc/crontab
changed: drwxr-xr-x root          4096 Apr 26 10:54:06 2002 /root/bin
### Phase 5:  Generating observed/expected pairs for changed files
###
### Attr          Observed (what it is)          Expected (what it should be)
### =====
/etc
      st_mtime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002
      st_ctime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002
----> File: '/etc'
----> Update entry? [YN(y)nh?] y

/etc/crontab
      st_size: 502                          500
      st_mtime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002
      st_ctime: Fri Apr 26 10:53:31 2002      Wed Apr 24 11:35:40 2002
      md5 (sig1): 0YGosX80ZEtDcX1EyFTZAO      3NKCCyVBbJ0863BV70I1QU
      snefru (sig2): 0z8pXkUuynwZLoRNSZbYQ8      1i9fjv72Y4.j1TMXFtFUST
----> File: '/etc/crontab'
----> Update entry? [YN(y)nh?] y

```



```

/root/bin
    st_mtime: Fri Apr 26 10:54:06 2002      Fri Apr 26 09:51:39 2002
    st_ctime: Fri Apr 26 10:54:06 2002      Fri Apr 26 09:51:39 2002
----> File: '/root/bin'
----> Update entry? [YN(y)nh?] y

Updating entry: /root/bin/test_tw
Updating entry: /etc
Updating entry: /etc/crontab
Updating entry: /root/bin
### Updating database...
###
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
Updating: add file: /root/bin/test_tw
Updating: update file: /etc
Updating: update file: /etc/crontab
Updating: update entry: /root/bin
### Phase 3:  Updating file information database
###
### Old database file will be moved to `tw.db_lnx4.old'
###           in './databases'.
###
### Updated database will be stored in './databases/tw.db_lnx4'
###           (Tripwire expects it to be moved to '/var/adm/tripwire/db'.)
###
###
### If you changed the tw.config file, remember to run `twdb_check.pl' to
### ensure database consistency.
### See the README file for details.

```

---

**Note:** We did not copy the new database for the `/var/adm/tripwire` directory because we are going to execute update again, but the correct thing to do would be to copy it into the directory.

## Update mode

Using the same database we used in the test of interactive mode, we executed in database update mode.

```
/var/adm/tripwire/bin/tripwire -update /root/bin/test_tw /etc/crontab
```

The result of the execution is shown in Example 4-6.

### Example 4-6 Updating the database in update mode

---

```

### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
Updating: add file: /root/bin/test_tw
Updating: update file: /etc/crontab
### Phase 3:  Updating file information database
###
### Old database file will be moved to `tw.db_lnx4.old'
###           in './databases'.
###
### Updated database will be stored in './databases/tw.db_lnx4'
###           (Tripwire expects it to be moved to '/var/adm/tripwire/db'.)
###

```

---

The database was created in `./databases/tw.db_lnx4`, and we copied the database for the specified directory:

```
cp databases/tw.db_lnx4 /var/adm/tripwire/db
```

Then we again ran the integrity check for validation:

```
/var/adm/tripwire/bin/tripwire
```

The result of this integrity check is shown in Example 4-7.

*Example 4-7 Running an integrity check*

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
### Phase 3:  Creating file information database
### Phase 4:  Searching for inconsistencies
###
###                Total files scanned:          52886
###                Files added:                  0
###                Files deleted:                0
###                Files changed:               52886
###
###                After applying rules:
###                Changes discarded:           52884
###                Changes remaining:          2
###
changed: drwxr-xr-x root          4096 Apr 26 10:53:31 2002 /etc
changed: drwxr-xr-x root          4096 Apr 26 10:54:06 2002 /root/bin
### Phase 5:  Generating observed/expected pairs for changed files
###
### Attr      Observed (what it is)           Expected (what it should be)
### =====  =====
/etc
    st_mtime: Fri Apr 26 10:53:31 2002       Wed Apr 24 11:35:40 2002
    st_ctime: Fri Apr 26 10:53:31 2002       Wed Apr 24 11:35:40 2002

/root/bin
    st_mtime: Fri Apr 26 10:54:06 2002       Fri Apr 26 09:51:39 2002
    st_ctime: Fri Apr 26 10:54:06 2002       Fri Apr 26 09:51:39 2002
```

---

This error message appeared because we forgot to inform directories, too, when:

```
/var/adm/tripwire/bin/tripwire -update /etc /root/bin
```

The result of the execution is shown in Example 4-8.

*Example 4-8 Updating the database in update mode*

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
Updating: update file: /etc
Updating: update entry: /root/bin
### Phase 3:  Updating file information database
###
### Old database file will be moved to `tw.db_lnx4.old'
###           in './databases'.
###
### Updated database will be stored in './databases/tw.db_lnx4'
###           (Tripwire expects it to be moved to '/var/adm/tripwire/db'.)
###
```

---

The database was created in `./databases/tw.db_inx4` and we copied the database for the specified directory:

```
cp databases/tw.db_inx4 /var/adm/tripwire/db
```

We again ran an integrity check for validation:

```
/var/adm/tripwire/bin/tripwire
```

The result of this integrity check is shown in Example 4-9.

*Example 4-9 Running an integrity check*

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
### Phase 3:  Creating file information database
### Phase 4:  Searching for inconsistencies
###
###          Total files scanned:          52886
###          Files added:                  0
###          Files deleted:                0
###          Files changed:                52886
###
###          After applying rules:
###          Changes discarded:            52886
###          Changes remaining:           0
###
```

---

Now, the database is correct with the modifications.

## Altering the `tw.config`

We decided to modify `tw.config` because it will be a current procedure in an installation site.

We removed the `test_tw` file, modified the `/root/bin` entry for `!/root/bin` in the `tw.config` file, and added a new file called `new_test_tw`.

Then we ran an integrity check for validation:

```
/var/adm/tripwire/bin/tripwire
```

The result of this integrity check is shown in Example 4-10.

*Example 4-10 Running an integrity check*

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
### Phase 3:  Creating file information database
### Phase 4:  Searching for inconsistencies
###
###          Total files scanned:          52884
###          Files added:                  0
###          Files deleted:                2
###          Files changed:                52884
###
###          After applying rules:
###          Changes discarded:            52886
###          Changes remaining:           2
###
deleted: drwxr-xr-x root          4096 Apr 26 10:54:06 2002 /root/bin
deleted: -rw-r--r-- root          14 Apr 26 10:54:06 2002 /root/bin/test_tw
```

---

Of the alterations made, only the removal of the test\_tw file was identified (in database exist the information for test\_tw), because we removed the control of the directory /root/bin.

We executed database update mode again:

```
/var/adm/tripwire/bin/tripwire -update /root/bin /root/bin/test_tw
```

The result of the execution is shown in Example 4-11.

*Example 4-11 Updating the database in update mode*

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
Updating: delete file: /root/bin
Updating: delete file: /root/bin/test_tw
### Phase 3:  Updating file information database
###
### Old database file will be moved to `tw.db_lnx4.old'
###           in './databases'.
###
### Updated database will be stored in './databases/tw.db_lnx4'
###           (Tripwire expects it to be moved to '/var/adm/tripwire/db'.)
###
```

---

Before the copy of the new database, we decided to execute the script twdb\_check.pl because an alert was shown in Example 4-5 on page 88 and also because of the observations mentioned in “Databases update considerations” on page 85.

```
/var/adm/tripwire/bin/twdb_check.pl databases/tw.db_lnx4 /var/adm/tripwire/tw.config
```

The result of the execution is shown in Example 4-12

*Example 4-12 Error in first execution of the twdb\_check.pl script*

---

```
In string, @dbaseversion now must be written as '@dbaseversion' at
/var/adm/tripwire/bin/twdb_check.pl line 77, near '^@dbaseversion'
Execution of /var/adm/tripwire/bin/twdb_check.pl aborted due to compilation errors.
```

---

We corrected the error and re-executed the command:

```
/var/adm/tripwire/bin/twdb_check.pl databases/tw.db_lnx4 /var/adm/tripwire/tw.config
```

No error message was received.

We copied the database for the specified directory:

```
cp databases/tw.db_lnx4 /var/adm/tripwire/db
```

Then we re-ran the integrity check for validation:

```
/var/adm/tripwire/bin/tripwire
```

The result of this integrity check is shown in Example 4-13.

*Example 4-13 Running an integrity check*

---

```
### Phase 1:  Reading configuration file
### Phase 2:  Generating file list
### Phase 3:  Creating file information database
### Phase 4:  Searching for inconsistencies
###
###
###           Total files scanned:          52884
###           Files added:                  0
```

```
###           Files deleted:           0
###           Files changed:           52884
###
###           After applying rules:
###           Changes discarded:       52884
###           Changes remaining:       0
###
```

---

## 4.2.6 Tips

Following are some handy hints and tips that we can pass on from our experience installing and testing Tripwire:

- ▶ In an environment using shared DASD R/O, there is no need to do a check for this DASD.
- ▶ There is only one copy of the Tripwire software. This being a test, we kept the database in the directory; however, for security reasons, we advise that the database be placed in a secure directory to prevent unauthorized alterations.
- ▶ The execution of integrity checks is slow. It is best to execute them during periods of low system use, or to execute with option `-i 2` (skip the Snefru signatures), as described in FAQ for Tripwire.
  - If alterations have been made in some files, the files can be recovered selectively from the last backup; it is not necessary to do a full restore.
- ▶ Since the codification of `tw.config` allows the use of preprocessing directives (`@ @ifhost HOSTNAME`), the preprocessing directives can be used to only have one `tw.config` that will be shared with all machines.
- ▶ The processes of checking and updating can be automated.





## Moodss

Modular Object-Oriented Dynamic Spread Sheet (moodss) is an open source system monitoring tool that was written by Jean-Luc Fontaine.

In the lab we ran moodss using RedHat Linux 7.2 with kernel 2.4.19 for S/390 installed on a VM-guest, but you can use any other distribution of Linux for S/390.

The version of moodss we used was #15.13.

**Note:** Like most other open source software, open source moodss comes with no warranty.

## 5.1 Prerequisites for moodss installation

Before installing the moodss software, check that the software required by moodss is installed on your Linux S/390 system. The following products should be installed:

- ▶ Tcl/Tc version 8.3.1 or above  
<http://tcl.sourceforge.net/>
- ▶ tkTable version 2.7 or above  
<http://tktable.sourceforge.net/>
- ▶ The latest BLT library version 2.4u or above
- ▶ If you want to send e-mails when a threshold is reached, you need:
  - tcllib version 1.0 or above  
<http://tcllib.sourceforge.net/>
- ▶ If you want to develop your own moodss modules in a language other than Tcl, you need:
  - tclperl library
  - tclpython or tclpython2 libraries  
<http://jfontain.free.fr>

You can get more information about the products listed by checking their respective Web sites. Most of the prerequisites are included in the Redhat 7.2 distribution for S/390, so the installation of this product is quite straightforward. However, for one product there was no RPMS S/390 available, so we had to get the source and build the RPMS ourselves.

### Tcl

Tcl-8.3.3 is shipped with the RedHat 7.2 distribution. It resides on disc1. The installation is straightforward; just use this rpm command:

```
rpm -ivh tcl-8.3.-65.s390.rpm
```

### Tk

tk-8.3.3 is shipped with the RedHat 7.2 distribution. It resides on disc1.

```
rpm -ivh tk-8.3.3-65.s390.rpm
```

### Tktable

Tktable is not included in the RedHat 7.2 distribution. There are a few steps to perform to install this product. First, download the source code of Tktable 2.7 from:

```
http://sourceforge.net/projects/tktable/  
http://sourceforge.net/project/showfiles.php?group\_id=11464&release\_id=41778
```

Choose the Tktable2.7.tar.gz file, which can be used for any architecture. Store it in the /usr/src/redhat/SOURCES/ sublibrary. Then download the specification file for Tktable from:

```
http://jfontain.free.fr/
```

Store the specification file in the /usr/src/redhat/SPECS/ sublibrary.

You are now ready to build the appropriate RPM file for your S/390 environment:

```
rpm -bb tktable-2.7-1.spec
```

When executing the command you will see the messages shown in Example 5-1.



### Example 5-1 Building the RPM file for tktable 2.7.1

---

```
[root@lnx19 SPECS]# rpm -bb tktable-2.7-1.spec
Executing(%prep): /bin/sh -e /var/tmp/rpm-tmp.98244
+ umask 022
+ cd /usr/src/redhat/BUILD
+
# $Id: tktable.spec,v 1.13 2001/07/10 20:29:00 jfontain Exp $
AutoReqProv: no
Requires: tk >= 8.3.1
Buildroot: /var/tmp/%{name}%{version}

%description
Tktable provides a table/matrix widget for Tk programs. Features:
multi-line cells, embedded windows, variable width columns/height rows
(interactively resizable), scrollbar support, tag styles per row,
column or cell, in-cell editing, works on UNIX, Windows and MacIntosh,
Unicode support with Tk 8.1 and above.

%prep

%setup -q -c

%build
cd Tktable%{version}/unix
./configure --with-tcl=%{directory}/lib --with-tk=%{directory}/lib
sed -e 's,$(SHLIB_LD) -o $@ $(OBJS) $(SHLIB_LD_LIBS) $(TK_LD_SEARCH_FLAGS),$(SHLIB_LD) -o
$@ $(OBJS),;' Makefile > $$; mv $$ Makefile
make TBL_CFLAGS=-O2

%install
cd Tktable%{version}
DIRECTORY=$RPM_BUILD_ROOT%{directory}/lib/%{name}%{version}
install -d $DIRECTORY
install unix/Tktable.so $DIRECTORY/Tktable.so.%{version}
install -m 644 unix/pkgIndex.tcl library/tkTable.tcl $DIRECTORY
install -d $RPM_BUILD_ROOT%{directory}/man/mann
install -m 644 doc/tkTable.n $RPM_BUILD_ROOT%{directory}/man/mann
install -m 644 ChangeLog README.txt README.blt TODO.txt UPGRADING.txt license.txt ..
install -d ../doc
install -m 644 doc/tkTable.html ../doc

PreReq: rpmlib(PayloadFilesHavePrefix) <= 4.0-1 rpmlib(CompressedFileNames) <= 3.0.4-1
Requires(rpmlib): rpmlib(PayloadFilesHavePrefix) <= 4.0-1 rpmlib(CompressedFileNames) <=
3.0.4-1
Requires: tk >= 8.3.1
Wrote: /usr/src/redhat/RPMS/s390/tktable-2.7-1.s390.rpm
Executing(%clean): /bin/sh -e /var/tmp/rpm-tmp.5533
+ umask 022
+ cd /usr/src/redhat/BUILD
+ cd tktable-2.7
+ rm -rf /var/tmp/tktable2.7
+ exit 0
[root@lnx19 SPECS]#
```

---

Once the file tktable-2.7-1.s390.rpm is built and stored in the /usr/src/redhat/RPMS/s390/ sublibrary, you are ready to install tktable:

```
rpm -ivh tktable-2.7-1.s390.rpm
```

**Note:** Tktable 2.7 needs XFree86-devel 4 or higher as a prerequisite!

## BLT library

blt-2.4u is shipped with the RedHat 7.2 distribution. It resides on disc 2.

```
rpm -ivh blt-2.4u-7.s390.rpm
```

## 5.2 Installing moodss

In this section we describe the installation process for moodss, from downloading the source to building the RPM packages.

### 5.2.1 Download the source code

Because Moodss was designed for platforms other than S/390, there are no RPM files for S/390 available. You therefore have to download the source code of moodss, and build the RPM files yourself.

Download the moodss source code moodss-15.13.tar.bz2 and the specification file moodss-15.13-1.spec from:

```
http://jfontain.free.fr/
```

Store the source code in /usr/src/redhat/SOURCE and the specification file in /usr/src/redhat/SPEC.

**Important:** Edit the moodss-15.13-1.spec file and delete the following line:

```
Icon: moodss.gif
```

Otherwise, the build will fail.

### 5.2.2 Build RPM packages

You are now ready to generate an RPM file for S/390 with the following command:

```
rpm -bb moodss-15.13-1.spec
```

This can take a few minutes.

When executing the command you will see the messages shown in Example 5-2.

*Example 5-2*

```
[root@lnx19 SPECS]# rpm -bb moodss-15.13-1.spec
+ umask 022
+ cd /usr/src/redhat/BUILD
+ cd /usr/src/redhat/BUILD
+ rm -rf moodss-15.13
+ tar -xf /usr/src/redhat/SOURCES/moodss-15.13.tar
+ cd moodss-15.13
++ /usr/bin/id -u
+ '[' 0 = 0 ']'
+ /bin/chown -Rhf root .
+ umask 022
```

```

+ cd /usr/src/redhat/BUILD
+ cd moodss-15.13
+ DOCDIR=/var/tmp/moodssmoomps-15.13/usr/share/doc/moodss-15.13
+ export DOCDIR
+ rm -rf /var/tmp/moodssmoomps-15.13/usr/share/doc/moodss-15.13
+ /bin/mkdir -p /var/tmp/moodssmoomps-15.13/usr/share/doc/moodss-15.13
+ cp -pr README LISEZMOI CHANGES INSTALL DEVELOPMENT COPYRIGHT BUGS TODO documentation/
/var/tmp/moodssmoomps-15.13/usr/share/doc/moodss-15.13
+ exit 0
Finding Provides: (using /usr/lib/rpm/find-provides)...
Finding Requires: (using /usr/lib/rpm/find-requires)...
Provides: libfilesystem.so.1.0 liblogging.so.1.0 libnetwork.so.1.4
PreReq: rpmlib(PayloadFilesHavePrefix) <= 4.0-1 rpmlib(CompressedFileNames) <= 3.0.4-1
Requires(rpmlib): rpmlib(PayloadFilesHavePrefix) <= 4.0-1 rpmlib(CompressedFileNames) <=
3.0.4-1
Requires: tk >= 8.3.1 blt >= 2.4u tktable >= 2.7 ld.so.1 libc.so.6 /usr/bin/wish
libc.so.6(GLIBC_2.0) libc.so.6(GLIBC_2.1.3)
Processing files: moomps-1.4.1-1
Executing(%doc): /bin/sh -e /var/tmp/rpm-tmp.59094
Wrote: /usr/src/redhat/RPMS/s390/moodss-15.13-1.s390.rpm
Wrote: /usr/src/redhat/RPMS/s390/moomps-1.4.1-1.s390.rpm
Executing(%clean): /bin/sh -e /var/tmp/rpm-tmp.44714
+ umask 022
+ cd /usr/src/redhat/BUILD
+ cd moodss-15.13
+ rm -rf /var/tmp/moodssmoomps-15.13
+ exit 0
[root@lnx19 SPECS]#

```

---

Now the file moodss-15.13-1.s390.rpm is built and stored in the /usr/src/redhat/RPMS/s390/ sublibrary.

### 5.2.3 Install

You can now install moodss with the following command:

```
rpm -ivh moodss-15.13-1.s390.rpm
```

Moodss is now ready for use.

## 5.3 Using moodss

In this section we describe how to use moodss and its options to monitor your system.

### 5.3.1 Starting moodss

You have to start an X-Window client before starting moodss. In the lab We used Hummingbird Exceed version 6.2 from Hummingbird Communications Ltd in the lab. You can get more information about this product from:

<http://www.hummingbird.com>

Log on to your Linux system as root. Check the IP-address of the workstation that you want to use moodss on. Export this IP-address (in our example it is 9.12.6.147) and then start moodss, as shown in Example 5-3.

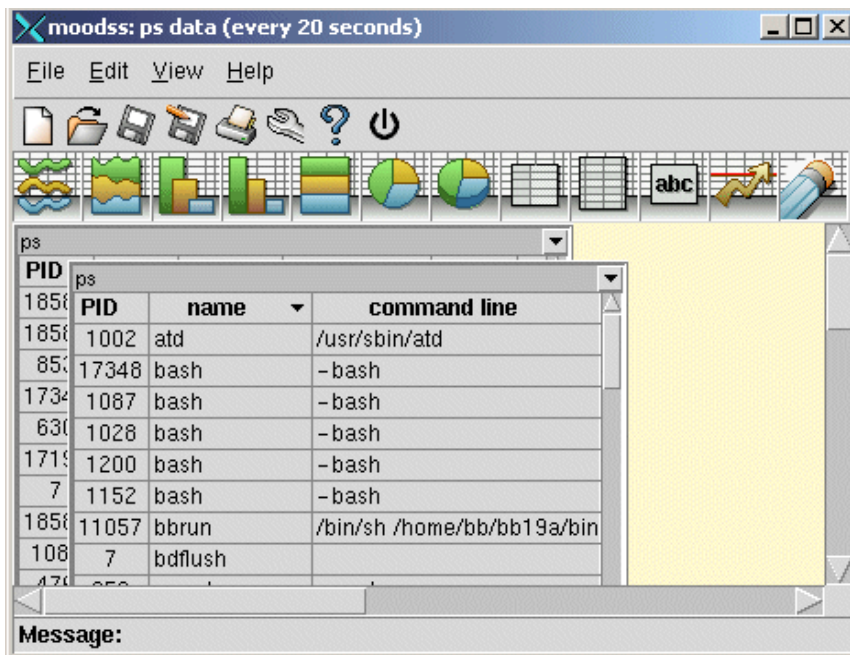
*Example 5-3 Launch moodss*

```
[root@lnx18 /]# echo $DISPLAY  
  
[root@lnx18 /]# export DISPLAY=9.12.6.147:0  
[root@lnx18 /]# echo $DISPLAY  
9.12.6.147:0  
[root@lnx18 /]# moodss random &  
[1] 17631  
[root@lnx18 /]#
```

Once the modules are loaded and initialized, the message in Figure 5-1 is displayed.



*Figure 5-1 Initial moodss window*



*Figure 5-2 Moodss is now initialized and the ps module is displayed*

Figure 5-2 shows moodss running with the ps module displayed. In the window title you can see which module is loaded and the time interval when the data is refreshed. The main menu includes four choices, each with a pull-down menu. The actions available are briefly described in this section. For more detailed information see the moodss manual on the Web site:

<http://jfontain.free.fr>

► **File**

- New** Save all unsaved change
- Open** Open an existing saved configuration
- Save** Save the current configuration for further use
- Save as** Save as above, choose file name for the configuration

## Modules

- Load** Load dynamically new modules to the already displayed modules
- Manage** See all loaded modules, reload or unload them, view its configuration
- Print** Print the current screen, WYSIWYG
- Exit** Quit moodss

### ► Edit

- Thresholds** The threshold menu, see “Thresholds” on page 108
- Configuration** Shows the configuration dialog box
- New** Allows the creation of empty viewers of any type
- Preference** Shows the preferences dialog box

### 8. View

- Refresh** Refresh the display of all loaded modules now
- Poll Time** Adjust the poll time interval, see chapter 6.3.4
- Trace** Opens an extra window containing the latest 20 messages (error, info, and so forth)
- Tool Bar** Shows or hides the tool bar

### 9. Help

- Global** Launches the complete help manual with an embedded HTML viewer
- Modules** Displays the help for one module
- About** About moodss

Now you can select additional modules to be launched. Select **File -> Modules-> Load**.

The message shown in Figure 5-3 will appear.

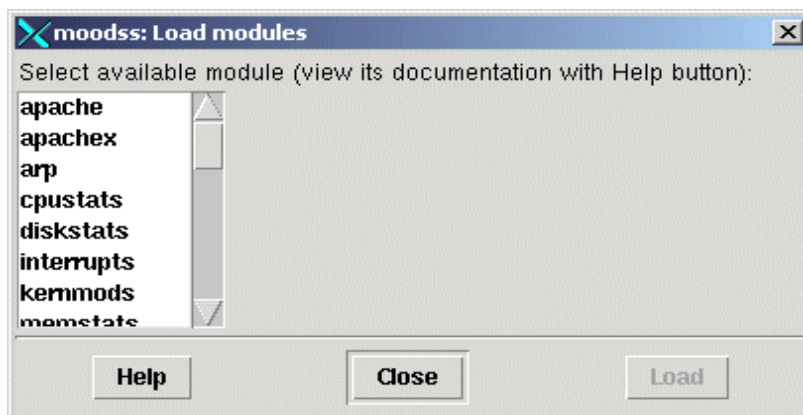


Figure 5-3 All available modules are displayed

Select a new module and press the **LOAD** button. The new module will now be started and displayed on the moodss screen. You can start as many modules as you like. Following (in alphabetic order) is a list of modules that are available:

- **apache** Displays the statistic of an Apache WWW server, like the number of accesses, amount of data served, CPU load, up time, average number of requests and bytes per second, and so forth.

- ▶ **apachex** Displays an extended statistic of an Apache WWW server. The data is displayed in 3 tables.
- ▶ **arp** Displays the network Address Resolution Protocol cache of the system.
- ▶ **cpustats** Displays the processor activity of the system.

*Figure 5-4 The cpustats module*

- ▶ **diskstats** Displays the disk statistics of the system. (The module for S/390 architecture is not yet tested.)
- ▶ **interrupts**

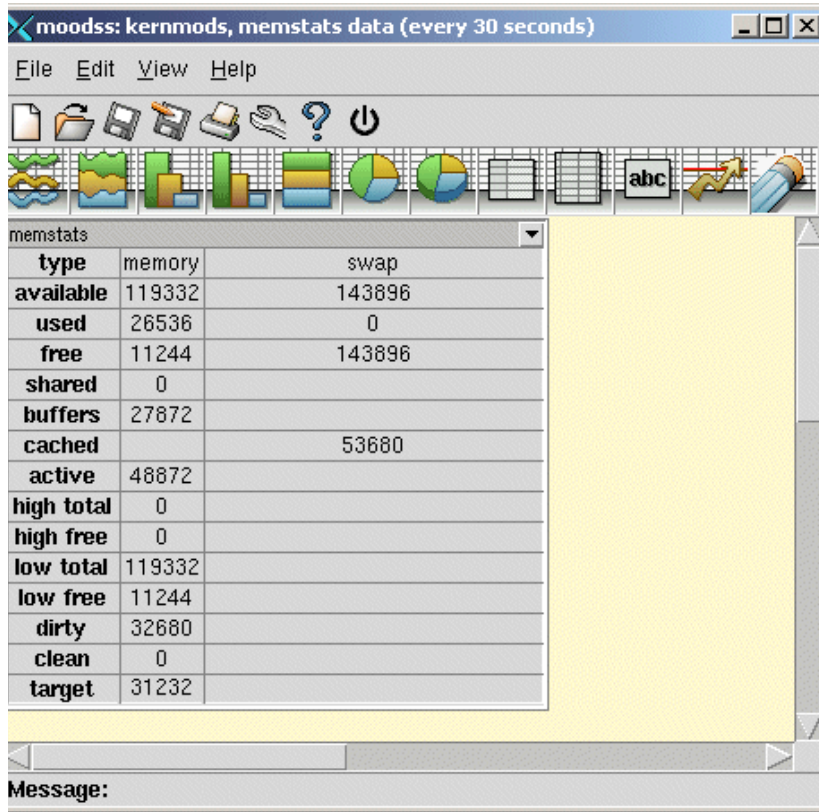


Figure 5-6 The memstat module

- ▶ **minimal** The minimal module is a skeleton module. It is used as a starting point for developing new modules.
- ▶ **Minimal** The Perl version of the minimal module.
- ▶ **minipy** The Python version of the minimal module.
- ▶ **mounts** Displays the mounted file systems of your Linux system.

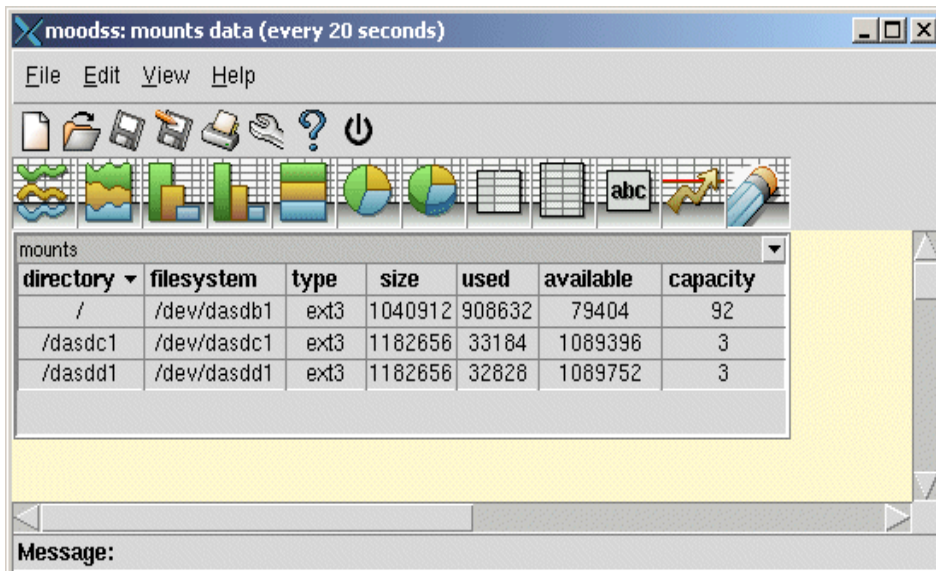


Figure 5-7 The mounts module.

- ▶ **myerrorlog** This module monitors the MySQL SQL database server error log.
- ▶ **myhealth** This module monitors the MySQL SQL database server health.
- ▶ **myprocs** This module monitors the MySQL SQL database server processes.
- ▶ **myquery** This module monitors any data from a MySQL SQL database server.
- ▶ **myreplication** This module monitors a pool of replicated master and slave MySQL SQL database servers and insuring that the servers are synchronized.
- ▶ **mystatus** This module monitors the MySQL SQL database server status.
- ▶ **myvars** This module monitors the MySQL SQL database server variables.
- ▶ **netdev** Displays the statistics about network devices. Data is displayed in two sets of two tables: two for data received by the interface with absolute and per-second values; the other for data transmitted by the interface, again with absolute and per-second values.

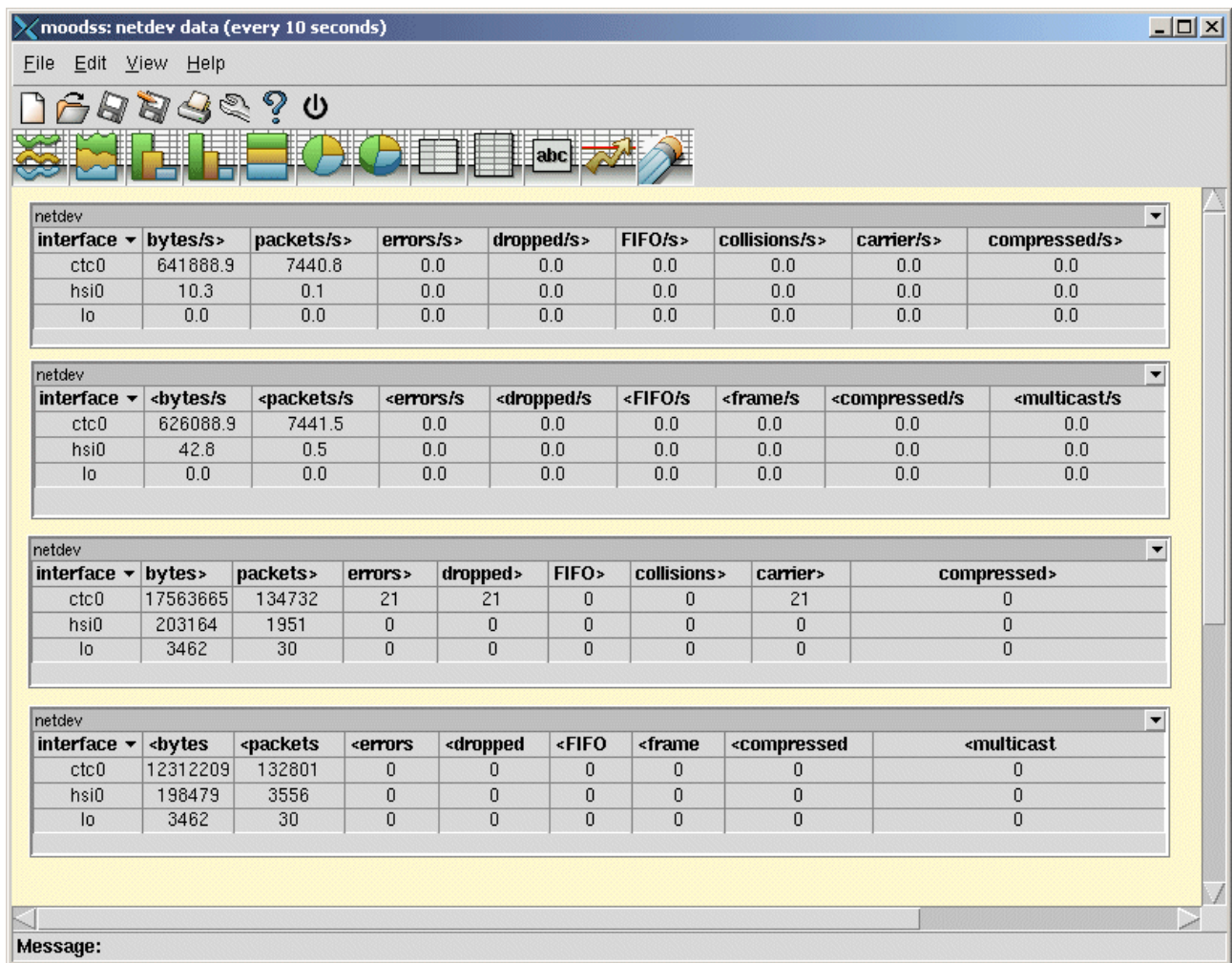


Figure 5-8 The netdev module

- ▶ **odbcquery** This module monitors any data from any database accessible through ODBC.
- ▶ **pci** This is a listing of all PCI devices found during the Linux kernel initialization, and their configurations. (This is not available for S/390 architecture).
- ▶ **ping** This module allows the monitoring of several hosts by pinging them; that is, sending one or more ICMP echo requests and reporting the round-trip time for each host.



- ▶ **ps** This is a view of processor activity, presented in two tables, one featuring the full command line. In the first table, tasks running on the system are initially sorted, with the most CPU-intensive first.

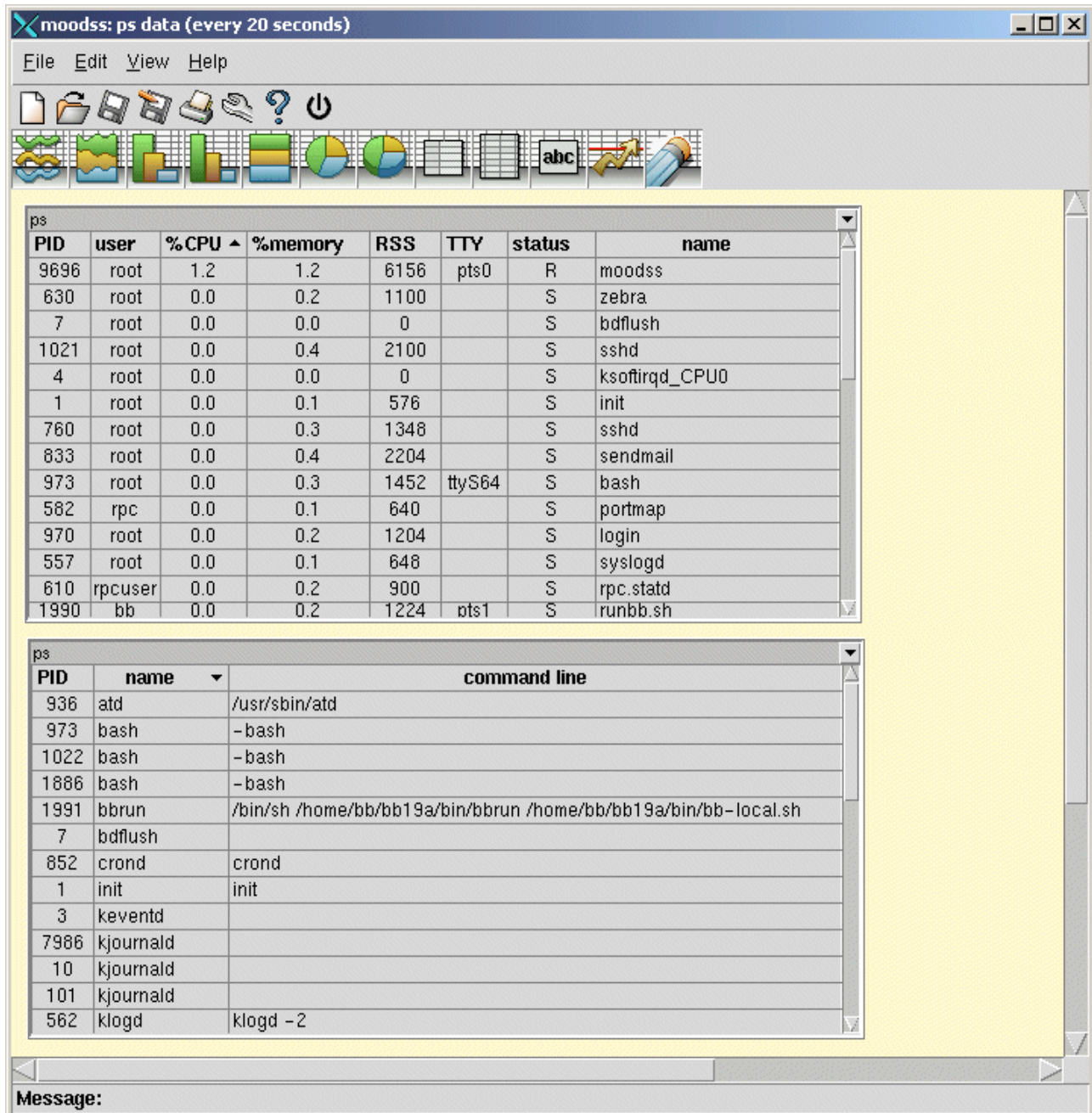


Figure 5-9 The ps module

- ▶ **random** This is a simple demonstration module for the moodss utility. Most of the data is randomly generated. Data is presented in two tables: one for the CPU and memory usage, the other for disk usage.
- ▶ **Random** The Perl version of the random module.
- ▶ **randpy** The Python version of the random module.

- **route:** Displays the network routes for the system (tested on IPv4 only). Networks and hosts addresses are looked up and converted to official names when possible.

The screenshot shows a window titled "moodss: route data (every 10 seconds)" with a menu bar (File, Edit, View, Help) and a toolbar with various icons. Below the toolbar is a table with the following data:

interface	destination	gateway	flags	mask	references	use	metric
lo	0.0.0.127		U	0.0.0.255	0	0	0
hsi0	0.1.1.10	11.2.1.10	GU	0.255.255.255	0	0	2
hsi0	0.2.1.10		U	0.255.255.255	0	0	0
ctc0	1.3.1.10		HU	255.255.255.255	0	0	0
hsi0	2.5.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
hsi0	4.5.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
hsi0	8.5.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
hsi0	9.5.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
hsi0	10.5.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
hsi0	12.4.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
hsi0	13.4.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
hsi0	17.4.1.10	11.2.1.10	GHU	255.255.255.255	0	0	2
ctc0	default	1.3.1.10	GU		0	0	0

Figure 5-10 The route module

- **sensors** This is a view of the current readings of all sensor chips available on your hardware, provided the sensors command is available on the system being monitored. (This is not available for S/390 architecture.)
- **snmp** This is a generic SNMP (Simple Network Management Protocol) module, which requires the Tnm Tcl extension package (from the Scotty Tcl extension software) to be installed on the computer.
- **snmptrap** This is an SNMP trap module, which requires the Tnm Tcl extension package to be installed on the computer, with the included straps daemon running.
- **system** This is a view of miscellaneous information for a Linux system. (This is not available for S/390 architecture.)
- **trace** This is the trace module for the moodss core. Whenever a loaded module issues a message to the user (through the internal moodss core/module interface), a new line is created in the table.(This is not available for S/390 architecture.)

Monitoring remote systems is also possible with each of these modules. To do this when loading a new module, type in the userid and the name of the remote host (rsh or ssh facilities must be properly set up, and rsh is used as default).

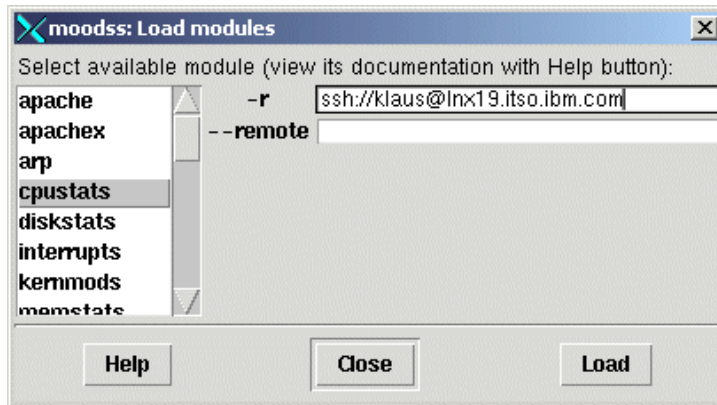


Figure 5-11 Get monitoring data from an remote system

### 5.3.2 Drag and drop

Drag and drop in moodss is the same function as drag and drop from Windows. To display data as a graphical plot, for example, select the data cell you want to display with the left mouse button, hold the button and drag over the icons. The icons that can be used to display the data are highlighted as you pass over them. Release the mouse button at the icon you have chosen and a BLT graph will be created. Once the graph is created, you can add as many data cells as you like, so it is possible to compare data in one graph (as illustrated in Figure 5-12).

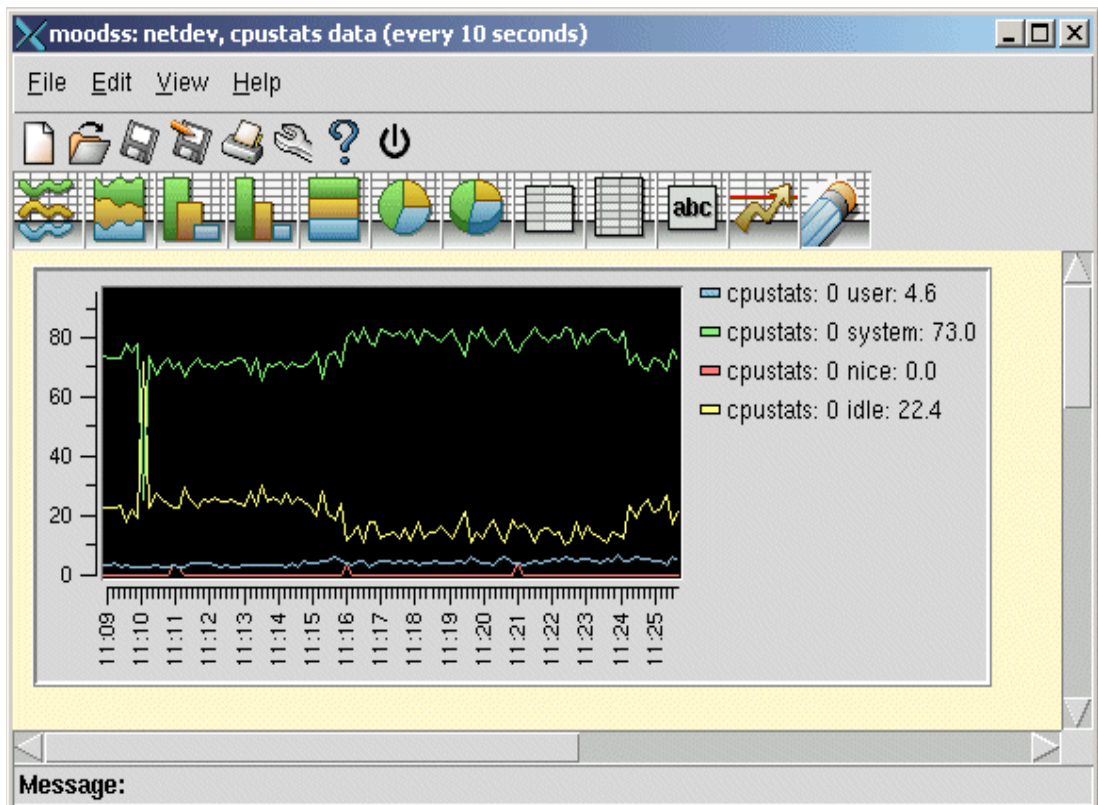


Figure 5-12 A graph with four data curves

Figure 5-13 shows the meaning of the available icons that can be used to display data.

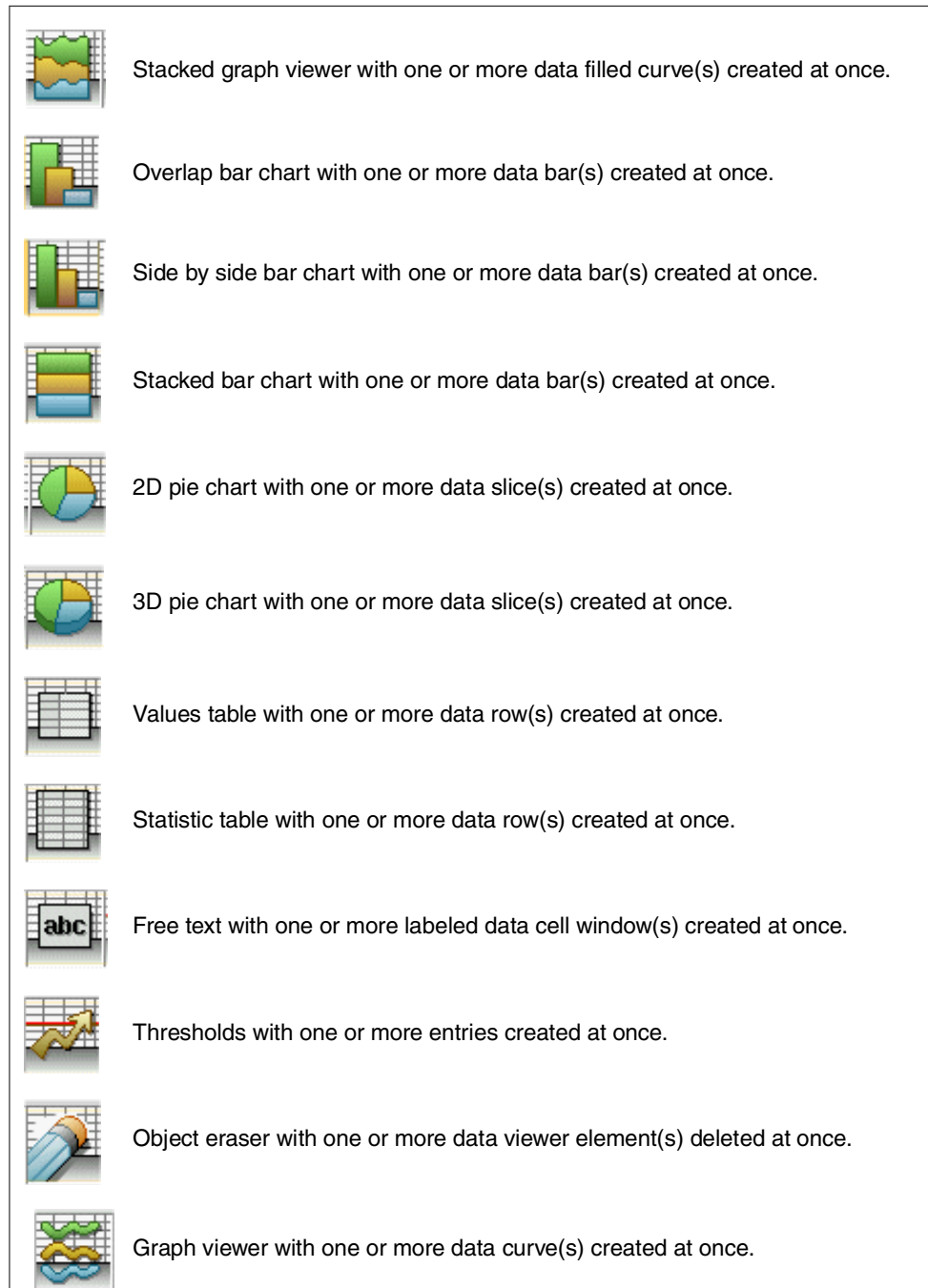


Figure 5-13 Data cell icons

### 5.3.3 Thresholds

To set or modify thresholds, select **Edit -> Thresholds**. A new window appears, as shown in Figure 5-14.

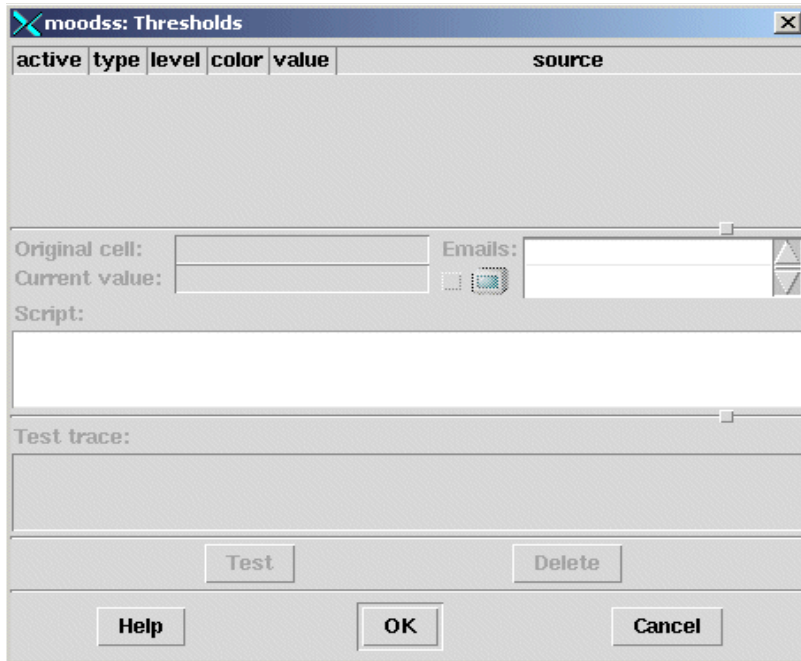


Figure 5-14 The threshold window

To add a new threshold, simply select the data cell you want to use and drop it in the threshold window. In the following example, we dropped the data files `cpu user` and `cpu system` from the `cpustats` module into the threshold window.

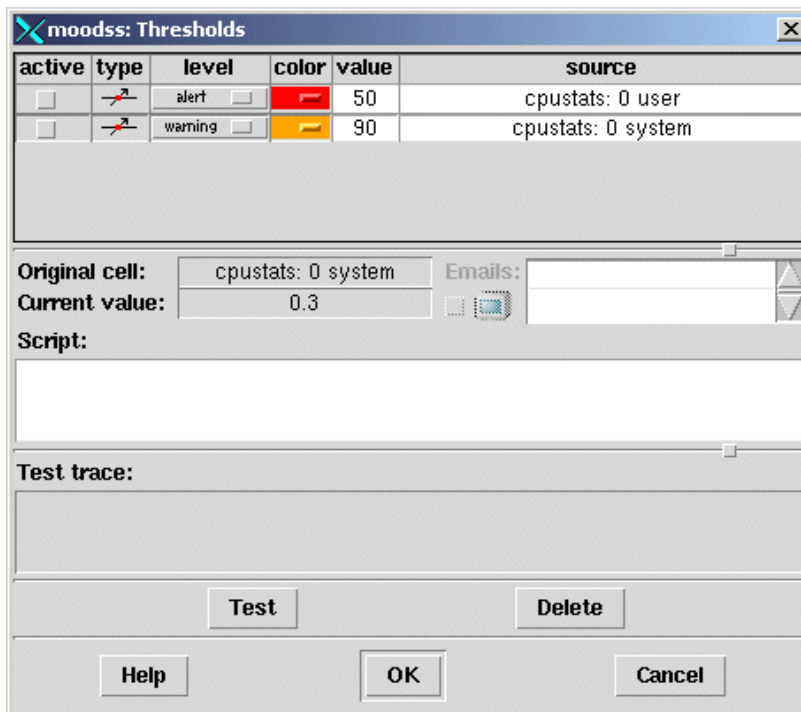


Figure 5-15 Threshold window with 2 entries

Now you can modify the threshold entries. First decide whether the entry should be active or not. When active, the shell script is invoked when the threshold condition occurs. Then select the threshold type. The following types are supported:

- differ**            Threshold and value must be different.
- down**            Cell value must be less than threshold.
- equal**            Threshold and cell value must be equal.
- unknown**        Cell value must be unknown.
- up**                Cell value must be greater than threshold.

The next setting is the level. In order of increasing importance, the following types are possible: debug, notice, warning, error, critical, alert, emergency.

Then select the color in which the source cell should be displayed when the threshold condition occurs.

The threshold value is used as a reference when comparing it with the data cell current value. In the source field you can enter a data cell label which would be displayed in the data viewer.

### 5.3.4 Poll time

To change the poll time, select **View ->Poll Time** and the window in Figure 5-16 will appear.



Figure 5-16 Poll time change window

You can select the new poll time either by using the up and down buttons, or by typing the new value directly. Then select **OK**.

### 5.3.5 Creating and saving configurations files

Once you have customized the view matching your needs (for example, loading the appropriate modules, resizing modules for a convenient view, changing colors, and so forth), it is a good idea to save this moodss configuration and reuse it again after moodss is restarted. To do this, select **File -> Save as** and the window shown in Figure 5-17 will appear.

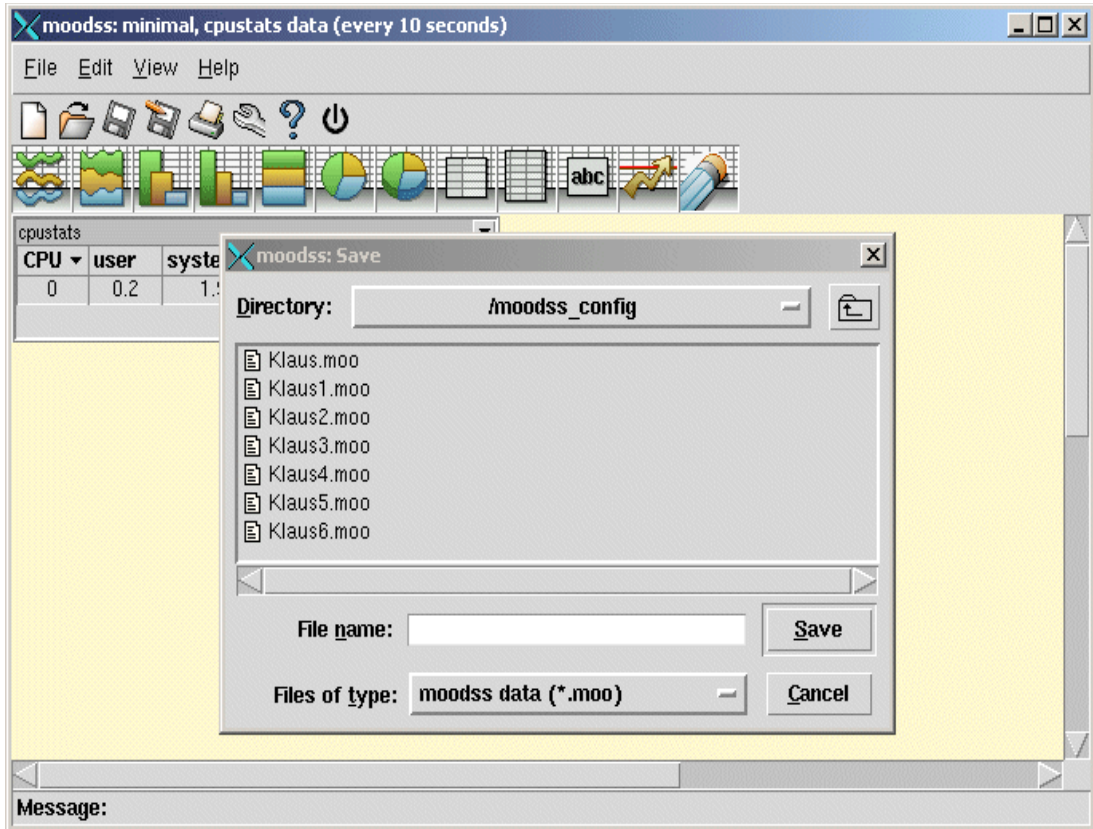


Figure 5-17 saving a configuration

The saved configurations can be used by selecting **File -> Open**.

Now you have to decide if you want to keep the configuration you are currently using. The prompt to do this is shown in Figure 5-18.

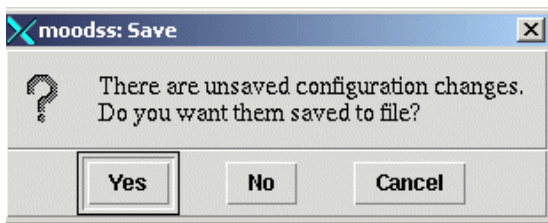


Figure 5-18 Configuration change confirmation

If you select **No** at this time, the window shown in Figure 5-19 is returned. Here you can choose which saved configuration file you want to use now.

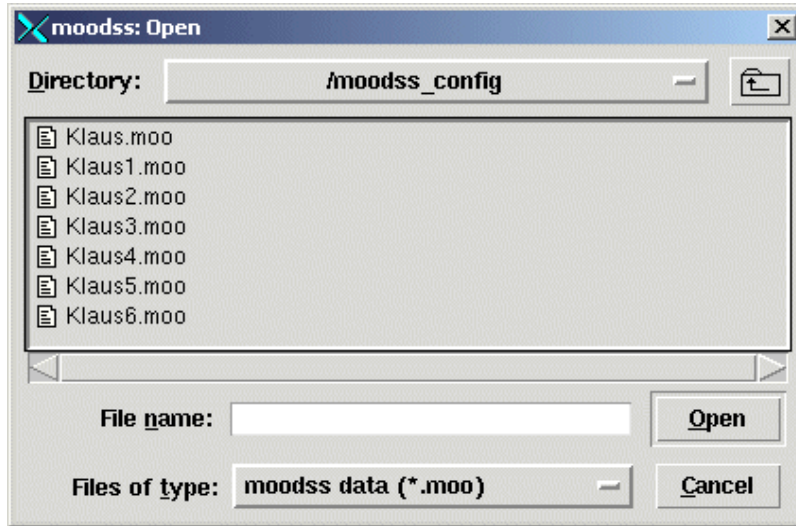


Figure 5-19 Open a saved configuration file

Double-click the file you want to use and the configuration will be restored.

### 5.3.6 Developing your own moodss modules

You can develop your own modules for moodss with Tcl, Perl, or Python. For more details about developing your own modules, see the moodss Web site.

### 5.3.7 Where to find more information

In this project at the ITSO we did not evaluate every single feature and possibility that moodss offers. When you need to find more information, especially on topics we do not cover in this redbook, visit the Web page of the creator of moodss, Jean-Luc Fontaine:

<http://jfontain.free.fr>

This Web page is also a good source for downloading the software you will need to run moodss.





## Amanda

Amanda (Advanced Maryland Automatic Network Disk Archiver) is one of the most common open source tools for data management. Both the SuSE and RedHat distributions we installed included Amanda. The version supplied with SuSE was 2.4.1p1; the version supplied with RedHat was 2.4.2p2.

In this chapter we describe how to install, configure, and use Amanda.

## 6.1 Amanda overview

Amanda is a tool that allows you to back up multiple client systems from a single backup server. It was developed at the University of Maryland and is now developed as an open source project. The Amanda web sites are:

<http://www.amanda.org/>  
<http://sourceforge.net/projects/amanda/>

Amanda uses standard tools such as `dump` and `tar` to perform its backups; however, it is more than just a front end for these tools. It manages the backups intelligently, and optionally maintains an index of files backed up. You can then restore files or disks as of a particular date, and Amanda will tell you which tapes are required for the restore.

The fact that Amanda uses standard tools to dump and restore means that the Amanda software is not required to recover a system.

Client backups are scheduled and controlled from the Amanda server. The server automatically switches between full, differential, and incremental backups of the clients to balance the workload and minimize the number of tapes required for a restore. You do not have control over the day in the cycle that full dumps are run. Amanda will spread the full dumps through the cycle to try to make the amount of data backed up in each backup run as even as possible. You define the length of the dump cycle that will be used, and Amanda ensures that a full dump of each system is done at least once per cycle.

## 6.2 Backup process

The Amanda server contacts the client to begin the backup. The first step is to gather information about how much data each client has to back up. Based on this information, the server determines what type of backup to perform on each client in this run.

The server then starts the backups on the clients. Normally the data is initially sent to a holding disk, and from there it is copied to tape. This means that the tape drive can be used at its maximum speed, without waiting for the data to be sent from the clients. Multiple clients can be backed up in parallel to the holding disk. If necessary, backups can also be defined to go direct to tape, and Amanda will also bypass the holding disk if it does not have enough space for the backup.

The backup data can be compressed on the client or the server, or written uncompressed. Amanda maintains a record of the compression achieved for each file system, which it uses when estimating the backup sizes.

### 6.2.1 Limitations

Amanda has some limitations. The major ones are:

- ▶ Dumps cannot span tapes. Amanda can use multiple tapes per run; however, each dump must fit on a single tape. If you need to back up very large file systems, use the GNUTAR backup option and include or exclude specific directories so that each backup will fit on a single tape.
- ▶ At the time of writing, the stable version only supports backup to tape, it does not support backups to a file. There is a beta version available which incorporates support for backing up to files. This gives extra flexibility for your backups – for example, you could back up to another system via NFS, then burn the images onto a CD.

## 6.2.2 Setting up the server

We did not have access to a tape drive, so we decided to install the beta version of the software and back up to disk. The version we installed was 2.4.3b3. We had a PC with a large amount of free space, so we shared its disk using NFS and used it for testing the backups.

We set up our server on LNX12, which is a SuSE system. There were some differences in the file locations used by SuSE and those of the default Amanda configuration.

### **Server configuration for SuSE**

These are the steps to configure the server when using Amanda from the SuSE distribution:

- ▶ Update `/etc/inetd.conf`.

Our `inetd.conf` contained entries for Amanda, but they were commented out. Uncomment the following entries:

```
# amanda backup server with indexing capabilities
amandaidx      stream tcp      nowait root    /usr/lib/amanda/amindexd amindexd
amidxtape      stream tcp      nowait root    /usr/lib/amanda/amidxtaped amidxtaped
```

Signal `inetd` to reread its configuration:

```
killall -HUP inetd
```

- ▶ Create a configuration.

Amanda configurations are defined in subdirectories in directory `/var/lib/amanda`. If necessary, you can create several configurations in different directories to support different requirements (for example, backup and long-term archiving).

The main configuration file is `amanda.conf` in the configuration subdirectory. This file then refers to other files to define disks to back up and other configuration details.

- ▶ If the server will be backing itself up – that is, if it is also a client – perform the client setup tasks.

**Differences when building from source:** The default file locations used if you build Amanda from source are different. These require some changes to the configuration.

`/etc/inetd.conf` entries should be:

```
# amanda backup server with indexing capabilities
amandaidx      stream tcp      nowait root    /usr/local/libexec/amindexd amindexd
amidxtape      stream tcp      nowait root    /usr/local/libexec/amidxtaped amidxtaped
```

The configuration directory is:

```
/usr/local/etc/amanda
```

The binaries are in directory:

```
/usr/local/sbin
```

This directory may need to be added to the path for your Amanda user.

## 6.2.3 Configuration details

We have one configuration which we named “normal.” We created a directory called `normal` in the configuration directory, and created the `amanda.conf` file there.

Since we did not have access to a tape drive, we configured Amanda for tapeless operation. For tapeless operations, a number of directories which will be used as virtual tapes must be

set up. Amanda will cycle through these in the same way it would for physical tapes. Each backup directory needs a subdirectory called data before Amanda will use it. Until that directory is created, Amanda will indicate that the tape is offline. Each directory needs to be labelled as if it were a tape using the `amlabel` program.

We created the directories using the serial numbers of our virtual tapes as the directory names.

```
cd /misc/backup/amanda
mkdir -p K302R01/data
mkdir -p K302R02/data
...
mkdir -p K302R25/data
```

The command `mkdir -p` creates any parent directories as required.

Example 6-1 shows the contents of our `amanda.conf` file.

*Example 6-1 amanda.conf*

---

```
org "LS-K302-R Linux Backups"
mailto "amanda"
dumpuser "amanda"

inparallel 5
netusage 10000 Kbps

dumpcycle 1 weeks
runspcycle 7
tapecycle 25 tapes

bumpsize 20 Mb
bumpdays 1
bumpmult 4

etimeout 300

runtapes 1
tpchanger "chg-multi"
changerfile "/usr/local/etc/amanda/normal/chg-multi.conf"

tapetype DISK-TOT25
labelstr "^K302R[0-9][0-9]*$"

holdingdisk hd1 {
    comment "main holding disk"
    directory "/amanda"
    use -0 Mb
    chunksize 0
}

reserve 30

infofile "/usr/local/etc/amanda/normal/curinfo"
logdir "/usr/local/etc/amanda/normal/"
indexdir "/usr/local/etc/amanda/normal/index"
tapelist "/usr/local/etc/amanda/normal/tapelist"

define tapetype DISK-TOT25 {
    comment "Backup to TOT25 DISK"
    length 10000 mbytes
```

```

        filemark 0 kbytes
        speed 1000 kbytes
    }

define dumptype global {
    comment "Global definitions"
    index yes
}

define dumptype root-tar {
    global
    program "GNUTAR"
    comment "root partitions dumped with tar"
    compress none
    priority low
}

define dumptype user-tar {
    root-tar
    comment "user partitions dumped with tar"
    priority medium
}

define dumptype comp-user-tar {
    user-tar
    compress client fast
}

define dumptype holding-disk {
    global
    comment "The master-host holding disk itself"
    holdingdisk no
    priority medium
}

define dumptype comp-user {
    global
    comment "Non-root partitions on reasonably fast machines"
    compress client fast
    priority medium
}

define dumptype comp-root {
    global
    comment "Root partitions with compression"
    compress client fast
    priority low
}

```

---

### ***amanda.conf parameters***

The `amanda.conf` parameters are well documented in the sample `amanda.conf` file, which can be found in directory `/usr/share/doc/packages/amanda` on SuSE, and `/usr/share/doc/amanda-server-2.4.2p2/examples` on RedHat. Some of the parameters require extra explanation regarding how they relate to our particular configuration.

**dumpcycle**            Defines the length of the dumpcycle. A full (level 0) dump will be taken at least once per dump cycle. We made the `dumpcycle` 7 days so that we would have a reasonable number of cycles in our limited testing time. Your dump cycle may be longer.

<b>runspercycle</b>	How often amdump will run per dump cycle.
<b>tapecycle</b>	How many tapes are available. This should be at least dumpcycle multiplied by runspercycle, plus a few more in case of extra runs. Tapes will not be overwritten while the backups they contain are still valid, so you need enough to last at least one cycle. We set tapecycle to 25 so that we had enough for extra runs for manual testing.
<b>bumpsize</b>	Defines the savings required before Amanda will switch to the next dump level. A level 0 dump is a full backup. Each dump level backs up everything modified since the last dump at a lower level. A level 1 dump backs up everything since the last level 0 dump, a level 2 dump backs up everything since the last level 1, and so forth. Going to a new dumplevel means that there is less data that needs backing up, but more tapes will be required in the event of a restore. Bumpsize 20 MB indicates that Amanda will not go to a new level unless it will reduce the size of the backup by at least 20 MB.
<b>bumpmult</b>	A multiplier for bumpsize used to make the change to each additional dump level less likely. Each additional level means that more tapes are required for a restore. Setting bumpmult to 4 means that bumping from level 2 to 3 requires 4 times the savings that it took to bump from 1 to 2.
<b>tpchanger</b>	Defines the tape changer to be used. chg-multi is a generic changer script supplied with Amanda that can be adapted to many different circumstances. We use it to change between our virtual tapes.
<b>changerfile</b>	Indicates the file that contains our configuration for the chg-multi script.
<b>holdingdisk</b>	Specifies an area to hold the backups before they are written to tape. There are two main reasons to do this. First, many tape drives are most efficient when they are supplied with an unbroken stream of data. Waiting for data can slow them down and even result in more tape being used. Amanda will gather data on the holding disk so it can write continuously to the tape. This is not an issue in our configuration, as we are writing to disk, not tape.  The second reason is that it allows many backups to be performed in parallel, whereas only one backup can be written to the tape at a time. This means that the overall backup window is likely to be much shorter. This is where we get a benefit from the holding disks.
<b>define tapetype</b>	This is where we define our virtual tapes. Since we are backing up to disk, these parameters are somewhat flexible.
<b>length</b>	Defines the length of a tape. Since we are backing up to disk, this effectively defines the maximum amount of data that will be backed up in a single run. This data may be split into a number of files.
<b>filemark</b>	The amount of space used by a filemark on a tape. Amanda will reduce its calculation of the amount of space remaining on the tape by this much when starting a new file. As we are backing up to disk, this is 0.
<b>speed</b>	The speed of the tape drive. Currently, Amanda does not use this parameter.

### ***Defining the tape changer***

Backing up to disk requires a tape changer script to be defined so Amanda can switch from one virtual tape directory to another. The supplied chg-multi script supports this function. We used the file chg-multi.conf to define the configuration.

### Example 6-2 *chg-multi.conf*

---

```
multieject 0

gravity 0

needeject 0

ejectdelay 0

statefile /usr/local/etc/amanda/normal/changer-status

firstslot 1
lastslot 25

slot 1 file:/misc/backup/amanda/K302R01
slot 2 file:/misc/backup/amanda/K302R02
slot 3 file:/misc/backup/amanda/K302R03
slot 4 file:/misc/backup/amanda/K302R04
slot 5 file:/misc/backup/amanda/K302R05
slot 6 file:/misc/backup/amanda/K302R06
slot 7 file:/misc/backup/amanda/K302R07
slot 8 file:/misc/backup/amanda/K302R08
slot 9 file:/misc/backup/amanda/K302R09
slot 10 file:/misc/backup/amanda/K302R10
slot 11 file:/misc/backup/amanda/K302R11
slot 12 file:/misc/backup/amanda/K302R12
slot 13 file:/misc/backup/amanda/K302R13
slot 14 file:/misc/backup/amanda/K302R14
slot 15 file:/misc/backup/amanda/K302R15
slot 16 file:/misc/backup/amanda/K302R16
slot 17 file:/misc/backup/amanda/K302R17
slot 18 file:/misc/backup/amanda/K302R18
slot 19 file:/misc/backup/amanda/K302R19
slot 20 file:/misc/backup/amanda/K302R20
slot 21 file:/misc/backup/amanda/K302R21
slot 22 file:/misc/backup/amanda/K302R22
slot 23 file:/misc/backup/amanda/K302R23
slot 24 file:/misc/backup/amanda/K302R24
slot 25 file:/misc/backup/amanda/K302R25
```

---

### ***chg-multi.conf* parameters**

The parameters specified in the *chg-multi.conf* script have the following meanings:

**multieject, gravity, needeject, ejectdelay**

These parameters all define characteristics of physical tape changers. For our virtual changer, they are set to 0.

**statefile**

This file stores the current status of the changer.

**firstslot, lastslot**

These parameters define the slot ranges for the tape changer. We defined a slot for each of our virtual tapes.

**slot**

The slot statements define the slots of our tapechanger. The third parameter in each slot statement is the device Amanda must use to access that slot. When Amanda calls *chg-multi*, the script returns this device name. This allow *chg-multi* to use several devices as a virtual tape changer. In our case we used the `file:/...` syntax to back up to disk, and listed each virtual tape directory as a different slot. The result

is that when Amanda calls for the tape in slot 23 to be mounted, `chg-multi` returns the device `file:/misc/backup/amanda/K302R23`.

### **Labelling the tapes**

Each tape, including our virtual tapes, needs to be labelled before Amanda will use it. To label our tapes we used the `amlabel` command. `Amlabel` will use our virtual tape changer to select the tape from the slot we specify.

```
amlabel normal K302R25 slot 25
```

```
Labeling tape in slot 25 (file:/misc/backup/amanda/K302R25):  
rewinding, reading label, not an amanda tape  
rewinding, writing label K302R25, checking label, done.
```

**Note:** While we have given our directories the same names as our virtual tapes, this is not required. Amanda does not check the directory names when it writes the label. It is quite possible to label slot 24 (directory `K302R24`) as tape `K302R25`. When Amanda uses the tape it refers to the label, so it would find tape `K302R25` in slot 24. To fix the problem we could rename the directories to correspond to the labels Amanda wrote, and Amanda would then find tape `K302R25` in slot 25.

## **6.2.4 Setting up the clients**

The work to set up each client is minimal. The 2.4 level clients and servers are compatible, so we did not have to change the client systems when we installed the new server software. The clients supplied with the SuSE and RedHat distributions worked without problems. Use the following steps to set up the clients.

1. Configure `inetd` (SuSE) or `xinetd` (RedHat)

- `inetd`

Update `/etc/inetd.conf`. Uncomment the entry:

```
# amanda backup client  
amanda dgram udp wait amanda /usr/lib/amanda/amandad amandad
```

Signal `inetd` to re-read its configuration with: `killall -HUP inetd`

- `xinetd`

Update file `/etc/xinetd.d/amanda` and change `disable = yes` to `disable = no`.

Signal `inetd` to re-read its configuration with: `killall -USR2 xinetd`

2. Add an entry to the `.amandahosts` file in the Amanda user's home directory to allow the server to connect and perform the backups. The format is the same as `.rhosts`:

```
<systemname> <user>
```

For example:

```
lrx12.itso.ibm.com root  
lrx12.itso.ibm.com amanda
```

3. On the server, add the client and its disks to the `disklist` file:

```
lrx13 dasdb1 comp-root
```

The RedHat systems needed the full name of the device:

```
lrx14 /dev/dasdb1 comp-root
```

4. Use the `amcheck` command to verify the configuration. Use `su` to switch to the `amanda` user, and issue command `amcheck normal` (`normal` is the name of our configuration). This command connects to the clients to make sure they are configured correctly.



**Difference when building from source:** The default location of the client software is also different when built from source.

The client `/etc/inetd.conf` entry should be:

```
# amanda backup client
amanda dgram udp wait amanda /usr/local/libexec/amandad amandad
```

## Normal operations

We added the following entries to `/etc/crontab` on the server so that the backups would run every day.

```
00 16 * * * amanda /usr/local/sbin/amcheck -m normal
30 1 * * * amanda /usr/local/sbin/amdump normal;/usr/local/sbin/amadmin normal export
> /misc/backup/amanda/db_export
```

At 16:00, `amcheck` is run. This checks the Amanda client and server status, and sends mail to root if any problems are found. Items checked include the availability of tapes, access to the clients, and sufficient space to allow the backups to run.

At 01:30 the `amdump` process is run. This performs the backups, and again any problems will be identified via email. The second part of the command exports the Amanda database after the backups to a text file on the same server as the backups. This file is human readable and can be used to find which tapes contain which backups if the Amanda database is lost, or imported to rebuild the database.

## 6.3 Recovering data

This section describes some sample recovery scenarios. More information about recovery can be found in the Amanda documentation. See file `/usr/share/doc/amanda-server-2.4.2p2/RESTORE` (RedHat) or `/usr/share/doc/packages/amanda/RESTORE` (SuSE) and the `amrestore` and `amrecover` man pages.

### *Recovering individual files or directories*

In this scenario we have decided we want to see two files as they were several days ago. We can use the `amrecover` command to query the backup index and select the date from which the files should be restored. We will restore the files into root's home directory so we don't overwrite the current files.

The two files are backed up to different tapes. We specify the restore device as an argument to the `amrecover` command. To allow us to change virtual tapes, we create a symbolic link in the backup directory, and use that as the restore device. When we need to change tapes, we can delete the link and recreate it pointing to the next virtual tape. Creating the symbolic link obviously requires a second session; for clarity, the commands for the second session are shown in italics.

1. Run `amrecover`, specifying the index server, the tape server, and the device to be used for recovery.

```
amrecover normal -s lnx12 -t lnx12 -d file:/misc/backup/amanda/restore
AMRECOVER Version 2.4.3b3. Contacting server on lnx12 ...
220 lnx12 AMANDA index server (2.4.3b3) ready.
200 Access OK
Setting restore date to today (2002-04-29)
200 Working date set to 2002-04-29.
```

```
200 Config set to normal.
200 Dump host set to lnx12.
Trying disk / ...
Trying disk root ...
Can't determine disk and mount point from $CWD '/'
```

2. Amanda didn't figure out which disk we are using, so set it manually.

```
amrecover> setdisk dasdb1
Warning: no log files found for tape K302R25 written 0-00-00
Warning: no log files found for tape K302R24 written 0-00-00
Warning: no log files found for tape K302R23 written 0-00-00
Warning: no log files found for tape K302R22 written 0-00-00
Warning: no log files found for tape K302R21 written 0-00-00
Warning: no log files found for tape K302R20 written 0-00-00
Warning: no log files found for tape K302R19 written 0-00-00
Warning: no log files found for tape K302R18 written 0-00-00
Warning: no log files found for tape K302R17 written 0-00-00
Warning: no log files found for tape K302R16 written 0-00-00
Warning: no log files found for tape K302R15 written 0-00-00
Warning: no log files found for tape K302R14 written 0-00-00
Scanning /amanda...
lost+found: skipping cruft directory, perhaps you should delete it.
200 Disk set to dasdb1.
```

3. List the backups for this disk.

```
amrecover> history
200- Dump history for config "normal" host "lnx12" disk "dasdb1"
201- 2002-04-29 1 K302R08 5
201- 2002-04-29 0 K302R09 7
201- 2002-04-29 1 K302R10 4
201- 2002-04-29 1 K302R11 3
201- 2002-04-29 1 K302R12 4
201- 2002-04-29 1 K302R13 3
201- 2002-04-28 1 K302R07 5
201- 2002-04-27 1 K302R06 5
201- 2002-04-26 1 K302R04 3
201- 2002-04-26 1 K302R05 6
201- 2002-04-25 0 K302R01 2
201- 2002-04-25 1 K302R02 5
201- 2002-04-25 0 K302R03 8
200 Dump history for config "normal" host "lnx12" disk "dasdb1"
```

4. Set the date so we recover the files as of the 27th April 2002.

```
amrecover> setdate 2002-04-27
200 Working date set to 2002-04-27.
```

5. Select the files to be restored.

We can move around the directory tree in the dump, and use **ls** to list files, **add** to add files to be recovered, **delete** to remove them from the list, and so forth. The **ls** listing gives the date each file was backed up. We can see that some were backed up on the day we are interested in; some had not changed, so they will be recovered from a previous dump.

```
amrecover> cd /etc
/etc
amrecover> ls
2002-04-27 .
2002-04-25 .pwd.lock
2002-04-25 DIR_COLORS
2002-04-25 HOSTNAME
2002-04-25 SuSE-release
```

```
...
2002-04-25 resolv.conf
2002-04-25 rmt
2002-04-27 route.conf
...
2002-04-25 wgetrc
2002-04-25 xinetd.conf
2002-04-25 ypserv.conf
2002-04-25 zshrc
lines 113-159/159 (END)
```

**q**

6. Add the files we want to recover.

```
amrecover> add route.conf resolv.conf
Added /etc/route.conf
Added /etc/resolv.conf
```

List shows which files will be recovered from which tape.

```
amrecover> list
TAPE K302R01 LEVEL 0 DATE 2002-04-25
    /etc/resolv.conf
TAPE K302R06 LEVEL 1 DATE 2002-04-27
    /etc/route.conf
lines 1-4/4 (END)
```

**q**

7. Check which directory we are recovering to, and ensure that we recover to /root so we don't overwrite the existing files. **lpwd** shows the current directory, **lcd** changes it.

```
amrecover> lpwd
/
amrecover> lcd /root
amrecover> lpwd
/root
```

8. Use **extract** to recover the files from the dump.

```
amrecover> extract
```

```
Extracting files using tape drive file:/misc/backup/amanda/restore on host lnx12.
The following tapes are needed: K302R01
                                K302R06
```

```
Restoring files into directory /root
Continue? [Y/n]: y
```

```
Load tape K302R01 now
```

9. With a second session, create the symbolic link to the required virtual tape:

```
cd /misc/backup/amanda
ln -s K302R01 restore
ls -l restore
lrwxrwxrwx  1 amanda  disk          7 Apr 30 10:12 restore -> K302R01
```

```
Continue? [Y/n/t]: y
set owner/mode for '.'? [yn] n
Load tape K302R06 now
```

10. Change the link to point to the next tape:

```
rm restore;ln -s K302R06 restore
ls -l restore
lrwxrwxrwx  1 amanda  disk          7 Apr 30 10:12 restore -> K302R06
```

```
Continue? [Y/n/t]: y
restore: ./etc: File exists
set owner/mode for './?' [yn] n
amrecover> quit
200 Good bye.
```

The directory structure is recreated under the point of restore, so our restored files are located in directory `/root/etc`.

**Tip:** Virtually rewinding a virtual tape.

Sometimes when running a restore, we got messages like the following:

```
amrestore: WARNING: not at start of tape, file numbers will be offset
amrestore: 0: reached end of information
restore: Tape is not a dump tape
```

The virtual tape emulation in Amanda maintains the concept of file numbers on a tape, and keeps track of a current position. This message means that the last Amanda operation to use the virtual tape left it positioned somewhere other than at the beginning. To avoid it, “rewind” the required tapes prior to the restore:

```
ammt -f file:/misc/backup/amanda/K302R01 rewind
```

### 6.3.1 Recovering a client system

This scenario involves a full restore of the root device of LNX13, one of our client systems. We start with a completely empty disk. Under VM, we can link the client’s disk to our server LNX12, and perform the restore there. This avoids having to create a running system on the client to perform the restore. However it did require a reboot of the server to recognize the new disk.

The DASD is attached to LNX12 as `/dev/dasde`, and it has been formatted with `dasdfmt`. The restore requires an empty file system created with `mke2fs`.

1. Create the new filesystem.

```
mke2fs /dev/dasde1
mke2fs 1.19, 13-Jul-2000 for EXT2 FS 0.5b, 95/08/09
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
132480 inodes, 264417 blocks
13220 blocks (5.00%) reserved for the super user
First data block=0
9 block groups
32768 blocks per group, 32768 fragments per group
14720 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376
```

```
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

2. Use the `amadmin info` command to find out which tapes contain the backup of the system to be restored. We could not use `amrecover` since we were restoring a disk from a different system, and `amrecover` does not have an option to restore data from another system. We needed to use `amrestore`.

### amadmin normal info lnx13

Current info for lnx13 dasdb1:

```
Stats: dump rates (kps), Full: 1501.0, 385.0, 137.0
      Incremental: 65.0, 6.0, 70.0
      compressed size, Full: 37.3%, 37.3%, -100.0%
      Incremental: 27.9%, 8.4%, 31.7%
Dumps: lev datestmp tape file origK compK secs
      0 20020429 K302R11 7 634943 634940 423
      1 20020430 K302R14 1 1655 461 7
```

The most recent backups for this system are on two tapes, with a level 0 full backup and a level 1 incremental backup. We needed to restore the level 0 backup first, then the level 1 backup on top of it.

3. Create a temporary mount point and mount the new file system. The restore *must* be started from the directory that is the target of the restore operation.

```
mkdir /tmp/restore
mount /dev/dasde1 /tmp/restore
cd /tmp/restore
ls -l
total 24
drwxr-xr-x 3 root root 4096 Apr 30 14:23 .
drwxrwxrwt 6 root root 4096 Apr 30 14:36 ..
drwxr-xr-x 2 root root 16384 Apr 30 14:23 lost+found
```

4. Restore the level 0 backup. The `amrestore` command reads the Amanda tape, and the data is passed to the normal Linux restore program.

```
amrestore -p file:/misc/backup/amanda/K302R11 lnx13 dasdb1 | restore -r -b2 -f -
```

5. Restore the level 1 backup on top of the level 0 backup.

```
amrestore -p file:/misc/backup/amanda/K302R14 lnx13 dasdb1 | restore -r -b2 -f -
```

This process would be repeated for each level if we had more backup levels.

6. The resulting filesystem contains all the data as of the last backup, but it is not bootable. We needed to run `silo` to write the IPL information. The easiest way we found to do this to a disk that is not the current boot disk is to use `chroot`. Chroot makes the directory appear to your session to be the root disk.

```
chroot /tmp/restore
cd /boot
silo -f image -d /dev/dasde -t2
o->image set to image
o->ipldevice set to /dev/dasde
Testonly flag is now 0
Testlevel is set to -2
IPL device is: '/dev/dasde'
bootsector is: '/boot/ipleckd.boot'...ok...
bootmap is set to: './boot.map'...ok...
Kernel image is: 'image'...ok...
original parameterfile is: '/boot/parmfile'...ok...tempfile is ./parm.T2SVmX
final parameterfile is: './parm.T2SVmX'...ok...
ix 0: offset: 01198b count: 0c address: 0x00000000
ix 1: offset: 011998 count: 80 address: 0x0000c000
ix 2: offset: 011a18 count: 80 address: 0x0008c000
ix 3: offset: 011a98 count: 72 address: 0x0010c000
ix 4: offset: 015bc9 count: 01 address: 0x00008000
Bootmap is in block no: 0x00015bca
```

At this point the DASD is detached from LNX12, and LNX13 is rebooted.

7. There are some minor housekeeping tasks that need to be done to complete the restore. In the root directory of the restored disk there is a file called `restoresymtable`. This is used by restore to keep track of the full and incremental restores correctly. Once the restore is complete it should be removed.

```
rm restoresymtable
```

The next backup of the restored file system must be a full backup. The inode information for the filesystem has changed, so a new incremental backup could not be applied to a full backup from before the restore. We used the `amadmin` command on the server to force the next backup of LNX13 to be level 0.

```
amadmin normal force lnx13
```

```
amadmin: lnx13:dasdb1 is set to a forced level 0 at next run.
```

### 6.3.2 Recovering the server

In this scenario we are recovering the root disk of the server, LNX12. This includes our Amanda databases and Amanda itself. We can perform the recovery using only standard Linux tools: `dd`, `gzip` and `restore`.

Again, we attach the DASD from the system to be restored to another Linux system. Other alternatives if there were no usable systems would be to install a system from scratch to run the restore, or have a basic system available via a VM backup.

This procedure would be very suitable for disaster recovery, where production systems could be recovered using a pre-built driver system.

1. The disk to restore is attached as `/dev/dasdc` and has been formatted. Create the filesystem.

```
mke2fs /dev/dasdc1
```

```
mke2fs 1.19, 13-Jul-2000 for EXT2 FS 0.5b, 95/08/09
```

```
Filesystem label=
```

```
OS type: Linux
```

```
Block size=4096 (log=2)
```

```
Fragment size=4096 (log=2)
```

```
132480 inodes, 264417 blocks
```

```
13220 blocks (5.00%) reserved for the super user
```

```
First data block=0
```

```
9 block groups
```

```
32768 blocks per group, 32768 fragments per group
```

```
14720 inodes per group
```

```
Superblock backups stored on blocks:
```

```
32768, 98304, 163840, 229376
```

```
Writing inode tables: done
```

```
Writing superblocks and filesystem accounting information: done
```

2. The disk containing the backups is accessible via NFS. Mount it from the recovery system.

```
mount tot25.itso.ibm.com:/export /tmp/backups
```

3. As part of our backups, we exported the Amanda database to a file on our backup disk. We can look at this file and find which backups are required to restore lnx12.

```
cd /tmp/backups/amanda
```

```
less db_export
```

```
CURINFO Version 2.4.3b3 CONF LS-K302-R Linux Backups
```

```
# Generated by:
```

```
# host: lnx12
```

```
# date: Wed May 1 01:30:53 2002
```

```
# command: /usr/local/sbin/amadmin normal export
```

```

# This file can be merged back in with "amadmin import".
# Edit only with care.
host: lnx12
disk: dasdb1
command: 0
last_level: 1
consecutive_runs: 7
full-rate: 234.000000 453.000000 158.000000
full-comp: 0.329574 0.328704 0.327870
incr-rate: 268.000000 251.000000 82.000000
incr-comp: 0.246453 0.235748 0.351500
stats: 0 705466 232503 991 1020087602 7 K302R09
stats: 1 14165 3491 13 1020231034 2 K302R16
//
...

```

The stats: lines for lnx12 show that we need two tapes to recover. K302R09 contains the level 0 backup, and K302R16 contains the level 1 backup.

4. Since the backups are on disk, we can access them directly. If they were on tape we would have to skip through to the correct file using the `mt` command. On disk, the filenames include the system and disk so it is easy to see which file we need to use.

```

cd /tmp/backups/amanda/K302R09/data
ls -l
...
-rw----- 1 amanda disk 238125056 Apr 29 10:00 00007.lnx12.dasdb1.0
...

```

5. The first record of the Amanda dump contains information about the dump, including instructions for restoring it. We can display it using the `dd` command.

```

dd if=00007.lnx12.dasdb1.0 bs=32k count=1
AMANDA: FILE 20020429 lnx12 dasdb1 lev 0 comp .gz program /sbin/dump
To restore, position tape at start of file and run:
    dd if=<tape> bs=32k skip=1 | /usr/bin/gzip -dc | sbin/restore -f... -

```

```

1+0 records in
1+0 records out

```

6. Mount the new filesystem and change to that directory.

```

mount /dev/dasdc1 /tmp/restore
cd /tmp/restore

```

7. Run the restore, using the commands from the Amanda dump header.

```

dd if=/tmp/backups/amanda/K302R09/data/00007.lnx12.dasdb1.0 bs=32k skip=1 |
/usr/bin/gzip -dc | restore -r -b2 -f -
restore: ./lost+found: File exists
./tmp/rstidir1020087602-d1XTmb: (inode 30517) not found on tape
./usr/local/etc/amanda/normal/index/lnx18/_dev_dasdb1/20020429_2.gz.tmp: (inode 209627)
not found on tape
./usr/local/etc/amanda/normal/index/lnx14/_dev_dasdb1/20020429_1.gz: (inode 209626) not
found on tape
expected next file 209599, got 209598
7266+0 records in
7266+0 records out

gzip: stdin: decompression OK, trailing garbage ignored

```

The error messages we receive are probably caused by activity on the file system when the dump was taking place. There are no major problems here.

- Repeat the process with the level 1 backup.

```
ls -l /tmp/backups/amanda/K302R16/data/
...
-rw----- 1 amanda disk 3637248 May 1 01:29 00002.lnx12.dasdb1.1
...
dd if=/tmp/backups/amanda/K302R16/data/00002.lnx12.dasdb1.1 bs=32k skip=1 |
/usr/bin/gzip -dc | restore -r -b2 -f -
restore: ./dev/log: No such file or directory
110+0 records in
110+0 records out
restore: ./var/run/printer: No such file or directory
restore: ./var/run/.nscd_socket: No such file or directory

gzip: stdin: decompression OK, trailing garbage ignored
```

At this point the file system has been restored. We need to run `si10` as documented for the client restore, then we can detach the DASD and reboot `lnx12`.

- When the server has been recovered, import the Amanda database that was exported after the last backup, to ensure that it has the most up-to-date information.

```
amadmin normal import < /misc/backup/amanda/db_export
```

- To complete the task, remove the `restoresymtable` file from the restored disk, and force the next backup to level 0.





# OpenLDAP

In this chapter we discuss using OpenLDAP and PAM to create a system in which userids and passwords can be centrally managed.

## 7.1 Managing users using OpenLDAP and pam\_ldap

One of the major issues when managing a number of systems is userid and password management. Often users, particularly systems administrators, have to be able to log on to many different systems. If each system has its own database of userids and passwords, management becomes very difficult. A user will either have to remember a different password for each system, or will have to change their password on many systems at the same time.

Storing user information using OpenLDAP and authenticating using pam\_ldap provides a large amount of flexibility, with reasonable security. With this configuration we were able to have a number of different types of users:

- ▶ Local users. These were users created in the normal fashion on individual systems, and administered locally by root.
- ▶ Global users with a network-wide home directory. These users were defined in the LDAP directory, and given an NFS-mounted home directory. They could then log on to any system, and they would have access to their data.
- ▶ Global users with local home directories. These users were defined in the LDAP directory, but with local home directories. We set the systems up so that the first time a user logged on to a system the home directory was created if it didn't already exist.

The global users are administered using LDAP commands. The authority to do this is defined in the LDAP configuration, and the users who perform userid and password administration do not need root access. Likewise, root on the client systems cannot change the passwords of the centrally administered users.

## 7.2 OpenLDAP overview

OpenLDAP is an open source implementation of LDAP (Lightweight Directory Access Protocol). The OpenLDAP Web page is:

<http://www.openldap.org/>

LDAP is a protocol used to access information stored in a directory. The information in the directory is arranged in a tree structure. Each object is identified by a unique distinguished name, which identifies the object and its position in the tree, for example:

“uid=andrew,ou=People,dc=itso,dc=ibm,dc=com”. The structure and rules for the objects in the tree are defined by schema. RFC2307 defines a schema for storing network information, including user and group details using LDAP. OpenLDAP comes with this schema.

### 7.2.1 PAM and pam\_ldap

PAM stands for Pluggable Authentication Module. The PAM API provides a flexible authentication mechanism. It allows different programs to be used to authenticate a user, without changing the application. Rules are set up defining the methods to be used, and the relationship between the different methods. PAM also allows other functions to be inserted into the authentication process, for example, creating a home directory.

PAM is included in standard Linux systems to perform the normal UNIX authentication functions. Tailoring it involves modifying the configuration files to change the modules called and/or the order they are called in.

Pam\_ldap was developed by PADL Software Pty. Ltd. It provides a module that authenticates a user with an LDAP server. The pam\_ldap Web page is:

[http://www.padl.com/OSS/pam\\_ldap.html](http://www.padl.com/OSS/pam_ldap.html)

Pam\_ldap takes the userid and password entered by the user and attempts to use them to bind to the LDAP server. If the LDAP server accepts the bind, the user is allowed to log in. This means that the password is stored on the LDAP server, and all the client ever gets is a success/failure response. As long as the clients are separated from the servers, not even the root userid on the clients can access or change the password information without actually knowing the password.

We initially tested using pam\_ldap-56-74 included in SuSE 7.0, but after experiencing some problems we switched to a SuSE SLES 7 system with pam\_ldap-105-29. The more recent pam\_ldap version solved the problems we experienced with the earlier version.

## 7.2.2 Configuring OpenLDAP

The OpenLDAP server is called slapd, and the configuration is defined in `/etc/openldap/slapd.conf`. The client configuration is defined in `/etc/openldap/ldap.conf`.

### ***slapd.conf***

Our initial slapd configuration is shown in Example 7-1.

#### *Example 7-1 slapd.conf*

---

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/nis.schema

pidfile /var/run/slapd.pid
argsfile /var/run/slapd.args

defaultaccess read
access to attr=userPassword
    by self write
    by anonymous auth
    by * none
access to *
    by * read

database ldbm

suffix "dc=itso,dc=ibm,dc=com"
rootdn "cn=Manager,dc=itso,dc=ibm,dc=com"
rootpw secret

directory /var/lib/ldap

index objectClass eq
index cn eq
index uid eq
index uidNumber eq
index gidNumber eq
```

---

**include** The include statements include several schemas which provide useful definitions. Schemas define the attributes of the objects that the database will contain. These particular schemas are supplied with OpenLDAP. Among other

things, they define NIS (Network Information Service) information, which includes the user information that we require.

**access** The access statements define ACLs to control access to information in our directory.

It is important that access to userPassword is defined correctly. The password should not be readable by everyone, as that would allow people to take it and run programs such as crack against it. Auth access allows a user to use it for authentication only. The user supplies a password and LDAP checks it against the stored encrypted password. The userPassword value itself is not supplied to the client. Allowing write access to “self” means that after a user has authenticated successfully, they have the ability to change their password.

It is also important that the user does *not* have access to change certain other attributes. In particular, they must not be able to change the uidNumber and gidNumber fields since these determine what access they have on a UNIX system.

Later we will add some rules to allow administrators to make changes.

**suffix** This defines the suffix of the entries this database will store.

#### **rootdn and rootpw**

This defines an LDAP “superuser.” This userid can be used to modify the database and no ACL checking is done. The user doesn’t have to exist in the database, so it is used to define the initial entries. Once some administrative users have been defined, rootdn can be removed. The password can be encrypted, but since we are going to remove it as soon as we define our first users we will leave it in plaintext.

**index** We added some extra index statements for attributes we felt would benefit from indexing.

To start slapd, issue:

```
/etc/rc.d/ldap start
```

Edit /etc/rc.config and set START\_LDAP=yes.

#### ***ldap.conf***

The ldap.conf file needs to define the LDAP server to be used, and the base name for the LDAP entries. Example 7-2 shows our initial ldap.conf file.

*Example 7-2 ldap.conf*

---

```
host lnx13.itso.ibm.com
base dc=itso,dc=ibm,dc=com
ldap_version 3
pam_passwd crypt
```

---

### **Adding initial entries**

PADL Software provides migration tools for transferring data to LDAP. The MigrationTools are a set of Perl scripts that read the data from the standard UNIX files and create LDAP entries. The MigrationTools can be found at:

<http://www.padl.com/OSS/MigrationTools.html>

We used the migration tools to generate LDAP entries based on our files, but we did not add the entries directly into LDAP. Instead we used the output from the migration tools as a

template to add other entries. To do this we ran scripts individually rather than the migrate\_all scripts which go on to add the users to LDAP.

Unpack the migration tools:

```
tar -zxvf MigrationTools.tgz
```

This creates a directory MigrationTools-40. Set the defaults in the file migrate\_common.ph. Since we only migrated the password and group files, the only important entry was \$DEFAULT\_BASE. This needs to match what was set in slapd.conf.

```
$DEFAULT_BASE = "dc=itso,dc=ibm,dc=com"
```

### ***Adding LDAP base entries***

Before we can add people and groups we need to set up the contexts for those entries, such as ou=People and ou=Group. The migrate\_base.pl script creates a file containing these entries.

```
./migrate_base > base.ldif
```

Then we can add our first entries to LDAP:

```
ldapadd -x -D "cn=Manager,dc=itso,dc=ibm,dc=com" -w secret < base.ldif
```

-D sets the LDAP user we connect as (the LDAP root user we created), and -w is the password. If we use a capital W we would be prompted for the password instead of specifying it on the command line.

### ***Adding an initial user***

We ran the migration scripts to produce the password and group entries:

```
./migrate_passwd.pl /etc/passwd passwd.ldif
```

```
./migrate_group.pl /etc/group group.ldif
```

This produced two files, passwd.ldif and group.ldif, which contained our passwd and group entries in LDIF form. LDIF stands for LDAP Data Interchange Format; it is the format used by utilities such as ldapadd.

*Example 7-3 /etc/passwd entry converted by migrate\_passwd.pl*

---

```
dn: uid=evandro,ou=People,dc=itso,dc=ibm,dc=com
uid: evandro
cn: evandro
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword: {crypt}JjNPOXXpJLSAA
shadowLastChange: 11795
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 500
gidNumber: 100
homeDirectory: /home/evandro
```

---

*Example 7-4 /etc/group entry converted by migrate\_group.pl*

---

```
dn: cn=users,ou=Group,dc=itso,dc=ibm,dc=com
objectClass: posixGroup
objectClass: top
cn: users
userPassword: {crypt}x
gidNumber: 100
```

---

Using the `ldif.passwd` entries as a template, we created an input file to allow us to add a new LDAP user; this is illustrated in Example 7-5.

*Example 7-5 New LDAP user*

---

```
dn: uid=andrew,ou=People,dc=itso,dc=ibm,dc=com
uid: andrew
cn: andrew
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword: {crypt}JjnPOXXpJLSAA
shadowLastChange: 11795
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 1001
gidNumber: 100
homeDirectory: /home/andrew
```

---

Then we can add the user to LDAP with:

```
ldapadd -x -D "cn=Manager,dc=itso,dc=ibm,dc=com" -w secret < andrew.ldif
```

Our `userPassword` entry contains the value from the UNIX shadow file, so the password is initially the same as the UNIX password. We can also set the password using LDAP functions:

```
ldappasswd -x -D "cn=Manager, dc=itso, dc=ibm, dc=com" -w secret \
-S "uid=andrew,ou=People,dc=itso,dc=ibm,dc=com"
New password:
Re-enter new password:
Result: Success (0)
```

**Creating an administrator**

Now that there is a normal user defined, we can give that user the authority to update entries, and remove the root user from the `slapd.conf` file. Comment out or delete the `rootdn` and `rootpw` entries, and change the ACLs to give the new user write access:

```
access to attr=userPassword
    by self write
    by dn="uid=andrew,ou=people,dc=itso,dc=ibm,dc=com" write
    by anonymous auth
    by * none
access to *
    by dn="uid=andrew,ou=people,dc=itso,dc=ibm,dc=com" write
    by * read
```

Restart the LDAP server:

```
/etc/rc.d/ldap restart
```

Now we can add information using the userid we just created:

```
ldapadd -x -D "uid=andrew,ou=People,dc=itso,dc=ibm,dc=com" -W < group.ldif
Enter LDAP Password:
adding new entry "cn=users,ou=Group,dc=itso,dc=ibm,dc=com"
```

### ***Listing entries in the directory***

We used the following to list the entries in our LDAP database:

---

```
ldapsearch -x -b 'dc=itso,dc=ibm,dc=com' 'objectclass=*' \
-D "uid=andrew, ou=People, dc=itso, dc=ibm, dc=com" -W
Enter LDAP Password:
version: 2

#
# filter: objectclass=*
# requesting: ALL
#

# itso,dc=ibm,dc=com
dn: dc=itso,dc=ibm,dc=com
dc: itso
objectClass: top
objectClass: domain

# People,dc=itso,dc=ibm,dc=com
dn: ou=People,dc=itso,dc=ibm,dc=com
ou: People
objectClass: top
objectClass: organizationalUnit
...
# andrew,People,dc=itso,dc=ibm,dc=com
dn: uid=andrew,ou=People,dc=itso,dc=ibm,dc=com
uid: andrew
cn: andrew
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
shadowLastChange: 11795
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 1001
gidNumber: 100
homeDirectory: /home/andrew
userPassword:: e1NTSEF9eUhXU1hgME94U0ZUbFB0bE1KNThJeU1JUzVrUHpUTjk=

# users,Group,dc=itso,dc=ibm,dc=com
dn: cn=users,ou=Group,dc=itso,dc=ibm,dc=com
objectClass: posixGroup
objectClass: top
cn: users
userPassword:: e2NyeXB0fXg=
gidNumber: 100
```

---

In this case we are connecting to the LDAP database using a particular userid and password, so we can see all the entries that user is allowed to see.

We can also run the command without supplying a userid and password. This shows what anybody connecting to LDAP can see. In our case we see the same entries, except for the userPassword field, because we have set up an ACL to restrict access to it.

---

```
ldapsearch -x -b 'dc=itso,dc=ibm,dc=com' 'objectclass=*'
...
# andrew,People,dc=itso,dc=ibm,dc=com
dn: uid=andrew,ou=People,dc=itso,dc=ibm,dc=com
uid: andrew
cn: andrew
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
shadowLastChange: 11795
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 1001
gidNumber: 100
homeDirectory: /home/andrew

# users,Group,dc=itso,dc=ibm,dc=com
dn: cn=users,ou=Group,dc=itso,dc=ibm,dc=com
objectClass: posixGroup
objectClass: top
cn: users
gidNumber: 100
```

---

## Diagnosing problems

To diagnose problems, you can turn on various logging options with the **loglevel** option in slapd.conf. Slapd will then log messages about its activities to syslog. By default these go to /var/log/messages, or you can send them to a separate file by changing the syslog configuration for **local4.\*** messages. Logging options available are:

```
1      trace function calls
2      debug packet handling
4      heavy trace debugging
8      connection management
16     print out packets sent and received
32     search filter processing
64     configuration file processing
128    access control list processing
256    stats log connections/operations/results
512    stats log entries sent
1024   print communication with shell backends
2048   entry parsing
```

The desired logging options are added together to produce the loglevel. **loglevel -1** will log everything.

**Note:** Some of these options produce a lot of data. We found that using them had a major impact on LDAP performance, and caused our syslog files to grow to many megabytes even when only performing limited testing.



## 7.2.3 Configuring Name Service Switch

The Name Service Switch (NSS) is a scheme to control where various functions go to find information. For example, many programs need to find information from the passwd file: usernames, home directories, and so forth. NSS allows the locations of the information to be changed without affecting the applications. The NSS configuration is stored in file `/etc/nsswitch.conf`.

To have the NSS functions go to LDAP for information, we change the passwd, group, and password entries to use LDAP:

```
passwd: files ldap
group: files ldap
shadow: files ldap
```

These entries mean that any search for information will check the local files first, and if the entry is not found, query LDAP.

We need to reload the name service cache daemon to make sure the changes are picked up:

```
/etc/rc.d/nscd reload
```

Then we should be able to perform a function that queries these entries:

```
echo ~andrew
/home/andrew
```

This command simply echoes the home directory of user andrew. As andrew is defined in LDAP but not on our local system, the information has to come from LDAP.

## 7.2.4 Configuring PAM

Once OpenLDAP is set up, we can set up the system to use it for authentication. To do this we need to change the PAM configuration. This tends to be set up very differently in different distributions. Here we describe the SuSE setup.

### Introduction to PAM

PAM on SuSE is configured using files in the `/etc/pam.d` file. Each program (login, ftp, passwd, su, and so forth) that needs to use PAM services has a configuration file. The default **login** PAM configuration is shown in Example 7-6. The files have a list of module types and modules. For each service the listed modules of the required type are called in order to determine the success or failure of the service.

*Example 7-6 /etc/pam.d/login*

---

```
##PAM-1.0
auth requisite /lib/security/pam_unix.so nullok #set_secrcp
#auth required /lib/security/pam_securetty.so
auth required /lib/security/pam_nologin.so
#auth required /lib/security/pam_homecheck.so
auth required /lib/security/pam_env.so
auth required /lib/security/pam_mail.so
account required /lib/security/pam_unix.so
password required /lib/security/pam_pwcheck.so nullok
password required /lib/security/pam_unix.so nullok use_first_pass use_authtok
session required /lib/security/pam_unix.so none # debug or trace
session required /lib/security/pam_limits.so
```

---

The first column defines the module type. There are currently four module types:

- auth** Identifies the user and checks that they are allowed to log in. This is typically by checking the password, although PAM allows other identity checks to be substituted.
- account** Checks other account management details - for example restricting the time or place a user may login.
- session** Set up a users session.
- password** Used to update a user's password or other means of authentication.

The second column defines the importance of each module.

- requisite** The module must succeed; if it fails the failure will be returned to the application. The following modules will not be called if it fails.
- required** The module must succeed; if it fails the failure will be returned to the application. However, modules following this one will still be called.
- sufficient** If this module indicates success, success will be returned to the application. No further modules will be called. Failure does not necessarily cause the service to fail.
- optional** The result of this module will be ignored, unless no other module returned a success or failure indication.

The rest of the line lists the module to be called and any parameters required.

The module type column indicates what function will call the module, but the module doesn't strictly have to perform that function. An example of this is the `pam_mail` module in the `auth` section of the default file. This module checks for mail when the user logs in. This isn't really an authentication function, and it would more typically be called for the session service. However, as long as it returns success it can be placed in the `auth` section.

More detail about administering PAM can be found in the *Linux-PAM System Administrator's Guide* at:

<http://www.kernel.org/pub/linux/libs/pam/Linux-PAM-html/pam.html>

**Important:** When modifying the PAM configuration, it is very easy to make a change that stops you from logging in. When making changes, ensure that you:

- ▶ Take a copy of your working configuration before you start.
- ▶ Have another root session already logged in that you can use for recovery.
- ▶ Before disconnecting, make sure you can still log in.

## Changes to `/etc/pam.d/login`

Our modified login file is shown in Example 7-7.

*Example 7-7 /etc/pam.d/login using pam\_ldap*

---

```
##PAM-1.0
#auth    required    /lib/security/pam_securetty.so
auth     required    /lib/security/pam_nologin.so
#auth    required    /lib/security/pam_homecheck.so
auth     sufficient  /lib/security/pam_unix.so
auth     required    /lib/security/pam_ldap.so          use_first_pass
account  sufficient  /lib/security/pam_unix.so
account  required    /lib/security/pam_ldap.so
password required    /lib/security/pam_pwcheck.so
```

password	sufficient	/lib/security/pam_unix.so	use_first_pass use_authtok
password	required	/lib/security/pam_ldap.so	use_first_pass use_authtok
session	required	/lib/security/pam_limits.so	
session	required	/lib/security/pam_mkhomedir.so	skel=/etc/skel/ umask=0022
session	required	/lib/security/pam_env.so	
session	required	/lib/security/pam_mail.so	
session	sufficient	/lib/security/pam_unix.so	
session	required	/lib/security/pam_ldap.so	

---

The `pam_unix` entries have been changed to *sufficient*, which means that if the user successfully authenticates using the local `/etc/passwd` information no other entries will be checked. The `pam_ldap` entries are *required* because if we reach them, authentication using the local files failed. The end result is that if either of the `pam_unix` or `pam_ldap` entries succeeds, the user will be able to log in.

One of the side effects of adding entries with an importance of *sufficient* is that we can't put other entries after them if we need them to be processed. The default login file in Example 7-6 on page 137 had `pam_env` and `pam_mail` modules to be called after the `pam_unix` authentication was successful. We have moved them to the session section so that they will still be called.

## Changes to `/etc/pam.d/passwd`

The file `/etc/pam.d/passwd` defines the configuration for the `passwd` command<sup>1</sup>. Example 7-8 shows our `passwd` definitions.

*Example 7-8 /etc/pam.d/passwd*

auth	sufficient	/lib/security/pam_unix.so	
auth	required	/lib/security/pam_ldap.so	use_first_pass
account	sufficient	/lib/security/pam_unix.so	
account	required	/lib/security/pam_ldap.so	
password	required	/lib/security/pam_pwcheck.so	
password	sufficient	/lib/security/pam_unix.so	use_authtok
password	required	/lib/security/pam_ldap.so	use_authtok
session	sufficient	/lib/security/pam_unix.so	
session	required	/lib/security/pam_ldap.so	

---

Again, we check the local files, and if the user is not found we go to LDAP. The `use_authtok` parameter on the password change modules forces the modules to use the password that was passed from the previous module. The result of that is that `pam_unix` and `pam_ldap` will use the new password that has passed the `pam_pwcheck` strength check. They will not allow the user to pick a different password.

## Other services

There are many other services such as `ftp`, `pop`, `ssh`, and `sudo` which use PAM. These all need to be configured in the same way if they are to use LDAP for authentication. The general principles to use when configuring them are:

- ▶ To allow authentication using either `pam_ldap` or `pam_unix`, the first one needs to be *sufficient* and the second should be *required*.
- ▶ Any modules of the same type after the first *sufficient* module will not be called if that module succeeds. Often a *required* module needs to be changed to *sufficient*, and there are other modules called after it. The order will need to be changed to ensure that the other modules are called.

<sup>1</sup> The definitions for the password service in the login file are only used if the user changes their password while logging in, not for password changes using the `passwd` command.

## 7.2.5 Securing the connection and verifying the server

The `pam_ldap` authentication process works by attempting to bind to the LDAP server using the `userid` and password supplied by the user. If the bind succeeds, the user is allowed to log in. This means that the `userid` and password need to be transmitted to the LDAP server.

By default the `userid` and password are transmitted unencrypted, which makes them vulnerable to packet sniffing-type attacks. To counter this, we set up our systems to use Transport Layer Security (TLS), which is similar and related to SSL.

Additionally, we want to be sure that the server that we are authenticating with is the server we think it is. Otherwise, it is possible for a bogus server to answer our requests, and allow unauthorized users to log into our system. To counter this we generate a certificate which we can use to verify the identity of the server.

Much of the information we used to set this up came from the paper “Security with LDAP” by Andrew Findlay, which is available at:

<http://www.skills-1st.co.uk/papers/afindlay.html#ldapsec20020214>

and linked from the PADL Software Web site.

## 7.2.6 Generating the keys and certificates

We used the `openssl` tools to generate the keys and certificates. We generate our own certificates rather than getting them from a supplier, which means that the client cannot go to an external site to confirm the identity of the server. Since we will be giving the client a copy of the Certification Authority key and we are only guarding against the substitution of the server, this is sufficient for our needs.

Generate the Certification Authority key and certificate:

```
openssl genrsa -des3 -out ca.key 2048
openssl req -new -x509 -days 365 -key ca.key -out ca.cert
```

You will be asked for various items of information. The most important is the pass phrase: this is used to encrypt the key that is generated. To use the key you will have to supply the pass phrase, so it is important to remember it. The rest of the information will be incorporated into the Certification Authority (CA).

Next we generate a key and certificate signing request (CSR) for the server, and sign it with the CA we just produced:

```
openssl genrsa -out ldap.key 1024
openssl req -new -key ldap.key -out ldap.csr
```

When asked for the Common Name, enter the fully qualified domain name of the server. This should be the same as the name the LDAP clients have in `ldap.conf`.

```
openssl x509 -req -in ldap.csr -out ldap.cert -CA ca.cert -CAkey ca.key \
-CAcreateserial -days 365
```

The server needs its certificate, its key, and the CA certificate. The client is also given a copy of the CA certificate, and it can then use it to verify the certificate of the server.

**Important:** Unlike the CA key we created, the server key is not encrypted and protected with a pass phrase. This is because the LDAP server needs to be able to access the key. The key needs to be protected, so ensure that the file containing the key is only readable by its owner.

## Enabling TLS

To enable TLS encryption of the transmitted data, copy the files `ldap.key`, `ldap.cert` and `ca.cert` to the `/etc/openldap/keys` directory. We had to create this directory on our system. Add the following lines to `/etc/openldap/slapd.conf` and restart the server:

```
TLSCertificateFile /etc/openldap/keys/ldap.cert
TLSCertificateKeyFile /etc/openldap/keys/ldap.key
TLSCACertificateFile /etc/openldap/keys/ca.cert
```

On the client, add the line:

```
ssl start_tls
```

The session between the LDAP client and server is now encrypted.

We verified this using `tcpdump`. We captured the traffic generated by a login to a file. Initially the `userid` and `password` were transmitted in clear text. Following this change there was no recognizable text transmitted.

## Enforcing TLS

While enabling TLS allows you to set up the client to encrypt the session with the server, you can also enforce the encrypted session at the server end. This will not stop a misconfigured client from sending passwords in the clear, but if the password is not encrypted the logon will always fail.

Enforcing the encrypted session is done using the `tls_ssf` parameter in the LDAP ACLs. This checks the *security strength factor* required for access to the attributes. This means we can deny access to the `userPassword` field over an unencrypted connection. We set up the following ACLs:

```
access to attr=userPassword
    by self tls_ssf=112 write
    by dn="uid=andrew,ou=people,dc=itso,dc=ibm,dc=com" tls_ssf=112 write
    by anonymous tls_ssf=112 auth
    by * none
access to *
    by dn="uid=andrew,ou=people,dc=itso,dc=ibm,dc=com" tls_ssf=112 write
    by * read
```

This means that any access to the `userPassword` field must be over an encrypted connection. Additionally, update access to other fields must be over an encrypted connection. We allow read access to other information without encryption.

## Verifying the identity of the server

To have the client verify the identity of the server, copy the `ca.cert` file to `/etc/openldap`. Add the following line to `/etc/openldap/ldap.conf`:

```
tls_checkpeer yes
tls_cacertfile /etc/openldap/ca.cert
```

The client should now verify the identity of the server, and if the server can't identify itself using the certificates we generated, LDAP users will not be allowed to log in.

**Note:** When we tested this function it did not work on our system: LDAP users could still log in if the certificate was invalid. We suspect that this function is too new to be included in the version of `pam_ldap` that came with SuSE.

## 7.2.7 User administration

User administration tasks for LDAP users obviously need to be done using LDAP commands. As the syntax of these commands is complex, and normally the same options are used, we created some simple scripts to simplify the tasks.

We examined the common tasks of creating users and changing passwords. Other tasks would be done in a similar fashion. All the administration tasks need to be done by a user with appropriate access defined in LDAP. The user's access on the UNIX system itself is irrelevant; in fact, with appropriate tools the administrator may not need access to the system at all.

### Creating users

Example 7-9 shows our simple script for adding users.

*Example 7-9 add\_ldap\_user*

---

```
#!/bin/bash
read -p "Enter new userid: " new_user
read -p "Enter uid: " new_uid
read -p "Enter gid: " new_gid
read -p "Enter home directory: " new_homedir
read -p "User information: " new_gecos

ldapadd -x -Z -D "uid=$USER, ou=people, dc=itso, dc=ibm, dc=com" -W << endofinput
dn: uid=$new_user,ou=People,dc=itso,dc=ibm,dc=com
uid: $new_user
cn: $new_user
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
shadowLastChange: 0
shadowMax: 60
shadowWarning: 7
loginShell: /bin/bash
uidNumber: $new_uid
gidNumber: $new_gid
homeDirectory: $new_homedir
gecos: $new_gecos
endofinput
```

---

The script creates the LDIF format file, and feeds it to the `ldapadd` program. The `$USER` variable plugs in the userid of the user running the script, and `-W` prompts for the password, adding a user using the script:

```
./add_ldap_user
Enter new userid: john
Enter uid: 2001
Enter gid: 100
Enter home directory: /home/john
User information: Test user
Enter LDAP Password:
adding new entry "uid=john,ou=People,dc=itso,dc=ibm,dc=com"
```

## Home directories

LDAP allows us to manage our users in a central directory, but they still need a home directory on the systems they log on to. There are two simple ways to manage this:

### ► pam\_mkhome

This module creates a home directory for the user if one doesn't already exist. Example 7-7 on page 138 showed our `/etc/pam.d/login` definitions. We added `pam_mkhome` to the session modules. This means that we can define a user in LDAP and when they first log on to a system their home directory will be created automatically. We now log on with the `ldap` user we created:

```
Welcome to SuSE SLES-7 (s390) - Kernel 2.4.7 (2).
```

```
lnx13 login: john
Password:
Creating home directory '/home/john'.
Have a lot of fun...
john@lnx13:~ >
```

### ► NFS-mounted home directories

We set up an NFS server for home directories, and set up `automount` to mount the directories. Our `automount` files were:

```
auto.master
/nfshome          /etc/auto.nfshome
auto.nfshome
*                 -rw,hard,intr      lnx12:/home/&
```

The result is that if we specify a home directory in `/nfshome` instead of `/home`, the home directory will be mounted from the NFS server. In this case we still need to create the home directory on the NFS server when we create the `userid`.

These two schemes can coexist. We can have some users with home directories of the type `/home/userid`, and some with `/nfshome/userid`. An NFS-mounted home directory has the advantage that the user's data is accessible from any system. A local home directory means that the user can log in even if the NFS server is unavailable. Depending on the `userid`, one or other of these considerations will be more important.

## Managing UID numbers

A centralized user directory requires some extra management of user's numeric uids. These numbers are the basis of security in UNIX, and it is important that each user has a different number. The `useradd` program which can normally be used to create a user defaults to picking the next unused uid, so on a single system this can effectively manage itself.

When using LDAP to manage users you need to specify the uid (`uidNumber` in the LDAP schema) to assign to the user. It is important that this is not the same as another user on any of your systems, because if there are two users with the same number they will have access to each other's files. For most purposes, Linux will treat them as if they were the same user.

The best policy is to reserve a currently unused range of uids for LDAP users. Leave some room to add local users to the systems as well. You also need a record of which uid numbers are already used. The LDAP directory itself may be sufficient. The following query lists all users with their uid numbers, sorted by uid number:

```
ldapsearch 'objectclass=posixaccount' uid uidNumber -S uidnumber
```

## Changing passwords

We also created a script to change passwords. The ability to change passwords is controlled by write access to the userPassword field. Our script was very simple, it just plugged values into the `ldappasswd` command.

*Example 7-10 change\_ldap\_passwd script*

---

```
#!/bin/bash
ldappasswd -x -ZZ -D "uid=$USER, ou=People, dc=itso, dc=ibm, dc=com" -W -S
"uid=$1,ou=People,dc=itso,dc=ibm,dc=com"
```

---

The administrator specifies the userid of the person whose password is to be changed as an argument to the script. The `ldappasswd` command prompts for the new password, and the password of the user binding to LDAP (the user running the script).

## 7.2.8 Replicating servers using slurpd

If user's passwords are all verified by one server, this server becomes a single point of failure as well as a potential performance bottleneck. To avoid this, the LDAP information can be replicated to a number of slave servers. The clients can then be divided among the servers, or the server addresses provided by some kind of round robin or workload balancing scheme.

All updates still need to be made to the master server. If a slave server receives an update request, it refers the client to the master. The master `slapd` writes all updates to a log file in LDIF format. The log file is read by a daemon called `slurpd`, which then connects to the slave servers and makes the changes using normal LDAP functions. `Slurpd` uses the same configuration file as `slapd`.

### Preparing for replication

`Slurpd` must connect to the slave servers with an ID that has authority to update the database. You can define the `rootdn` in the slave's `slapd.conf` file and use that user, or define a user to perform the updates. We chose to define a new ID, which we called `ldap_update`.

We created the following definition in file `ldap_update.ldif`.

```
dn: cn=ldap_update,dc=itso,dc=ibm,dc=com
objectClass: simpleSecurityObject
userPassword: x
```

We added it to the directory and updated the password:

```
ldapadd -Z -x -D "uid=andrew, ou=people, dc=itso, dc=ibm, dc=com" -W < ldap_update.ldif
Enter LDAP Password:
adding new entry "cn=ldap_update,dc=itso,dc=ibm,dc=com"
```

```
ldappasswd -x -Z -D "uid=andrew, ou=People, dc=itso, dc=ibm, dc=com" -W \
-S "cn=ldap_update,dc=itso,dc=ibm,dc=com"
New password:
Re-enter new password:
Enter bind password:
Result: Success (0)
```

### Configuring the master

We made the following changes to the master:

In `slapd.conf`, we added entries for the replica and log file.

```
replica host=lnx4.itso.ibm.com tls=critical bindmethod=simple
```



```
binddn="cn=ldap_update,dc=itso,dc=ibm,dc=com" credentials=secret
```

```
repllogfile /var/run/openldap/repllog/repllog.ldif
```

We had to create the directory for the repllogfile. Ensure that the permissions are 700 since the directory will contain sensitive information. The binddn and credentials are used by slurpd to connect to the slave ldap server. We used simple (userid/password) authentication, but more secure methods are possible. Since there is now a password in the slapd.conf file, ensure the file is protected adequately.

Change /etc/rc.config and set START\_SLURPD=yes

## Configuring the slave

We copied the master slapd.conf file to the slave and made the following changes:

- ▶ Removed the replica and repllogfile statements.
- ▶ Added entries:

```
updatedn      "cn=ldap_update,dc=itso,dc=ibm,dc=com"
updateref     ldaps://lnx13.itso.ibm.com
```

Updatedn specifies the ID that will be allowed to connect to the slave and make updates. This is the ID specified in the replica binddn on the master.

Updateref refers any update requests to the master server.

- ▶ Changed the ACLs to allow write access for the ID in updatedn. Our ACLs were:

```
access to attr=userPassword
    by self tls_ssf=112 write
    by dn="uid=andrew,ou=people,dc=itso,dc=ibm,dc=com" tls_ssf=112 write
    by dn="cn=ldap_update,dc=itso,dc=ibm,dc=com" tls_ssf=112 write
    by anonymous tls_ssf=112 auth
    by * none
access to *
    by dn="uid=andrew,ou=people,dc=itso,dc=ibm,dc=com" tls_ssf=112 write
    by dn="cn=ldap_update,dc=itso,dc=ibm,dc=com" tls_ssf=112 write
    by * read
```

The slave also needs a new TLS certificate and key. We used the CA created previously:

```
openssl genrsa -out lnx4.key 1024
openssl req -new -key lnx4.key -out lnx4.csr
openssl x509 -req -in lnx4.csr -out lnx4.cert -CA ca.cert -CAkey ca.key \
-Acreateserial -days 365
```

We copied lnx4.key, lnx4.cert and ca.cert to the slave, and updated the TLSCertificateFile and TLSCertificateKeyFile entries in slapd.conf.

## Copying the data

Initially, the database files need to be copied from the master to the slave. The OpenLDAP software levels and configuration must be the same. Ensure that the master slapd is shut down or running in read-only mode. Copy all the files in the database directory to the slave. We created a tarfile with the contents of the directory and transferred that.

```
tar -cvf ldap.tar /var/lib/ldap
```

Transfer the tar file to the slave, and extract the files.

```
tar -xvf ldap.tar
```

## Activating the servers

Start LDAP on each system:

```
/etc/rc.d/ldap start
```

On the master, both slapd and slurpd should start. Now clients can be configured to send the LDAP requests to any of the LDAP servers.

**Problem:** During our testing, we encountered a problem with password changes using a slave server. When the request was referred to the master server, it appeared that the new connection did not use TLS. This meant that the password change failed because access was denied by our ACLs enforcing TLS. If we removed the `tls_ssf` requirement from our ACLs, password changes worked correctly.



## System Installation Suite

System Installation Suite (SIS) is a collaboration of three open source projects designed to work together to automate cloning and configuration of Linux images. It is typically used to set up clusters and server farms. Due to its modular design, SIS currently works on IA-32, IA-64, PPC, and s390 architecture, with support of more architectures on the way.

In this chapter we describe the features of SIS and identify some of the functional details of the components.

For more information about SIS, see:

<http://www.sisuite.org>

## 8.1 Features of SIS

The collaboration of System Installation Suite was established to achieve the following objectives:

### Architecture independence

SIS is modular in design. Support for a new architecture could be added easily without making modification to the existing frame work.

### Distribution independence

The user interface is consistent for all installations. Thus, the user doesn't have to be conscious about which distribution he is installing. Among the distributions supported by SIS are RedHat, SuSE, Debian, Turbo, Mandrake, Kondara, and many more.

**Note:** The differences in distributions are taken care by System Configurator under the cover. See "System Configurator" on page 150.

### Package- and image-based installation

Two types of installation methods currently exist. One is package-based (for example, RPM and dpkg), and the other is image-based. Package-based installation takes a list of predefined packages, and based on the list provided, proceeds with the installation. This is very useful for installation of a single system. The package manager takes care of dependencies among packages, so the user does not need to worry about software compatibility.

There are limitations to package-based installations, however, when it comes to installing a scalable number of systems, such as clusters and server farms. Package-based installations generally don't have an automated way for dealing with "non-packaged" files, such as customized kernel, configuration files, files that reside on /usr/local and /home directories (usually these are user-created files), and so on.

SIS is a hybrid of both package-based and image-based installation. As the first step of installation, the user defines a list of packages to install. *System Installer* takes the list and populates a user-defined directory (usually /var/lib/systemimager/images/*image\_name*) on a *System Imager* server by doing full package installation. When the installation of packages is completed, this directory contains an image of a full blown system. At this point the user can make modifications to the image, such as changing configuration files. Then, the image is cloned and installed on a number of systems with minimal interaction.

### More than a cluster installation tool

The image-based nature of SIS allows incremental updates of system images. This means SIS can be effectively used to back up a snapshot of a system image. SIS uses the rsync protocol as the underlying file transport mechanism. Since rsync only updates files that are changed, updating and backup is done with great efficiency.

### Automated installation

Once initial configuration for installation is defined, the whole installation process is completely unattended. All disk partitioning, setting up of networking, and detection of hardware is done automatically by components of System Installation Suite.

## 8.2 SIS components

The three components of SIS are System Installer, System Imager, and System Configurator. In this section we describe the functions of each component.

### System Installer

System Installer is responsible for creation of a system image. System Installer works in one of two ways:

- ▶ The first method is package-based installation of a “golden image.” In this method, System Installer uses a list of packages that define a system. It does the full installation of them under a directory defined by the user as the “image directory,” for example `/var/lib/systemimager/images/image_name`. The dependencies among the packages are checked by the appropriate package manager (RPM, dpkg, and so forth), and a system image is created under it.
- ▶ In the second method, the user first builds a “golden client” system using the native installer of the distribution he is using, or even using SIS. After the installation is complete and its image is hardened (such that the user feels confident to run production applications on it), the user invokes the System Installer to clone that working system so it can be copied into the image directory (`/var/lib/systemimager/images/image_name`) on System Imager.

When a system image is populated into the image directory, information about its disk partitions is also obtained. This information is used by System Imager to set up disk drives in the second step of installation.

### System Imager

System Imager automates the installation process and provides the image transfer mechanism. It is also responsible for setting disk partitions and file systems.

System Imager consists of a server and a client. A client is the node on which the installation takes place. System Imager server holds all the images populated by System Installer, plus partition schemes of all the disk devices.

Before the installation take place, the user has to specify the host name of the client, plus the name of the image (from `/var/lib/systemimager/images`) to install. This information is used to make an association between the host and an autoinstall script.

**Note:** An autoinstall script is a script created when an image is populated on the System Imager server. It is used to automate the installation process.

An installation process is initiated by the client as it boots up with System Imager initial ramdisk + kernel. The System Imager ramdisk holds a customized and stripped (due to the space constraint on IA-32 platform) system image. It uses BusyBox, an extremely light-weight shell used on embedded devices, and its utilities. The kernel contains all the modules necessary to bring up the network using either static IP or DHCP.

After the network is brought up, the autoinstall script is downloaded and executed to set up disks and create file systems. When disk setup is complete, the image on the server is pulled onto the client using the `rsync` utility. Then System Configurator is run.

**Note:** System Imager kernel + ramdisk is called BOEL, Brian’s Own Embedded Linux, name after the creator of System Imager.

**Note:** The medium on which System Imager initial ramdisk and kernel reside can vary depending on the user's environment. For instance, on an IA-32 system the user can boot from a floppy, CD-ROM, or even from the network card.

On s390, users are provided with three files to be punched into the VM reader. These are first stage ramdisk, second stage ramdisk, and kernel.

## System Configurator

System Configurator is executed after System Imager completes cloning of the image. Its main purpose is the configuration of network devices, bootloader, and hardware, independent of the architecture and Linux distribution it runs on.

System Configurator is extremely modular. It exports consistent API, such that support for an additional architecture and a new Linux distribution is fairly trivial. For example, when support for PPC architecture was added, the only major piece of work needed was to write a module to set up the bootloader, YaBoot.

Figure 8-1 is a simplified illustration of how these components fit together.

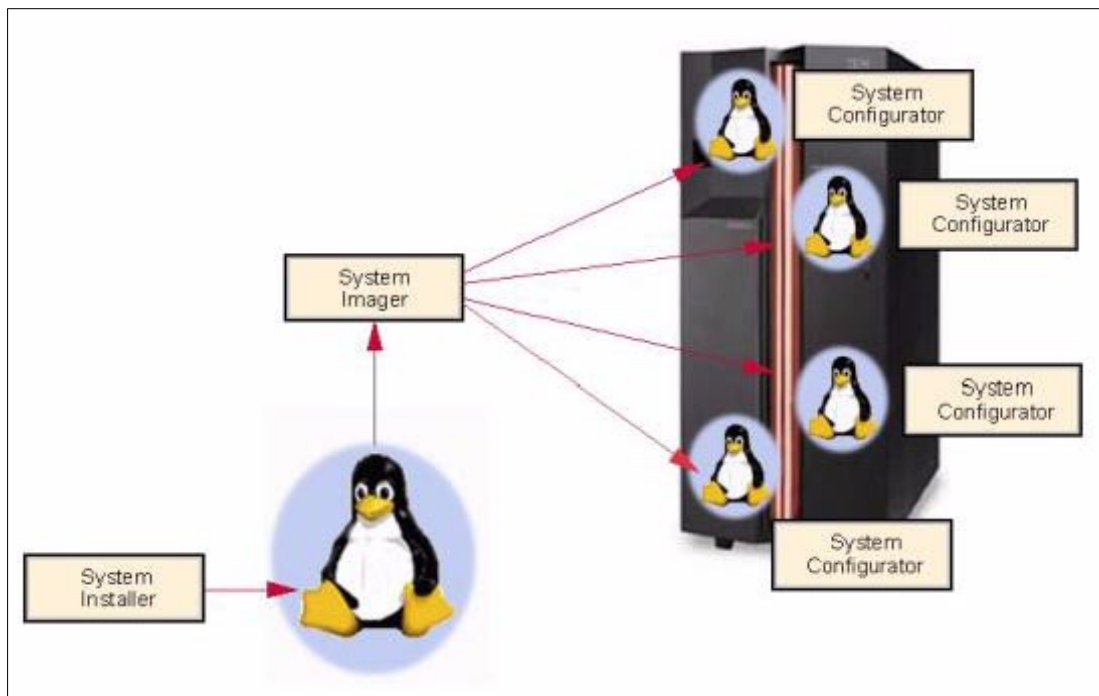


Figure 8-1 System Installation Suite

First System Installer creates a Linux image, which is then pushed onto a System Imager server. After System Imager client boots up with System Imager kernel and ramdisk, it pulls down the image from the server.

## 8.3 SIS on s390

In a typical Linux on s390 VM environment, there are tens to hundreds of Linux images running. Each image, or group of images, is deployed to serve a specific business need. For instance, there may be a number of images running DB2, another number of images running MySQL for in-house accounting, and some other number of images running WebSphere, and

so on. Frequent backups and incremental updates of each instance of images are crucial to minimizing loss of data and service downtime. Doing the necessary backups and updates, however, can be very time consuming. It can also be expensive. Think of all the DASD space required to store backup images. This is where SIS comes in handy.

SIS can be a cost-effective mechanism to back up all the instances of Linux images running on s390 VM systems. Since the image server, namely System Imager server, doesn't have to be running on an s390 system, the images can be backed up on inexpensive PC hard drives with as much redundancy as needed. Plus, since most of the process of SIS is automated, backing up or updating images is easy, and fast.

Furthermore, SIS is *free*! It is an open source project under GPL license. If you need to run it in a specialized environment, go ahead, change the code. The code is in clean and modular Perl. Just make sure to feed back the change.

**Note:** Note SIS on s390 is in beta state at the time of writing.

## 8.4 Obtaining SIS

The source files and documentation for the components of SIS can be found at:

<http://www.sisuite.org>





## Part 3



# Tivoli

The following Tivoli products are described in this part:

- ▶ IBM Tivoli Storage Manager
- ▶ IBM Tivoli Access Manager for e-business
- ▶ IBM Tivoli Identity Manager
- ▶ Tivoli Distributed Monitoring

The next two chapters describe how to set up a Tivoli environment, and how to use Tivoli software for system management.





## Setting up the IBM Tivoli environment

In this chapter we describe the IBM Tivoli products used to manage Linux/390.

Figure 9-1 shows where we deployed Tivoli and what we managed. All systems had a Tivoli endpoint installed; however, not all systems were managed while writing this book. We used three different “flavors” of Linux, and we did encounter some problems while using Tivoli on the distributions which were not supported as per the release notes. Time didn’t permit us to explore why some of the systems were not working as expected.

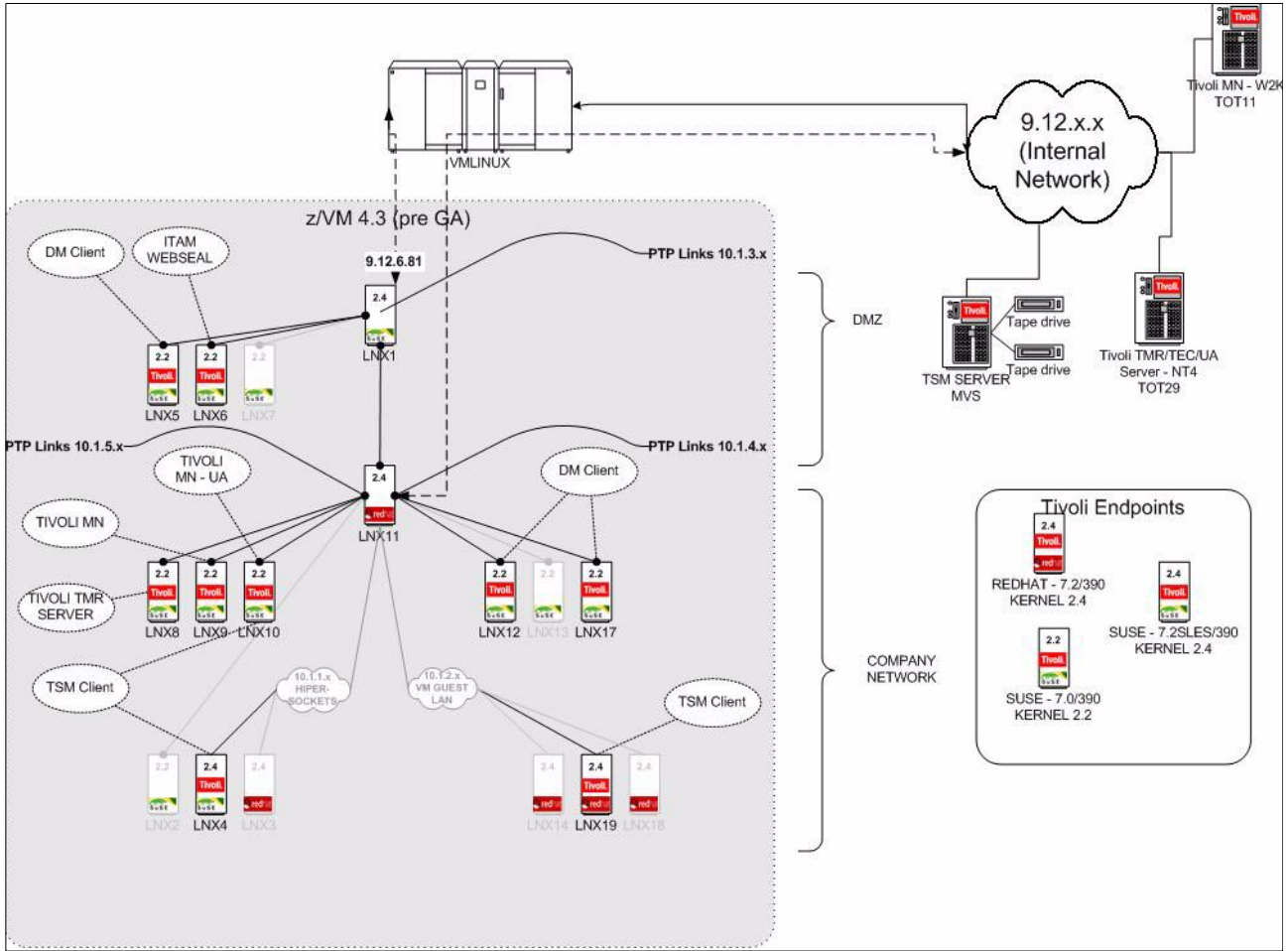


Figure 9-1 Tivoli Management infrastructure

## 9.1 Tivoli Storage Manager

IBM Tivoli Storage Manager (TSM) includes extensive and flexible centralized management, customizable and complete automation, broad cross-platform and storage device support, and smart data-transfer and storage technologies. These capabilities allow TSM to scale up to manage, administer, and automate data protection for any size organization. The base TSM provides two important data protection functions:

► Backup/Restore

Data *backups* are copies of your active online data stored on offline storage. Should an online storage device fail, a data error occur, or someone accidentally delete a file, the offline copy of that data can be copied back to online storage – or *restored*. IBM Tivoli Storage Manager includes multiple techniques to reduce data transfer sizes to their minimums to make backups and restores as fast as possible.

► Archive/Retrieve

Some percentage of your data is *inactive*. Other data must have copies kept for many years. IBM Tivoli Storage Manager will move that data to much less expensive offline storage, the *archive*, to free online disk space for more important active data or to keep copies available for other reasons. If you need that data again, IBM Tivoli Storage Manager will retrieve it for you.

TSM is currently supported on Linux/390 systems only as a TSM client with the server portion of TSM running on another platform (such as AIX, MVS, Windows, and so forth). In the future, the server portion of TSM will also run on Linux/390 (and Linux/Intel). The current supported server platforms are described in the TSM release notes.

For the purpose of this writing, we assume that you have a TSM server somewhere available to you and that it is currently up, running, and operational.

It is recommended that you get the latest Tivoli Storage Manager client, which is available from:

[http://www.tivoli.com/support/storage\\_mgr/clients.html](http://www.tivoli.com/support/storage_mgr/clients.html)

At the time of this writing, the TSM v4.20 client was available for Linux/390. Generally speaking, the TSM client version you use can be earlier or later than the TSM server version that you use. Review the *README* that comes with the TSM client to make sure you will have a compatible and interoperable client with your version of the TSM server.

In our environment, we set up and successfully backed up the following Linux/390 systems:

Table 9-1 TSM clients

System	Linux distribution	Kernel version
LNX4	SUSE 7.0 SLES	2.4.7
LNX10	SUSE 7.0	2.2.16
LNX19	RedHat 7.2	2.4.7

We backed up our Linux/390 clients to an MVS v4.2.1 TSM server that was running on another host in our 9.12.x.x network.

**Restriction:** While the TSM Client *Release Notes* indicated that the client was only supported on SuSE 7.0 with a 2.2 kernel, it appeared to operate without problems on a RedHat 7.2 with a 2.4 kernel, and the SuSE 7.0/SLES 2.4 kernel. We recommend that you check for a later version of the client that is supported for RedHat and/or 2.4 kernels, or at least check with Tivoli support.

See “Configuring the TSM client” on page 158 for information about recognized file systems and how to back up unrecognized ones.

## 9.1.1 TSM Backup/Archive client

### Installing the TSM client

Download the latest TSM client (it is normally available as a compressed tar archive) and put it into your /tmp directory.

**Tip:** The TSM client was compiled with some older Linux libraries that are normally on Linux 2.2 systems. These libraries are normally available for 2.4 systems via a *compat* RPM on your distribution CD.

- ▶ For RedHat 7.2 the compat RPM is called `compat-libstdc++-2.10.0-1.s390.rpm`.
- ▶ For SuSE 7.0/SLES the compat RPM is called `compat-2001-10-30-0.s390.rpm`.

For the SuSE 7.0 systems (Linux 2.2) the required libraries will probably have been installed already (from the `gppshare-2.95.3.s390.rpm`).

Make sure these libraries are installed before you continue with the TSM installation.

Example 9-1 shows the steps that we performed to install the Linux/390 TSM client on our RedHat 7.2 system. In a later section (10.1.2, “Software deployment with Tivoli Software Distribution” on page 218) we describe how to deploy the TSM Linux/390 client on all our Linux systems.

#### *Example 9-1 Installation of the TSM Client on RedHat 7.2*

---

```
[root@lnx19 /tmp]# tar xzf TSM420_LINUX390.tar.Z
[root@lnx19 /tmp]# cd linux390
[root@lnx19 linux390]# rpm -ivh TIVsm-BA.s390.rpm
Preparing...          ##### [100%]
   1:TIVsm-BA          ##### [100%]
Postinstall of the Backup Archive client

TSM Linux client installation complete.

Be sure to set up the system configuration file
before starting the client!
[root@lnx19 /tmp]#
```

---

### Configuring the TSM client

Once the TSM client is installed, you need to create the configuration files `dsm.sys` and `dsm.opt` with details about what will be backed up and where to located the TSM server. These files need to be placed in the `/opt/tivoli/tsm/client/ba/bin` directory. Example 9-2 and Example show what was used for all Linux images in our environment.

### Example 9-2 TSM Client Configuration *dsm.sys*<sup>1</sup>

---

```
SErvername          TSM_SERVER
  COMMmethod        TCPip
  ERRORLOGName      /var/log/tsmerror.log
  ERRORLOGRetention 14D
  MANAGEDServices   webclient schedule
* NODename          lnx19
  PASSWORDAccess    generate
  SCHEDLOGName      /var/log/tsmsched.log
  SCHEDLOGRetention 14D
  SCHEDMODE         P0lling
  TCPPort           1500
  TCPServeraddress  tsm.itso.ibm.com
  VIRTUALMountpoint /
  exclude           /var/log/tsm*.log
```

---

**Important:** The TSM 4.2.0 client can only recognize *ext2* and *nfs* file systems. If you are using other types of file systems (for example, *ext3*, *reiserfs*, *jfs*, or *xfs*), you need to add *VIRTUALMountpoint* statements in your *dsm.sys* file for each mount point that is not *ext2* or *nfs* (if you want to back up those file systems).

### Example 9-3 TSM Client Configuration *dsm.opt*

---

```
DATEformat          12
SUBdir              Yes
```

---

Review your TSM Client documentation (or launch the TSM client interactively and type `help`) to get an explanation of these options.

**Tip:** Try and keep your *dsm.opt* and *dsm.sys* files as generic as possible. This will enable you to distribute the same files to all your virtual Linux systems (or where you have shared read-only file systems, place the option files on that shared file system).

The TSM server also enables you to *profile* clients and give common configuration options to similar clients via the *client option sets*. Refer to your TSM documentation for more details on this feature.

## Starting the TSM client

TSM can back up your system interactively by executing the following command:

```
dsmc i
```

The session will be displayed to your screen as the backup is processed, with a summary of the session upon completion.

Alternatively, by starting the scheduler daemon (which is the most common way to use TSM) you can let the TSM server determine when the system is backed up. The simplest way of using the scheduler is by putting the following in your */etc/rc.d/rc.local* script:

```
/opt/tivoli/tsm/client/ba/bin/dsmcad
```

The *rc.local* script is called when the system is booted, and this line will ensure that the TSM scheduler client is started at that time. (This daemon automatically releases from the terminal, so there is no need to background this process.)

<sup>1</sup> The \* at the beginning of a line indicates that the line is a comment.

<sup>2</sup> For those that use a date format of DD-MM-YYYY, use 2 instead of 1

If the daemon is not running, you can start it by entering the command `dsmc i` while logged in as root.

## 9.1.2 Installation and configuration of the TSM Web client

The TSM Web client allows administrators to control the backup/restore and archive/retrieve functions using a Web browser. Thus these administrators can be located anywhere in your network and only need access to the TSM client Web port on your virtual machines.

Your TSM administrators can also have a link from the TSM Server Web interface to each of the clients by updating the URL field of the node configuration. Figure 9-2 shows an example of the TSM Web administration page. Our Linux systems with the lightning bolt next to them have the URL field updated, and clicking on the lightning bolt will take us to the client's TSM Web interface.

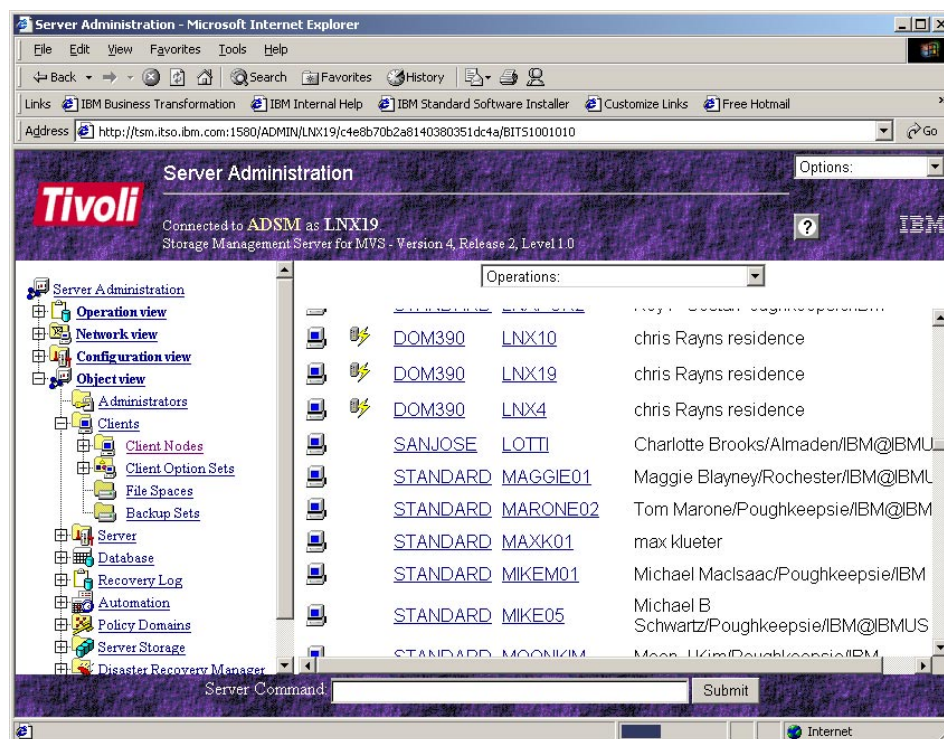


Figure 9-2 TSM server Web administration

The Web client is provided as part of the normal backup/archive client and is called via the `dsmcad` daemon and the configuration setting of the `MANAGEDServices` option in your `dsm.sys` configuration file. There are no additional installation steps necessary.

If you only use the `dsmcad` daemon to launch the TSM client services (as described previously), then the Web client can be enabled or disabled by updating the `MANAGEDServices` section of your `dsm.sys` configuration file, adding or removing the `webclient` option as desired (our configuration file is shown in Example 9-2 on page 159).

## 9.2 IBM Directory Server

IBM Directory Server (previously called IBM Secureway Directory) provides a powerful Lightweight Directory Access Protocol (LDAP) identity infrastructure that is the foundation for



deploying comprehensive identity management applications and advanced software architectures like Web services.

## 9.2.1 Installation

The installation followed the procedures listed in the `lparent.pdf` file that comes with the product. Those steps are highlighted here. The operating system was SuSE Linux/390 7.0 (2.2.16 kernel).

The first thing to do is remove other installed LDAP packages. We neglected to do this when installing the product, and luckily, there were no problems.

The first package that we installed was the IBM Secureway Directory Client for Linux/390. There are two packages. The `ldap-client` uses 56-bit encryption and the `ldap-clientd` uses 128-bit encryption. We installed only the 128-bit encryption package.

### *Example 9-4 Install LDAP client*

---

```
rpm -hiv ldap-clientd 3.2.2-1.s390.rpm
```

---

We modified `/etc/profile.local` to include the entries shown in Example 9-5.

### *Example 9-5 Set environment variables for LDAP client*

---

```
export NLSPATH=/usr/share/i18n/en_US
export LANG=en_US
export LC_ALL=en_US
```

---

Next, we installed the IBM Global Security Kit.

### *Example 9-6 Install IBM Global Security Toolkit*

---

```
rpm -hiv gsk5bas-5.0.4-67.s390.rpm
```

---

We then installed IBM DB2 version 7.1. We followed the instructions given in the *IBM Secureway Directory Version 3.2.2 for Linux: Installation, Configuration, and Administration Guide*. The only exceptions to the installation instructions were that we included the DB2 Connect Enterprise Edition component in addition to those required by the installation guide, and we added the following to `~/.profile` instead of `~/.bashrc`

### *Example 9-7 Set environment variables for LDAP DB2 instance*

---

```
export DB2INSTANCE=ldapdb2
export LD_LIBRARY_PATH=/usr/IBMDB2/V7.1/lib:/usr/ldap/lib:$LD_LIBRARY_PATH
```

---

After installing DB2, we installed the IBM Directory Server.

### *Example 9-8 Install LDAP server*

---

```
rpm -hiv ldap-serverd-3.2.2-1.s390.rpm
rpm -hiv ldap-msg_en_US-3.2.2-1.s390.rpm
slapd
```

---

## 9.2.2 Configuration

Our server configuration was simply the default. ACLs on attributes would need to be set in a production environment. For information on setting ACLs, see the section on OpenLDAP. Here is our `/usr/ldap/etc/slapd32.conf` file:

*Example 9-9 /usr/ldap/etc/slapd32.conf*

---

```
# IBM SecureWay Directory Server Configuration File Version 3.2 for Linux
#
#
dn: cn=Configuration
objectclass: top
objectclass: ibm-slapdTop
cn: Configuration
# The default administrator DN may be changed
ibm-slapdAdminDn: cn=root
# You MUST set the administrator password
ibm-slapdAdminPW: ibmdb2
#ibm-slapdSysLogLevel must be one of l/m/h (l=terse, h=verbose)
ibm-slapdSysLogLevel: m
ibm-slapdErrorLog: /tmp/slapd.errors
ibm-slapdPort: 389
ibm-slapdTimeLimit: 900
ibm-slapdSizeLimit: 500
#ibm-slapdPwEncryption must be one of none/imask/crypt/sha
ibm-slapdPwEncryption: imask

dn: cn=Front End, cn=Configuration
objectClass: top
objectClass: ibm-slapdFrontEnd
cn: Front End
#

dn: cn=Schemas,cn=Configuration
objectclass: top
objectclass: container
cn: Schemas

dn: cn=IBM SecureWay,cn=Schemas,cn=Configuration
objectclass: top
objectclass: ibm-slapdSchema
cn: IBM SecureWay
ibm-slapdIncludeSchema: /etc/ldapschema/V3.system.at
ibm-slapdIncludeSchema: /etc/ldapschema/V3.ibm.at
ibm-slapdIncludeSchema: /etc/ldapschema/V3.user.at
ibm-slapdIncludeSchema: /etc/ldapschema/V3.system.oc
ibm-slapdIncludeSchema: /etc/ldapschema/V3.ibm.oc
ibm-slapdIncludeSchema: /etc/ldapschema/V3.user.oc
ibm-slapdIncludeSchema: /etc/ldapschema/V3.ldapsyntaxes
ibm-slapdIncludeSchema: /etc/ldapschema/V3.matchingrules
ibm-slapdSchemaAdditions: /etc/ldapschema/V3.modifiedschema
#ibm-slapdSchemaCheck must be one of V2/V3/V3_lenient
ibm-slapdSchemaCheck: V3_lenient

dn: cn=LDCF Backends,cn=IBM SecureWay,cn=Schemas,cn=Configuration
objectclass: top
objectclass: container
cn: LDCF Backends
```

```
dn: cn=SchemaDB,cn=LDCF Backends,cn=IBM SecureWay,cn=Schemas,cn=Configuration
objectclass: top
objectclass: ibm-slapdLdcfBackend
cn: SchemaDB
ibm-slapdSuffix: cn=schema
```

```
dn: cn=RDBM Backends,cn=IBM SecureWay,cn=Schemas,cn=Configuration
objectclass: top
objectclass: container
cn: RDBM Backends
```

```
dn: cn=Directory,cn=RDBM Backends,cn=IBM SecureWay,cn=Schemas,cn=Configuration
objectclass: top
objectclass: ibm-slapdRdbmBackend
cn: Directory
# The following attributes must match the the database being used
ibm-slapdDbInstance: ldapdb2
ibm-slapdDbName: ldapdb2
ibm-slapdDbUserId: ldapdb2
# You MUST set the DB2 user password
ibm-slapdDbUserPW: ibmdb2
# The following suffix is used by /usr/ldap/examples/sample.ldif
ibm-slapdSuffix: o=ibm,c=us
ibm-slapdPlugin: database /lib/libback-rdbm.so rdbm_backend_init
ibm-slapdDbConnections: 6
ibm-slapdSuffix: cn=localhost
ibm-slapdReadOnly: FALSE
```

---

To enable Linux to use the LDAP directory for login authentication, we needed to add the RFC 2307 object classes and attributes, and needed to alter some system files. The system files that we modified are shown in Example 9-10 through Example 9-13.

*Example 9-10 /etc/nsswitch.conf*

---

```
passwd: compat ldap
group: compat ldap
shadow: compat ldap
```

---

*Example 9-11 /etc/pam.d/login*

---

```
##PAM-1.0
auth sufficient /lib/security/pam_ldap.so
auth requisite /lib/security/pam_unix.so nullok #set_secrcp
#auth required /lib/security/pam_securetty.so
auth required /lib/security/pam_nologin.so
#auth required /lib/security/pam_homecheck.so
auth required /lib/security/pam_env.so
auth required /lib/security/pam_mail.so
account sufficient /lib/security/pam_ldap.so
account required /lib/security/pam_unix.so
password sufficient /lib/security/pam_ldap.so
password required /lib/security/pam_pwcheck.so nullok
password required /lib/security/pam_unix.so nullok use_first_pass use_authtok
session sufficient /lib/security/pam_ldap.so
session required /lib/security/pam_unix.so none # debug or trace
session required /lib/security/pam_limits.so
```

---

*Example 9-12 /etc/pam.d/passwd*

---

```
##PAM-1.0
auth    sufficient /lib/security/pam_ldap.so    nullok
auth    required   /lib/security/pam_unix.so      nullok
account sufficient /lib/security/pam_ldap.so    nullok
account required   /lib/security/pam_unix.so
password sufficient /lib/security/pam_ldap.so    nullok
password required   /lib/security/pam_pwcheck.so  nullok
password required   /lib/security/pam_unix.so    nullok use_first_pass use_authtok
session sufficient /lib/security/pam_ldap.so    nullok
session required   /lib/security/pam_unix.so
```

---

*Example 9-13 /etc/pam.d/su*

---

```
##PAM-1.0
auth    sufficient /lib/security/pam_rootok.so
auth    sufficient /lib/security/pam_ldap.so
auth    required   /lib/security/pam_unix.so    nullok #set_secrc
account sufficient /lib/security/pam_ldap.so
account required   /lib/security/pam_unix.so
password sufficient /lib/security/pam_ldap.so
password required   /lib/security/pam_unix.so
session sufficient /lib/security/pam_ldap.so
#session required /lib/security/pam_homecheck.so
session required   /lib/security/pam_unix.so    none # debug or trace
```

---

The object classes and attributes were added to our server schema using the Directory Management Tool (DMT) that ships with IBM Directory Server. This tool was installed on a Windows platform and connected to the IBM Directory Server via TCP/IP.

The object classes added were:

- ▶ posixAccount
- ▶ posixGroup
- ▶ shadowAccount

The attributes added were:

- ▶ uidnumber
- ▶ gidnumber
- ▶ memberuid
- ▶ shadowExpire
- ▶ shadowFlag
- ▶ shadowInactive
- ▶ shadowLastChange
- ▶ shadowMax
- ▶ shadowMin
- ▶ shadowWarning

The Add attribute window is shown in Figure 9-3 on page 165.

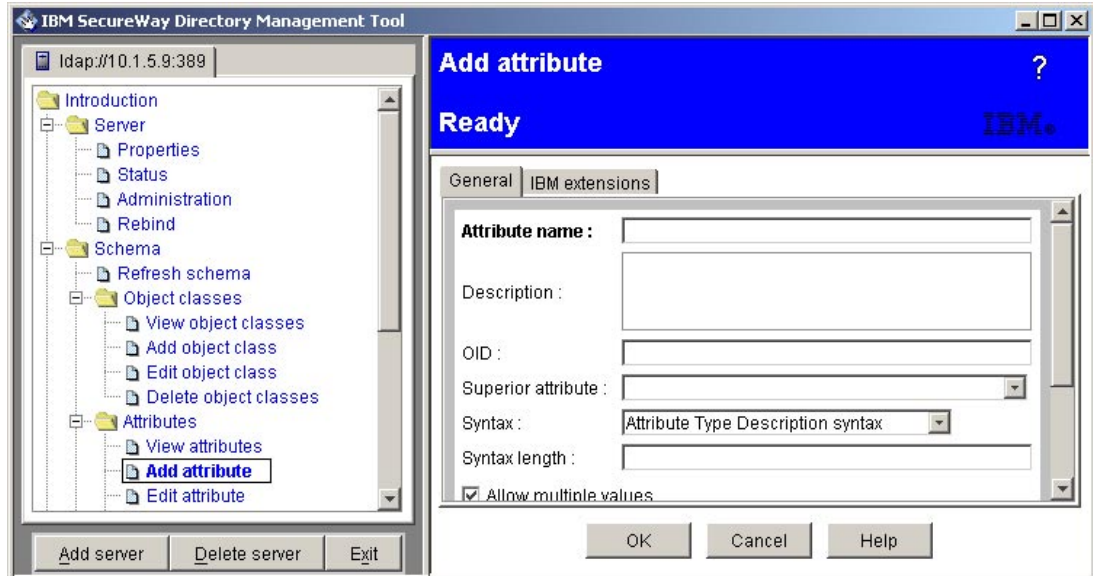


Figure 9-3 IBM Directory Server Management Tool (DMT) Add attribute

Each attribute was added and could then be seen under the View attributes link. The attributes added are shown in Example 9-14 through Example 9-23.

*Example 9-14 uidnumber attribute*

---

```
uidnumber
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :UIDNUMBER
    DBTableName :UIDNUMBER
    LENGTH :240
  Attribute name : 'uidnumber'
  OID : uidnumber-OID
  Syntax : Integer syntax - Integral number
```

---

*Example 9-15 gidnumber attribute*

---

```
gidnumber
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :GIDNUMBER
    DBTableName :GIDNUMBER
    LENGTH :240
  Attribute name : 'gidnumber'
  OID : gidnumber-OID
  Syntax : Integer syntax - Integral number
```

---

*Example 9-16 memberuid attribute*

---

```
memberuid
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :MEMBERUID
    DBTableName :MEMBERUID
    LENGTH :240
  Attribute name : 'memberuid'
  OID : memberuid-OID
  Syntax : Integer syntax - Integral number
```

---

*Example 9-17 shadowExpire attribute*

---

```
shadowExpire
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :SHADOWEXPIRE
    DBTableName :SHADOWEXPIRE
    LENGTH :240
  Attribute name : 'shadowExpire'
  OID : shadowExpire-OID
  Syntax : Attribute Type Description syntax
```

---

*Example 9-18 shadowFlag attribute*

---

```
shadowFlag
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :SHADOWFLAG
    DBTableName :SHADOWFLAG
    LENGTH :240
  Attribute name : 'shadowFlag'
  OID : shadowFlag-OID
  Syntax : Attribute Type Description syntax
```

---

*Example 9-19 shadowInactive attribute*

---

```
shadowInactive
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :SHADOWINACTIVE
    DBTableName :SHADOWINACTIVE
    LENGTH :240
  Attribute name : 'shadowInactive'
  OID : shadowInactive-OID
  Syntax : Attribute Type Description syntax
```

---

*Example 9-20 shadowLastChange attribute*

---

```
shadowLastChange
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :SHADOWLASTCHANGE
    DBTableName :SHADOWLASTCHANGE
    LENGTH :240
  Attribute name : 'shadowLastChange'
  OID : shadowLastChange-OID
  Syntax : Attribute Type Description syntax
```

---

*Example 9-21 shadowMax attribute*

---

```
shadowMax
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :SHADOWMAX
    DBTableName :SHADOWMAX
    LENGTH :240
  Attribute name : 'shadowMax'
  OID : shadowMax-OID
  Syntax : Attribute Type Description syntax
```

---

*Example 9-22 shadowMin attribute*

---

```
shadowMin
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :SHADOWMIN
    DBTableName :SHADOWMIN
    LENGTH :240
  Attribute name : 'shadowMin'
  OID : shadowMin-OID
  Syntax : Attribute Type Description syntax
```

---

*Example 9-23 shadowWarning attribute*

---

```
shadowWarning
  ibm attribute fields
    ACCESS-CLASS :NORMAL
    DBColumnName :SHADOWWARNING
    DBTableName :SHADOWWARNING
    LENGTH :240
  Attribute name : 'shadowWarning'
  OID : shadowWarning-OID
  Syntax : Attribute Type Description syntax
```

---

The object classes were added with DMT using the [/Introduction/Schema/Object classes/Add object class link](#).

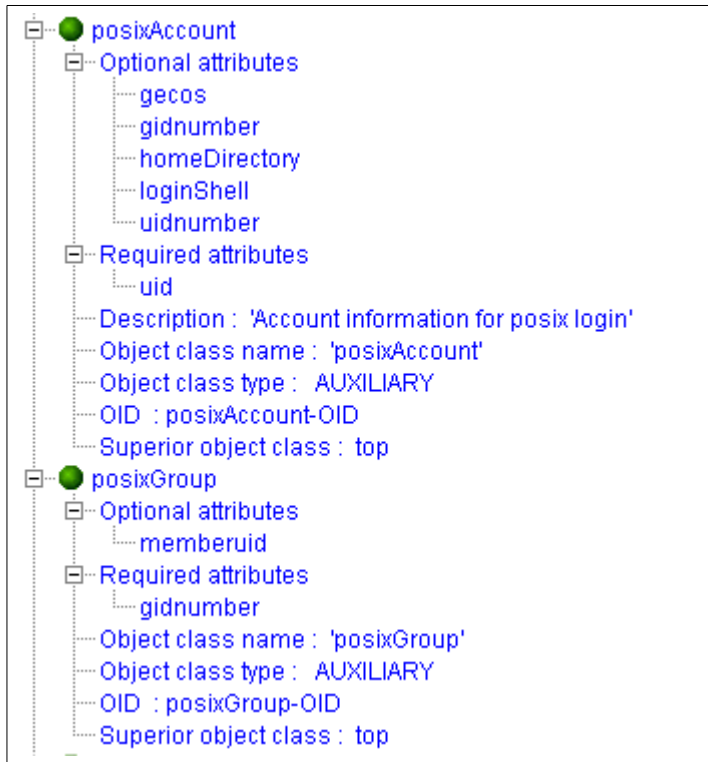


Figure 9-4 *posixAccount and posixGroup object classes*



Figure 9-5 *shadowAccount object class*

**Tip:** The following websites have good information on system authentication and LDAP:

<http://www.skills-1st.co.uk/papers/security-with-ldap-jan-2002/security-with-ldap.html>  
[http://staff.pisoftware.com/bmarshal/publications/system\\_auth/sage-au/system\\_auth.html](http://staff.pisoftware.com/bmarshal/publications/system_auth/sage-au/system_auth.html)  
<http://www.tldp.org/HOWTO/LDAP-Implementation-HOWTO/index.html>



## 9.3 IBM Tivoli Access Manager for e-business

Tivoli Access Manager for e-business allows you to define a comprehensive policy and administer security based on that policy – giving your employees, partners, suppliers, and customers specific access based on each user’s responsibilities. You can group users and assign permissions to groups, simplifying administration of access control across multiple applications and resources. And for those applications that require it, there is support for dynamic roles, dynamic business entitlements, and authorization decisions based on external data.

Tivoli Access Manager for e-business helps:

- ▶ Centrally define and manage security policy for a broad range of e-business applications and other managed resources.
- ▶ Provide flexible SSO to Web-based applications that can span multiple sites or domains with a range of SSO options, including a highly secure Web SSO lockbox.
- ▶ Leverage a common security policy model with the Tivoli Access Manager family of products to extend support to other resources such as MQSeries applications and UNIX system resources.
- ▶ Secure your e-business environments that have demanding scalability requirements across a broad range of platforms.

### Product layout

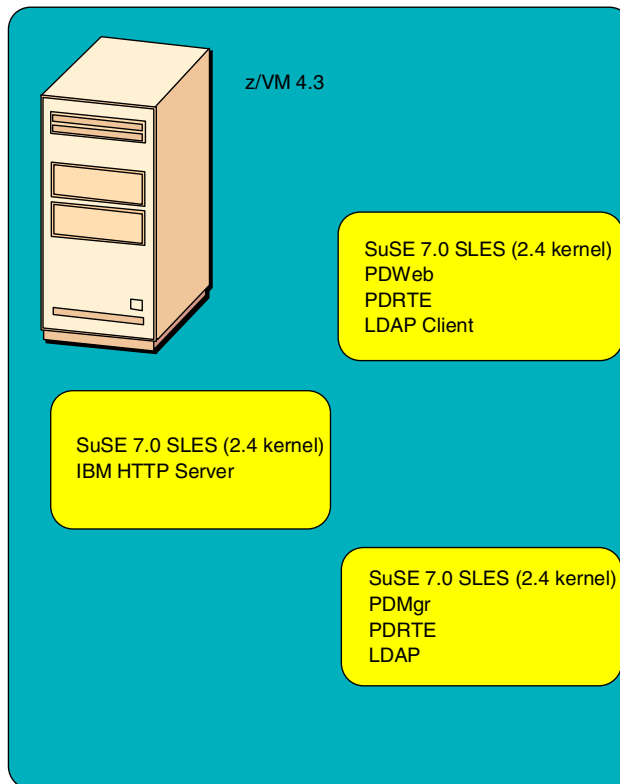


Figure 9-6 IBM Tivoli Access Manager for e-business layout

Our Access Manager for e-business layout consisted of three VMs. One contained the Policy Server (PDMgr) to which Access Manager blades can authenticate, one contained the

WebSEAL (PDWeb) blade, and then we had one with IBM HTTP Server. All VMs were running SuSE 7.0 SLES (2.4 kernel).

## Installation

The IBM HTTP Server VM did not need any Access Manager components to be installed, so these instructions apply to only the WebSEAL and Policy Server VMs.

Our install of SuSE SLES did not have the compat prerequisite installed, so we installed that first, with the command shown in Example 9-24.

### *Example 9-24 Install prerequisite compat package*

---

```
rpm -hiv compat-2001.10.30-0.s390.rpm
```

---

We installed the IBM Global Security Kit (GSKit) for SSL/TLS functionality, the Access Manager Run Time Environment (PDRTE), and the IBM Directory Server client. We removed another LDAP package that conflicted with the IBM Directory Server client. These steps are shown in Example 9-25.

### *Example 9-25 Install PDRTE and prerequisites*

---

```
rpm -hiv gsk5bas-5.0-4.67.s390.rpm
echo 'export LD_PRELOAD=/usr/lib/libstdc++-libc6.1-2.so.3' >> /etc/profile.local
. /etc/profile.local
rpm -ev nss_ldap
rpm -hiv ldap-clientd-3.2.2-1.s390.rpm
rpm -hiv PDRTE-PD-3.9.0-0.s390.rpm
```

---

We installed IBM Directory Server on the Policy Server VM, following instructions similar to the ones outlined in the IBM Directory Server section in this chapter. In addition to setting the passwords in `/usr/ldap/etc/slapd32.conf`, we added the entry in Example 9-26 to the last section in the file.

### *Example 9-26 Addition to /usr/ldap/etc/slapd32.conf for Access Manager*

---

```
ibm-slapdSuffix: secAuthority=Default
```

---

We also modified `/etc/init.d/ldap` since that script started the OpenLDAP server that ships with SuSE SLES. Our new `/etc/init.d/ldap` file looked like Example 9-27.

*Example 9-27 /etc/init.d/ldap*

---

```
#!/bin/sh
#
#
# /etc/init.d/ldap
#
case "$1" in
  start)
    echo "Starting IBM Directory Server"
    if [ -x /usr/bin/slapd ] ; then
      /usr/bin/slapd
    fi
    ;;
  stop)
    echo "Shutting down ldap-server:"
    if [ -f /etc/slapd.pid ] ; then
      slapd_pid=`cat /etc/slapd.pid`
      kill -9 $slapd_pid
    fi
    ;;
  restart)
    ## Stop the service and regardless of whether it was
    ## running or not, start it again.
    $0 stop
    $0 start
    ;;
  *)
    echo "Usage: $0 {start|stop|restart}"
    exit 1
esac
```

---

**Policy Server**

We then installed the Policy Server (PDMgr).

*Example 9-28 Install PDMgr*

---

```
rpm -hiv PDMgr-PD-3.9.0-0.s390.rpm
```

---

We ran the pdconfig utility that comes with PDRTE to configure our run-time environment on our Policy Server. This is shown in Example 9-29.

*Example 9-29 Configure PDRTE on PDMgr system*

---

Access Manager for e-business Configuration Menu

1. Access Manager Runtime Configuration
2. Access Manager Policy Server Configuration
- x. Return to Access Manager for e-business Setup Menu

Please select the menu item [x]: **1**

Enter the LDAP server hostname: **1nx7**

Enter the LDAP server port number [389]: **389**  
This package has been successfully configured.

Press <enter> to continue ...

---

We used the same tool to configure the Policy Server itself.

*Example 9-30 Configure PDMgr*

---

Access Manager for e-business Configuration Menu

- 1. Access Manager Policy Server Configuration
- x. Return to Access Manager for e-business Setup Menu

Please select the menu item [x]: **1**  
Enter the LDAP administrative user DN [cn=root]: **cn=root**  
Enter the LDAP administrative user password: **ibmdb2**  
Access Manager Policy Server and the LDAP server (y/n) [Yes]? **n**  
Enter the LDAP DN for GSO database: **o=ibm,c=us**

You are required to provide a password for the  
Access Manager Administrator account.  
The administrator login name is sec\_master and cannot be changed.

Enter the password for the Access Manager Administrator: **AMeb3.9**  
Re-enter the password for confirmation:

Enter the SSL server port for Access Manager Policy Server [7135]:  
Enter the Policy Server SSL certificate lifetime [365]:

Selecting the Enable root CA Certificate download option simplifies the  
configuration of the Runtime on subsequent machines. Enabling this option  
may introduce a security exposure if a non-trusted host can impersonate the  
Access Manager Policy Server in the network.  
Enable root CA Certificate download (y/n) [No]? **n**

\* Configuring server

Generating Server Certificates, please wait.

Creating the SSL certificate. This may take several minutes...

The SSL configuration of the Access Manager Policy Server has completed successfully.  
The Policy Server's signed SSL certificate is base-64 encoded and saved in text file  
/var/PolicyDirector/keytab/pdcacert.b64  
This file is required by the configuration program on each machine in your  
secure domain.

SSL Configuration completed successfully

\* Starting server

Access Manager Policy Server v3.9.0 (Build 020412)

Copyright (C) IBM Corporation 1994-2002. All Rights Reserved.

2002-05-13-21:44:02.516+00:00I----- 0x1354A0A0 pdmgrd NOTICE ivc general ivmgrd.cpp 710  
0x00000400  
Server startup  
2002-05-13-21:44:02.517+00:00I----- 0x1354A0A0 pdmgrd NOTICE ivc general ivmgrd.cpp 715  
0x00000400  
Loading configuration  
This package has been successfully configured.  
Press <enter> to continue ...

---

### **WebSEAL Server**

Before installing the WebSEAL Server package (PDWeb), we had to configure our run-time environment. Before doing that, we copied the /var/PolicyDirector/keytab/pdcacert.b64 file from the Policy Server to the WebSEAL Server.

#### *Example 9-31 Copy SSL/TLS encryption key file*

---

```
scp 1nx7:/var/PolicyDirector/keytab/pdcacert.b64 /var/PolicyDirector/keytab/pdcacert.b64
pdconfig
```

---

Then we configured PDRTE, as shown in Example 9-32.

#### *Example 9-32 Configure PDRTE on WebSEAL system*

---

```
Access Manager for e-business Configuration Menu

1. Access Manager Runtime Configuration
x. Return to Access Manager for e-business Setup Menu

Please select the menu item [x]: 1

Will the Access Manager Policy Server be installed on this machine (y/n) [No]: n

Enter the LDAP server hostname: 1nx7

Enter the LDAP server port number [389]: 389

Enter the hostname of the Policy Server machine: 1nx7

Enter the SSL listening port used by Policy Server [7135]: 7135

The file containing the Access Manager CA certificate is required.
This file is created during the Policy Server configuration

Enter the name of the file containing the CA certificate:
/var/PolicyDirector/keytab/pdcacert.b64
This package has been successfully configured.

Press <enter> to continue ...
```

---

Once PDRTE was configured, we could install and configure the WebSEAL package, as shown in Example 9-33.

#### *Example 9-33 Configure WebSEAL*

---

```
#rpm -hiv PDWeb-PD-3.9.0-0.s390.rpm
#pdconfig
...
Access Manager for e-business Configuration Menu

1. Access Manager WebSEAL Configuration
x. Return to Access Manager for e-business Setup Menu

Please select the menu item [x]: 1
Enter the password for the Access Manager Administrator: AMeb3.9
Do you want to enable SSL communication between the
Access Manager server and the LDAP server (y/n) [Yes]? n
Please check Web Server configuration:

1. Enable TCP HTTP?          Yes
```

- 2. HTTP Port 80
- 3. Enable HTTPS? Yes
- 4. HTTPS Port 443
- 5. Web document root directory /opt/pdweb/www/docs

a. Accept configuration and continue with configuration

x. Exit configuration

Select item to change: **a**

\* Configuring the Web Server

Configuration of server webseald is in progress. This may take several minutes...

SSL configuration has completed successfully for the server.

\* Starting server

Access Manager WebSEAL Version 3.9.0 (Build 020412)  
 Copyright (C) IBM Corporation 1994-2002. All Rights Reserved.  
 ERROR: Could not start server

Press <enter> to continue ...

---

The error shown was due to an Apache server having bound itself to port 80 before the WebSEAL server could. We killed the Apache server, removed its links in the rc directories, and then started WebSEAL. There were no problems after removing Apache.

*Example 9-34 Disable Apache and verify WebSEAL*

---

```
# /etc/rc.d/apache stop
# rm /etc/init.d/rc?.d/???apache
# /etc/rc.d/pdweb start
# netstat -lpn
...
tcp      0      0 0.0.0.0:80          0.0.0.0:*          LISTEN   1848/webseald
...
```

---

## 9.4 IBM Tivoli Identity Manager

IBM Tivoli Identity Manager provides a secure, automated, and policy-based user management solution that helps address key business issues across both legacy and e-business environments. It includes tools to get users online and productive quickly. Tivoli Identity Manager provides:

- ▶ An intuitive Web administrative interface
- ▶ A sophisticated role-based administration model for delegation of administrative privileges
- ▶ Web self-service and challenge/response interfaces
- ▶ An embedded workflow engine for automated submission and approval of user requests
- ▶ An application management toolkit for extending the management model to new and custom environments

## 9.4.1 Component layout

For the IBM Tivoli Identity Manager examples, the Tivoli Identity Director 1.1 product was used. Since the Tivoli Identity Director 1.1 server cannot be installed on Linux/390, we installed the server component on an x86 system running Windows NT Server 4.0 SP6a.

One Linux/390 image was configured as an endpoint gateway, and a Tivoli Management Agent (TMA) was placed on that image.

We installed only a TMA on another Linux/390 image.

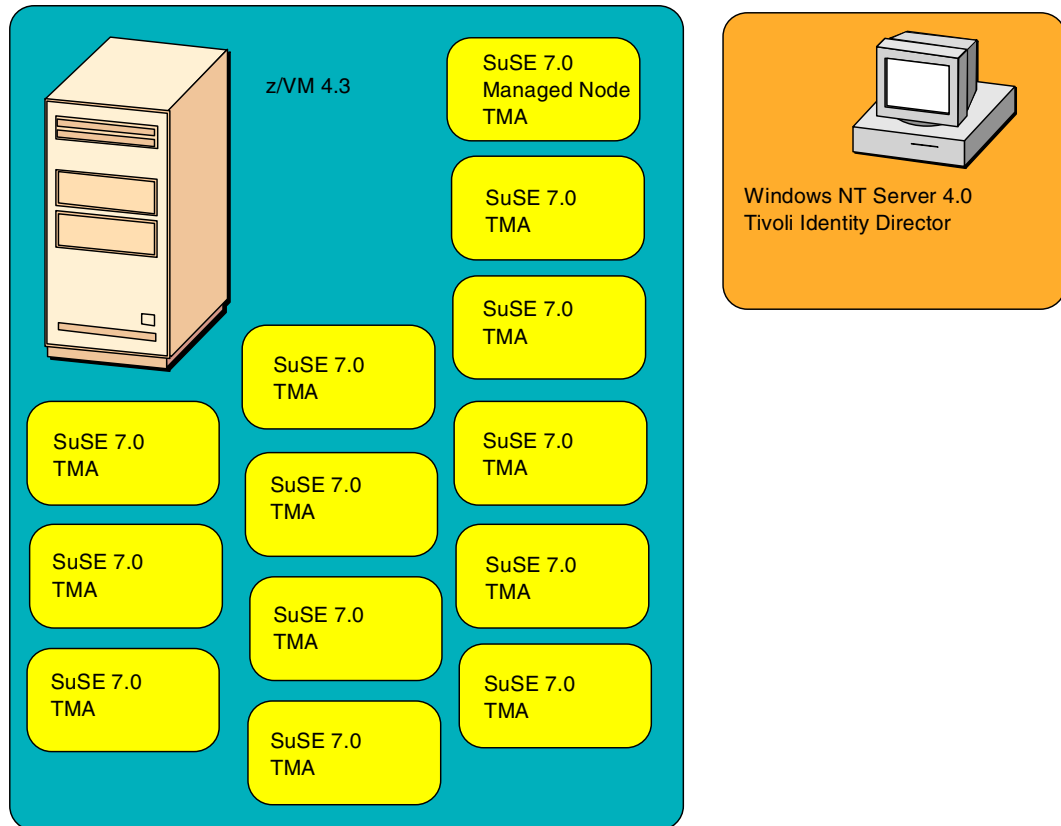


Figure 9-7 IBM Tivoli Identity Manager component layout

## 9.4.2 Installation

We installed all Tivoli Identity Director 1.1 components using the Install Shield Multi-Platform (ISMP) program. ISMP failed to install patches 1 and 3 for Tivoli User Administration 3.8, so those were added manually.

After every product or patch install we ran `wckddb -u` until we saw clean output and backed up the oserv database with `wbkupdb`. These commands are not shown in the examples for this section.

*Example 9-35 Apply Tivoli User Administration patches to server*

```
wpatch -c D:/ -i 38ADM01
wpatch -c D:/ -i 38ADM03
```

Patches for supporting tier 2 operating systems came next (see Example 9-36).

---

*Example 9-36 Apply patches for tier 2 OS support*

---

```
wpatch -c D:/ -i TMF36134
wpatch -c D:/ -i 361TMF61
wpatch -c D:/ -i 361TMF62
wpatch -c D:/ -i 37TMF020
wpatch -c D:/ -i 37TMF021
wpatch -c D:/ -i 371TMF32
odadmin reexec all
wpatch -c D:/ -i 371TMF33
```

---

Tivoli Identity Director 1.1 does not install any Tivoli User Administration gateway products by default, so two more products were installed. (Example 9-37)

---

*Example 9-37 Install User Administration Gateway products for distributing profiles*

---

```
winstall -c D:/3.8-ADM -i ADMIN_GW
winstall -c D:/3.8-ADM -i LINUX_GW
```

---

Tivoli User Administration needed patch 5 to support Linux/390 endpoints, so more patches were added. (Example 9-38)

---

*Example 9-38 Install Tivoli User Administration patches for Linux/390 support*

---

```
wpatch -c D:/ -i 38AGW01
wpatch -c D:/ -i 38LNX01
wpatch -c D:/ -i 38ADM04
wpatch -c D:/ -i 38AGW04
wpatch -c D:/ -i 38ADM05
wpatch -c D:/ -i 38LNX05
```

---

To install a managed node on SuSE 7.0 for s/390, we had to uncomment the exec line in /etc/inetd.conf, and then restart the inetd daemon. (Example 9-39)

---

*Example 9-39 Enable rexec for managed node installation*

---

```
vi /etc/inetd.conf
/etc/rc.d/inetd restart
```

---

The **wclient** command was run from the Windows NT TMR server to install the managed node. We also added the root login of the Linux/390 VM to our Tivoli administrator. (Example 9-40)

---

*Example 9-40 Create managed node on Linux/390*

---

```
wclient -c D:/ -P -d -p tot29-region DB=/usr/local/Tivoli/db @CreatePaths@=1 lnx10
wsetadmin -l root@lnx10.itso.ibm.com Root_tot29-region
odadmin reexec all
```

---

We next created an endpoint gateway on the Linux/390 VM and installed a TMA. These command were run from the Linux/390 VM. (Example 9-41)

---

*Example 9-41 Create endpoint gateway on Linux/390*

---

```
wcrtgate -h lnx10 -p 9123
winstlcf -g lnx10+9123
```

---

We then installed more patches and products from the NT TMR server. Some patches are for Tivoli User Administration support, others are prerequisites. (Example 9-42)



*Example 9-42 Install Tivoli User Administration patches on managed node and TMR server*

---

```
winstall -c D:/<TMF 3.7b CD> -i ADE tot29
winstall -c D:/<TMF 3.7b for Linux CD> -i ADE lnx10
wpatch -c D:/ -i 37TMF023
wpatch -c D:/ -i 37TMF024
winstall -c D:/ -i ADMIN_GW lnx10
wpatch -c D:/ -i TMF36134 lnx10
wpatch -c D:/ -i 361TMF61 lnx10
wpatch -c D:/ -i 361TMF62 lnx10
winstall -c D:/ -i LINUX_GW lnx10
wpatch -c D:/ -i 38AGW01 lnx10
wpatch -c D:/ -i 38LNX01 lnx10
wpatch -c D:/ -i 38AGW04 lnx10
wpatch -c D:/ -i 37TMF020 lnx10
wpatch -c D:/ -i 37TMF021 lnx10
wpatch -c D:/ -i 38LNX05 lnx10
```

---

At this point the repeater infrastructure was in place, so to manage other Linux/390 VMs a TMA was all that was needed. We installed a TMA on lnx9 with this command on lnx10.

*Example 9-43 Create TMA*

---

```
winstlcf -g lnx10+9123 lnx9
```

---

## 9.5 Tivoli Management Framework

Today's network computing enterprise requires an open, scalable, cross-platform approach that is also truly integrated. The Tivoli Management Framework is the foundation for a suite of management applications that are making systems and network management easy.

The Tivoli Management Framework makes it possible to:

- ▶ Shield administrators from platform-specific details of day-to-day operations. Common operations, such as deploying applications and routine network maintenance, can be performed with a single action; administrators are no longer required to repeat the same operation for each platform on your enterprise.
- ▶ Deploy applications to literally thousands of machines with one operation, all the while ensuring the applications remain available.
- ▶ Integrate with third-party applications, because the Tivoli Management Framework is an open solution. The Business Partner Program provides many of these third-party solutions to your enterprise management challenges; challenges such as job scheduling, intrusion detection, and backup and restore, all of which snap-in to the Tivoli Management Framework.

By providing a truly open and comprehensive foundation for network computing management, the Tivoli Management Framework simplifies today's most complex network computing challenges.

The following section describes how we installed our Tivoli Management Framework and software so that we could manage our Linux/390 virtual machines (and even our other IT systems if we wanted to). This section is by no means an authoritative guide on the best way of deploying Tivoli, nor is there sufficient detail for you could deploy it within your infrastructure. Furthermore, we did not include the latest software patches.

Our aim is to demonstrate that the Tivoli Management Framework could be used to manage our Linux/390 infrastructure (and our Penguin farm if it grew that big). We strongly recommend that you review other Tivoli Framework Redbooks and consult with Tivoli services to ensure that the right Tivoli Management Framework configuration and architecture is used for your environment.

The following instructions should be sufficient for you to install the Tivoli Management Framework and become more familiar with it.

In our network of Linux virtual machines, we used LNX8 as our Tivoli Management Framework Server (TMR) and LNX9 as our Tivoli Management Framework Managed Node (MN). All our other 17 Linux systems had a Tivoli Endpoint (LCF) agent installed so that they could be managed.

The Tier 2 Tivoli release notes state that only SuSE 7.0 is supported, so we installed SuSE 7.0 (kernel 2.2) on LNX8 and LNX9.

Here's how we did it.

## 9.5.1 Installation of the Tivoli Management Framework Server

Since the Linux/390 version of the Tivoli Management Framework does not include any of the Xwindow GUI consoles, the installation can only be performed using the command line. To prepare for a command line mode installation, you need to export the DOGUI variable as follows:

```
export DOGUI=false
```

You also need to be the root user before you can install the Tivoli Management Framework because Tivoli needs to be able to open port 94 once installed.

We mounted our CDROM via NFS under /mnt/tmp on LNX8, and then ran the installation as shown in Example 9-44.

### *Example 9-44* Installation of the Tivoli Management Framework

---

```
lnx8:/tmp # export DOGUI=false
lnx8:/tmp # mount -t nfs 9.12.6.143:/export /mnt/tmp
lnx8:/tmp # /mnt/tmp/tmf37b-linux/WPREINST.SH
to install, type ./wserver -c /mnt/tmp/tmf37b-linux

lnx8:/tmp # ./wserver -c /mnt/tmp/tmf37b-linux LK=<INSERT LICENSE KEY> BIN=/opt/tivoli/bin
LIB=/opt/tivoli/lib ALIDB=/opt/tivoli/db MAN=/opt/tivoli/man CAT=/opt/tivoli/cat RN=itso
@CreatePaths@=1
```

using the interpreter type linux-s390 as set in your environment.  
Using command line style installation...

Unless you cancel, the following operations will be executed:

```
need to copy the CAT (generic) to:
  lnx8:/opt/tivoli/cat
need to copy the CSBIN (generic) to:
  lnx8:/opt/tivoli/bin/generic
need to copy the GBIN (generic) to:
  lnx8:/opt/tivoli/bin/generic_unix
need to copy the BUN (generic) to:
  lnx8:/opt/tivoli/bin/client_bundle
need to copy the SBIN (generic) to:
  lnx8:/opt/tivoli/bin/generic
```

```

need to copy the LCFNEW (generic) to:
  lnx8:/opt/tivoli/bin/lcf_bundle.40
need to copy the LCF (generic) to:
  lnx8:/opt/tivoli/bin/lcf_bundle
need to copy the LCFTOOLS (generic) to:
  lnx8:/opt/tivoli/bin/lcf_bundle.40/bin
need to copy the LIB (linux-s390) to:
  lnx8:/opt/tivoli/lib/linux-s390
need to copy the BIN (linux-s390) to:
  lnx8:/opt/tivoli/bin/linux-s390
need to copy the ALIDB (linux-s390) to:
  lnx8:/opt/tivoli/db/lnx8.db
need to copy the MAN (linux-s390) to:
  lnx8:/opt/tivoli/man/linux-s390
need to copy the CONTRIB (linux-s390) to:
  lnx8:/opt/tivoli/bin/linux-s390/contrib
Continue(y/n)? y
Executing queued operation(s)
Distributing machine independent Message Catalogs --> lnx8
... Completed.
Distributing machine independent generic Codeset Tables --> lnx8
..... Completed.
Distributing architecture specific Libraries --> lnx8
..... Completed.
Distributing architecture specific Binaries --> lnx8
..... Completed.
Distributing architecture specific Server Database --> lnx8
..... Completed.
Distributing architecture specific Man Pages --> lnx8
... Completed.
Distributing machine independent Generic Binaries --> lnx8
... Completed.
Distributing machine independent Client Installation Bundle --> lnx8
.... Completed.
Distributing machine independent generic HTML/Java files --> lnx8
... Completed.
Distributing architecture specific Public Domain Contrib --> lnx8
... Completed.
Distributing machine independent LCF Images (new version) --> lnx8
..... Completed.
Distributing machine independent LCF Images (old version) --> lnx8
..... Completed.
Distributing machine independent LCF Tools --> lnx8
..... Completed.
Registering installation information...Finished.

```

---

Example 9-45 shows a simple test that you can perform so that you know the Tivoli installation went successfully. You should get some similar output.

*Example 9-45 Test to make sure the TMR installed correctly*

```

lnx8:/tmp # odadmin odlist
Region      Disp  Flags  Port      IPAddr    Hostname(s)
1091227347  1     ct-    94        10.1.5.8  lnx8.itso.ibm.com,lnx8

```

---

Once the installation is complete, you will be able to use the Tivoli commands and Tivoli desktop on a Windows platform.

## Some installation hints

- ▶ Make regular backups!

The Tivoli Framework provides a backup tool that can be used to back up the Tivoli internal database. It is suggested (and strongly recommended) that you back up after you perform anything major to Tivoli. For example, perform a backup after you:

- Install any Tivoli products (perform a backup after *each* product installation)
- Install any Tivoli patches
- Deploy any Managed Nodes
- Deploy the Tivoli Endpoint to many systems

You should also perform a backup regularly, either nightly if possible, or at least weekly!

You can use the Tivoli Desktop or the command line tool **wbkupdb** to start a Tivoli database backup.

- ▶ Read the latest release notes! You'll find them at:

[http://www.tivoli.com/support/public/Prodman/public\\_manuals/td/ManagementFramework3.7B.html](http://www.tivoli.com/support/public/Prodman/public_manuals/td/ManagementFramework3.7B.html)

- ▶ Install the patches recommended by the release notes. You can download them from the Tivoli website; you will need your own login ID and password to get these patches.

<https://www.tivoli.com/secure/support/patches>

- 3.7-TMF-0023
- 3.7-TMF-0024 (Requires ADE to be installed first)
- 3.7-TMF-0025 (Requires the MDIST GUI to be installed first)

See "Installation of the Tivoli patches" on page 184 for examples of how to do this.

- ▶ A Linux TMR server cannot be a RIM Host (used for Inventory, Software Distribution, and the Tivoli Event Console). You'll need a Tier 1 platform (that is, AIX, Sun, NT, and so forth) configured as a Managed Node for that function.
- ▶ You can't install and use Tivoli Software Installation Service (SIS) when you have a Linux TMR server. To be able to use SIS, you need a Tier 1 platform as the TMR server.
- ▶ There is no Tivoli desktop available on Linux/390. You are able to use a Tivoli Desktop on another support platform to connect to the Linux/390 TMR server.
- ▶ Tivoli TMR Server and Managed Nodes require your UNIX environment to be set up if you are using Tivoli via the command line. You can set up your environment by sourcing the following script (you can put this in your `.profile`):

```
. /etc/Tivoli/setup_env.sh
```

## 9.5.2 Installation of the Tivoli Management Framework Managed Node

To remotely install the MN component of the Tivoli Management Framework, you need to install and enable **rexec**. Tivoli uses the `rexec` function of UNIX to remotely execute some commands during installation.

While some system administrators may frown at using `rexec` (especially using `rexec` for the root user), it will only be required for the installation. Once the installation is complete, you can remove or disable it.

**Restriction:** Removing rexec will prevent you from begin able to perform an **odadmin start n** (or **odamin reexec n**) to restart a remote Tivoli Managed Node. There are two workarounds for this:

- ▶ Log onto the remote system as root and execute **odadmin start**.
- ▶ Enable and restrict rexec so that it will only accept connections from the TMR server.

To quickly test that your rexec service is running and ready for the Tivoli install, try the following (you must be logged in as root):

```
lnx8:/tmp # rsh lnx9 date
Wed May 1 17:14:07 EDT 2002
```

Executing **rsh <HOST> date** should return the system date of the remote machine. If you get **Permission denied**, then rexec is being denied by the remote machine and you will have to resolve this before you continue.

**Tip:** The easiest way to enable access for rsh is to put the IP address of the TMR server in the file **.rhosts** in root's home directory. You can remove it once the install is complete.

SIS is not available for our use on the Linux TMR server, so the only installation options available are:

- ▶ Via the Tivoli Desktop (running on a Windows machine)
- ▶ Using the command line.

Example 9-46 shows how we did the installation using the command line.

*Example 9-46 Installation of the Tivoli Manage Node via command line.*

```
lnx8:/tmp # rsh lnx9 date
Wed May 1 17:31:53 EDT 2002
lnx8:/tmp # wclient -d -p itso -c /mnt/tmp/tmf37b-linux lnx9
Inspecting node lnx9...
```

Unless you cancel, the following operations will be executed:

For the machines in the independent class:

hosts: lnx9

need to copy the CAT (generic) to:

lnx9:/opt/tivoli/cat

need to copy the CSBIN (generic) to:

lnx9:/opt/tivoli/bin/generic

lnx9 already has the X11 Resource Files installed (from lnx9.itso.ibm.com).

need to copy the GBIN (generic) to:

lnx9:/opt/tivoli/bin/generic\_unix

need to copy the SBIN (generic) to:

lnx9:/opt/tivoli/bin/generic

need to copy the LCFNEW (generic) to:

lnx9:/opt/tivoli/bin/lcf\_bundle.40

need to copy the LCF (generic) to:

lnx9:/opt/tivoli/bin/lcf\_bundle

need to copy the LCFTOOLS (generic) to:

lnx9:/opt/tivoli/bin/lcf\_bundle.40/bin

For the machines in the linux-s390 class:

hosts: lnx9

need to copy the LIB (linux-s390) to:

lnx9:/opt/tivoli/lib/linux-s390



We also need to install a Tivoli Managed Node on a Windows platform to support our RIM functions used later by Tivoli Software Distribution. If you don't plan on using the MDIST database features of Tivoli Software Distribution, you can skip the next part.

To install a Managed Node onto a Windows platform, you need to first install Tivoli Remote Execute Program (also known as TRIP). TRIP provides a REXEC-type function for Windows platforms.

If you have a Tivoli Endpoint already deployed to a Windows machine, you could use Tivoli Software Distribution to deploy and install TRIP. If not, you'll need a method of getting TRIP installed. Refer to the Tivoli Framework Users Guide for other methods of installing TRIP.

We mounted the Tivoli Framework CD (Tier 1 CD) into TOT11 and manually installed TRIP by running **setup.exe** and choosing the defaults. You can check that TRIP is installed successfully by running the following test:

*Example 9-48*

---

```
Inx8: # rexec -l Administrator tot11 cmd.exe /c 'dir C:\'  
Password:  
Volume in drive C is Winn-2000  
Volume Serial Number is 1475-1901  
  
Directory of C:\  
  
05/20/2002  07:51a      <DIR>          WINNT  
...  
                8 File(s)          240,528 bytes  
                27 Dir(s)    25,244,274,688 bytes free
```

---

If you get Login incorrect, you'll need to resolve this before you continue.

Once TRIP was installed, we used the same installation process that we described previously for LNX9. (Don't forget to use the -U Administrator and provide the Administrator password.)

**Note:** To install the Tivoli Framework onto a Tier 1 platform (such as Windows), you'll need the Tier 1 CD that has the appropriate binaries for the Tier 1 platform.

Mount the Tier 1 CD somehow (via NFS if it is available) so that the Linux TMR Server can read it.

### 9.5.3 Installation of other Tivoli Framework products

Also on the Tivoli Framework CDRom are the Tivoli ADE and Tivoli AEF products. These products can be installed via the command line by using the **winstall** command.

Alternatively, these products can also be installed using the Tivoli desktop on a Windows system. For this section, we installed them with the Tivoli desktop on a Windows system.

**Note:** The Tivoli desktop is available on a Tier 1 Framework CD. The installation is as simple as running **setup.exe** and installing it in a location of your choice on your Windows system.

Follow these steps to install the products:

1. Start your Tivoli desktop and log in to your TMR server.

2. On the Tivoli desktop, select **Desktop -> Install -> Install Patch**. Use the Select Media button to point to the correct directory where you have your CDROM mounted. You should have a desktop similar to Figure 9-8.



Figure 9-8 Installing additional Tivoli Framework products

3. For each product available (ADE or AEF), select the TMR Server and Managed Nodes you want to install the product to, and then select **Install**.

During the installation, you will get a new dialog pop up that indicates the progress and results of the installation.

**Tip:** Don't forget to perform your Tivoli backup before *and* after the installation of any Tivoli products.

## 9.5.4 Installation of the Tivoli patches

The installation of Tivoli patches is similar to the installation of Tivoli products. There are at least three Tivoli patches that need to be installed for Tivoli on Linux/390 (see "Some installation hints" on page 180).

Again, you can perform the patch installation either using the Tivoli desktop on a Windows platform, or using the command line tool **wpatch**.

Following are the steps to install the three required Tivoli patches using the Tivoli desktop on Windows.

1. Download the Tivoli patches from the Tivoli support site. Store them on the TMR server in a temporary location (that is, /tmp).
2. Untar the patches using:
 

```
tar xf <PATCH FILE>
```
3. Start the Tivoli Desktop and log in to your TMR server.



4. Select **Desktop -> Install -> Install Patch**. Use the Select Media button to point to the correct directory where you have untared the patch.
5. Select the nodes that you need to apply the patch to (these details are in the README file that comes with each patch tar archive).
6. Select **Install** to install the patch, then watch the progress to ensure that it works OK.

The patch installation should look similar to Figure 9-9.

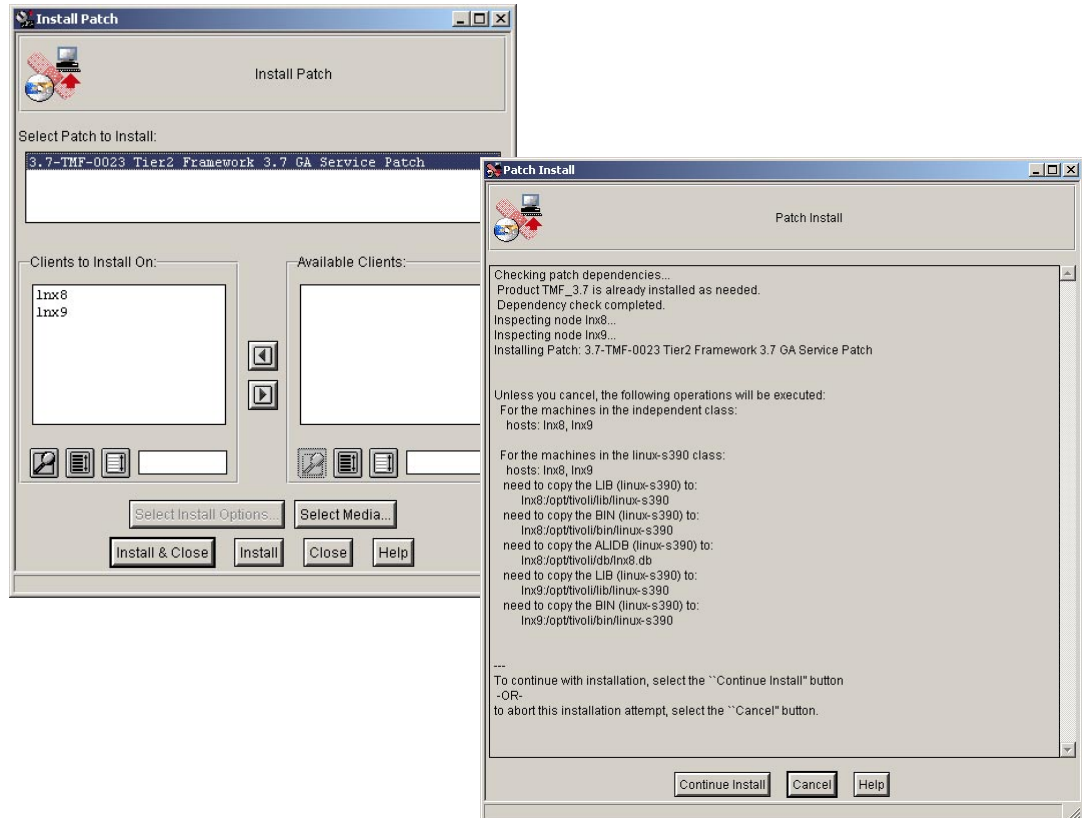


Figure 9-9 Installing Tivoli patches with the Tivoli Desktop

**Tip:** Don't forget to perform your Tivoli backup before *and* after the installation of any Tivoli patches.

### 9.5.5 Installation of a Tivoli Endpoint gateway

Before we can install our Tivoli Endpoints, we need a Tivoli gateway for them to contact and be managed by. The Tivoli gateway function can be provided by any Managed Node (or even the TMR server, although it is not recommended).

Since LNX8 will be our gateway, we ran these commands to enable the gateway service:

```
lnx8:/tmp # wcrtgate -h lnx9 -p 9494 -n gw.itso.lnx9
1091227347.2.19#TMF_Gateway::Gateway#
```

And to test that it worked, we issued the following command:

```
lnx8:/tmp # wlookup -ar Gateway
gw.itso.lnx9 1091227347.2.19#TMF_Gateway::Gateway#
```

LNx9 is now ready to manage endpoints!

## 9.5.6 Installation of the Tivoli Endpoints

There are many ways to deploy your Tivoli Endpoints throughout your organization, and if you are already using Tivoli, you probably have this process organized.

For simplicity (and because we could), we enabled `rexec` on all our Linux/390 systems (as we required for our Managed Node install) and deployed the Tivoli Endpoint using the supplied Tivoli `winstlcf` command. In Example 9-49 we show how we deployed the Tivoli Endpoint to LNx9.

*Example 9-49 Installation of Tivoli Endpoint using `winstlcf`*

---

```
lnx8:/tmp # rsh lnx9 date
Wed May 1 17:50:29 EDT 2002
lnx8:/tmp # winstlcf -d /opt/tivoli/lcf -e -g 10.1.5.9+9494 lnx9

Trying lnx9...
locating files in /opt/tivoli/bin/lcf_bundle.40...
locating files in /opt/tivoli/bin/lcf_bundle...

Ready to copy files to host lnx9:
  destination: lnx9:/opt/tivoli/lcf
  source: lnx8:/opt/tivoli/bin/lcf_bundle.40
  files:
    generic/lcfd.sh
    generic/epinst.sh
    generic/as.sh
    generic/lcf_env.sh
    generic/lcf_env.csh
    generic/lcf_env.cmd
    bin/linux-s390/mrt/lcfd
    lib/linux-s390/libmrt272.so
    lib/linux-s390/libcpl272.so
    lib/linux-s390/libdes272.so
    lib/linux-s390/libmd2ep272.so
  source: lnx8:/opt/tivoli/bin/lcf_bundle
  files:

  Continue? [yYna?] y
Tivoli Light Client Framework starting on lnx9
May 01 17:50:33 1 lcfd Command line argv[0]='/opt/tivoli/lcf/bin/linux-s390/mrt/lcfd'
May 01 17:50:33 1 lcfd Command line argv[1]='-Dlcs.login_interfaces=10.1.5.9+9494'
May 01 17:50:33 1 lcfd Command line argv[2]='-Dlib_dir=/opt/tivoli/lcf/lib/linux-s390'
May 01 17:50:33 1 lcfd Command line argv[3]='-Dload_dir=/opt/tivoli/lcf/bin/linux-s390/mrt'
May 01 17:50:33 1 lcfd Command line argv[4]='-C/opt/tivoli/lcf/dat/1'
May 01 17:50:33 1 lcfd Command line argv[5]='-Dlcs.machine_name=lnx9'
May 01 17:50:33 1 lcfd Command line argv[6]='-Dlcs.login_interfaces=10.1.5.9+9494'
May 01 17:50:33 1 lcfd Starting Unix daemon
Performing auto start configuration
Tivoli LCF daemon master autostart file is /sbin/init.d/Tivoli_lcfd1.
Done.
```

---

And, finally, to test that the Endpoint successfully logged into our gateway, we issued the following command:

```
lnx8:/tmp # wep ls
G      1091227347.2.19 gw.itso.lnx9
      1091227347.3.517+#TMF_Endpoint::Endpoint# lnx9
```

The `winstlcf` command can be used to deploy more than one Endpoint at a time; just include each hostname on the command line that you want to install the Endpoint to.

### 9.5.7 The view from the Tivoli Desktop

After deploying the Tivoli Endpoint to most of our systems, Figure 9-10 shows how it looked from the Tivoli Desktop.

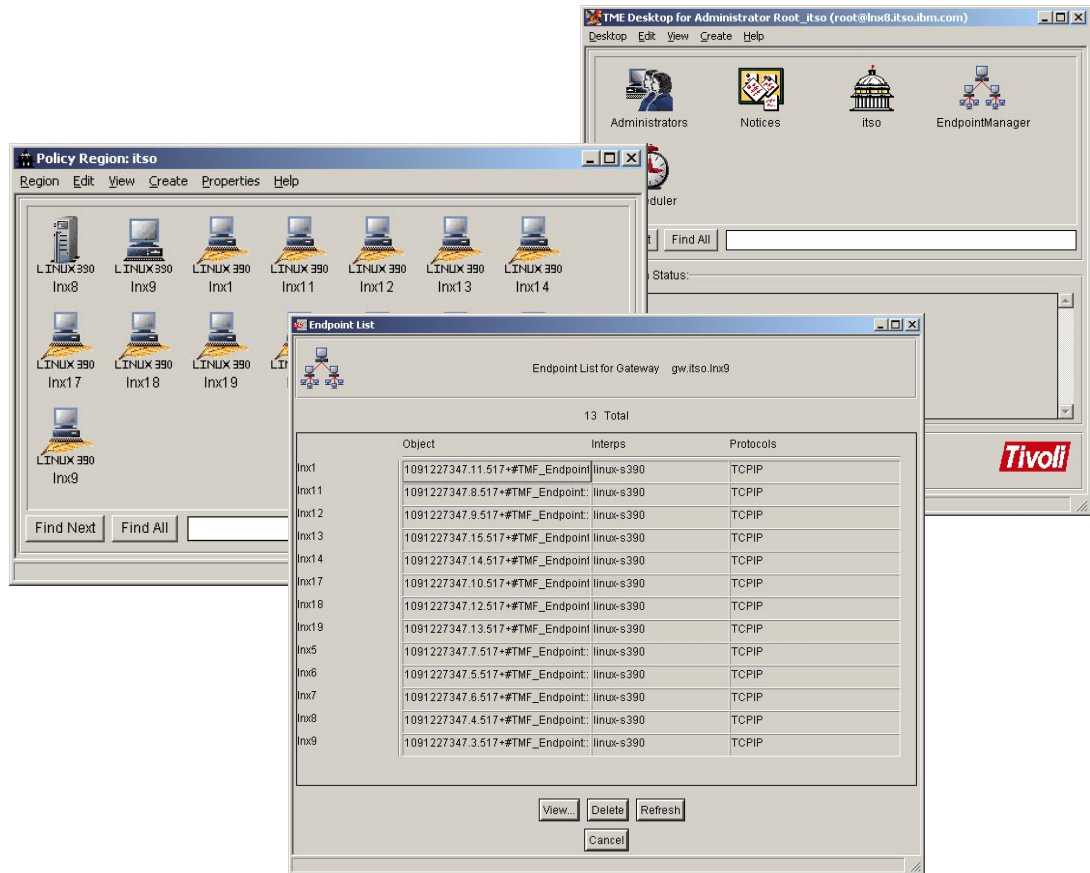


Figure 9-10 The view from the Tivoli Desktop

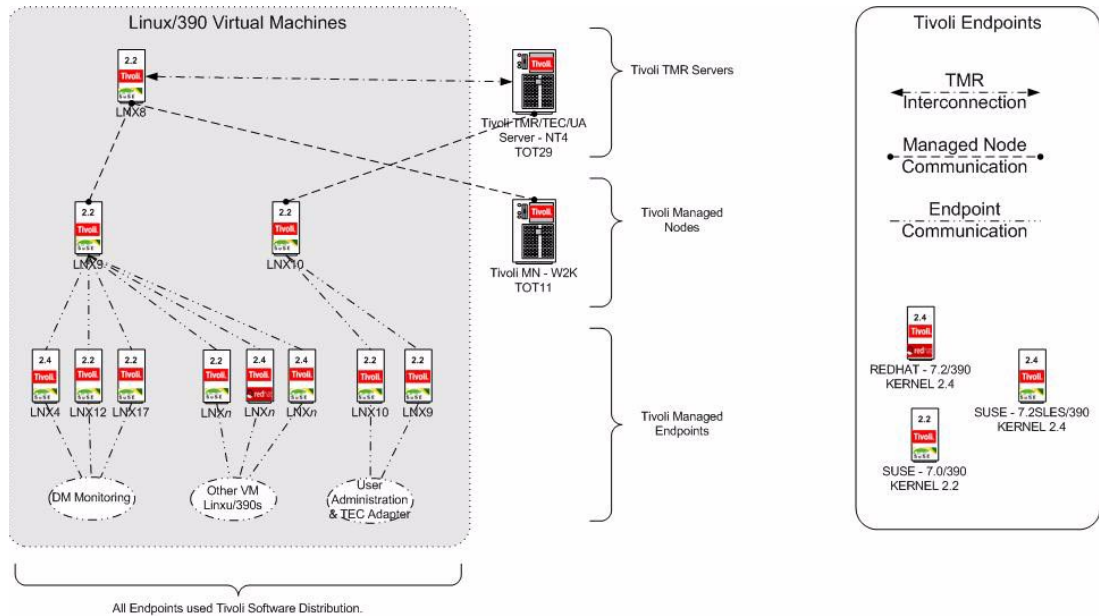


Figure 9-11 Tivoli Managed Environment

## 9.6 Tivoli Software Distribution

Tivoli Software Distribution (SD) is part of the Configuration and Operations suite of tools provided by IBM Tivoli. Software Distribution can do the following:

- Deploy applications at a pre-determined time and have those application installed. The installation can occur during the deployment or could be scheduled for a later date, enabling a synchronized application deployment.

The systems that you deploy the applications to need not be online when submitting the deployment (or the later installation). If there are systems that are offline, or go offline during the application deployment, they will receive it (or the remainder of it), when they next come back online.

- Deploy any arbitrary file or files, and if required execute a set of instructions before or after the deployment of the files.
- With Tivoli Inventory installed, SD can deploy files and/or data based on specific conditions. For example, you could specify that it only deploy to systems with a specific amount of memory, that have a prerequisite file or software application installed, or are of a particular make, module, and so forth.

**Note:** Tivoli Inventory is not supported on Linux/390 systems, so we will not be able to use SD this way without some customization.

- It is also possible to have SD deploy only to systems that are grouped by a custom method that fits within your organization; for example, systems belonging to a department, systems that have a particular role, or systems that have a particular name.

Software or data to be deployed with SD can be placed on a source host, and then staged at appropriate points within your network (for example at the other end of slow links), so that the deployment can be optimized and be network bandwidth friendly.

Tivoli Software Distribution can take care of software:

- ▶ Deployment – getting data/applications out to your systems
- ▶ Removal – removing data/applications from your systems
- ▶ Verification – verifying that data/applications are installed correctly and are intact
- ▶ Repairs – fixing up data/applications when the verify determines that they are not installed/deployed correctly

## 9.6.1 Preparing to install Tivoli Software Distribution

For our environment, we deployed Tivoli Software Distribution with the following configuration:

- ▶ All our Linux/390 Endpoints are managed by LNX9 (configured as a Tivoli Managed Node/Gateway). Tivoli Software Distribution Gateway is installed on LNX9 so that we can deploy to those Endpoints.
- ▶ All our Windows Endpoints are managed by TOT11 (a Windows 2000 system), also configured as a Tivoli Managed Node/Gateway.

**Note:** We installed a Windows Managed Node/Gateway for a couple of reasons:

- ▶ To use the Tivoli MDIST graphical user interface. The Tivoli MDIST GUI provides a graphic interface to determine what distributions are occurring, which routes they are taking, and their status. The Tivoli MDIST GUI is not required for SD to work, but it does provide additional control if it is installed. Since the GUI is not supported on Linux/390, we needed a Tier 1 Managed Node to host this function.
- ▶ RIM (the interface between Tivoli and a database) is not available or supported for Linux/390, so we need a Tier 1 platform to host our database and RIM connection. The RIM host is used by the Tivoli MDIST graphical user interface, and also is optional.

- ▶ We installed the Tivoli MDIST graphical user interface on TOT11.
- ▶ We installed DB2 for Windows v6.1 on TOT11.

DB2 will hold our distribution status data as displayed by the Tivoli MDIST GUI (it is accessed via RIM).

Here are the steps we took:

1. First, we installed a Managed Node on TOT11 in our Windows 2000 system using the same procedure described previously (see “Installation of the Tivoli Management Framework Managed Node” on page 180).
2. We then installed DB2 on TOT11 using the DB2 installation defaults when prompted.
3. We created a database for MDIST to use.

It is easier to create a database for use by MDIST2 using DB2’s Control Center. After DB2 was installed successfully, we started the DB2 Control Center. (If we hadn’t changed the user name used for the install, it would have created user *db2admin* with the password *db2admin*.)

Once logged in, we created a database by drilling down in the Explorer view side of the Control Center, starting from **Systems -> PC NAME -> Instances -> DB2 -> Databases -> Right Click -> Create -> Database using Smart Guide**. Figure 9-12 shows the path used to create the database.

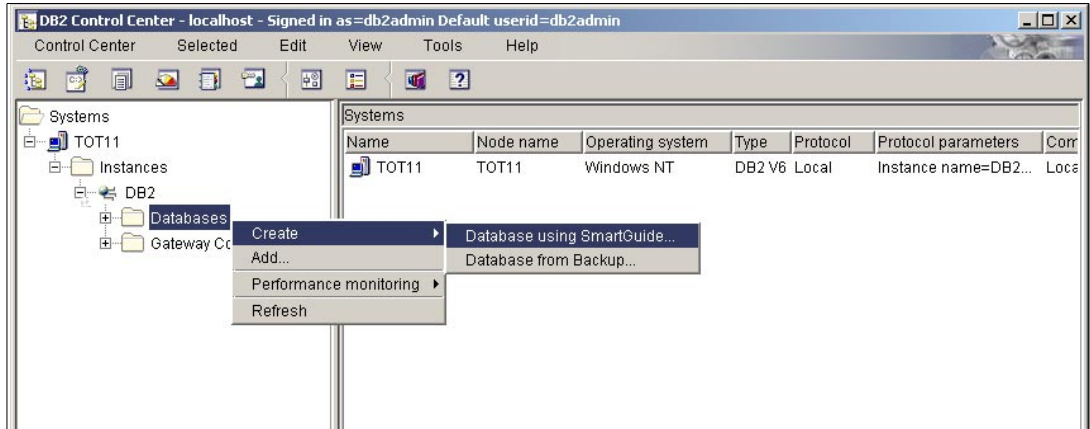


Figure 9-12 Creation of MDIST2 database on TOT11.

We gave the database a name (MDIST2) and then clicked **Done**. (We reviewed the other details first, to make sure they were appropriate for our environment.)

Figure 9-13 below shows our created database.

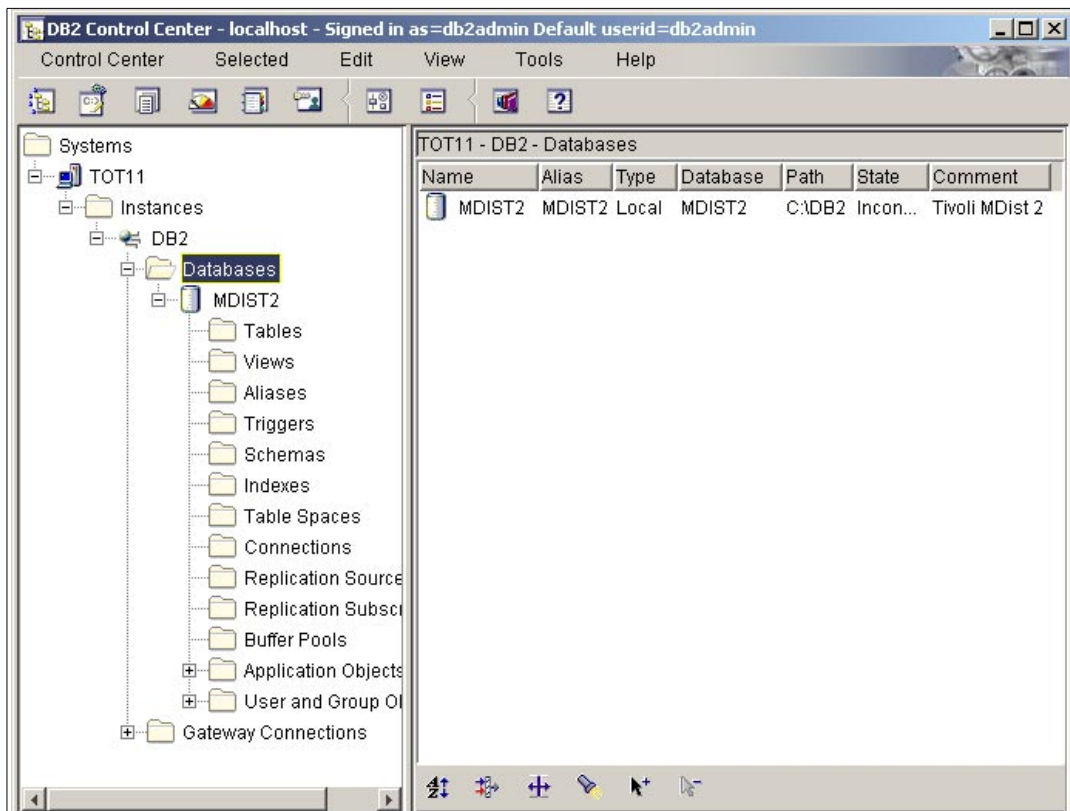


Figure 9-13 MDIST2 database created

## 9.6.2 Installation of Tivoli Software Distribution

We installed Tivoli SD on the TMR server and our two managed nodes as per the configuration described previously.

We followed the same process to install the Tivoli Software Distribution product and patches as were used for the Tivoli Management Framework, described in “Installation of other Tivoli Framework products” on page 183 and “Installation of the Tivoli patches” on page 184.

Figure 9-14 shows part of our installation of Tivoli Software Distribution.

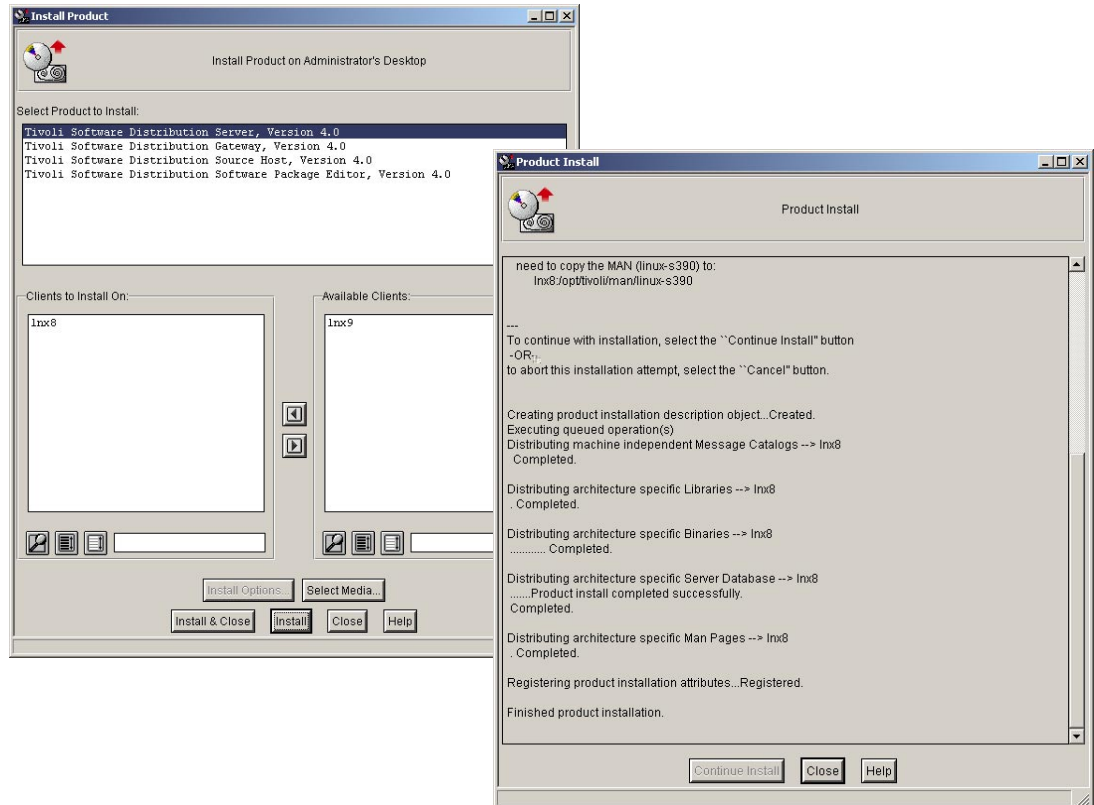


Figure 9-14 Installation of Tivoli Software Distribution

### Some installation notes

- ▶ Make regular backups!
- ▶ Read the latest release notes! You'll find them at:  
[http://www-internal.tivoli.com/support/public/Prodman/public\\_manuals/td/SoftwareDistribution4.0.html](http://www-internal.tivoli.com/support/public/Prodman/public_manuals/td/SoftwareDistribution4.0.html)
- ▶ Install the patches recommended in the release notes. You can download them from the Tivoli Web site; you will need your own login ID and password to get these patches.
  - 4.0-COU-0028
  - 4.0-COU-0029
  - 4.0-COU-0030
  - 4.0-COU-0031

Example 9-50 lists the products installed and their locations after our installation was complete.

*Example 9-50 Installed products for Tivoli Software Distribution from `ws1inst -ah`.*

---

```
*-----*
                                Product List
*-----*
Tivoli Java Client Framework 3.7
  lnx8      linux-s390
  tot11     w32-ix86

Java for Tivoli 3.7
  lnx8      linux-s390
  tot11     w32-ix86

Tivoli Java RDBMS Interface Module (JRIM) 3.7
  lnx8      linux-s390
  tot11     w32-ix86

JavaHelp for Tivoli 3.7
  lnx8      linux-s390
  tot11     w32-ix86

Swing for Tivoli 3.7
  lnx8      linux-s390
  tot11     w32-ix86

Tivoli MDist 2 Graphical User Interface
  lnx8      linux-s390
  tot11     w32-ix86

Tivoli Software Distribution Gateway, Version 4.0
  lnx9      linux-s390
  tot11     w32-ix86

Tivoli Software Distribution Source Host, Version 4.0
  lnx8      linux-s390

Tivoli Software Distribution Server, Version 4.0
  lnx8      linux-s390

*-----*
                                Patch List
*-----*
Tivoli Software Distribution Server, Tier2 GA Patch 4.0-COU-0028
  lnx8      linux-s390

Tivoli Software Distribution Tier2 GA gateway patch, 4.0-COU-0029
  lnx9      linux-s390

Tivoli Software Distribution Source Host, Tier2 GA Patch 4.0-COU-0030
  lnx8      linux-s390
```

---

### 9.6.3 Prepare the MDIST2 database

The installation of SD v4.0 gateway on TOT11 will have put the SQL scripts that are used to create the tables in the database.

Access a DB2 Command Window by selecting **Start -> Programs -> DB2 for Windows NT -> Command Windows**. Create the DB2 schema for MDIST2 as shown in Example 9-51.



*Example 9-51 Create the DB2 Schema for MDIST2*

---

```
C:\SQLLIB\BIN>c:\winnt\system32\drivers\etc\Tivoli\setup_env
C:\SQLLIB\BIN>cd %BINDIR%\TME\MDIST2\SQL
C:\Tivoli\bin\w32-ix86\TME\MDIST2\sql>db2 connect to MDIST2 user db2admin using db2admin
Database Connection Information
```

```
Database server      = DB2/NT 6.1.0
SQL authorization ID = DB2ADMIN
Local database alias = MDIST2
```

```
C:\Tivoli\bin\w32-ix86\TME\MDIST2\sql>db2 -tf mdist_db2_admin.sql
DB21034E The command was processed as an SQL statement because it was not a
valid Command Line Processor command. During SQL processing it returned:
SQL0204N "MDIST_DATA_TS" is an undefined name.  SQLSTATE=42704
```

```
DB20000I The SQL command completed successfully.
```

```
DB21034E The command was processed as an SQL statement because it was not a
valid Command Line Processor command. During SQL processing it returned:
SQL0204N "MDIST_TEMP_TS" is an undefined name.  SQLSTATE=42704
```

```
DB20000I The SQL command completed successfully.
```

```
C:\Tivoli\bin\w32-ix86\TME\MDIST2\sql>db2 -tf mdist_db2_schema.sql
DB21034E The command was processed as an SQL statement because it was not a
valid Command Line Processor command. During SQL processing it returned:
SQL0204N "DB2ADMIN.DIST_STATE" is an undefined name.  SQLSTATE=42704
```

```
DB20000I The SQL command completed successfully.
```

```
DB20000I The SQL command completed successfully.
```

```
DB21034E The command was processed as an SQL statement because it was not a
valid Command Line Processor command. During SQL processing it returned:
SQL0204N "DB2ADMIN.DIST_NODE_STATE" is an undefined name.  SQLSTATE=42704
```

```
DB20000I The SQL command completed successfully.
```

```
DB20000I The SQL command completed successfully.
```

```
DB20000I The SQL command completed successfully.
```

---

**Tip:** Don't worry about the four error messages you receive when running the db2 command. These errors are because the scripts are trying to drop tables before creating them. Since this is a new database, the tables don't exist yet!

And, finally, create a RIM object so that MDIST can talk to the database. (Example 9-52)

*Example 9-52 Create MDIST2 RIM object*

---

```
C:\Tivoli\bin\w32-ix86\TME\MDIST2\sql>wcrtrim -v db2 -h tot11 -d MDIST2 -u db2admin -H
C:/sqllib -I db2 mdist2
RDBMS password: <ENTER PASSWORD>
```

---

To ensure that it works, you can use `wrintest -l mdist` to make sure you get a connection. This will at least tell you that Tivoli can find the database and log into it.

### Example 9-53 Test RIM connection

```
C:\Tivoli\bin\w32-ix86\TME\MDIST2\sql>wgetrim mdist2
RIM Host:          tot11
RDBMS User:       db2admin
RDBMS Vendor:     DB2
Database ID:      MDIST2
Database Home:    C:/SQLLIB
Server ID:
Instance Home:    DB2

C:\Tivoli\bin\w32-ix86\TME\MDIST2\sql>wrimtest -l mdist2
Resource Type   : RIM
Resource Label  : mdist2
Host Name       : tot11
User Name       : db2admin
Vendor          : DB2
Database        : MDIST2
Database Home   : C:/SQLLIB
Server ID       :
Instance Home   : DB2
Opening Regular Session...Session Opened
RIM : Enter Option >x
Releasing session
```

**Note:** Tivoli MDIST graphical user interface (from TMF CD, not available for Linux/390).

We are now ready to start deploying software to our Endpoints.

## 9.7 Tivoli Distributed Monitoring (Advanced Edition)

Tivoli Distributed Monitoring (Advanced Edition) is part of the Tivoli performance and availability suite of tools that applies pre-configured, automated best practices to the automated monitoring of essential system resources. Tivoli Distributed Monitoring (DM) detects bottlenecks and other potential problems and provides for automatic recovery from critical situations, which eliminates the need for system administrators to manually scan through extensive performance data.

DM events can be sent to:

- ▶ Tivoli Enterprise console, where the events can be correlated with other external factors to help with root cause analysis.
- ▶ Tivoli Business Systems Manager, where the impact of an event can be viewed from a business service perspective.
- ▶ The DM Health Console, provided as part of Tivoli Distributed Monitoring (Advanced Edition). The DM Health Console displays the “health” of the system by assigning each potential problem as a numeric value between 100 (perfect health) and zero (the conditions for the corresponding event have been met). Intermediate values indicate the presence of a condition, that has not yet fully materialized.

The previous version of this product was called Tivoli Distributed Monitoring for Windows; in the v4.1 release it has been extended to include UNIX and Linux systems.

Tivoli Distributed Monitoring (Advanced Edition) provides the following features:

- ▶ An off-the-shelf solution for monitoring Windows, UNIX, and Linux systems.

- ▶ Ready-to-use resource models that report on specific aspects of a system's status. For example, the Process resource model provides information on the status of processes, CPU usage, and so on. Resource monitoring is an implementation of the Common Information Model (CIM). CIM is an approach to system and network management that applies object-oriented techniques to model the system.
- ▶ Data collection and problem analysis performed locally on the system.
- ▶ Resource models that can easily be added (point and click) to a Tivoli profile, which can be distributed to multiple systems simultaneously.
- ▶ The ability to view both real-time and historical data for any system from a centralized monitoring application called the Health Console, which is supplied with the product. Only the results of the data collection and problem analysis are retrieved by the Health Console.
- ▶ Availability of options to send the results from the data collection and analysis to the Tivoli Enterprise Console or to the Tivoli Business Systems Manager.
- ▶ The ability to specify automatic corrective or preventative actions to resolve situations that could develop into real problems.
- ▶ The ability to modify resource models; for example, by changing threshold levels to match a users' own requirements.
- ▶ A scheduling feature that allows monitoring to take place at user-specified times.
- ▶ A heartbeat function, running at gateways, that regularly checks the availability and status of attached endpoints and makes the information available to the Tivoli Enterprise Console server, Tivoli Business Systems Manager or the Tivoli Distributed Monitoring (Advanced Edition) Notice Group.

## 9.7.1 Understanding Tivoli Distributed Monitoring (Advanced Edition)

It is important to understand the concepts behind DM. Following are brief explanations of the terms used by DM.

Resource	A resource is anything that can affect the operation of a computer system, including physical and logical disks, memory, CPU, printers, processes, events in log files (syslog) and TCP/IP. Tivoli Distributed Monitoring (Advanced Edition) monitors resources.
Resource Model	Resources models specify which resource data is accessed from a system at runtime, and how the data is processed. For example, the Process resource model obtains data related to processes running on a system. Performance data is automatically collected by the resource model and processed by an appropriate algorithm to determine whether or not the system is performing to your expectations.  Full details about the resource model supplied are given in the <i>Tivoli Distributed Monitoring (Advanced Edition): Resource Model Reference</i> , SH19-4564.
Cycles	When a resource model is run, it gathers data in regular intervals, known as cycles; the duration of a cycle is the cycle time. The data collected is a snapshot of the status of the resources specified in the resource model at that time.
Threshold	Each resource model defines one or more thresholds. A threshold is a named property of a resource with a value that represents a level of a performance-related entity, which, if exceed or not reached, a system administrator might want to know.

	<p>However, some thresholds can be used by the algorithm to limit the scope of a resource model. The UNIX resource models don't appear to use thresholds this way.</p>
Parameter	<p>Some resource models have one or more parameters. Each parameter can take the form of a list of strings, numeric values, a boolean list of predetermined values from which you can make any combination of selections, or a choice list of mutually exclusive alternatives.</p> <p>For example, the Process resource model can take a list of process names that should be monitored and alert when they are stopped, killed, or no longer existing.</p>
Indication	<p>Each resource model will generate an indication if certain conditions implied by the resource model's thresholds are not satisfied in a given cycle. Each resource model has its own algorithm to determine which combinations of thresholds should generate an indication.</p> <p>Full details about the specific definitions of the indications that have been created for each resource model are document in <i>Tivoli Distributed Monitoring (Advanced Edition): Resource Model Reference</i>, SH19-4564.</p>
Occurrence	<p>An occurrence is the term used to refer to a cycle during which an indication occurs for a given resource.</p>
Hole	<p>A hole is the term used to refer to a cycle during which an indication does <i>not</i> occur for a given resource model. In other words, none of the conditions specified for the generation of any indication have been met. This does not mean that none of the thresholds have been exceeded.</p>
Event	<p>An event is used to verify the persistence of a given indication, meaning the indication has occurred for the last number of occurrences (taking into account any holes).</p> <p>For example, an indication that the CPU is running unusually high in one cycle may not be considered serious. However, if it happened frequently in a brief time frame, and we ignore some (few, if any) samples in the same time frame where the CPU was running normally, it may be an indication of a problem.</p>
Clearing event	<p>An event that is generated by the resource model, if enabled, that closes any previous error event, indicating that the previous condition is no longer present.</p>

## 9.7.2 Preparing for Tivoli Distributed Monitoring (Advanced Edition)

After you perform the installation of the Tivoli Management Framework as detailed in “Tivoli Management Framework” on page 177, you are ready to install Tivoli Distributed Monitoring (Advanced Edition).

There are three Tivoli installable products on the installation media for Tivoli Distributed Monitoring (Advanced Edition):

- ▶ Tivoli Distributed Monitoring (Advanced Edition), Version 4.1

This product is installed on your TMR Servers first, then on all Managed Nodes that have endpoints that you want to monitor. (In our environment, we installed this product on LNX8 first; then on LNX9, for our Linux/390 Endpoints; and on TOT11, for our Windows Endpoints).

- ▶ Tivoli Distributed Monitoring (Advanced Edition) TDS Configuration, Version 4.1.  
This component enables you to integrate to the Tivoli Decision Support Server Prediction Guide with a Tivoli Decision Support server. Since we are not using a Tivoli Decision Support Server, we did not install this component.
- ▶ Tivoli Distributed Monitoring (Advanced Edition) TBSM Adapter, Version 4.1.  
This component enables a Tivoli Business Systems Manager (TBSM) server to received Tivoli Distributed Monitoring (Advanced Edition) events. Since we do not have a TBSM Server, we did not install this component.

There is also the Tivoli Distributed Monitoring (Advanced Edition) Health Console, which can be installed directly on a Windows PC by running **setup.exe**. We installed this component on TOT11.

### 9.7.3 Installation of Tivoli Distributed Monitoring (Advanced Edition)

Using the same process described in “Installation of other Tivoli Framework products” on page 183, we installed the Tivoli Distributed Monitoring (Advanced Edition), Version 4.1 from the Tivoli Distributed Monitoring (Advanced Edition) installation CD.

Figure 9-15 shows part of our installation.

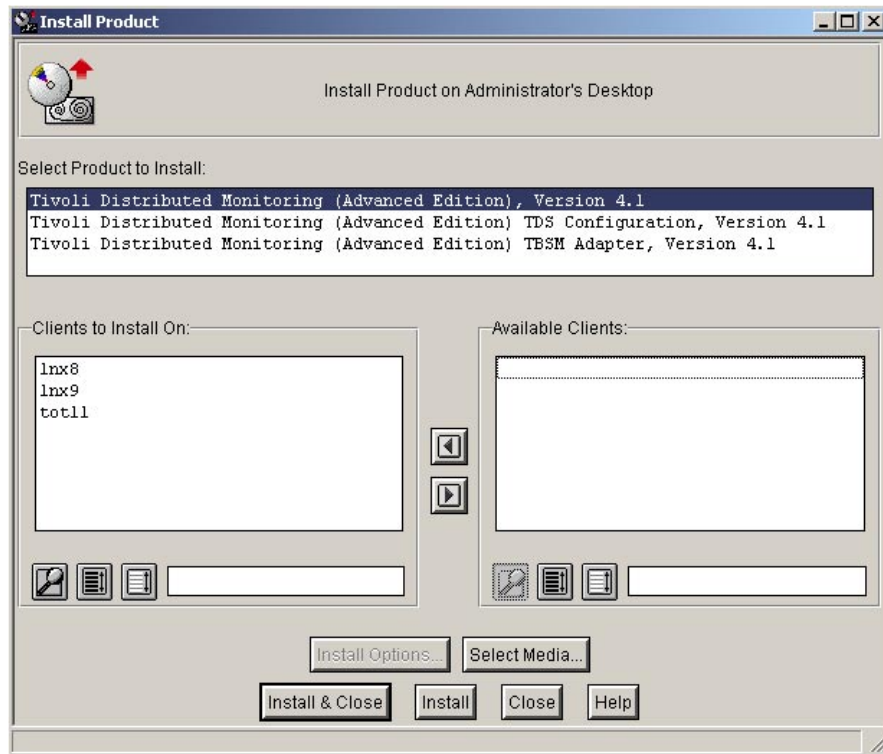


Figure 9-15 Installation of Tivoli Distributed Monitoring (Advanced Edition).

The products installed and their locations after the installation finished are as follows:

lnx8	linux-s390
lnx9	linux-s390
tot11	w32-ix86

## 9.7.4 Preparing the Linux/390 Endpoints

Linux/390 Endpoints need to have a Java Runtime Environment (JRE) installed before they can receive Tivoli Distributed Monitoring (Advanced Edition) Profiles. If you have already installed a JRE, you can skip this section

**Note:** See the redbook *IBM Tivoli Monitoring Version 5.1: Advanced Resource Monitoring SG24-5519* for more information on Installing JRE Section 4.2.4

One way to deploy the JRE for DM is to use Tivoli Software Distribution. Details on how to deploy software with Tivoli Software Distribution is covered in “Software deployment with Tivoli Software Distribution” on page 218. It would be beneficial to read that chapter before continuing with the following instructions. The next few steps will assume that you have set up and understand Tivoli Software Distribution.

We deployed the JRE by creating a SoftwarePackage profile in our sd.itso.misc Profile Manager. We called this `sp.itso.misc.jre^1.3.0`, as shown in Figure 9-17 on page 200.

Our `sp.itso.misc.jre^1.30` SoftwarePackage had the following components:

- ▶ Disk Space Check

Since the JRE is approximately 40 Mb in size, and it requires approximately 100 Mb after installation, we added a disk space check to make sure we have 140 Mb of available disk space on the appropriate file systems (we only have */*).

This will stop the installation from sending down the JRE if there isn't sufficient space.

- ▶ Add Folder

We stored the JRE `IBMJava2-JRE-13.[1].s390.tgz` (found on the Tivoli Distributed Monitoring (Advanced Edition) CD in the `/tools/jre/linux-s390` directory) in `/opt/tivoli/SoftwarePackage/sp.itso.misc` directory on our TMR server.

When we sent the JRE archive to each Endpoint, we dropped it into `/tmp`, since it will only be there for the installation. (We renamed it `IBMJava2-JRE-13.1.s390.tgz` so that the square brackets don't cause any problems with our installation program.)

- Add File

We added the `IBMJava2-JRE-13.1.s390.tgz` file with the File Properties dialog.

Figure 9-16 shows the Directory and File properties we used.

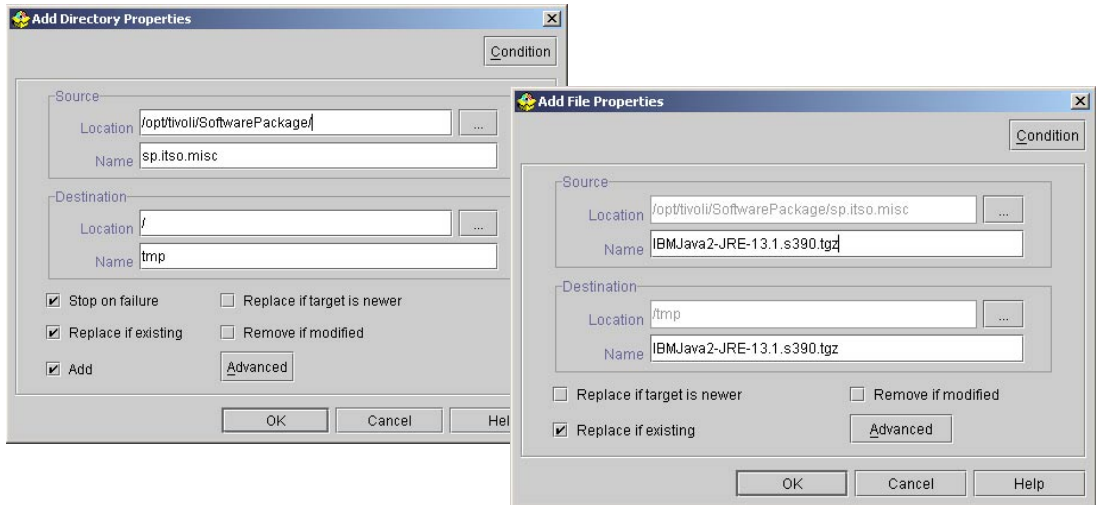


Figure 9-16 Directory and File Properties

**Tip:** Don't forget to click the **Temporary** attribute located on the Advanced screens.

► Execute Program

We added options to our execution commands so that the JRE is installed after distribution. Table 9-2 shows the options we used.

Table 9-2 Execute Program options

Stage	Command	Options
Install	/bin/tar	Arguments: xzf /tmp/IBMJava2-JRE-13.1.s390.tgz Working Directory: /opt
Remove	/bin/rm	Arguments: -rf /opt/IBMJava2-s390-13
Verify	/bin/ls	-al /opt/IBMJava2-s390-13

Figure 9-17 shows our final SoftwarePackage.

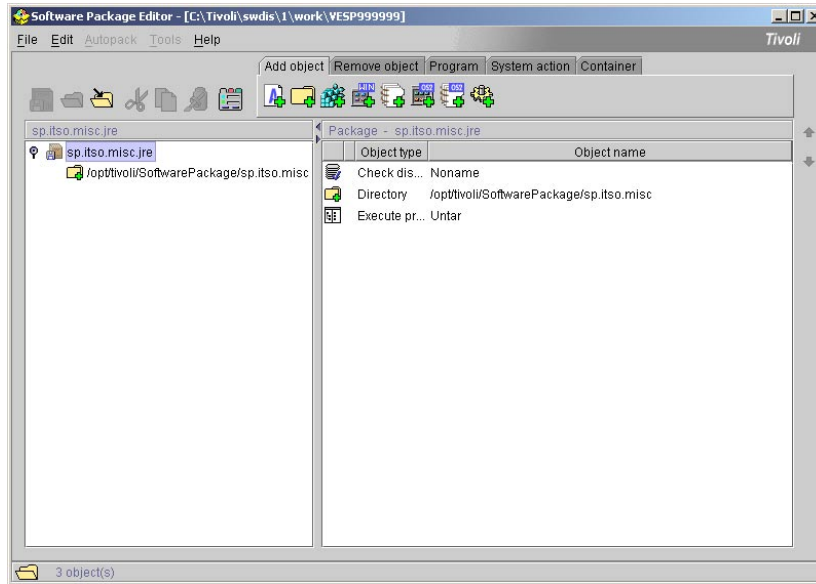


Figure 9-17 SoftwarePackage for the JRE

You can now distribute the JRE to your Tivoli Endpoints that will be monitored using Tivoli Distributed Monitoring (Advanced Edition). Details on how to distribute software is explained in 10.1.2, “Software deployment with Tivoli Software Distribution” on page 218.

### 9.7.5 Installation of the Health Console

The health console can be installed on many different platforms, we decided to install it on Windows because that was what we have available, for more information on what platforms it can be installed on refer to IBM Tivoli Monitoring Version 5.1: Advanced Resource Monitoring redbook SG24-5519-02 Section 4.5. We installed the Windows client directly from the Tivoli Distributed Monitoring (Advanced Edition) installation cd, by running the **setup.exe** from the **DMWHC** directory.

The installation will automatically install a Java Runtime Environment if you do not have one already installed on your Windows system. Figure 9-18 shows the installation process.





Figure 9-18 Tivoli Distributed Monitoring (Advanced Edition) Health Console installation

You are now ready to start using Tivoli Distributed Monitoring (Advanced Edition).

## 9.8 IBM Tivoli Enterprise Console

IBM Tivoli Enterprise Console forms the hub of Tivoli performance and availability management solutions. Designed specifically for enterprise computing environments, Tivoli Enterprise Console consolidates and processes the thousands of events that occur daily from network devices, hardware systems, relational databases, and applications.

Sophisticated event grouping and filtering can reduce the number of events displayed to your operators, helping them zero in on the most critical events and helping enable them to manage even the largest, most complex environments.

Tivoli Enterprise Console can also analyze and correlate these events to efficiently guide your support staff to the root cause of each problem, responding automatically whenever possible. and helping operators track problems down to the network layers allowing them to easily perform complex network diagnostics and take corrective actions.

The IBM Tivoli Enterprise Console delivers the following capabilities:

- ▶ Comprehensive event integration
- ▶ Enterprise-wide event correlation (beyond simple filtering)
- ▶ Automated event notification and response with robust security features
- ▶ Integrated network management and diagnostics features

## 9.8.1 Component layout

The IBM Tivoli Enterprise Console (T/EC) Server product was available for Linux/390 at the time this book was written, while the RDBMS Interface Module (RIM) objects were not. In a recommended configuration, the Enterprise Console Server, its database, and the RIM object that connects them are all on the same system. Installing a RIM object on a system outside of the z/VM environment would require network traffic between the Enterprise Console Server and its database and cause poor performance.

To keep the Enterprise Console Server and its database on the same system, we chose to use an Intel-based Windows NT 4.0 Server system for TMR and T/EC servers.

We installed a managed node on one of the Linux/390 images, and a TMA on each Linux/390 image.

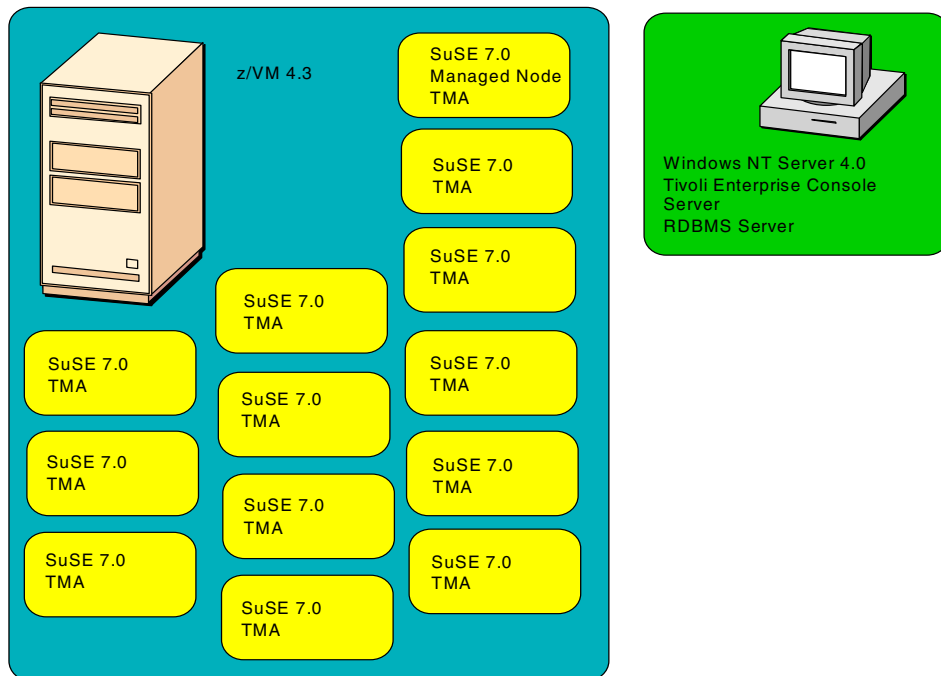
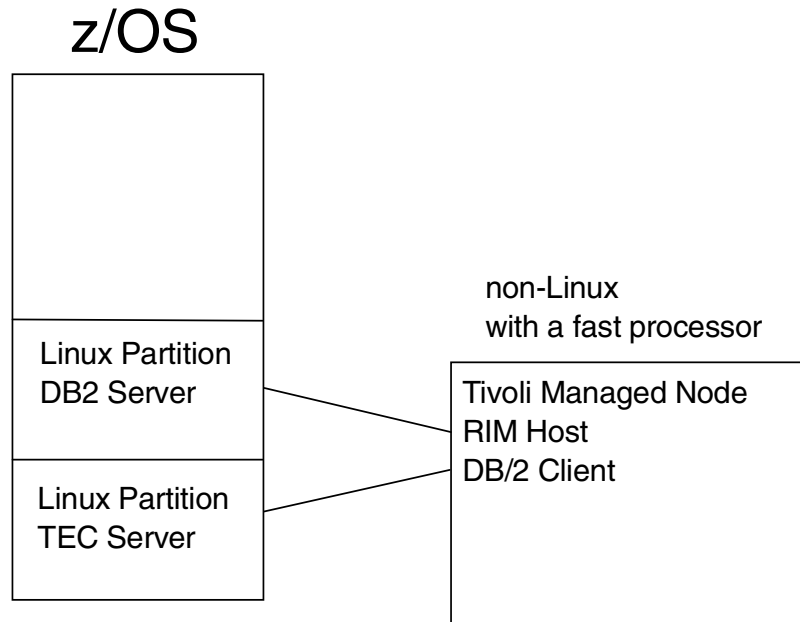


Figure 9-19 IBM Tivoli Enterprise Console component layout

Figure 9-20 shows a scenario where we have DB2 running on Linux and the Tivoli RIM running on TEC(linux z/OS) RIM/DB2 client(non Linux) DB2 server (Linux z/OS). This would have been an interesting setup, but due to time constraints and availability we went with the setup in Figure 9-19.



It is best if the Managed Node could be in the same subnet as the z/OS machine to minimize the network latency.

Figure 9-20 TEC Server on z/OS

## 9.8.2 Installation

### T/EC Server

The installation of the TMR server and T/EC server followed the user guides and release notes for each product. To integrate Linux/390 managed nodes and TMAs into the TMR, several patches were added.

*Example 9-54 Install T/EC patches for Linux/390 support*

---

```
wpatch -c D:/ -i TMF36134
wpatch -c D:/ -i 361TMF61
wpatch -c D:/ -i 361TMF62
wpatch -c D:/ -i 37TMF020
wpatch -c D:/ -i 37TMF021
wpatch -c D:/ -i 371TMF32
odadmin reexec all
wpatch -c D:/ -i 371TMF33
```

---

We then installed the managed node and set up an endpoint gateway. (Example 9-55)

*Example 9-55 Create managed node and gateway on Linux/390 system*

---

```
wclient -c D:/ -P -d -p tot29-region DB=/usr/local/Tivoli/db @CreatePaths@=1 lnx10
wsetadmin -l root@lnx10 Root_tot29-region
wrtgate -h lnx10 -p 9700
```

---

We installed a TMA on the Linux/390 managed node. (Example 9-56)

*Example 9-56 Create TMA on Linux/390 system*

```
winstlcf -g lnx10+9700
```

To enable Linux/390 endpoint distribution of Adapter Configuration Profiles (ACPs), a patch was added to both the T/EC Server and the Linux/390 managed node. (Example 9-57)

*Example 9-57 Apply T/EC patch for ACP distribution to Linux/390*

```
wpatch -c D:/ -i 371TEC04
```

## TME Adapter

To give an example of installing a TME Adapter, we installed the `tecad_logfile_linux-s390` adapter. The installation process follows. We only included the details for steps two and three because we assume you are familiar with the other steps.

1. Create a profile manager.
2. Create the Adapter Configuration Profile (ACP).
  - a. In your profile manager select **Profile** from the **Create** menu.
  - b. Name the profile `Linux390` and select type **ACP**.
3. Customize the ACP.
  - a. Double-click the `Linux390 ACP` icon.
  - b. Click **Add Entry**.
  - c. Select `tecad_logfile_linux-s390`.

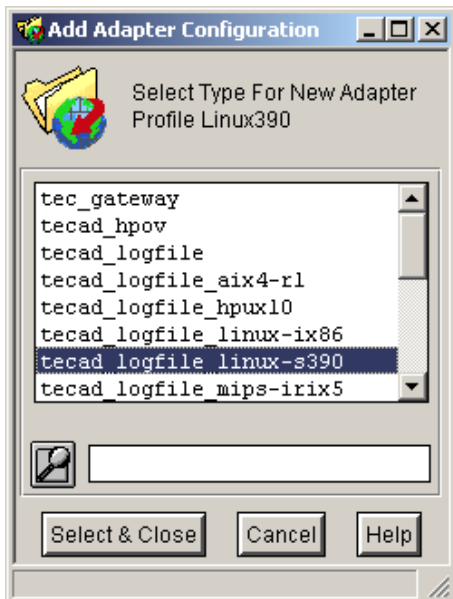


Figure 9-21 Select `tecad_logfile_linux-s390` adapter for ACP

- d. Select the **Filters** category and disable the filters (allow events to pass) that you want. We disabled the **Logfile\_Base** filter, which would let `Logfile_Base` events be generated.

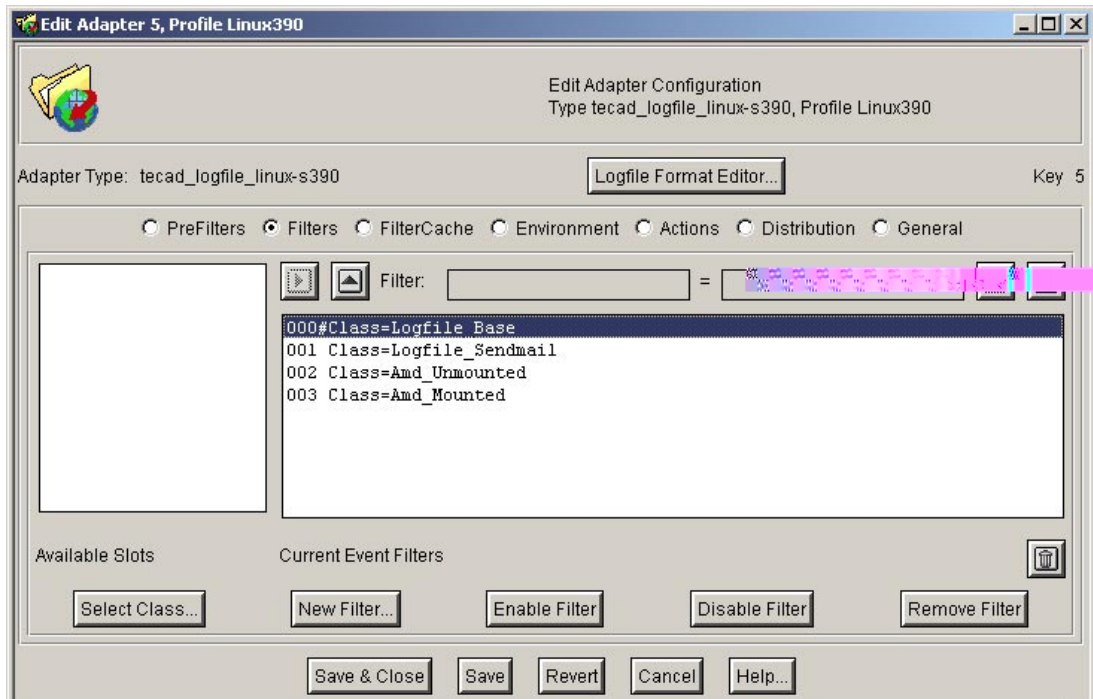


Figure 9-22 Disable Logfile\_Base filter

- e. Click the **Actions** category and add the following code to the “After file distribution” text box:

```
ln -s /etc/rc.d/tecad_logfile /etc/rc.d/rc2.d/K01tecad_logfile
ln -s /etc/rc.d/tecad_logfile /etc/rc.d/rc2.d/S98tecad_logfile
```

**Note:** The SuSE 7.0 (2.2 kernel) s/390 distribution uses runlevel three for xdm login, and the default multi-user/network runlevel is two. We needed to link the logfile adapter’s run control scripts to the rc2.d directory for standard operation.

The default run control directory links are K50 and S99. We observed some unwanted behavior and changed the startup to use S98 so that it would start before the TMA, and observed which were running on the same system. We changed the kill to K01 so that it would shut down completely. At K50 in rc2.d, the Linux system would sometimes hang while shutting down or rebooting.

- f. Click **Save and Close**. Your profile is now ready to distribute to a TMA.
4. Subscribe an endpoint to the profile manager.
5. Distribute the profile.

### Non-TME Adapter

To give an example of installing a Non-TME Adapter, we installed the tecad\_logfile\_linux-s390 adapter. The installation process is:

1. Copy LOGFILE.TAR to endpoint:

```
mkdir -p /opt/Tivoli/tecad
cp LOGFILE.TAR /opt/Tivoli/tecad
```

2. Untar LOGFILE.TAR:

```
cd /opt/Tivoli/tecad
```

```
tar -xvf LOGFILE.TAR
```

3. Set the TECADHOME variable:

```
export TECADHOME=/opt/Tivoli/tecad
```

4. Run the tecad\_logfile.cfg script to configure the logfile adapter.

```
./bin/tecad_logfile.cfg
```

5. Link the run control script to the rc2.d directory (see previous note).

```
ln -s /etc/rc.d/tecad_logfile /etc/rc.d/rc2.d/K01tecad_logfile
```

```
ln -s /etc/rc.d/tecad_logfile /etc/rc.d/rc2.d/S98tecad_logfile
```



# System management using IBM Tivoli Software

In this chapter we present examples of how to perform selected management actions with the Tivoli Software suite of products. The specific tasks included are:

- ▶ Availability management using Tivoli Distributed Monitoring (Advanced Edition)
- ▶ Software deployment using Tivoli Software Distribution
- ▶ Data management using Tivoli Storage Manager
- ▶ Security using Tivoli Identity Directory and Tivoli User Administration

## 10.1 Operations

### 10.1.1 Availability management with Tivoli Distributed Monitoring (Advanced Edition)

Tivoli Distributed Monitoring (DM) is an availability management product that enables you to proactively monitor your Tivoli Endpoints and react quickly to any upcoming issues.

In the previous chapter we described how to install Tivoli Distributed Monitoring; in this section we create one profile with some resource monitors to monitor three of our Linux systems (LNX5, LNX12, and LNX17). We then view the health of these systems using the supplied DM Health Console and the Tivoli command line tools from the TMR server.

It is also possible to have DM events sent directly to a Tivoli Enterprise Console (TEC) if you have one. These events can then be correlated with other events that are occurring within your environment to help determine the root cause of a current situation. Details on how to set up the integration to TEC are thoroughly covered in *Tivoli Distributed Monitoring (Advanced Edition) Users Guide 4.1 SH19-4565-00*, so we haven't covered that integration in this book.

We also didn't cover how to set up automatic actions, so that when events are generated (for example, a file system filling up), appropriate action can automatically be executed to possibly help resolve the situation. Details on how to set up automatic actions are also in *Tivoli Distributed Monitoring (Advanced Edition) Users Guide 4.1 SH19-4565-00*

#### Creating a Distributed Monitoring Profile

Use the following steps to create a distributed monitoring profile.

1. On the Tivoli Desktop, select **Create -> Region** and name your new region `dm.itso`.
2. Assign the Profile Manager and Tmw2kProfile resources to your new region. Figure 10-1 shows an example.



Figure 10-1 Assigning managed resources to our `dm.itso` Policy Region

This will permit the new Policy Region to manage Distributed Monitoring Profiles.

3. Open up the new policy region and create a Profile Manager to hold your DM Profile by selecting **Create -> Profile Manager** and giving it a name. In our example, we called it



dm.itso.linux-s390. Select “Dataless Endpoint Mode” if you want to directly subscribe endpoints to this profile manager. (We did.)

4. Open up the new profile manager, and create a profile by selecting **Create -> Profile** and giving it a name. We called ours `dm.itso.linux-s390.all` to represent all the monitors for our Linux/390 systems. (This is shown in Figure 10-4 on page 211.)
5. Right-click the new profile and select “Open Distributed Monitoring.” Your new profile manager will be empty, so now you can add some monitoring resource modules. Select **Add** and you will be able to choose from the following:
  - **DMXCpu** - This resource model detects problems with the CPU of a system (for example, how long processes wait in the queue to be processed). It sends alerts if the system has a very low percentage of CPU idle time or a high percentage of CPU system time.
  - **DMXFile** - This resource model gives information about files on a system. You can use this model to alert when files are modified (for example, `/etc/passwd`).
  - **DMXFileSystem** - This resource model alerts on the inefficiency of file system usage. It sends alerts for the following conditions: Low space available, Fragmented file system, and Low percentage of available i-nodes.
  - **DMXMemory** - This resource model provides information on how memory is being utilized. It gives alerts when you have Low storage space (percentage of total free space), Low swap space, or the System appears to be thrashing.
  - **DMXNetworkInterface** - Detects problems with all installed network interfaces and identifies events when the performance of interfaces appears critical.
  - **DMXProcess** - looks for bottlenecks in running processes, including: a process using too much CPU time, too many zombies, a named process stopping or being killed, a named process not existing.
  - **DMXSecurity** - provides information about files and users logged on to a system. It highlights a number of logins by the same user, a suspect super user, and an account that is not valid for root.

More information about these resources models can be found in *Tivoli Distributed Monitoring (Advanced Edition) Resource Model Reference*.

We added one of each, so our profile looks like Figure 10-2 on page 210.

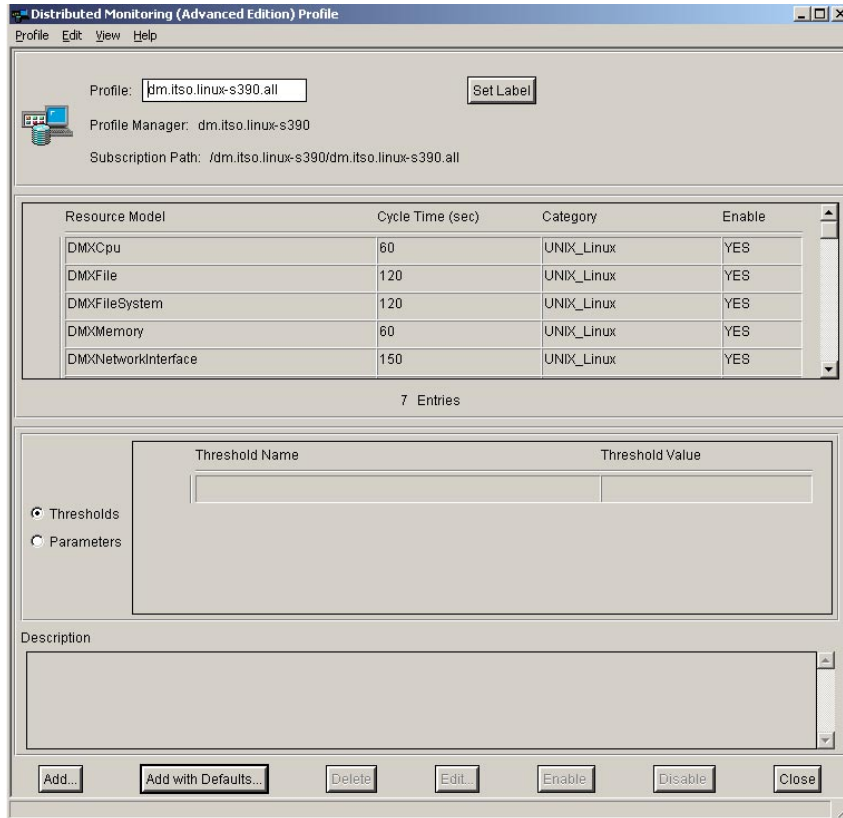


Figure 10-2 Distributed Monitoring Profile for Linux/390 systems

- After making any tuning adjustments to the Resource Models, you can **Close** this profile.
- Subscribe the Endpoints that will receive this profile. In our case, we sent it to LNX5, LNX12, and LNX17. In the profile manager select **Profile Manager -> Subscribers** and select “Endpoints to subscribe to this Profile Manager.” Figure 10-3 shows the Subscribers dialog box, and Figure 10-4 shows our final DM Profile Manager.

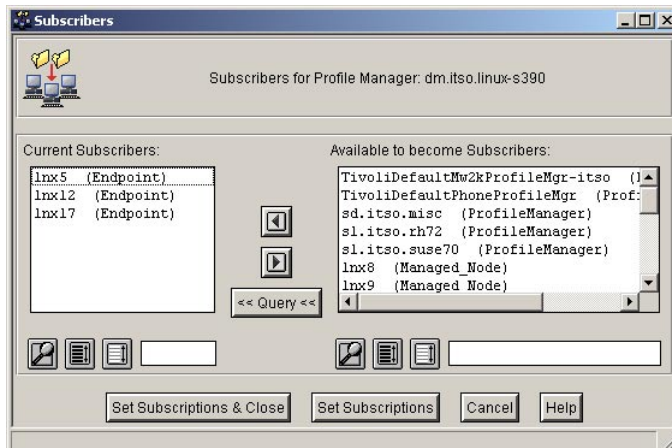


Figure 10-3 Subscribing Endpoints to a Profile Manager



Figure 10-4 Tivoli Distributed Monitoring Profile Manager

## Preparing to distribute the DM Profile Manager

Now you are ready to distribute this profile to the endpoints. Since this is the first time that these endpoints will receive a DM profile, you must first run a Tivoli task against each endpoint, which sets up the Java Runtime Environment that Tivoli will use. (You probably have already installed a JRE on the Endpoints since this was covered in 9.7.4, “Preparing the Linux/390 Endpoints” on page 198.)

The installation of DM provides a Tivoli task that will prepare the endpoint so that it is ready to run Distributed Monitoring. (If you don’t perform this task, you will get an error if you distribute the profile).

The required task is located in a TopLevel Policy Region, which can be found by going to the main Tivoli desktop and selecting **Desktop -> TMR Connections -> TopLevel Policy Regions**.

A new window will open with Top Level Policy Regions. Select the **TivoliDefaultMw2kRegion-itso -> Tivoli Distributed Monitoring (Advanced Edition) Tasks**. Figure 10-5 shows the Task Library for DM.

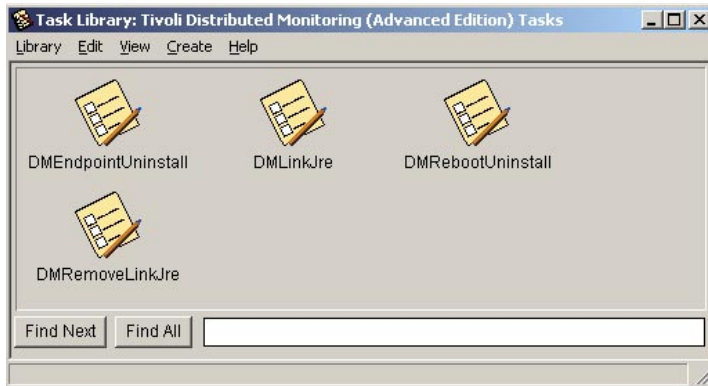


Figure 10-5 Tivoli Tasks for Tivoli Distributed Monitoring (Advanced Edition)

Right-click the **DMLinkJre** task and choose **Execute**. When you execute the task DMLinkJRE, make sure you select “Display on Desktop” in the Output Destination area of the Execute Task window. This way you’ll see the results of the task when it completes.

Select the Endpoints that will run this task in the Execution Targets section, then click **Execute & Dismiss**. Figure 10-6 shows the Execution dialog with our Endpoints selected for execution.

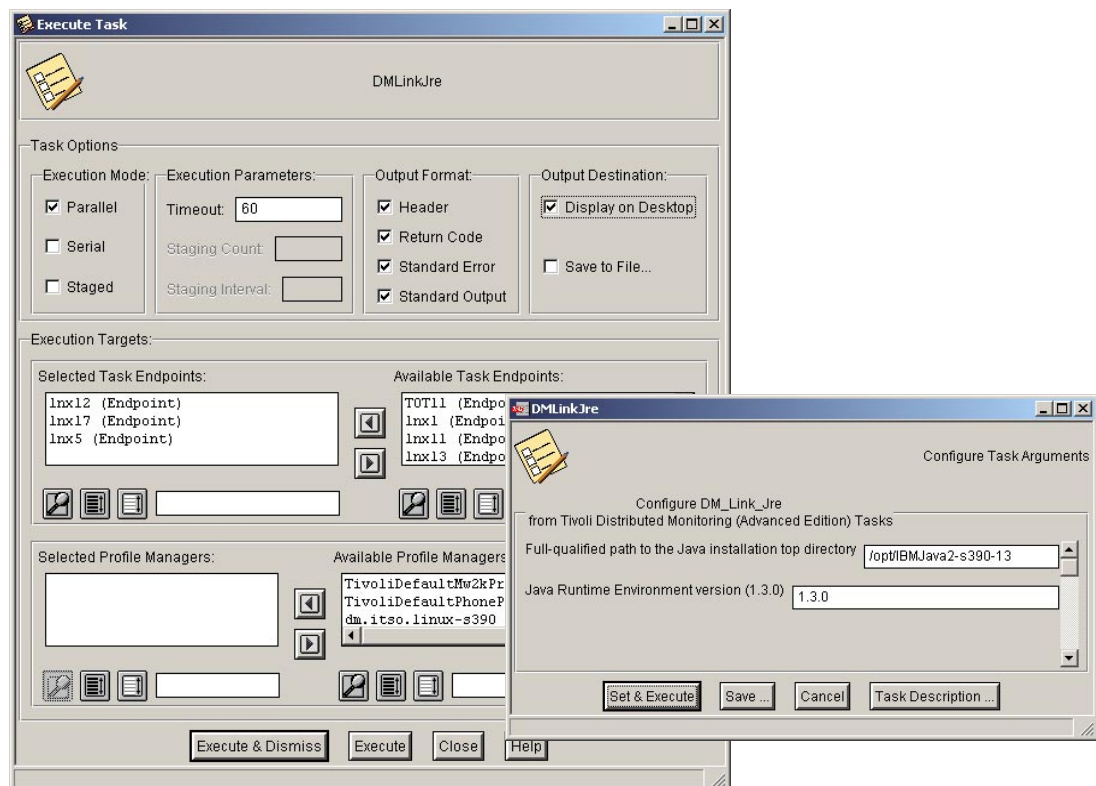


Figure 10-6 Executing the DMLinkJRE task

A new window will pop up, requesting some parameters to be passed to the task. Enter the location of the JRE and the version. (If you obtained the JRE from the Tivoli CD and installed it as described in 9.7.4, “Preparing the Linux/390 Endpoints” on page 198, it is version 1.3.0). If all systems are the same, you can complete the path (excluding the /bin) and the version,

and click **Set & Execute**. (Since we distributed our JRE with Tivoli Software, the path of the JRE was `/opt/IBMJava2-s390-13`).

The status of the task execute will show up in a new window if you chose the desktop as the output destination (see Figure 10-6). If the task execution is successful you should see the line `Link created successfully` in the execution output window. Ours is shown in Figure 10-7.



Figure 10-7 Successful execution of task DMLinkJRE

Your endpoints are now ready to receive the DM profile.

### Distributing the DM Profile

With the Tivoli desktop, there are a few ways of distributing the DM profile. The following steps describe a simple way of distributing the profile.

1. Return to the Profile Manager with the DM Profile and the subscribed Endpoints (`dm.itso.linux-s390`).
2. Holding your CTRL key, select each Endpoint that should receive the profile. We selected our three Endpoints, LNX5, LNX12 and LNX17.
3. Click the DM Profile and drag it over to the selected Endpoints. Your cursor will change to an hourglass to show that the distribution is occurring, and return to a mouse cursor when it is complete. If there is an error, you'll get an error dialog box describing the error to you, which you will need to resolve.

If you do get an error, it will most likely be because DM can't find the Java; double check that you performed the task as described in the previous section. If you are not sure, execute the `DMRemoveLinkJRE` task in the Tivoli Distributed Monitoring (Advanced Edition) Task Library and redo the previous section.

**Note:** The first distribution will take longer than subsequent ones. This is because the Endpoint will be downloading the DM application from its gateway for the first time. Subsequent distributions should be much quicker.

You can use the Tivoli command line tool `wdmlseng` to query the DM engine on an Endpoint to see what monitors it has, and what state the monitor is in. Example 10-1 shows a sample output of the `wdmlseng` command.

*Example 10-1 Command line tool to query DM engine.*

---

```
lnx8:/root # wdmlseng -e lnx12
```

Forwarding the request to the engine...

The following profiles are running:

```
dm.itso.linux-s390.all#itso
  DMXProcess: Running
  DMXFileSystem: Running
  DMXNetworkInterface: Running
  DMXSecurity: Running
  DMXMemory: Running
  DMXFile: Running
  DMXCpu: Running
```

---

## Enabling DM heartbeat

DM includes a heartbeat function, which monitors the basic signs of life at Endpoints attached to the gateway that has the heartbeat enabled.

Heartbeat will send alerts when:

- ▶ The heartbeat has stopped on an Endpoint
- ▶ Resource models have errors
- ▶ The DM engine has stopped
- ▶ An Endpoint is no longer on the network

## Configuring the heartbeat

The heartbeat configuration is controlled by the `wdmconfig` command line, which is available on any Managed Node and the TMR server that has DM installed. You can choose to have your heartbeat events go to a TEC server if you have one, otherwise they can be sent to a Tivoli Notice board and be readable from the Tivoli Desktop.

We configured our DM heartbeat with the following command:

```
wdmconfig ALL -D heartbeat.send_events_to_tec=true -D heartbeat.tec_server=tot29 -D
heartbeat.send_events_to_notice=true
```

This configured the heartbeat to send events to our TEC server `tot29` and also to the Tivoli Notice group.

## Controlling the heartbeat

The `wdmheartbeat` command controls the status of the heartbeat. To start the heartbeat, issue the following command:

```
wdmheartbeat -s 60 -m all
```

This will start the heartbeat on all Managed Nodes and poll the endpoints at 60 second intervals.

To check the status of the heartbeat, use the `wdmheartbeat` command with the `-q` option. Example 10-2 shows a sample output from querying the status of the heartbeat.

*Example 10-2 Status of the DM heartbeat engine.*

```
lnx8:~ # wdmheartbeat -m all -q
Processing ManagedNode tot11...
HeartBeat processor status: STARTED, time interval: 60
Processing ManagedNode lnx9...
HeartBeat processor status: STARTED, time interval: 60
Processing ManagedNode lnx8...
HeartBeat processor status: STARTED, time interval: 60
```

To stop the heartbeat, you can use the **wdmmn** command with the **-h** option:

```
wdmmn -stop -m all -h
```

To view the heartbeat status of each endpoint, you can use the **wdmmngcache** command. Example 10-3 shows the status output of our endpoints.

*Example 10-3 Heartbeat status of Endpoints*

```
lnx8:~ # wdmngcache -m all -l
Processing ManagedNode tot11...
Processing ManagedNode lnx9...
Processing ManagedNode lnx8...
Endpoint | HB status
-----+-----
lnx17 | Alive
lnx12 | Alive
lnx5 | Alive
```

Figure 10-8 shows an example of the messages you may receive if you enable the heartbeat to send status information to the Tivoli notice board.

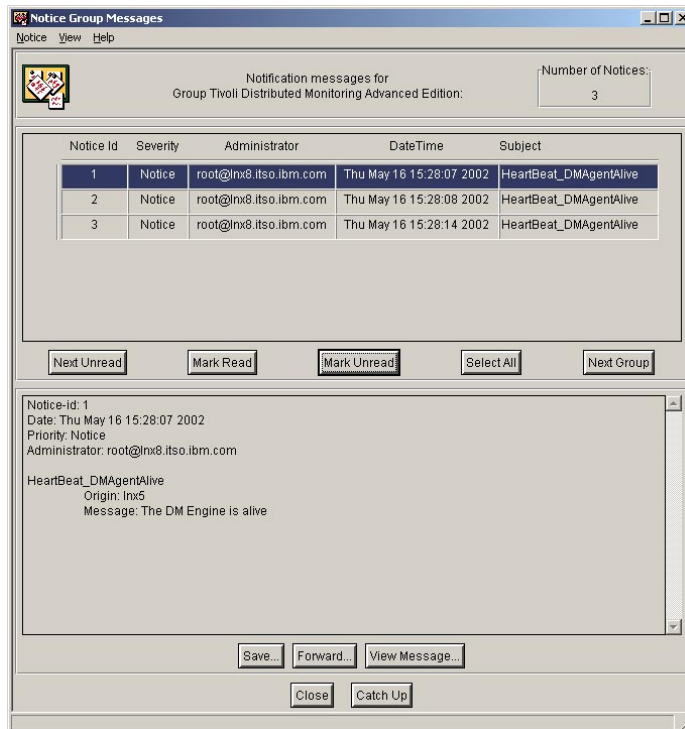


Figure 10-8 Tivoli Notice board with heartbeat status information

## Health monitoring

You can use the health console to check, display, and analyze the health and status of any endpoint to which DM profiles and resource models have been distributed. It is an ideal tool to use after you have been alerted of a problem (say via TEC) and you want to explore the problem further.

The health console shows the status of the resource models in real time; you can also view the historical data stored at an endpoint if it has been enabled.

You can use the health console to perform the following:

- ▶ Select resource models running on an endpoint
- ▶ Start and stop the DM engine on an endpoint
- ▶ Start and stop resource models
- ▶ Remove a profile from an endpoint
- ▶ View the health of all resource models on an endpoint
- ▶ Browse the health of all instances of a particular event
- ▶ Display real-time data of an instance of a problem
- ▶ Display up to 24 hours of recent history data of a resource model
- ▶ View the messages issued by the health console

## What is health?

The health of a resource is a calculated number between 0 and 100, where 100 indicates the system is in perfect condition, and 0 shows that an event has been generated indicating an issue. Numbers in between are indications of problems arising on the endpoint.

The health is calculated by determining the number of holes and occurrences that make up an event cycle. (The number of holes or occurrences is configured within the DM resource model of a monitoring profile.)

For example, if an event is defined by 3 occurrences (O) allowing for 1 holes (H), then an event is determined over 4 monitoring cycles. Thus during monitoring, when there are *more than* 1 holes, the indication will be reset to 100 indicating that the system is operating at perfect health.

The number of occurrences, up to 3 during a 4 monitor cycle period, will calculate a health, where 3 indications will result in 0, indicating an event has been generated. Here is an example:

Cycles	Health
O O O O	0 (no holes, event generated)
O H O O	0 (event generated)
H O H O	33 (2 occurrences separated by 1 hole cannot trigger event)
O H H O	66 (2 consecutive holes resets the counter to 100, this only one occurrence after the reset.)
O H O H	100 (2 holes restarts the count of occurrences)
H H O H	66 (final hole is ignored)
H H H O	66 (new occurrence)
H H H H	100 (no occurrences)



## Starting the Health Console


To start the DM health console on your Windows PC, select **Start -> Programs -> Distributed Monitoring (Advanced Edition) 4.1-> Health Console**. You will be prompted with a login box similar to Figure 10-9.



Figure 10-9 DM Health Console login

The first time you start the Health Console, you'll have an empty screen, where you can add the endpoints that you wish to monitor.

Choose **File -> Add Endpoint** to add each of the endpoints that are being monitored with DM. Once added, you'll be able to navigate to each endpoint and view their current health.

We added our three endpoints (LNX5, LNX12, and LNX17) to our Health Console and we were able to immediately observe a problem with our LNX12 and LNX17 systems (note the  next to the endpoint name in Figure 10-10).

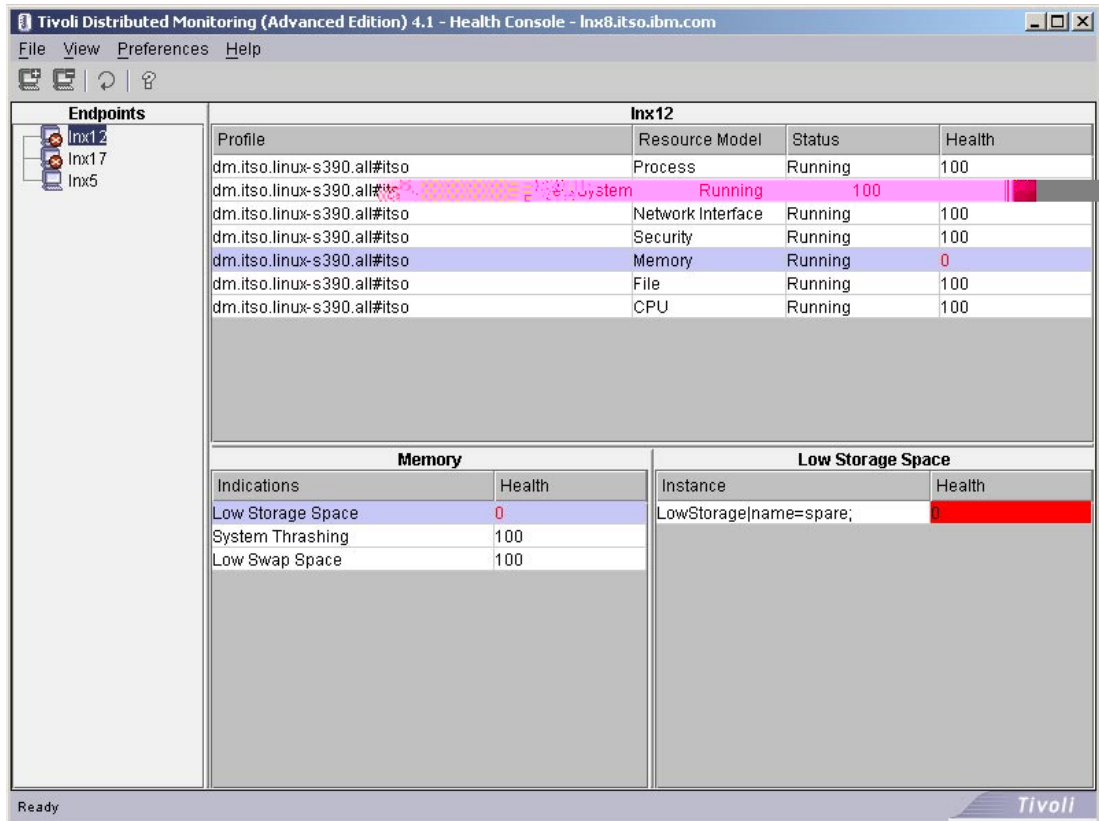


Figure 10-10 Health Console showing problems with LNX12 and LNX17

**The reason:** LNX12 and LNX17 were both showing 0 (zero) health on the Memory resource model. When we drilled down further we found that the Low Storage Space indicator had generated the event.

The Low Storage Space indicator (by default) triggers when there is less than 40 % virtual storage available (also known as swap). The indicator generated an event because we didn't actually have any swap on these systems – we had forgotten to set up the swap partition!

As soon as we set up the swap, both systems went back to perfect health (100).

## 10.1.2 Software deployment with Tivoli Software Distribution

Software deployment management is provided by Tivoli Software Distribution (SD). SD enables you to deploy software (which is actually just files and data) to systems and optionally install it (or execute any process after deployment).

SD can use an optimized network route to deploy your software throughout your network (which really doesn't apply to your virtual machines running under z/VM). Software can be scheduled for deployment at any time in the future, and the recipients of the software need not be online to receive it. When they come online, their computers will automatically check for any outstanding software deployments and initiate them if required.

In this section, we describe how to create a TSM client software package and deploy it to all your Linux/390 systems. (We assume that you have already installed SD as described in 9.6, “Tivoli Software Distribution” on page 188.)

## Installing the Software Package Editor

One of the first applications you will want to deploy with SD is the Software Package Editor (SPE). The SPE is a Java client tool that enables you to create software packages which contain the applications or data that you want to deploy to systems. It is much easier to create packages with this tool. (This tool is not supported on Linux/390, so you can deploy it to a Windows 2000 system that has a Tivoli endpoint in your TMR).

First, launch the Tivoli desktop and log in as the **root** Tivoli Administrator.

1. Create a policy region to hold your Software Distribution profiles and profile managers by selecting **Create -> Region**. Assign a name (we called ours `sd.itso`), then click **Create & Close**.
2. Change the Managed Resources so that it includes Profile Manager and SoftwarePackage profiles. Right-click `sd.itso` and select **Managed Resources**.



Figure 10-11 Defining Managed Resources to `sd.itso`

3. Open the `sd.itso` Policy Region by double-clicking it.
4. Create a Profile Manager to hold your SoftwarePackage profiles. Select **Create -> ProfileManager**. Give it a name (we called ours `sd.itso.misc`) and click **Create & Close**. There is an option for “Dataless Endpoint Mode;” select this option if you will have endpoints directly subscribed to this profile manager (we did).
5. Open the new profile manager `sd.itso.misc` and create a profile for your SPE. Select **Create -> Profile**, give it a name (we used `sp.itso.misc.spe^4.0`), and click **Create & Close**.

Our final Profile Manager is shown in Figure 10-18 on page 226. The new SoftwarePackage profile looks like a yellow, open box.

6. Import the SPE from the SD v4.0 CD. Right-click the SP profile, choose **Import**, and then point to the `/PACKAGE/PREPSITE/` directory on the SDv4.0 CD. Select the

SD40EP\_NT.spb package to import and click **Select File & Close** (see Figure 10-12). Choose where you want to import the SoftwarePackage to. (We created the directory /opt/tivoli/SoftwarePackage to store all our packages.)

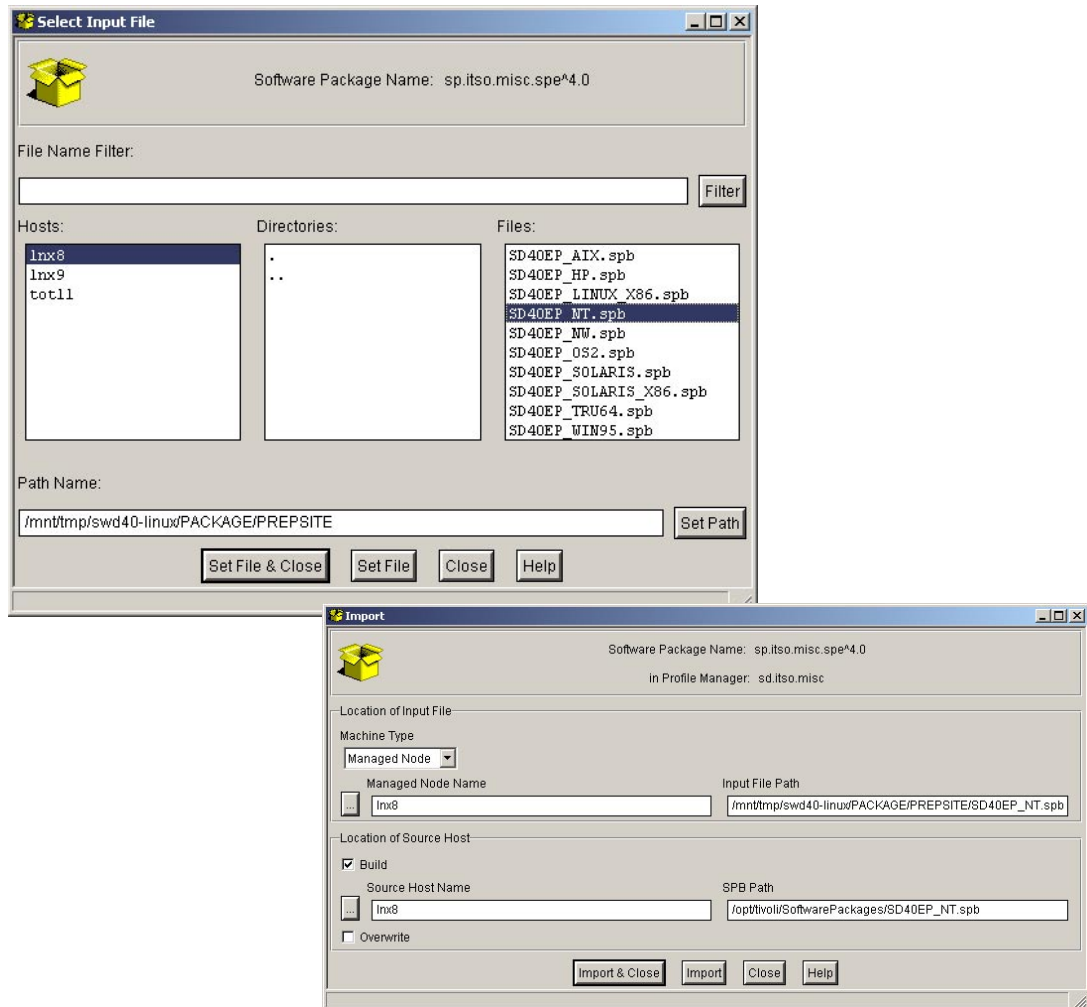


Figure 10-12 Importing the SPE Editor from the SD v4.0 CD.

7. Now subscribe your Windows workstation to this profile manager. Select **Profile Manager** -> **Subscribers**.
8. Now you can distribute the SPE to your desktop by clicking the **SPE SoftwarePackage** and dragging it onto the Windows workstation. If all went well, you should have the SPE tool on your desktop and under Start -> Programs -> Software Distribution 4.0

## Building a package with the Software Package Editor

You are now ready to create a SoftwarePackage for the TSM Client. This section describes how to create the package in your sd.itso.misc Profile Manager.



1. First, create a SoftwarePackage profile for your software, and call it sp.itso.misc.tsm^4.20. Do this by selecting **Create** -> **Profile** and entering sp.itso.misc.tsm^4.20. This will create an icon like the one shown at left – an open, empty box indicating that this is a SoftwarePackage profile with no contents.
2. Right-click the new SoftwarePackage icon and choose **Properties**.
3. Click **Launch Software Package Editor**.

4. Select the Source host (our source host was LNX8) and click **Set & Close**. The SPE should automatically be launched and on your desktop.
5. Right-click the package name and select **Properties**; update the details as appropriate.
 

Make sure you have “Stop on failure” selected. This ensures that the package deployment will be stopped if any conditions you put in place for it are not met. (For example, in the next step you will specify that a minimum amount of disk space must be available.)

You can scroll through the other options and review anything that you would like to change.

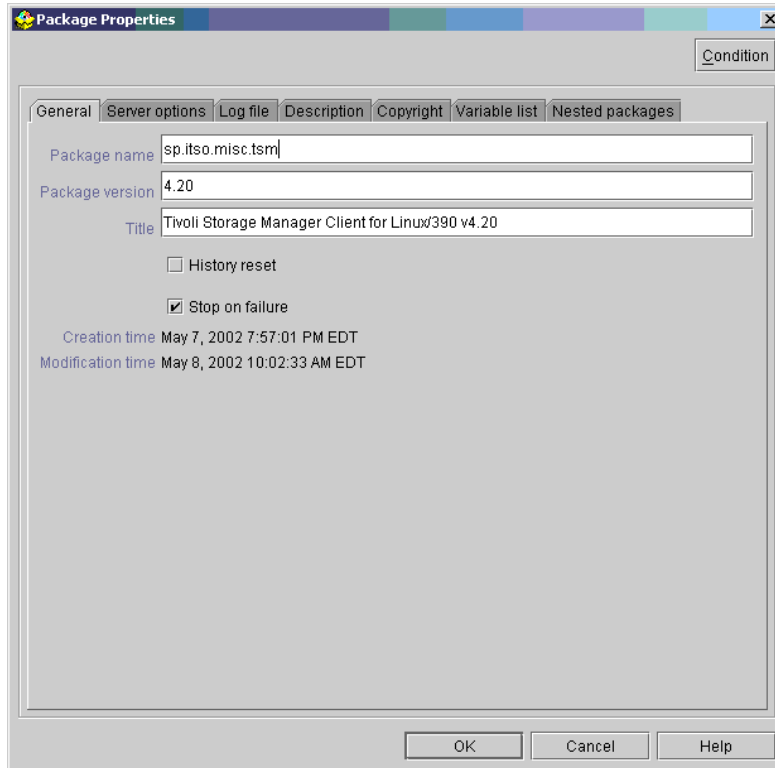


Figure 10-13 Software package editor: package properties

6. Add a disk space check since the Release Notes indicate you need 30 MB of space on the file system where /opt/tivoli/tsm/client/ba resides. Also, make sure that you have 10 MB free on your /tmp file system since you will distribute the TIVsm-BA.s390.rpm there.
 

(Actually, **rpm** will also do disk space checks for you, and won't install the package if there is insufficient space.)

Right-click the package name, then select **Insert -> System Action -> Check disk space**.

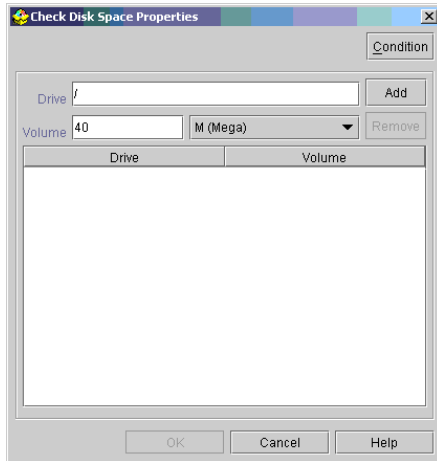


Figure 10-14 Software Package system action: Disk space check

**Note:** On our Linux virtual machine, we only had one file system / (root), so our disk space check makes sure we have 40 MB free (30 MB for the install and our 10 MB for /tmp).

7. Add the RPM to the package as the file to distribute. Right-click the package name, then select **Insert -> Add Object -> Directory**. We stored our RPM in the following directory on LNX8 (our Source Host):

/opt/tivoli/SoftwarePackage/sd.itso.misc.tsm/TIVsm-BA.s390.rpm

On the Advanced dialog tab, mark the files “Temporary” to ensure that SD removes any transferred files after successful installation.

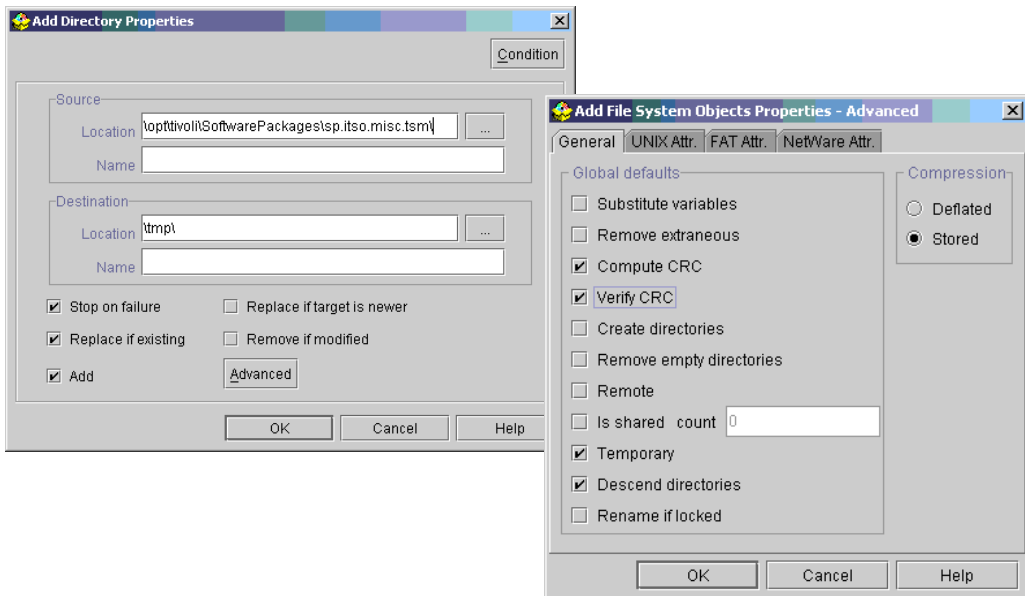


Figure 10-15 Defining the files to distribute

Now right-click the directory name, then select **Insert -> File** and add TIVsm-BA.s390.rpm.

8. Add commands to have the package installed during distribution. Right-click the package name, then select **Insert -> Program -> Execute Program**.

Fill in the following tabs.

Table 10-1 Install, remove and verify actions

Action	Program & Path	Arguments	Working Directory
Install	/bin/rpm	-ivh TIVsm-BA.s390.rpm	/tmp
Remove	/bin/rpm	-e TIVsm-BA	N/A
Verify	/bin/rpm	-V TIVsm-BA	N/A

**Tip:** For debugging, select the “Reporting standard error file on Server” option on the Advanced tab, and the same for “Standard output,” so that standard error and standard output are logged on the TMR Server. The log file name is specified by the **Package Properties -> Log File** option.

**Install vs Commit:** Tivoli Software distribution has 2 modes of operation for installation.

► Installation

The *installation* mode is normally associated with moving data from a source host to the destination and, optionally, *preparing* the data for installation. This mode of operation is useful if you have a large software deployment that needs to take place, but you want to *install* the application on a set date (or over a short period of time).

Since network links (especially slow ones) can affect the deployment and installation time of an application over a large network, with this phase of Tivoli Software deployment, you could send out an application ahead of the install date (weeks in advance, if appropriate), and have the application on the local destination host ready for *install*.

The SD install actions may be used to unzip the application once it arrives, clean up some space, or do pre-installation/preparation processing that doesn't involve the actual installation of the application.

► Commit

The *commit* mode is normally used to activate or *install* an application. It doesn't involve moving of the data from the source host to the destination, but instead assumes the application is already on the destination system, dormant, ready for install.

With these two modes of operation, you can initiate an SD Install distribution, which would be used to send out the application to all your systems, and then later, send out an SD commit distribution to actually perform the install.

Since the commit distribution doesn't involve sending of the actual files that make up the application, it operates and completes much faster and enables you to perform the actual install over a short period of time.

If you are not concerned with installing (deploying) and later committing (installing), then your install action could do both (deploy and install) and your commit action can be blank.

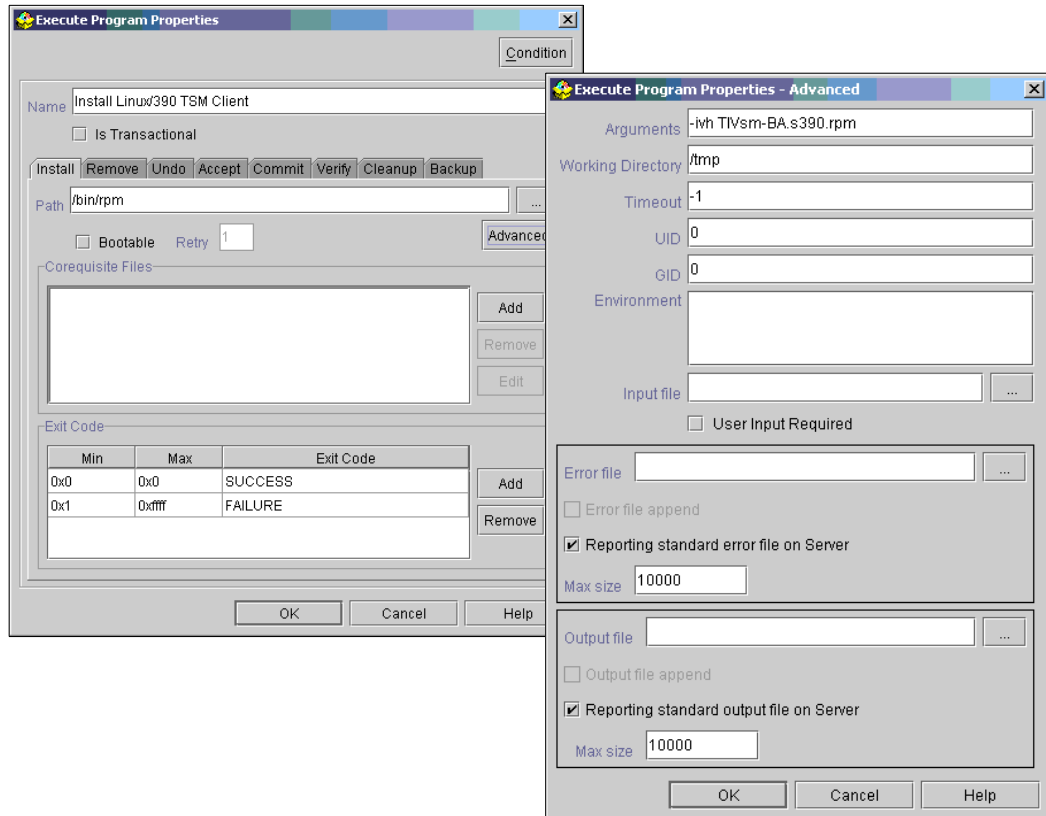


Figure 10-16 Package installation commands and attributes

- Next, add your dsm.sys and dsm.opt configuration files. If the installation in the previous step was successful, your configuration files can go directly into the /opt/tivoli/tsm/client/ba/bin directory. Right-click the package name, then select **Insert -> Add Object -> Directory**. Then right-click the directory, select **Insert -> File** and add two entries, one for dsm.opt and one for dsm.sys. (We used our TSM client configuration files from Example 9-2 on page 159 and Example on page 159.)



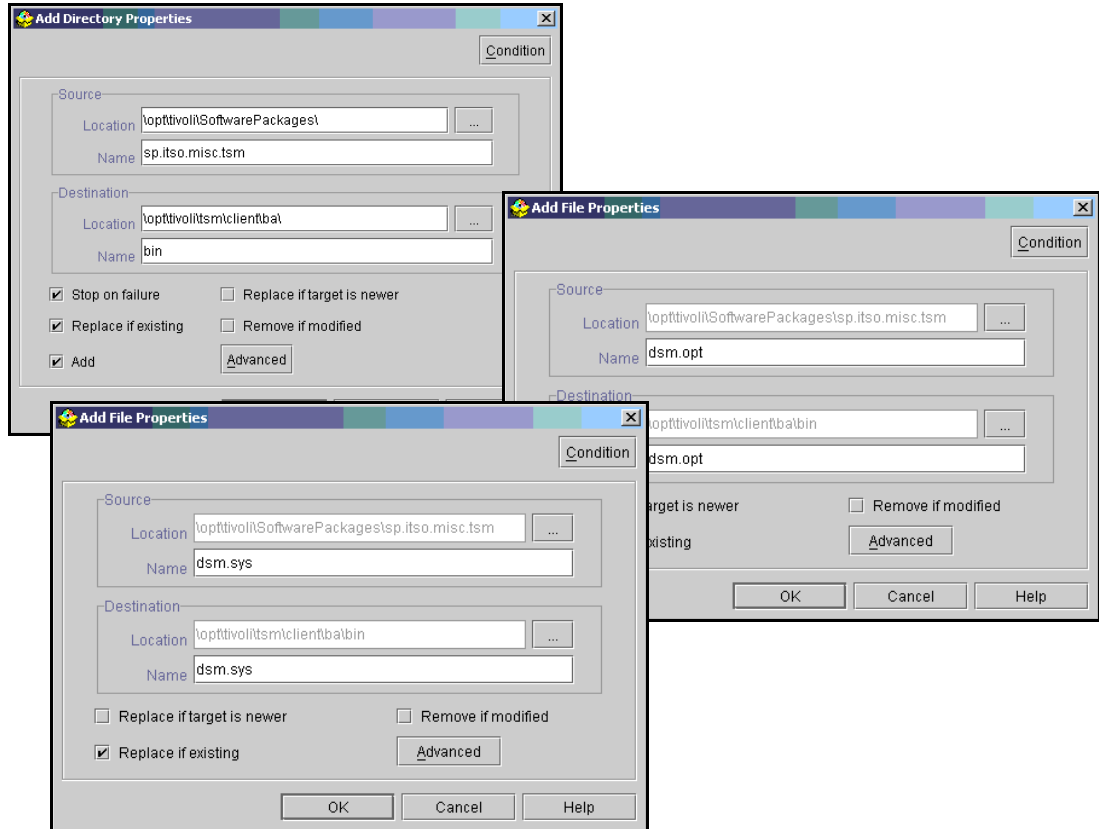


Figure 10-17 Adding *dsm.opt* and *dsm.sys* files

**Note:** The only thing missing from your TSM client deployment is a script that can be used to start the TSM client at system boot. Unfortunately, there isn't one provided with the rpm, so we should have added one to our distribution to make it complete.



10. Select **File -> Return** to save your options back in the profile, and you are ready to distribute the package. When you return to the Tivoli Profile Manager, your yellow box icon will have changed to a box with a CD inside, indicating that the package now has contents.

With the box left in the *open* or *unbuilt* state, the files required for the distribution will be collected at distribution time, bundled, and sent to each endpoint.

You could *build* the package, which would collect all files that make the package, bundle them together, and effectively freeze the contents and use them for distribution. Thus, any changes made (in the `/opt/tivoli/SoftwarePackage/sp.itso.misc.tsm`) won't be included.

To build the package, right-click the SoftwarePackage icon and select **Convert**, then input the directory where the built software package (SPB) will be stored. We stored ours as `/opt/tivoli/SoftwarePackages/sp.itso.misc.tsm^4.20`. And finally, convert and close.



The icon for the built software package is a closed box (with tape and all), indicating that it's contents are complete, ready for deployment, and cannot be changed. You can convert packages between the built and unbuilt stages at any time though.

11. You need to subscribe your Endpoints to the profile manager so that they are applicable targets for your TSM client. In the profile manager choose **Profile Manager -> Subscribers** and add the Endpoints that you want to deliver this to. Figure 10-18 shows how ours looked.

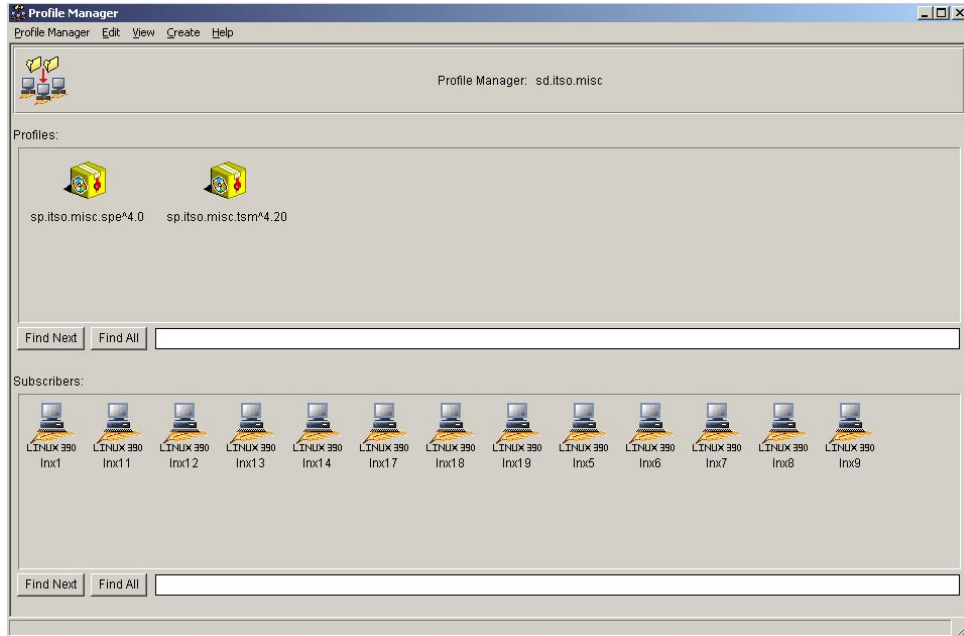


Figure 10-18 Profile manager sd.itso.misc and its profiles and subscribes.

12. To distribute your package to all the subscribers, right-click sp.itso.misc.tsm^4.20 and select **Install**, then click **Install & Close**.

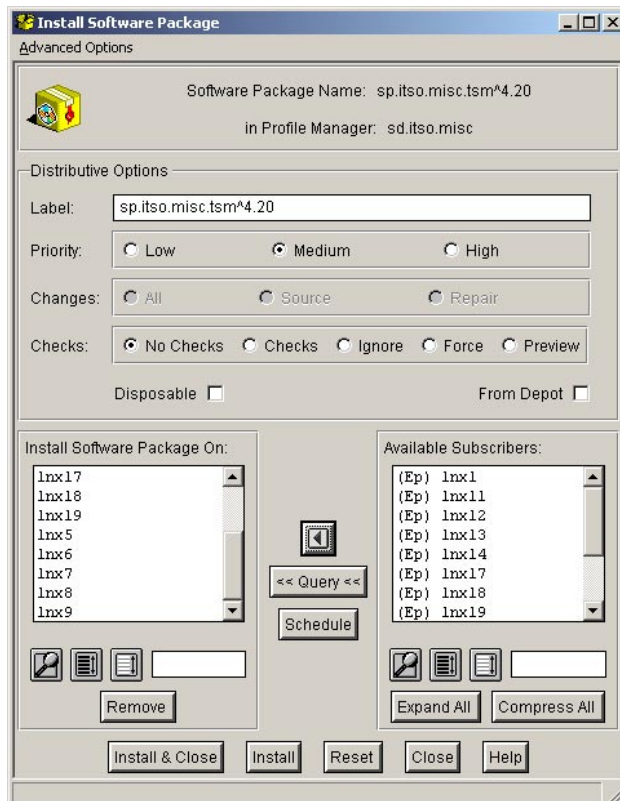


Figure 10-19 Installation dialog

There are two ways to verify that your installation was successful:

- ▶ Review the log file as defined in the SoftwarePackage profile.  
We set ours to /opt/tivoli/swdist/work/sd.itso.misc.tsm^4.20.log. Reviewing it showed that our distribution was successful to 9 of the 13 nodes. The reasons for the four failures were as follows:
  - LNX1 didn't have the required libraries installed, so the RPM installation part failed.
  - LNX8, LNX9 and LNX17 didn't have the minimum space requirement of 40 MB free on the root file system.
- ▶ Review the distribution status using the Distribution Monitor; this is covered in the next section.

## Tivoli Distribution Status Monitor

The Tivoli MDIST GUI or Status Monitor provides a Java GUI display of the current MDist2 distributions. Currently only Tivoli Software Distribution uses MDist2.

The MDIST GUI interface is only available on Managed Nodes, but it is not available on Linux/390 Managed Nodes. We installed our MDIST GUI components on TOT11, our Windows Managed Node.

Normally you can start the MDIST GUI using the Tivoli Desktop, by clicking on the distribution status icon. Alternatively, you can start the MDIST GUI by using the command line of your Managed Node (where it is installed).

Start a DOS command window and use the following commands to start the MDIST GUI:

```
C:\>c:\winnt\system32\drivers\etc\tivoli\setup_env
Tivoli environment variables configured.
C:\>bash
bash$ wmdistgui
Launching MDist 2 GUI, please wait ...
```

You should then be presented with a login interface. Use your TMR server's hostname (or IP address) and log in as you would to the Tivoli Desktop. You will be presented with the MDist GUI and you'll be able to see the status of your distributions, as shown in Figure 10-20 on page 228.



Figure 10-20 Distribution Status

## 10.2 Data Management

### 10.2.1 File backup and restore

In the previous chapter we described how to install, configure, and start the TSM client scheduler, and how to start the TSM client Web interface. In this section we briefly cover how to use the client via the command line, as well as the Web interface.

In a normal environment, it is assumed that your Linux/390 images would be backed up via a backup schedule configured on the TSM server. That configuration and setup is not covered in this book (it's a server configuration). We recommend that you review your TSM server documentation on how to configure client schedules.

## TSM client command line dsmc

The TSM client has four modes of operation, two of which we discuss here (Backup and Restore).

### ► Backup

The backup mode enables you to incrementally (the default) back up files on your file system, or selectively back up files (re-send files to the server, even though they have not changed) on your file system. The mode is used when you invoke the client with the **I** or **S** options, respectively.

Example 10-4 shows an incremental backup; Example 10-5 is a selective backup.

#### *Example 10-4 TSM client incremental backup*

---

```
[root@lnx19 root]# dsmc i
Tivoli Storage Manager
Command Line Backup Client Interface - Version 4, Release 2, Level 0.0
(C) Copyright IBM Corporation, 1990, 2001, All Rights Reserved.

Node Name: LNX19
Session established with server ADSM: MVS
  Server Version 4, Release 2, Level 1.0
  Server date/time: 29-04-2002 14:30:52  Last access: 29-04-2002 13:48:53

Incremental backup of volume '/'
Successful incremental backup of '/'

Total number of objects inspected:      123420
Total number of objects backed up:      0
Total number of objects updated:        0
Total number of objects rebound:        0
Total number of objects deleted:         0
Total number of objects expired:         0
Total number of objects failed:         0
Total number of bytes transferred:      0 Bytes
Data transfer time:                      0.00 sec
Network data transfer rate:              0.00 KB/sec
Aggregate data transfer rate:            0.00 KB/sec
Objects compressed by:                   0%
Elapsed processing time:                  00:03:05
```

---

Example 10-4 starts an incremental, full file system backup. The first time we ran this backup, it sent all 123420 objects to the TSM server. Since we ran this a second time, and no files have changed, there was nothing to do and it complete much quicker than the first time.

#### *Example 10-5 TSM client selective backup*

---

```
[root@lnx19 root]# dsmc s /home/
Tivoli Storage Manager
Command Line Backup Client Interface - Version 4, Release 2, Level 0.0
(C) Copyright IBM Corporation, 1990, 2001, All Rights Reserved.

Selective Backup function invoked.

Node Name: LNX19
Session established with server ADSM: MVS
  Server Version 4, Release 2, Level 1.0
  Server date/time: 29-04-2002 14:35:27  Last access: 29-04-2002 14:35:19
```

```

Directory-->          4,096 /home/ [Sent]
Directory-->          4,096 /home/deon [Sent]
Directory-->          4,096 /home/deon/tmp [Sent]
Normal File-->        24 /home/deon/.bash_logout [Sent]
Normal File-->        191 /home/deon/.bash_profile [Sent]
Normal File-->        124 /home/deon/.bashrc [Sent]
Normal File-->        820 /home/deon/.emacs [Sent]
Normal File-->        3,511 /home/deon/.screenrc [Sent]
Normal File-->         76 /home/deon/.bash_history [Sent]
Normal File-->       1,048,576 /home/deon/tmp/lmg [Sent]
Selective Backup processing of '/home/*' finished without failure.

```

```

Total number of objects inspected:      10
Total number of objects backed up:      10
Total number of objects updated:        0
Total number of objects rebound:       0
Total number of objects deleted:        0
Total number of objects expired:        0
Total number of objects failed:         0
Total number of bytes transferred:      1.00 MB
Data transfer time:                     2.04 sec
Network data transfer rate:             503.92 KB/sec
Aggregate data transfer rate:           255.52 KB/sec
Objects compressed by:                  0%
Elapsed processing time:                 00:00:04

```

---

Example 10-5 performs a selective backup beginning at /home. A selective backup is a way of re-sending files to the TSM server even though they have not changed since they were backed up previously (via an incremental or selective backup process). You wouldn't normally use the selective backup.

► **Restore**

The TSM restore function is used to get backed up data from TSM and place it back on your file system. The backed up data could have been backed up with either the incremental backup method or the selective backup method. In Example 10-6 we deleted the file /home/deon/tmp/lmg that was backed up previously with the selective backup, and then restored it.

*Example 10-6* TSM client restore

---

```

[root@lnx19 root]# dsmc restore /home/deon/tmp/lmg
Tivoli Storage Manager
Command Line Backup Client Interface - Version 4, Release 2, Level 0.0
(C) Copyright IBM Corporation, 1990, 2001, All Rights Reserved.

Restore function invoked.

Node Name: LNX19
Session established with server ADSM: MVS
  Server Version 4, Release 2, Level 1.0
  Server date/time: 29-04-2002 14:56:52  Last access: 29-04-2002 14:35:34

Restoring      1,048,576 /home/deon/tmp/lmg [Done]

Restore processing finished.

Total number of objects restored:        1
Total number of objects failed:          0
Total number of bytes transferred:      1.00 MB
Data transfer time:                     1.21 sec

```

Network data transfer rate: 843.84 KB/sec  
Aggregate data transfer rate: 224.00 KB/sec  
Elapsed processing time: 00:00:04

---

The other two client modes of operation, *archive* and *retrieve*, are used for longer term storage of data.

Archived data is expired from the TSM server by using the number of days since the archive is created. This gives you the ability to keep archived data for several years if it is required (for example, financial data).

Incremental and selective backup files are expired only when they have been replaced by a newer version of the file, and are controlled either by the number of versions stored on the TSM server, the number of days since the files have changed on the TSM server, or a combination of both settings.

Generally speaking, archived data is sent to slower media for longer term storage (and sometimes is sent offsite), while backup/restore data is kept on faster media (including disk) for quicker backup and restore time.

### TSM Web client interface

The TSM Web client interface is available by pointing your a Web browser to the client using port 1581 (unless you changed it by including the HTTPport option in the client dsm.sys file). To see the Web interface for LNX19, we used the following URL:

<http://lrx19.itso.ibm.com:1581>

This brought up the interface shown in Figure 10-21 (this interface will be familiar to those who have used the Windows TSM Client).



Figure 10-21 TSM Web client

Performing a restore with the Web GUI is quite intuitive. Figure 10-22 is an example of restoring the file /home/TIVsm-BA.s390.rpm after it was deleted.

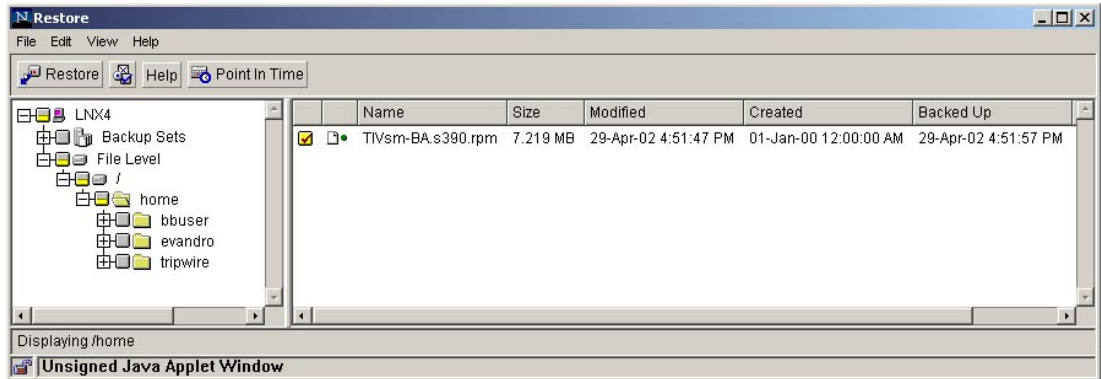


Figure 10-22 Selecting a file for restore.

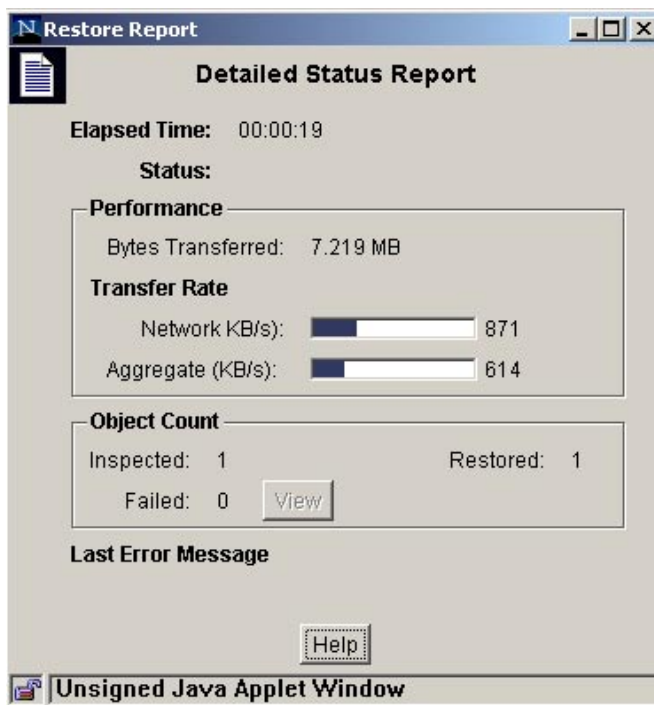


Figure 10-23 TSM Web GUI restore complete



**Problems with the TSM Web GUI:** We had a couple of problems when using the Tivoli Web Interface on our systems:

▶ RedHat 7.2 (LNX19)

It appears that the Java Interface didn't recognize our file systems (it was blank) even though we had a `VIRTUALMountpoint` directive in our `dsm.sys` file (our file systems were EXT3). Thus we weren't able to perform any backups with the Web GUI.

We were able to successfully restore with the Web GUI (which probably is what you would use it for most, anyway).

▶ SuSE 7.0 SLES (LNX4) & SuSE 7.0 (LNX10)

While SuSE installs the SuSE 7.0 SLES with the ext2 file system, we got an error dialog when selecting Backup -> Local -> /. Thus we weren't able to successfully perform a backup using the Web GUI.

After some research we discovered that this is a known bug with the TSM v4.2.0 client for Linux/390. The fix is scheduled in v4.2.2, which had an estimated release date of May 2002. (The APAR number is IC30292.)

As with the RedHat system, we were able to successfully perform a restore with the Web GUI.

## 10.3 Security

### 10.3.1 User definition and administration

In this section we describe two scenarios for managing user accounts on Linux/390 VMs. The first uses IBM Tivoli Identity Manager to manage `/etc/passwd` and other local files; the second uses IBM Tivoli Identity Manager to manage an LDAP directory that the Linux/390 VMs are configured to use for authentication.

In both cases, Identity Manager maintains user account information in user profile objects that are stored in the TMR server's object database. These user profile objects are distributed to either the endpoint object representing the Linux/390 VM or to an LDAP connection object representing the LDAP server.

To create a user, the administrator logs in to Identity Manager with a supported browser, the returned screen is shown in Figure 10-24.



Figure 10-24 IBM Tivoli Identity Manager administrator login page

The left-hand column presents the administrator with standard user management features like create, manage, and delete users.

Regular users can also log in to Identity Manager using a Web browser and receive a subset of management tools. The **Update My Account** option allows the user to change only data that has been allowed by the administrator. Some examples of changeable data are login shell, home phone number, and home page URL.

Information which has been changed by a user can be reviewed by an administrator before taking effect. Using a feature called *Workflow*, Identity Manager can be configured to hold changes made by users, send an email to a specified administrator, and wait for the administrator's consent or modifications before committing the user's requested change.

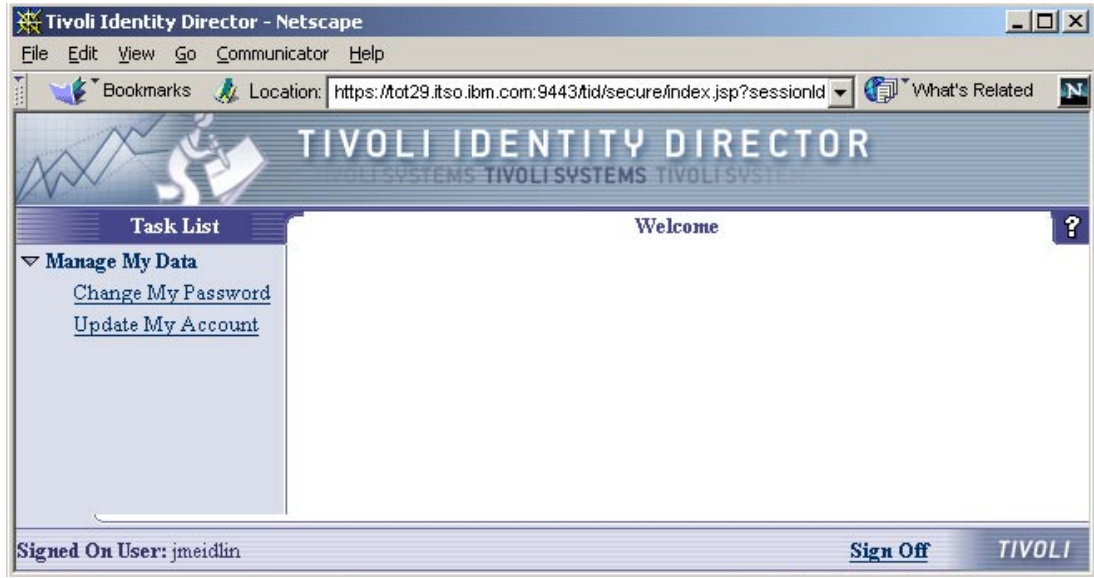


Figure 10-25 IBM Tivoli Identity Manager regular user login page

Create a user profile object called Linux390 to hold the data of your users, placed in the Identity profile manager.

*Example 10-7 IBM Tivoli Identity Manager create profile*

---

```
wcrtprf @ProfileManager:Identity UserProfile Linux390
```

---

Log in to the Identity Manager server and create user accounts by following these steps:

1. Log in to the browser as tid\_admin.  
<https://tot29.itso.ibm.com:9443/tid>
2. Click **Create User**.
3. Select the Linux390 profile and click **OK**.

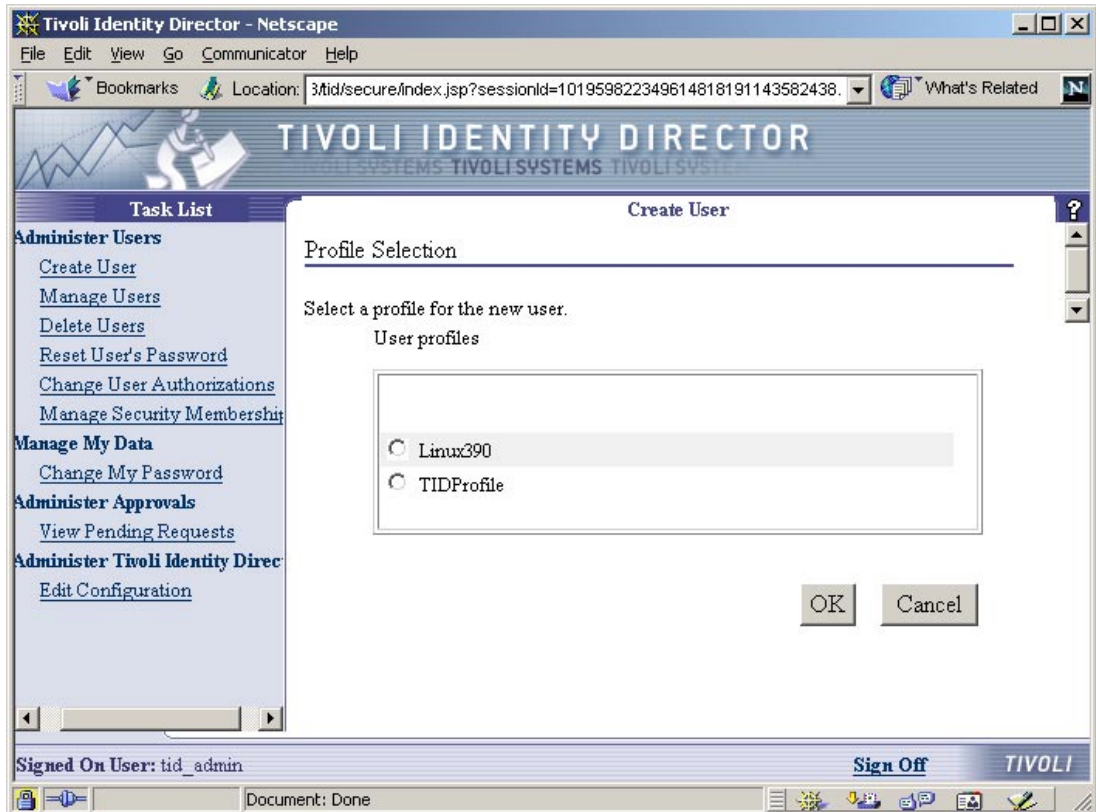


Figure 10-26 IBM Tivoli Identity Manager profile selection

4. Fill in the user information and click **Create**.

At this point, the user information is in the user profile object, and the methods of distributing it to system files and the LDAP directory are explained in the next section.

## Identity Manager managing files

To get the user profile information into the Linux/390 system files, you need to first subscribe the TMAs of the Linux/390 VMs to the Identity profile manager. (Example 10-8)

*Example 10-8 IBM Tivoli Identity Manager subscribe TMAs to profile manager*

---

```
wsub @ProfileManager:Identity @Endpoint:lnx10
wsub @ProfileManager:Identity @Endpoint:lnx9
```

---

Once the subscription hierarchy is in place, the profiles can be distributed to the endpoints and the system files are updated. (Example 10-9)

*Example 10-9 Distribute profile to endpoints*

---

```
wdistrib -m -l maintain @UserProfile:Linux390 @Endpoint:lnx10
```

---

Once the distribution is complete, you will be able to see the new user account records in the Linux/390 system files.

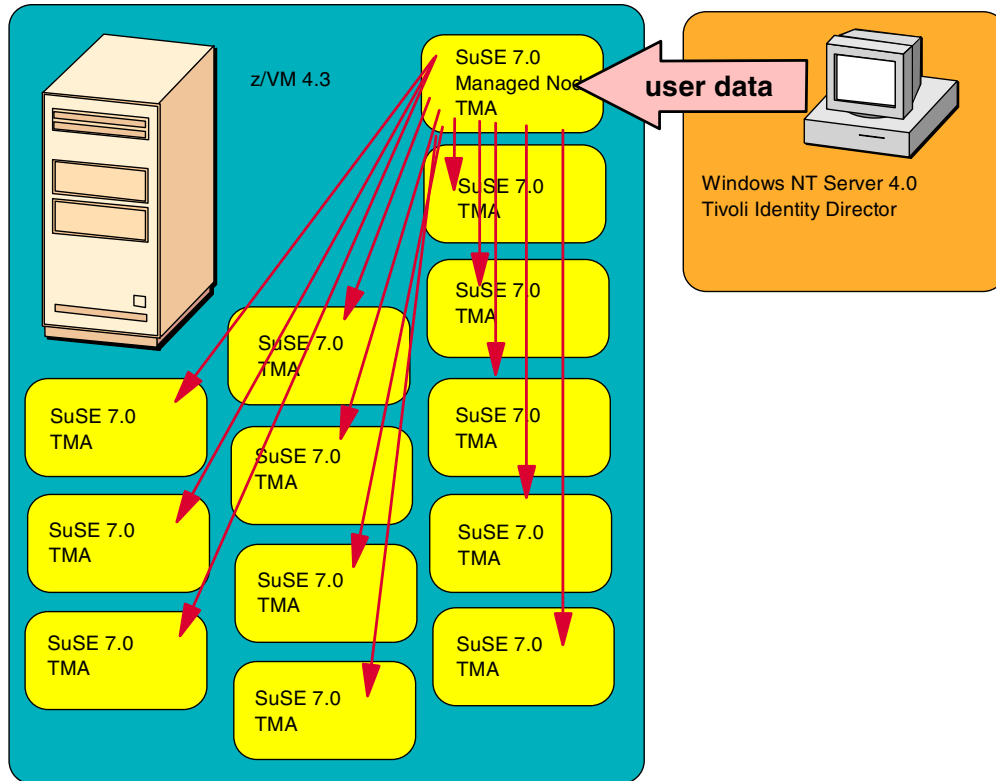


Figure 10-27 Distribution of user account data to local files (like /etc/passwd)

If all subscribing endpoints use the same user profile, only one copy of the profile needs to be distributed to the managed node VM, and then it distributes a copy to each subscribing VM.

## Identity Manager managing LDAP

To get the user profile information into the LDAP directory, we needed to create an LDAP connection object. You must add *LDAP connection* to the list of managed resources for your policy region, and then LDAP Connection will appear on the Create menu. Once you select it, you can create the connection object by defining the managed node where the object will reside and the name of the object.

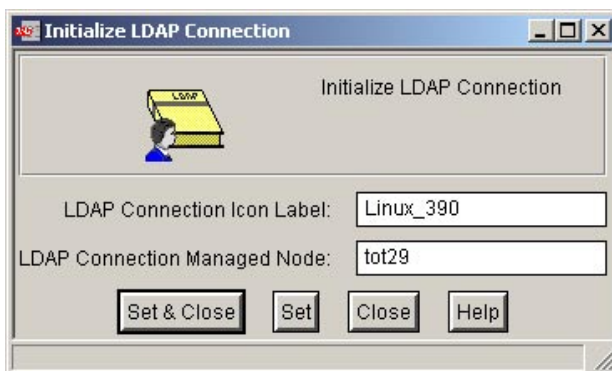


Figure 10-28 IBM Tivoli Identity Manager Create LDAP Connection Object

In this example, we created the LDAP Connection object on the TMR server. A distribution to the LDAP server will go directly from the managed node with the connection object, in this case the TMR server, to the LDAP server.

The Linux/390 VMs are then configured to use LDAP for login authentication and send their queries to the LDAP server.

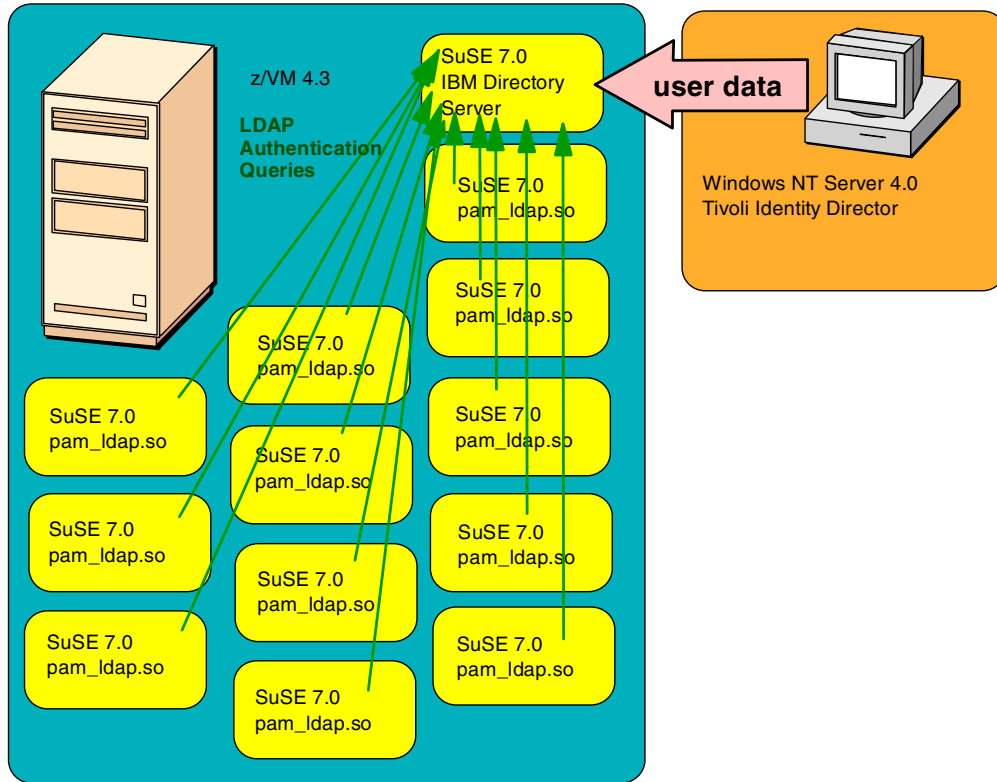


Figure 10-29 Distribution of user data to LDAP server in Linux VM

### Attribute maps

Attribute maps must be defined to tell the LDAP connection object which parts of the LDAP schema each piece of user profile data must go to. Also, the object classes *posixAccount* and *shadowAccount* must be defined so that these object classes are attached to each distributed user record.

#### Example 10-10 LDAP connection object attribute maps

```
C:\>wldap -g @LDAP:Linux_390
Label: @LDAP:Linux_390
host: 10.1.5.9
port: 389
sslport: 636
binddn: cn=root
password: 5HuZtEORGplU5tsj2YF
basedn: ou=Linux390,o=ibm,c=us
class: ePerson
ldapversion: 3
searchscope: one
usessl: FALSE
objectclass: top
objectclass: inetOrgPerson
objectclass: posixAccount
objectclass: shadowAccount
attributemap: city <--> 1
attributemap: comment <--> description
attributemap: department <--> ou
```

```

attributemap: employee_id <--> employeeNumber
attributemap: fax <--> facsimileTelephoneNumber
attributemap: gcos <--> gecos
attributemap: gen_qual <--> generation
attributemap: gen_qual <--> generationQualifier
attributemap: gid <--> gidnumber
attributemap: given_name <--> givenName
attributemap: hd_local_path <--> homeDirectory
attributemap: initials <--> initials
attributemap: last_name <--> sn
attributemap: login_name <--> userid
attributemap: mail_address <--> mail
attributemap: mail_address <--> textEncodedORaddress
attributemap: office <--> physicalDeliveryOfficeName
attributemap: office <--> roomNumber
attributemap: pager <--> pager
attributemap: phone <--> telephoneNumber
attributemap: real_name <--> cn
attributemap: sso_login <--> uid
attributemap: sso_password <--> userpassword
attributemap: state <--> st
attributemap: street <--> street
attributemap: title <--> title
attributemap: uid <--> uidnumber

```

---

Once the connection object has been defined, profile distributions work just like distributing to a TMA or managed node.

### 10.3.2 Access control

IBM Tivoli Access Manager for Operating Systems provides a centrally administered solution to help prevent unauthorized access to your Linux/390 systems.

Network security solutions such as firewalls and application security systems typically only focus on guarding against external, Web-based attacks. Tivoli Access Manager for Operating Systems helps you prevent security violations that can occur during everyday use of the system, especially issues that arise from the privileges of root users. By implementing access control lists (ACLs) on resources on a per-user or per-group basis, you can compartmentalize access to application and operating system resources, regardless of a user's privilege status. You can also track the login process and apply policies that improve login security, such as defining the number of permitted failed login attempts before a user is locked out.

IBM Tivoli Access Manager for Operating Systems acts as an extra layer of security over the standard permission bits provided by UNIX. It hooks into the operating system to make access control decisions on security-sensitive operations (such as opening a file, logging in, or using a TCP/IP port).

Because it hooks into system calls, you can audit what goes on and track users accessing resources. Because IBM Tivoli Access Manager for Operating Systems can secure files, we can increase the security of audit files and application logs by preventing unauthorized access. All of our controls apply to root as well as any other user. This allows us to partition the capabilities of an administrator and continue to protect critical resources to avoid accidental or deliberate damage.

IBM Tivoli Access Manager for Operating Systems provides the same capabilities, managed in the same way, across all the major UNIX platforms. This becomes more significant with

Linux because even single-vendor UNIX implementations are quite likely to add Linux systems – increasing the complexity due to the management of multiple system types.

IBM Tivoli Access Manager for Operating Systems can share the management infrastructure with the rest of the Access Manager family (although all the components are included in IBM Tivoli Access Manager for Operating Systems – there are no pre-requisites). Because IBM Tivoli Access Manager for Operating Systems used to be a part of Tivoli Security Manager, and Security Manager is now part of the provisioning piece in IBM Tivoli Identity Manager, there is very good integration between IBM Tivoli Access Manager for Operating Systems and IBM Tivoli Identity Manager. IBM Tivoli Access Manager for Operating Systems extends the provisioning in IBM Tivoli Identity Manager to become a part of the role-based access control management that IBM Tivoli Identity Manager provides.

### Typical uses

IBM Tivoli Access Manager for Operating Systems improves the security of the environment under which an application is running. This could be viewed as a form of virtual partitioning, where only authorized users can affect the operation of a protected application. IBM Tivoli Access Manager for Operating Systems authorization policy can be defined to provide application security by:

- ▶ Protecting the files themselves – determining exactly who can access them via a specific program if desired.
- ▶ Preventing root from un-mounting or otherwise destroying the underlying file system. For example, with basic database security, table access controls within a database will prevent basic access violations by logged in users, but it will not prevent the root user from deleting the database files. With IBM Tivoli Access Manager for Operating Systems, the deletion of the database files can be protected via appropriate access controls.
- ▶ Ensuring the integrity of privileged user accounts, ensuring root and others do not switch to application IDs. (Under default UNIX security, root can switch to any ID in the system without the password of that ID.)
- ▶ Extending this protection to the TCP/IP service (port) used to gain access to the application. For example, IBM Tivoli Access Manager for Operating Systems can ensure that clients can only connect from specific systems or using pre-determined TCP services.

**Note:** As of this writing, the Linux/390 version of IBM Tivoli Access Manager for Operating Systems had not been released. Check the Web site for availability information.

## 10.3.3 Firewalls

### Beyond packet filtering

Firewall systems begin with screening routers that filter packets based on type, source, and destination. What if you want to restrict the access to certain URLs to a limited number of users, but these users do not have static addresses?

IBM Tivoli Access Manager for e-business provides control over authentication and authorization of users of your Web servers and services. It does this by providing a policy enforcer which sits between the user and the server, and intercepts requests from the user. The policy enforcer routes the request to a policy manager, which maintains user account information. If the policy manager approves the request, it is forwarded to the server.

A real example of the front end, policy enforcer, back end arrangement is the way in which IBM Tivoli Access Manager for e-business can be used to protect Web resources. In this case



the front end is a Web browser, and the back end is one or more Web servers. The policy enforcer, known as WebSEAL, acts as a reverse proxy sitting between the two.

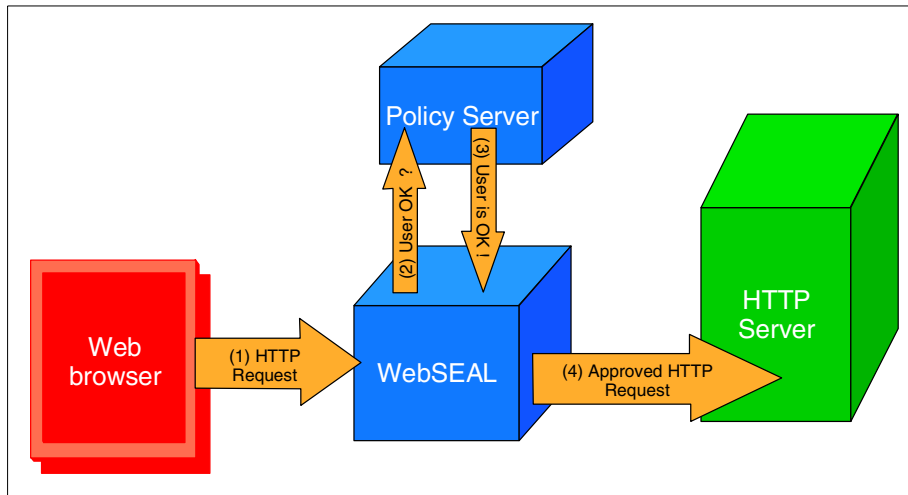


Figure 10-30 Basic WebSEAL functionality

Instead of making requests directly to the Web servers, the client is forced (using a firewall, for example) to make requests to WebSEAL (Step 1 in Figure 10-30). When WebSEAL receives a request, it first authenticates the user, then it decides whether that user is allowed to access the requested resource (Steps 2 and 3). If the user is authorized, then WebSEAL passes the original request to the backend server (Step 4) and then passes the response back to the user. If the user is not permitted, an error page is returned to the user and the request is discarded.

In the simplest case, neither the browser nor the back-end server has to be modified to achieve this functionality.

## User management

User accounts for accessing Web resources are stored in a directory. This directory can be an LDAP server, a Microsoft Active Directory, a Lotus Domino directory, or others. User accounts can be managed by the `pdadmin` command line utility on the WebSEAL server or the Policy Server, an HTTP-based console called the Web Portal Manager, or by IBM Tivoli Identity Manager.

The Web Portal Manager is a JSP application that runs on IBM Websphere Application Server. The Web Portal Manager is not yet available for Linux/390.

To manage Access Manager for e-business users with IBM Tivoli Identity Manager, you need to install the Tivoli User Administration for Policy Director product in your TMR and create a Policy Director Connection object (similar to the LDAP Connection object) on a managed node within the Identity Manager's TMR. You then need to define the Access Manager (Policy Director) user information in your user profiles. Then simply distribute the user profile to the Policy Director Connection object. Since Identity Manager can manage users of multiple systems in addition to Access Manager for e-business, there is a distinct advantage in using it over the other options.

See the *Tivoli Secureway User Administration Supplement for Policy Director* for more information.

When a user tries to access a URL protected by Access Manager for e-business, the login prompt looks like Figure 10-31.



Figure 10-31 Access Manager for e-business login prompt

A successful login is rewarded with access to the protected URL, a failed login receives a message like Figure 10-32.

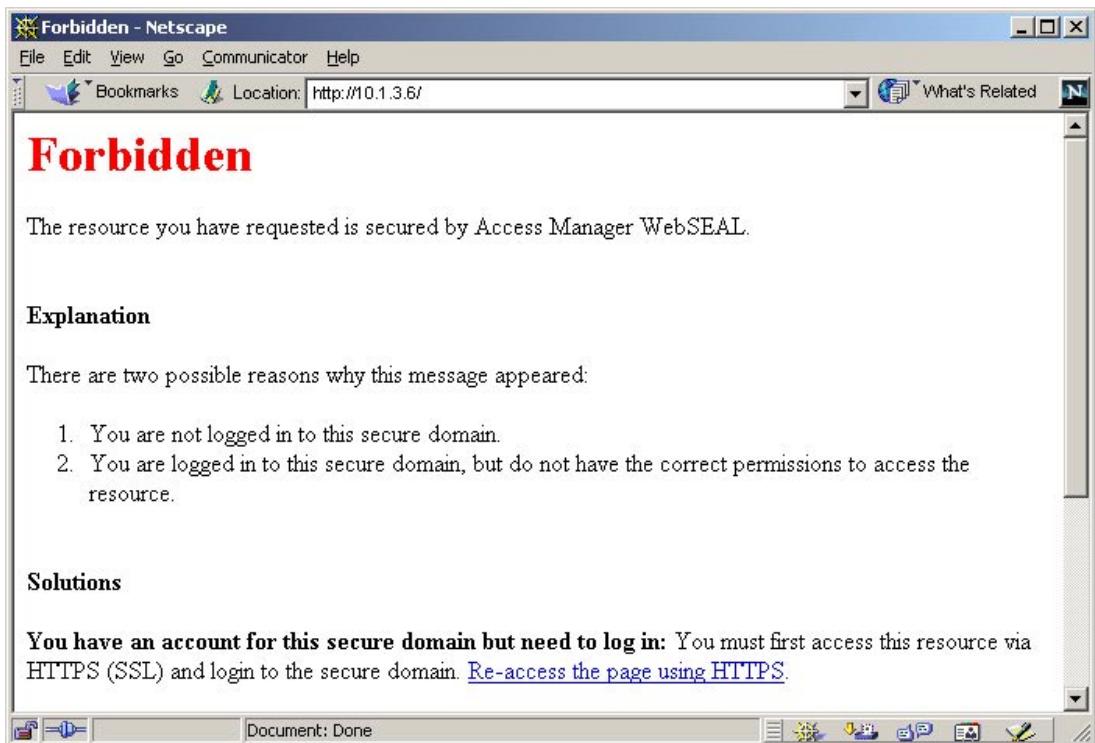


Figure 10-32 Access Manager for e-business denied access message

## Web server management

Management of information can be simplified if the company organizes corporate Web resources in a logical Web namespace, in which content is accessed through an Internet address, or URL, that reflects a logical structure chosen by the organization. This allows information to be organized logically, such as by department or on a project basis, instead of by the physical location of the resource.

To create a logical Web space, WebSEAL is positioned in front of an existing Web server and its corporate Web resource tree. WebSEAL associates a user-defined logical name (as part of the logical URL) to refer to the Web server content. When a user requests a resource (using the logical URL), the server intercepts the request and uses Smart Junctions to match the logical name with the physical address of the Web server. In effect, WebSEAL translates the logical URL, locates the information, and returns it to the user, who remains unaware of the physical location of the information.

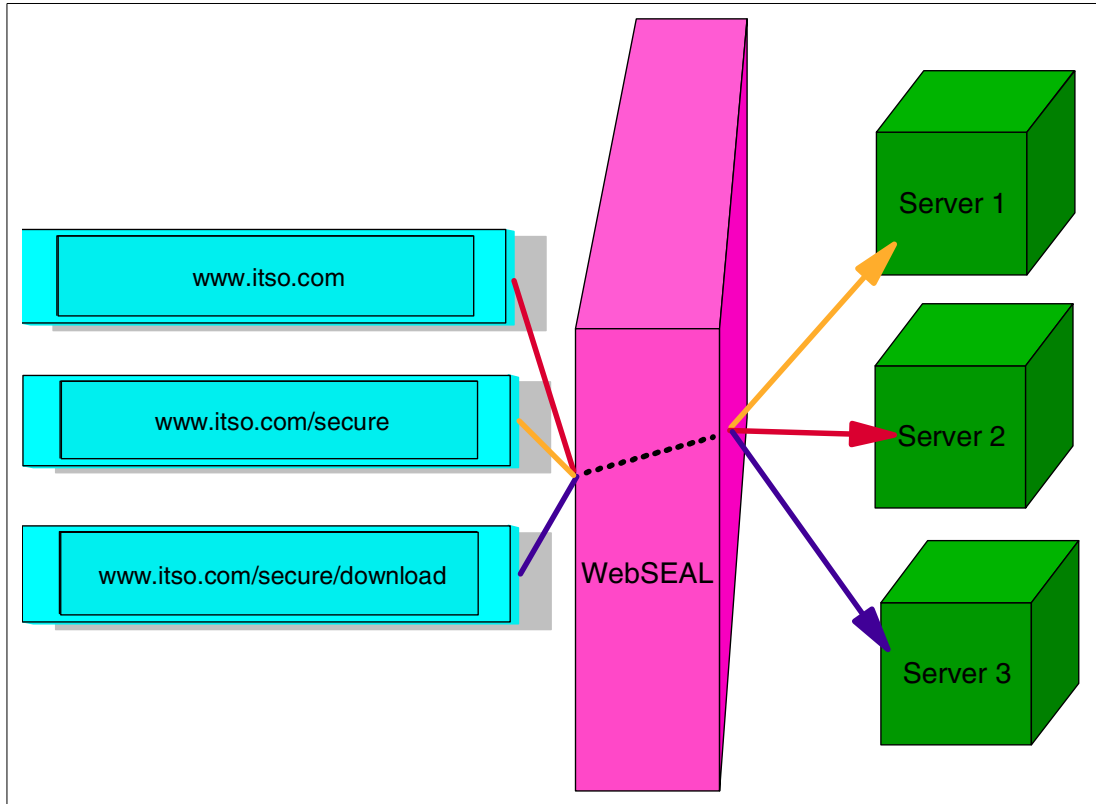


Figure 10-33 Smart Junctions combining several servers into a single logical name space

This structure allows access policy to be set at WebSEAL, rather than individually at the physical servers on which the information resides. Smart Junctions enable support for any back-end Web server.

In addition to transparently supporting any Web server, the Access Manager for e-business logical Web space can also include resources accessed by Web-enabled applications, such as PeopleSoft 7.5 and SAP. This means Access Manager for e-business controls access to information accessed from legacy databases and other back-end applications in exactly the same manner as static Web resources.

The logical addressing scheme also makes it easier to make changes to the network. If information must be moved between servers, or a new server added, the Web administrator can make the change and then adjust Smart Junctions. Users never know a change took place – unless they realize it as greater speed and efficiency.

The use of a logical Web namespace also simplifies security administration since WebSEAL can set access policy against the logical Web space instead of at each server.

Smart Junctions can be used to mount multiple Web servers with replicated contents at the same point in the logical Web space. When this is done, WebSEAL performs intelligent load balancing across the replicated servers for improved performance and fault recovery. This allows security administration of Web resources to be available at all times, even in the event of system maintenance or failure. Using Smart Junctions, Web server capacity can be added in a linear fashion as demand increases on the corporate Web infrastructure.

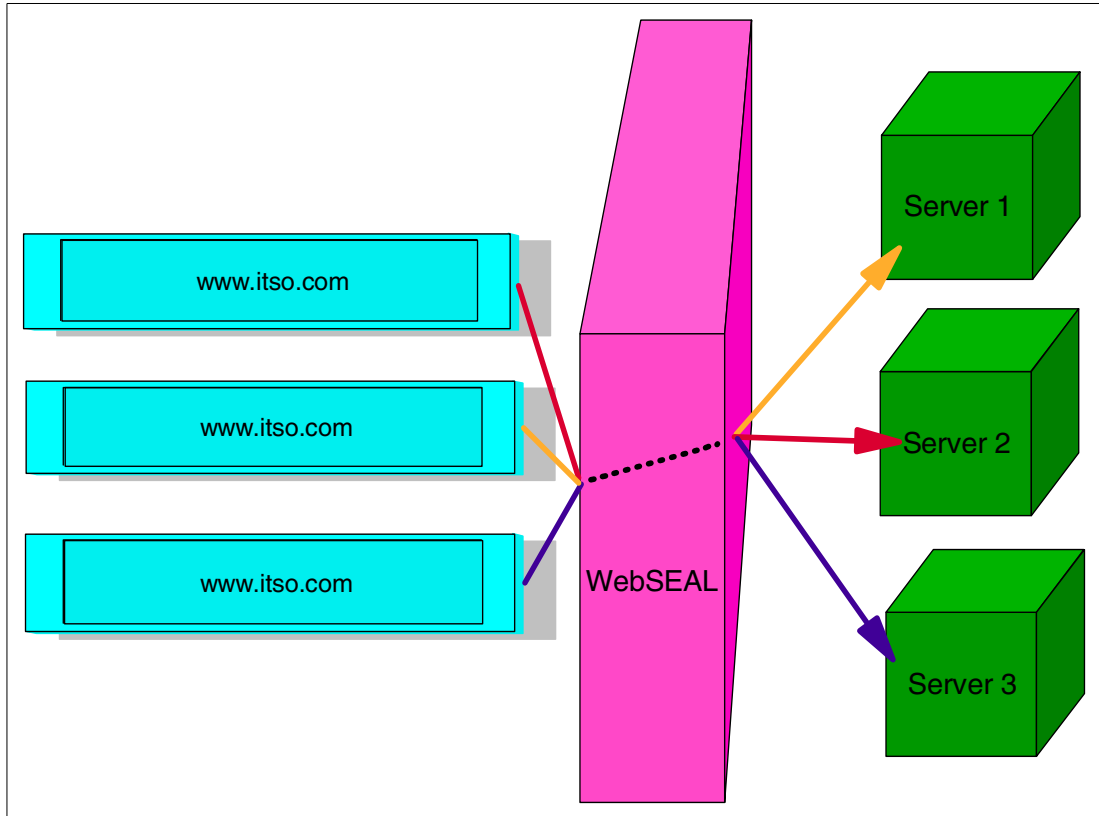


Figure 10-34 Smart Junctions mapping several servers to the same point for load balancing

All Access Manager for e-business services can be similarly replicated, providing high availability and fail-over.

### 10.3.4 Managing for audit

Log files are used by security administrators to detect break-ins before they occur and to assess the damage done after a break-in. Unsuccessful attacks against an HTTP server appear in the HTTP server logs, and failed login attempts appear in system logs. Since successful attacks can give the attacker the ability to edit these log files, it is a wise practice to log the events of one system into a file on a second. The attacker then needs to compromise two systems to hide his activities.

When you have a z/Series server with dozens of Linux VMs, you can create a complicated network of remote logging, or centralize the collection of data. The IBM Tivoli Event Console (T/EC) provides not only an event collection service, but event correlation and response as well.

**Note:** Event correlation is a process by which you can eliminate unwanted messages by comparing events to predefined rules. If you do not receive a heartbeat message from a file server, it may appear that this file server is down. But if you also receive a message that the router separating the Event Console server and the file server is down, you can ignore the fact that the file server is down, since it will always appear that it is down when the router is not functioning.

Also, duplicate events can be summarized in a single entry. This entry will include a note that the message has been repeated a given number of times. This capability of T/EC to streamline information presented to a security administrator makes it more valuable than standard log files for auditing.

Automated responses to T/EC events can range from attempting to restart a service that has crashed, to altering firewall rules, to paging a human. Anything that can be scripted or programmed can be made an automatic response to a T/EC event.

### The T/EC adapter

The primary source for T/EC events is the T/EC adapter. The T/EC adapter is a program that runs on a system and has the capability to send events to T/EC. The one that is most useful in this section is the logfile adapter. The logfile adapter is a program that intercepts system log events before they are written to files, and decides whether they are forwarded to the T/EC server based on the administrator's configuration.

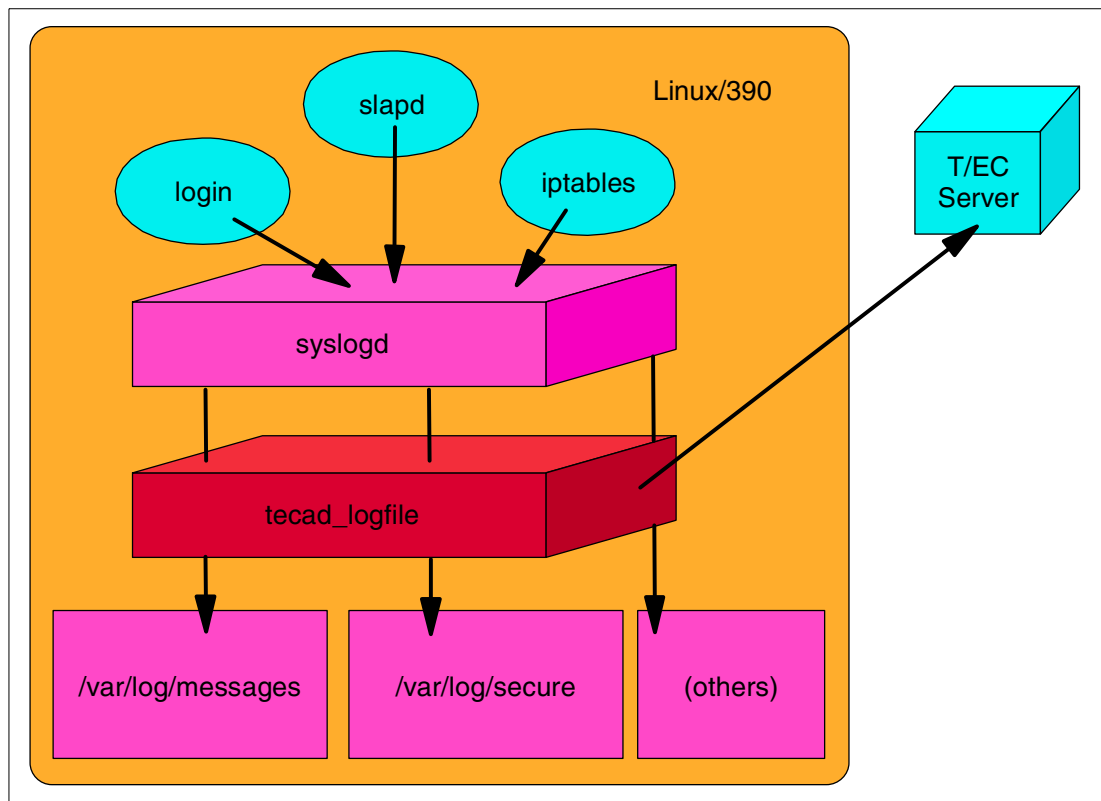


Figure 10-35 T/EC logfile adapter intercepts system events and sends them to the T/EC server

**Note:** Events can come from other sources besides the T/EC adapter. Tivoli Monitoring (Tivoli Distributed Monitoring) can generate T/EC events, as can any Tivoli-ready certified application.

To configure the adapter to send only selected events, you only need to modify the `tecad_logfile.conf` file. In Example 10-11, you see a configuration file which has been modified to send only *su failure* events to the T/EC server. The `FilterMode` is set to `IN`, which means that no events are sent to the T/EC server unless they are defined in this configuration file. The only events that are defined are the ones meeting the definition:

```
Filter:Class=Logfile_Base;msg=re:'su: F.*'
```

This filter allows events of the class `Logfile_Base` to pass through if they meet the criteria that follows the semicolon. In this example, we have directed our filter to look at the `msg` slot, and only allow events to be forwarded that meet the following regular expression (re):

```
su: F.*
```

To find a match for succeeding `su` events, simply change the filter to look for `msg` slots that contain `'su: S.*'`

**Important:** Although the *su failure* event is part of the `Su_Failure` class, the logfile adapter sends this event to the T/EC server as a member of the `Logfile_Base` class, which is a parent of the `Su_Failure` class. The filter must be defined using the class that the adapter uses to form the event sent to the T/EC server, even if this is not a leaf event class.

#### Example 10-11 `tecad_logfile.conf`

---

```
# TE/C LogFile adapter configuration file, for Linux9.
#
# (C) COPYRIGHT TIVOLI Systems, Inc. 1999.
# Unpublished Work
# All Rights Reserved
# Licensed Material - Property of TIVOLI Systems, Inc.
#
#TestMode=YES
ServerLocation=tot29
BufEvtPath=/etc/Tivoli/tec/logfile.cache
ServerPort=5529
EventMaxSize=4096
PollInterval=30

FilterMode=IN
Filter:Class=Logfile_Base;msg=re:'su: F.*'
#Filter:Class=Logfile_Sendmail
#Filter:Class=Amd_Unmounted
#Filter:Class=Amd_Mounted
```

---

If you wanted to send in everything *except* `su` failure events, you would simply change the `FilterMode` parameter to `OUT`.

### Troubleshooting the logfile adapter

If events from your logfile adapter are not appearing in your T/EC console, consider the following items:

- ▶ If you set the `TestMode` parameter to `YES` in your `tecad_logfile.conf` file, events are written to a file named `$TECADHOME/etc/$HOSTNAME`.

- ▶ In non-TestMode=YES mode (TestMode=NO), events are written to /etc/Tivoli/tec/logfile.cache if the server is not available. Once the server can be reached by the logfile adapter, it sends the events that have been stored in logfile.cache.
- ▶ If the T/EC server is Windows NT, then you must specify ServerPort=5529 in your tecad\_logfile.conf file.
- ▶ Running **wtdumpr1** on the T/EC server will show which events have reached the T/EC server's reception log.
- ▶ To make sure the T/EC server is functioning properly, use **wpostemsg** (on the T/EC server):

```
wpostemsg TEC_DB TEC
```

or **postemsg** (on the Linux/390 VM):

```
postemsg -f /tmp/tecad_logfile.conf TEC_DB TEC
```

with a TEC\_DB event instead of a Logfile\_Base event, where /tmp/tecad\_logfile.conf contains:

```
ServerPort=5529
ServerLocation=tecserver.domain.net
```

## Removing the logfile adapter

If you need to remove the Linux/390 logfile adapter for any reason, complete the following:

- ▶ If you installed the logfile adapter by distributing an ACP, delete all the entries in the ACP so that it is empty, and then distribute the empty profile to the Linux/390 TMA.
- ▶ If you installed the logfile adapter by running the **tecad\_logfile.cfg** non-TME adapter install script, you can run the **tecad\_remove\_logfile.sh** script.





# Part 4



# BMC

The following BMC products are described in this part:

- ▶ BMC Mainview
- ▶ BMC Patrol





## BMC products

This chapter describes several products from BMC Software. We explain the purpose of each product, as well as how to install, configure, and customize each one. The products covered are:

- ▶ MAINVIEW for Linux Servers
- ▶ PATROL for Linux Enterprise Server
- ▶ PATROL Internet Server Manager

## 11.1 MAINVIEW for Linux Servers

MAINVIEW for Linux Servers is a system management application that provides services and functions to help you monitor and control your Linux servers. Built on the BMC Software MAINVIEW architecture, MAINVIEW for Linux Servers uses a traditional MAINVIEW interface to provide access to Linux data.

MAINVIEW for Linux Servers provides monitoring and management of Linux for S/390 systems. MAINVIEW for Linux Servers can:

- ▶ Display Linux performance and availability information in real time.
- ▶ Provide over 60 different views that display Linux performance, availability, usage, and configuration information.
- ▶ Integrate views from the MAINVIEW console of S/390 subsystems and Linux applications.
- ▶ Monitor system and process resource usage.
- ▶ Alert support personnel of exception conditions through alarm management.
- ▶ Discover Linux systems automatically.
- ▶ Provide the ability to view network statistics.
- ▶ Let you customize the environment to provide logical or performance-scaled division of Linux images.
- ▶ Let you choose between several user interfaces (3270 MAINVIEW or the browser-based MAINVIEW Explorer).

### 11.1.1 Configuration/layout

The installation of MAINVIEW for Linux Servers in our lab is on zSeries hardware, OS/390 release 2.9.00, host named BBSYSC. The Linux components of the product are installed on the same physical hardware in a SuSE 2.2.16 image named LNXSUSE. There are also several other SuSE Linux 2.2.16 images running – named WEBxx, DEVxx, MAILxx, and SECxx – which will be used for purposes of explaining customization and product usage. All images are running in a VM 4.3 system.

The OS/390 components in this installation consist of five started tasks. They are the communication server (LNxzRTS), communication server companion task (RTL0CT), MAINVIEW Coordinating Address Space (LNxzCAS), MAINVIEW Product Address Space (LNxzPAS), and the MAINVIEW Explorer address space (LNxzMXP).

With the exception of the RTL0CT task, all task names can be changed. The RTL0CT task is required by the communication server and is started automatically when the communication server is started.

There are two processes that run on the Linux image: a data server with process name mmlsrv and a data collector with process name bgscollect.

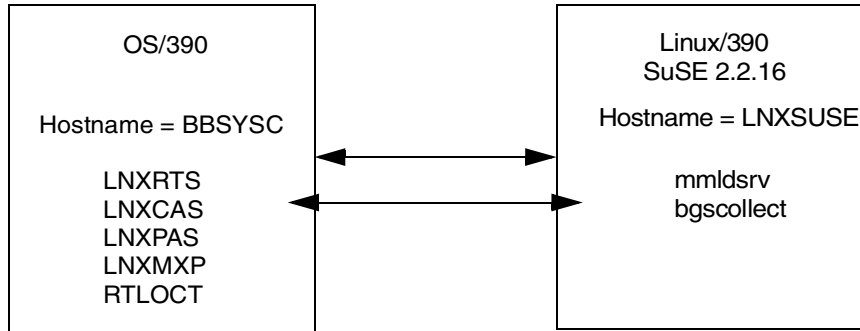


Figure 11-1 OS/390 and Linux/390 components

## 11.1.2 Installation

MAINVIEW products can be installed in either a full SMP/E format or in a non-SMP/E format. The non-SMP/E format is referred to as the “Standard Install”; it uses IEBCOPY to unload the products, and also provides an SMP/E environment for applying maintenance. We chose the full SMP/E Install format for our installation. Installation of the MAINVIEW product was accomplished following the procedures outlined in the *OS/390 and z/OS Installer Guide*, BMC part number 100040949.

Installation is started by unloading two data sets from tape as described in the *OS/390 and z/OS Installer Guide*. The bulk of the installation steps are executed via TSO CLIST. Not every panel is included here since each is covered in the installation manual. The panels described in this section are of particular interest to our installation.

The Install Repository/Profile Options panel is used to create and modify the installation repository and profile information. This information can be used for restarting an incomplete installation and can also be used to re-install products with saved values.

For Installation System User Options, we have the choice of selecting either the Basic or Advanced form of installation. For a new installation such as ours, we selected the Basic option. The Advanced option would be necessary if an installation wanted to migrate data from a previous installation, use user-defined VSAM data sets, change BMC product object attributes or modify default option values.

Many products can be available to install from BMC tapes. Since we are only interested in the MAINVIEW for Linux Servers product, we selected MAINVIEW Infrastructure and MAINVIEW for Linux Servers from the Install System Product Selection panel.

Continuing with the installation, we verified that the installation tape VOLSER's job card information, selected to create new SMP/E zones in separate CSIs, and then accepted the default file allocation for the installation libraries.

We generated the jobs necessary to install the product, and after successful completion of these jobs, we were ready to begin product customization, which is selected from the CLIST main panel. Our list of products contained the following:

- ▶ MAINVIEW Alarm
- ▶ MAINVIEW for Linux Servers
- ▶ MAINVIEW Infrastructure

We did not specifically select to install the MAINVIEW Alarm Manager product, but it is included when installing most MAINVIEW products. Customization of the MAINVIEW

Infrastructure and MAINVIEW Alarm Manager are not covered in this publication. Refer to the *MAINVIEW Common Customization Guide*, BMC part number 100042266, for assistance with customizing these products. In our experience, customizing the products was not difficult and was accomplished with little reference to the manual.

### 11.1.3 Customization

Customization of the MAINVIEW infrastructure was done using the *MAINVIEW Common Customization Guide*. For customization assistance for the MAINVIEW for Linux Servers product, see the *MAINVIEW for Linux Servers Customization Guide 1.1*, BMC part number 100041294.

There are twenty steps involved in customizing the MAINVIEW for Linux Servers product. Most of these steps are self-explanatory. The steps of particular interest are discussed in this section.

Several steps on the following screens have a plus (+) or minus sign (-) in the **Status** field, indicating those steps have been completed or bypassed. We completed Step 1 through Step 4 following the instructions given in those steps.

On-line product authorization can be done either on line or in batch. In Step 5, we selected **O** for online and entered `mm1` for the MAINVIEW for Linux Servers product code.

```
GLOBAL STEP TABLE ----- PRODUCT AUTHORIZATION ----- CUSTOMIZATION
COMMAND ==>

You are installing one or more BMC products that require a Product
Authorization password. If you have not already specified passwords
during the installation process, you should do so now. This dialog will
assist you in creating a password table for each product or updating an
existing password table. If all products are to be authorized using BBKEYS,
enter "KEY". Entering "KEY" when there are no products that use keys for
authorization bypasses this step.

Do you want to authorize a product now? ==> YES (YES, NO OR KEY)

    * YES continues the dialog to authorize a product.
    * NO bypasses this step.

If you reply YES, select either the Online Authorization dialog, or the
Batch Authorization dialog: ==> O (O or B)

Enter the three character Product code: ==> mm1 _

Verify existence of Password data set: ==> YES (YES or NO)

To continue, press ENTER
```

Figure 11-2 MAINVIEW for Linux Servers customization: Product authorization step

```
                                BMC Product Authorization Primary Menu
COMMAND ===> _____
Select an option. Type additional information if applicable. Then press Enter.

Options

1 1. Process password (Requires password library and password)
   2. Display product authorization (Requires password library only)
   3. Display current processor information
   4. Help about...
   5. Exit

Additional information

Password library . . . . . 'ROHLXW.OZIZOS.D0409.BMCPSWD'

Authorization password . . jyp tkp uh9 308 _
```

Figure 11-3 BMC Product Authorization Primary Menu

We selected option 1 (shown in Figure 11-3) and entered the password provided in the BMC tape cover letter. The CPU serial number and model number are required and are entered in the BMC Product Authorization Primary Menu.

After successfully completing Step 5, we continued through Step 10 following the instructions given in each step. Step 11 creates the CAS startup procedure.

We changed the default name of the CAS subsystem ID from BBCS to LNXZ; we changed the procedure name from BBMCAS to LNXZCAS. Steps 12 and 13 were completed using the instructions provided for each of those steps. We saved that procedure and continued with Step 14, creating the RTSERVER startup procedure.

The name for our TCPIP address is TCPIP, so we did not change that. If the TCPIP address space had been different, it would be specified here. The field for RTHOME.HLQ is limited to two qualifiers. For this installation we chose ROHLXW.RTHOME, which resulted in a data set name of ROHLXW.RTHOME.STANDARD.CM being allocated. It should be noted that another data set is created using this HLQ; its name is ROHLXW.RTHOME.STANDARD.TXT. These data sets are used by the RTSERVER address space.

We also changed the name of the RTSERVER procedure to LNXZRTS for our installation.

Continuing with Step 15, we created the RTLOCT procedure. The name of this procedure must not be changed since it is hard coded within the RTSERVER code.

In Step 16, we created the SYSIN member with the parameters needed by the Product Address Space (PAS) in the Create SYSIN Member panel (shown in Figure 11-4).

```
MAINVIEW FOR LINUX ----- CREATE SYSIN MEMBER ----- CUSTOMIZATION
COMMAND ==>

Specify a UBBPARM data set member name for the MAINVIEW for Linux - Servers
PAS start-up (SYSIN) parameters.
Specify the MAINVIEW for Linux - Servers PAS sub-system ID.
Specify the network host name and port number of the Communication Server.

SYSIN MEMBER NAME ==> MMLPRM00 (1-8 character name, default=MMLPRM00)
PAS SUBSYSTEM ID ==> MML0 (1-4 character name, default=MML0)
COMM. SERVER HOST (up to 60 characters)
==> BBSYSC
COMM. SERVER PORT ==> 5101 (numeric value, default=5101)

Press ENTER to continue, HELP for more information, or END for previous panel.
```

Figure 11-4 Creating the SYSIN member

The member name MMLPRM00 was not changed. If this name is changed, it must also be reflected in the PAS procedure environment variable, MMLSYSIN. We also did not change the PAS subsystem ID, opting for the default of MML0. COMM. SERVER HOST requires the name of the host system where the RTSERVER procedure will run. Our RTSERVER procedure name is LNXZRTS; it runs on host BBSYSC. A TCPIP port is required for communication between the PAS, RTSERVER, and the Linux image data server. The default port number of 5101 was selected.

Figure 11-5 shows the values for the parameters that will be saved in member MMLPRM00.



```

EDIT ----- ROHLXW.OZIZOS.D0409.IMAGSJSC.UBBPARAM(MMLPRM00)----- COLUMNS 000 000
COMMAND ==> _ SCROLL ==> PAGE

AutoCUSTOMIZATION of MMLPRM00 is complete.
Make further modification if necessary.

Enter END to save member and continue.
Enter CANCEL to return to the previous panel without saving the member.
-----
032512 *
032612 * Change Log:
032712 *
032812 * Created by ROHLXW on 02/05/07 at 09:27
032912 *
033012 *-----
033112 SUBSYSID=MML0, PAS subsystem name (default=MML0)
033212 SUBSTR=(1,8,B), How to form image names from host names
033316 COMMHOST=BBSYSC,
033412 COMMPORT=5101, Communication Server port number
033512 SSID=LNxz, CAS subsystem id to connect to
033615 LOGLEVEL=400, Communication sub-task logging level
033715 MXCOMMWT=1000, Communication sub-task max wait (ms)
034012 END End configuration parameters

```

Figure 11-5 Custom values of MMLPRM00 to be saved

Steps 17 and 18 were completed according to their provided instructions. In Step 19, creating the PAS procedure, we changed the name of the procedure from MMLPAS to LNXPAS.

In Step 20 we built an installation script and a batch job, used to FTP files from the installed libraries to the Linux images. Figure 11-6 illustrates the creation of a Linux user account called **bmcuser** as our *run user as* account. (The data server and collector programs have the setuid bit “on,” which will allow the programs to access files normally available only to the *root* user.)

```

MAINVIEW FOR LINUX ----- CREATE JOB TO FTP RPMS FILES ----- CUSTOMIZATION
COMMAND ==>

Specify the Linux user name which the MAINVIEW for Linux - Servers
Data Server should run as.
Specify whether or not to start the Data Server after installation
on the Linux system is completed.

DATA SERVER RUNUSER ==> bmcuser (1-20 characters)
START DATA SERVER? ==> n (Y or N)

When you press ENTER, the customization dialog creates a modified version of
sample member MMLDINST using the options you provided. It also places you in
an edit session to confirm the values and make further changes, if necessary.

Press ENTER to continue, HELP for more information, or END for previous panel.

```

Figure 11-6 Creating a job to FTP the RPM files

In Figure 11-7, parameters are displayed that will be used by the installation script MMLDINST for installation of the data server. This will link the data server communications to the RTSERVER running on host BBSYSC.

```

EDIT ----- ROHLXW.OZIZOS.D0409.UBBSAMP(MMLDINST)----- COLUMNS 000 000
COMMAND ==> _ SCROLL ==> CSR

                AutoCUSTOMIZATION of MMLDINST is complete.
                Make further modification if necessary.

Enter END to save member and continue.
Enter CANCEL to return to the previous panel without saving the member.
-----
000003 # Configuration parameters for MAINVIEW for Linux PAS
000004 #
000005 #-----
000006 #
000007 # Change Log:
000008 #
000009 #         Created by ROHLXW on 02/05/07 at 09:39
000010 #
000011 #-----
000012 RTSERVHOST=BBSYSC
000013 RTSERVPOR=5101
000014 DATASRVUSER=bmcuser
000015 INSTSTART=n
000016 #

```

Figure 11-7 Custom values of MMLDINST to be saved

The FTP job is created with the parameters in Figure 11-8.

```

MAINVIEW FOR LINUX ----- CREATE JOB TO FTP RPMS FILES ----- MMLDINST SAVED
COMMAND ==>

Specify the network host name, username, and password for the Linux system
to which you want to FTP the MAINVIEW for Linux - Servers install package.

REMOTE LINUX SERVER NAME (up to 60 characters)
==> LNXSUSE
USER ==> bmcuser (1-20 characters)
PASSWD ==> _ (1-20 characters)

Specify the library name and member name where the FTP job JCL is to be
saved.

TARGET LIBRARY ==> 'ROHLXW.OZIZOS.D0409.UBBSAMP'
MEMBER ==> MMLFTPJB (Member name for FTP job)
REPLACE MEMBER? ==> NO (If duplicate member in TARGET LIBRARY)

When you press ENTER, the FTP job is created and you are placed in an edit
session to make further changes, if necessary.

```

Figure 11-8 Creating a job to FTP the RPM files

The host name of the target Linux image in our case is LNXSUSE. We entered our Linux userid, *bmcuser*, along with the password, which is masked. Figure 11-9 shows the FTP JCL,

which can be modified if necessary. Note that the password is no longer masked. After completion of the FTP execution, this should be removed for security reasons.

```

File Edit Confirm Menu Utilities Compilers Test Help
EDIT      ROHLXW.OZIZOS.D0409.UBBSAMP(MMLFTPJB) - 01.00  Columns 00001 00072
Command ==> Scroll ==> CSR
000016 //STEP1 EXEC PGM=FTP,PARM=(EXIT',REGION=2048K
000017 //SYSPRINT DD SYSOUT=*
000018 //OUTPUT DD SYSOUT=*
000019 //INPUT DD *
000020 LNXSUSE
000021 bmcuser
000022 passwd
000023 TYPE A
000024 cd /tmp
000025 LCD 'ROHLXW.OZIZOS.D0409.UBBSAMP'
000026 PUT MMLDINST mmlldinst
000027 SITE chmod 755 mmlldinst
000028 TYPE I
000029 LCD 'ROHLXW.OZIZOS.D0409.RPMS'
000030 PUT MMLUD390 mmlud390
000031 PUT MMLUD386 mmlud386
000032 SITE chmod 644 mmlud390
000033 SITE chmod 644 mmlud386
000034 QUIT

```

Figure 11-9 Custom FTP job parameters to be saved

After running the FTP job, we logged into our Linux host, LNXSUSE, as root. The FTP job dropped the installation script and the RPM package in the /tmp directory. Notice that the FTP job puts both an S/390(MMLUD390) and an Intel(MMLUD386) version of the product in /tmp. The installation script will determine which package to use. To install the RPM, issue the following command:

```
MMLDINST -V
```

At this time, we started the mainframe started tasks described earlier in this chapter. These components were started in the following order: LNXZRTS, LNXZCAS, LNXZPAS, LNXZMXP and LNXZALM. The only requirement for starting these tasks is that the CAS should be running before the PAS is started. All other tasks can be started in any order. Once these are up and running we moved back to the Linux system.

During the installation process we opted not to have the data server start automatically after installation or on subsequent reboots of the system. We issued the `mmldsrv start` command, followed a few seconds later by a `mmldsrv stat` command to display the processes. Two processes, `mmldsrv` and `bgscollect`, should be running. Log files are created in `/var/BMCS/mml` and `/var/BMCS/mml/bgs/log` for the data server and collector respectively.

With all required tasks running, we brought up a MAINVIEW console by executing `MAINVIEW CLIST` and selecting our LNXZ CAS id. The initial screen displayed is that of the first Linux image to connect to our PAS. The view name is EZLNX. Hyperlink on **.Image Status** to see all Linux images connected to our PAS.

Scrolling to the right produces the display shown in Figure 11-10.

```

14MAY2002 13:34:16 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> I ALT WIN ==>
+w1 =SYLOVERZ=====DEVO=====*=====14MAY2002==13:32:35====MVLNX====D====13
Image %Swap I/O-rate PageRate Matched Host Name
----- 0.....50...100 ----- Rule -----
DEV0 0.00 1.85 $DEFAULT Dev0
DEV1 0.00 1.85 $DEFAULT Dev1
DEV2 0.00 1.85 $DEFAULT Dev2
LNXSUSE 0.00 1.85 $DEFAULT LNXSUSE
MAIL0 0.00 1.85 $DEFAULT MAIL0
MAIL1 0.00 1.85 $DEFAULT MAIL1
MAIL2 0.00 1.85 $DEFAULT MAIL2
SEC0 0.00 1.85 $DEFAULT SEC0
SEC1 0.00 1.85 $DEFAULT SEC1
SEC2 0.00 1.85 $DEFAULT SEC2
WEB0 0.00 1.85 $DEFAULT WEB0
WEB1 0.00 1.85 $DEFAULT WEB1
WEB2 0.00 1.85 $DEFAULT WEB2

```

Figure 11-10 SYLOVERZ view

What we are after here is the *Matched Rule* field. All Linux instances are using the \$DEFAULT rule definition, which controls data sampling intervals, time-outs, and heartbeat data. Since not all Linux images will necessarily have the same type of workload, a user might want to change some of the values for certain systems. To do that, we create a new rule.

In this example, we went back to the EZLNX view for image DEV0 and clicked the **.Add/Edit Monitor Rules** field under **Utilities**. This displays the ADLRULE view (shown in Figure 11-11).

```

w1 =ADLRULE=====DEVO=====*(00 BROWSE )====MVLNX====D====1
CMD Image Monitor HeartBeat Reply Backgrnd Demand Min Demand
--- Pattern (Y/N) Interval TimeOut Samp Int Samp Int Mode Dur
$DEFAULT Yes 60 20 60 10 300

```

Figure 11-11 ADLRULE view in Browse mode

The status line shows 00 BROWSE. We typed EDIT on the command line to change the status to 00 EDIT. We typed add in the CMD field next to \$DEFAULT and overtyped \$DEFAULT with WEB\*, leaving the other parameter values unchanged, and press Enter.

```

W1 =ADLRULE=====DEVO=====*(00 EDIT )====MVLNX====D====1
CMD Image      Monitor HeartBeat  Reply Backgrnd  Demand Min Demand
--- Pattern    (Y/N)      Interval TimeOut Samp Int Samp Int  Mode Dur
add WEB*       Yes          60      20      60      10      300

```

Figure 11-12 ADRULE view in Edit mode

The result of this command is shown in Figure 11-13.

```

15MAY2002 09:48:40 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==> SCROLL ==> PAGE
CURR WIN ==> I ALT WIN ==>
W1 =ADLRULE=====DEVO=====*(00 EDIT MOD )====MVLNX====D====2
CMD Image      Monitor HeartBeat  Reply Backgrnd  Demand Min Demand
--- Pattern    (Y/N)      Interval TimeOut Samp Int Samp Int  Mode Dur
$DEFAULT Yes          60      20      60      10      300
WEB*       Yes          60      20      60      10      300

```

Figure 11-13 Result of the Add command in ADRULE view

Returning to our Image Status screen, view SYLOVERZ, once a sampling interval has passed we see that our rule for Linux Images that start with hostname WEB\* now have a Matched Rule name of WEB\*. We can customize the values in that rule and they will be applied to those systems.

```

14MAY2002 13:39:35 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==> SCROLL ==> PAGE
CURR WIN ==> I ALT WIN ==>
+w1 =SYLOVERZ=====DEVO=====14MAY2002==13:39:34====MVLNX====D====1
Image      %Swap      I/O-rate PageRate Matched Host Name
----- 0.....50...100 ----- Rule -----
DEV0      |          | 0.00 0.00 $DEFAULT Dev0
DEV1      |          | 0.00 0.00 $DEFAULT Dev1
DEV2      |          | 0.00 0.00 $DEFAULT Dev2
LNXSUSE   |          | 0.00 0.00 $DEFAULT LNXSUSE
MAIL0     |          | 0.00 0.00 $DEFAULT MAIL0
MAIL1     |          | 0.00 0.00 $DEFAULT MAIL1
MAIL2     |          | 0.00 0.00 $DEFAULT MAIL2
SEC0      |          | 0.00 0.00 $DEFAULT SEC0
SEC1      |          | 0.00 0.00 $DEFAULT SEC1
SEC2      |          | 0.00 0.00 $DEFAULT SEC2
WEB0      |          | 0.00 0.00 WEB*   WEB0
WEB1      |          | 0.00 0.00 WEB*   WEB1
WEB2      |          | 0.00 0.00 WEB*   WEB2

```

Figure 11-14 SYLOVERZ view

## 11.2 PATROL for Linux Enterprise Server

PATROL for Linux Enterprise Server ensures server performance and availability across the enterprise for the Linux OS. It increases availability by reducing unplanned downtime and by setting thresholds for memory, CPU usage, disks, file systems, logs, processes, security, and swap file utilization. The user can establish recovery actions for threshold attainment, anticipating problems before the Linux OS can be impacted.

The key benefits of PATROL for Linux Enterprise Server include its ability to:

- ▶ Provide monitoring and management for Red Hat and SuSE Linux distributions
- ▶ Monitor key system health indicators and take action before a problem impacts the system workload
- ▶ Provide automated event notification to the IT staff
- ▶ Automatically discover and display a graphic presentation of the Linux operating system components
- ▶ Offer real-time graphic views of the operating system and related components
- ▶ Provide proactive, configurable monitoring
- ▶ Reduce system administration workload
- ▶ Provide history retention for reporting and data analysis
- ▶ Take corrective action during a system failure or problem

PATROL for Linux Enterprise Server provides these benefits by:

- ▶ Monitoring size and content of user-specified log files
- ▶ Allowing alert conditions to be defined based on logical combinations of parameters and constants
- ▶ Providing easy-to-read, customized information through console views, info boxes, reports, charts, and graphs
- ▶ Providing an operator-oriented console and advanced event-driven management functions
- ▶ Monitoring system-wide CPU usage
- ▶ For file system resources, determining which disks are heavily used or running out of space
- ▶ For log files, monitoring the size and content of specified log files
- ▶ For memory activity, monitoring paging, I/O caching, and swapping
- ▶ For network activity, monitoring TCP/IP traffic levels
- ▶ Monitoring the print queue and activity
- ▶ Tracking network traffic related to remote procedure calls and Network File System (NFS) activity

**Note:** The Linux kernel on zSeries does not support all the commands or provide all of the statistics that the PATROL KM for UNIX requires. Specifically, on zSeries:

- `nfsstat` is not available.
- The kernel returns empty `proc` entries for disk and swap; thus, the `DISK` and `SWAP` parameters in PATROL display values of zero.

Using the PATROL Classic Console (shown in Figure 11-15), you can click the Linux OS icons and drill down to access detailed information, menu commands, and reports.

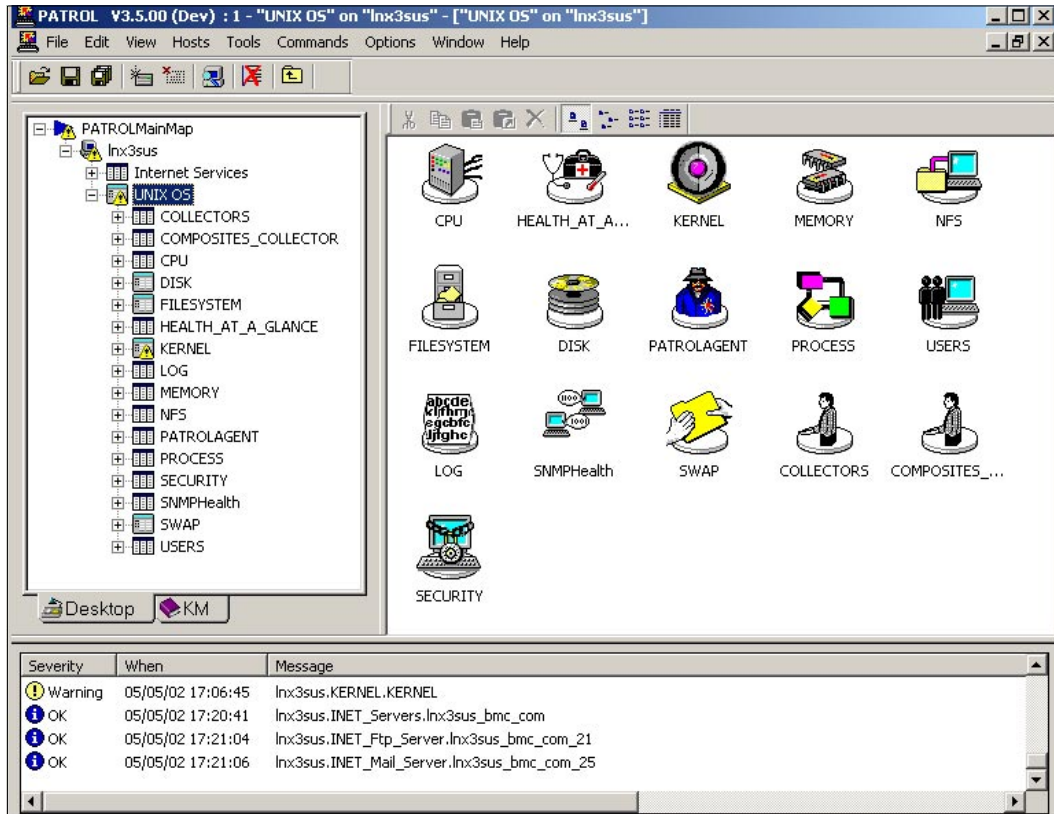


Figure 11-15 PATROL Classic Console: PATROL for Linux Enterprise Server

### 11.2.1 Configuration/layout

To monitor and manage SuSE Linux Enterprise Server 7 for S/390 and zSeries using the PATROL for Linux Enterprise Server product, we must first determine the role of the system.

For the system that we will use to manage Linux system resources, the PATROL Agent for Linux Enterprise Server version 3.5.00 and the PATROL Knowledge Module for UNIX version 8.3.06 (agent components) are installed on a SuSE Linux Enterprise Server 7 for S/390 and zSeries image (hostname agent390, Kernel 2.2.16) on a z/VM 4.3 system.

For the system that we will use to view data collected by the PATROL Agent, the PATROL Console for Microsoft Windows version 3.5 and the PATROL Knowledge Module for UNIX version 8.3.06 (console components) are installed on a PC running Microsoft Windows Advanced Server with SP2.

### 11.2.2 Installation of PATROL for Linux Enterprise Manager

We used the PATROL for Linux Enterprise Server Kit (KIT-1033) to install PATROL products on a SuSE Linux system (managed system). This kit contains the PATROL for Linux Enterprise Server v1.1.00 product CD, which was used to install the following PATROL components on our SuSE Linux system:

- ▶ PATROL Agent for Linux Enterprise Server v3.5.00
- ▶ PATROL Knowledge Module for UNIX v8.3.06 (agent components only)

Also included in this kit is the PATROL for Linux Enterprise Server Documentation CD. The documentation CD contains additional information about the products that comprise the kit.

We used the PATROL Solutions for UNIX Kit (KIT-1031) to install PATROL products on a Windows 2000 Advanced Server system (console system). This kit contains the PATROL Solutions for UNIX Product CD and the PATROL Consoles and Migration Tools CD. The PATROL Solutions for UNIX Product CD contains the PATROL Knowledge Module for UNIX v8.3.06 product. The PATROL Consoles and Migration Tools CD contains the PATROL Console for Microsoft Windows v3.5.00 product. Also included in this kit is the PATROL Solutions for UNIX Documentation CD. The documentation CD contains additional information about the products that comprise the kit.

Before you begin installing PATROL, you must ensure that the accounts that you want to use for PATROL meet the following conditions.

### **SuSE Linux installation account**

BMC Software recommendations for the Linux account you create for installing PATROL products on your SuSE Linux Enterprise Server 7 system are as follows:

- ▶ The account .login, .profile, .cshrc, and .kshrc files should contain as little user customization as possible. Specifically, there should be no aliases, the prompt should be set to the default, and there should be no commands in these files to change the umask setting. The recommended umask setting for the installation account is 022.
- ▶ Do not use root to install PATROL products because this may create security risks.
- ▶ Be sure the account has permission to create directories in the directory where you will install PATROL products. The computers on which you want to install PATROL must have ftp and telnet enabled.
- ▶ PATROL configuration requires privileges usually reserved by the system administrator. These privileges include access to a root account on the hardware where you want to install PATROL.
- ▶ Install PATROL on local partitions, not on NFS-mounted partitions. If you do install PATROL on NFS-mounted partitions, the root account must have been granted root access privileges on the NFS server.
- ▶ The account that you use to install PATROL must have permission to write the installation logs to the \$HOME and /tmp directories on the computer where you are installing products.

The installation account (patqa1) that we used for the installation on the SuSE Linux Enterprise Server 7 for S/390 and zSeries system meets these requirements.

### **Windows 2000 Installation Account**

The account that you use for installing PATROL products on your Windows 2000 system must be a member of the Administrators group. The account can be either a local account or a domain account. During the installation, PATROL will assign the following user rights to the account you specify:

- ▶ Act as part of operating system
- ▶ Debug programs
- ▶ Increase quotas
- ▶ Log on as a service
- ▶ Log on locally
- ▶ Profile system performance
- ▶ Replace a process level token



The installation account (patrol) that we used for the installation on the Windows 2000 Advanced Server system meets these requirements.

## Installing PATROL for Linux Enterprise Server on Linux

Use the following steps to install PATROL for Linux Enterprise Server v1.1.00 on your SuSE Linux Enterprise Server 7 for S/390 and zSeries system.

1. Mount the PATROL for Linux Enterprise Server v1.1.00 CD and type `./setup.sh` this is the root of the CD,once this is done the Welcome to the Installation Utility dialog will be displayed.

**Note:** If you do not have a browser installed on your SuSE Linux system, a message is displayed telling you that a Web server was started and to use a browser on another system. Start the browser and point to the following URL:

```
http://<IP Address of SuSE Linux System>:50001
```

2. Click **Next** to continue. The Review License Agreement dialog is displayed.
3. Select **Accept** to accept the license agreement, then click **Next** to continue.

**Note:** If you do not accept the license agreement, the installation program will exit when you click **Next**.

The Select Type of Installation dialog is displayed.

4. Select the **Typical** installation path and click **Next** to continue. The Specify Installation Directory dialog is displayed.
5. Specify the base installation directory to use for the PATROL for Linux Enterprise Server components. We used the home directory for user patqa1 (/export/home/patqa1/PATROL3500\_82688). On UNIX systems, the default installation directory is /opt/BMC. Be sure you have write permissions for this directory before you install or the installation will fail.

Once you have specified a directory, click **Next** to continue. The Select System Roles dialog is displayed.

6. Select **Managed System** and click **Next** to continue. We selected a Managed System because we want PATROL to manage our SuSE Linux system's resources. The Select Products and Components to Install dialog is displayed.

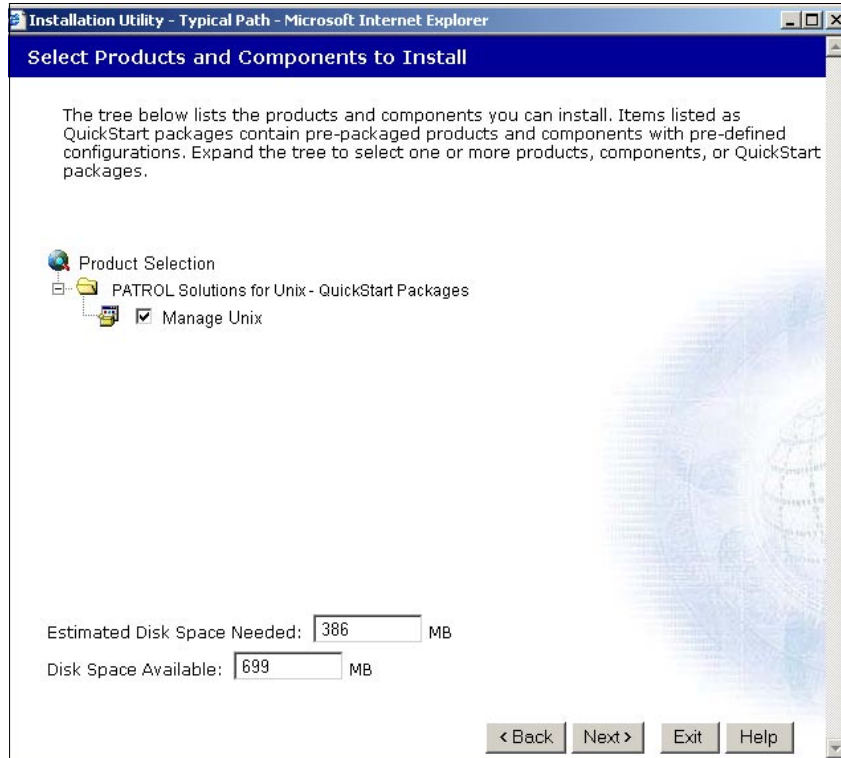


Figure 11-16 Installation Utility: Select Products and Components to Install dialog

7. Expand **PATROL Solutions for UNIX - QuickStart Packages**, select the **Manage UNIX** package, and then click **Next** to continue. The System Root Account Properties dialog is displayed.
8. Enter the Root Login Name and the Root Login Password in the appropriate fields. Click **Next** to continue. The PATROL Default Account Properties dialog is displayed.
9. Enter the PATROL default account name and Password in the appropriate fields. Click **Next** to continue. The Information for the PATROL Agent dialog is displayed.
10. Select **Start the PATROL Agent automatically**, then click **Next** to continue. The Review Selections and Install dialog is displayed.

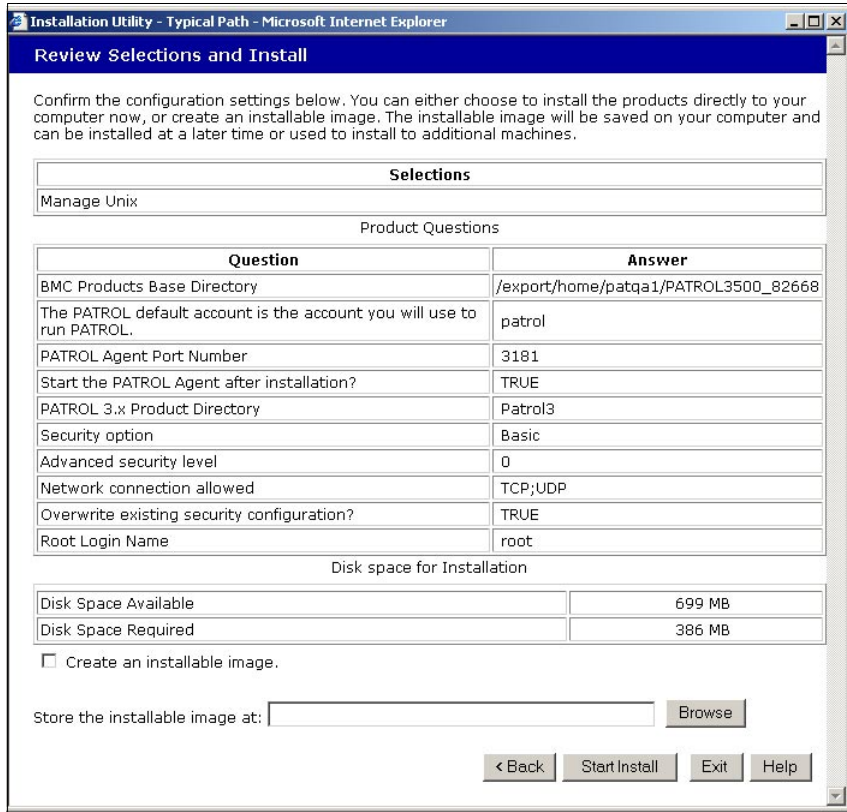


Figure 11-17 Installation Utility: Review Selections and Install Dialog

The Review Selections and Install dialog shows the PATROL solution we are installing, the configuration settings that we specified, and the approximate amount of disk space required by the install.

11. Select **Start Install** to begin the installation of the PATROL for Linux Enterprise Server on your SuSE Linux system. The Installation Status dialog is displayed. This dialog shows the progress of the installation.
12. When the installation is 100% complete, click **Next** to continue. The Success dialog is displayed. The Success dialog shows which PATROL products and components were installed and the location of the installation log file.
13. To view the installation log, click **View Log**. The installation log is displayed in a scrollable window that allows you to review the messages that were generated by the PATROL Installation Utility during the installation of PATROL for Linux Enterprise Server on your SuSE Linux system.

When you have finished reviewing the installation log, close the window. You will be returned to the Success dialog.

14. Click **Finish**. You have completed installing PATROL for Linux Enterprise Server v1.1.00 on your SuSE system. Now you need to install a PATROL Console so you can view the data collected by the PATROL Agent on your system.

## Installing PATROL Console for Microsoft Windows

To install the PATROL Console for Microsoft Windows v3.5.00 on a Microsoft Windows 2000 Advanced Server system, perform the following:

1. Insert the PATROL Console and Migration Tools CD into the CD-ROM drive on your machine. To start the PATROL Installation Utility, do one of the following:

- Use the Windows Explorer to open the CD directory and double-click **setup.exe**. The Welcome to the Installation Utility dialog is displayed.
  - Open a DOS command window and change to the <CD-ROM\_drive>:\setup.exe directory. Type setup.exe at the prompt, then press Enter. The Welcome to the Installation Utility dialog is displayed.
2. Click **Next** to continue. The Review License Agreement dialog is displayed.
  3. Select **Accept** to accept the license agreement, then click **Next** to continue.

**Note:** If you do not accept the license agreement, the installation program will exit when you click **Next**.

4. Select the **Typical** installation path and click **Next** to continue. The Specify Installation Directory dialog is displayed.
5. We used the default installation directory (C:\Program Files\BMC Software) to install PATROL components on our Windows 2000 system. If you want to install PATROL in a different directory, type the directory path in the BMC Products Installation Directory field. Click **Next** to continue. The Select System Roles dialog is displayed.
6. Select **Console System** and click **Next** to continue. We selected a Console System because we want to use this system to view data collected by the PATROL Agent on our SuSE Linux system. The Select Products and Components to Install dialog is displayed.

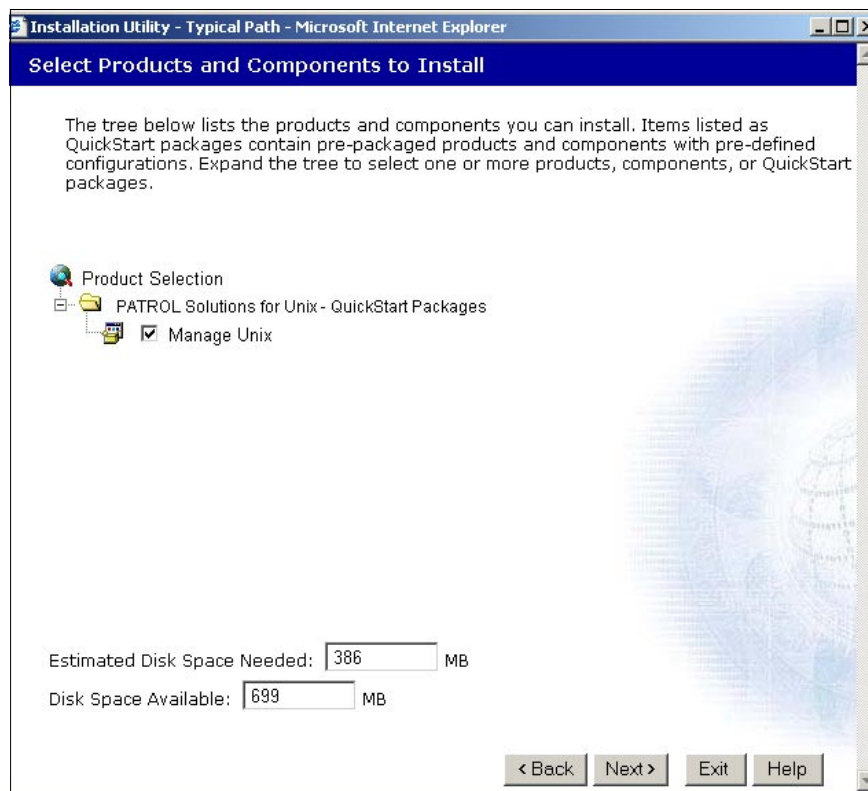


Figure 11-18 Installation Utility: Select Products and Components to Install Dialog

7. Expand **PATROL Classic Consoles (PATROL 3)**, select **PATROL Console for Microsoft Windows**, and then click **Next** to continue. The PATROL Default Account Properties dialog is displayed.

8. Enter the PATROL default account name and Password in the appropriate fields. Click **Next** to continue. The Information for the PATROL Agent dialog is displayed.
9. Select **Start the PATROL Agent automatically**, then click **Next** to continue. The Review Selections and Install dialog is displayed.

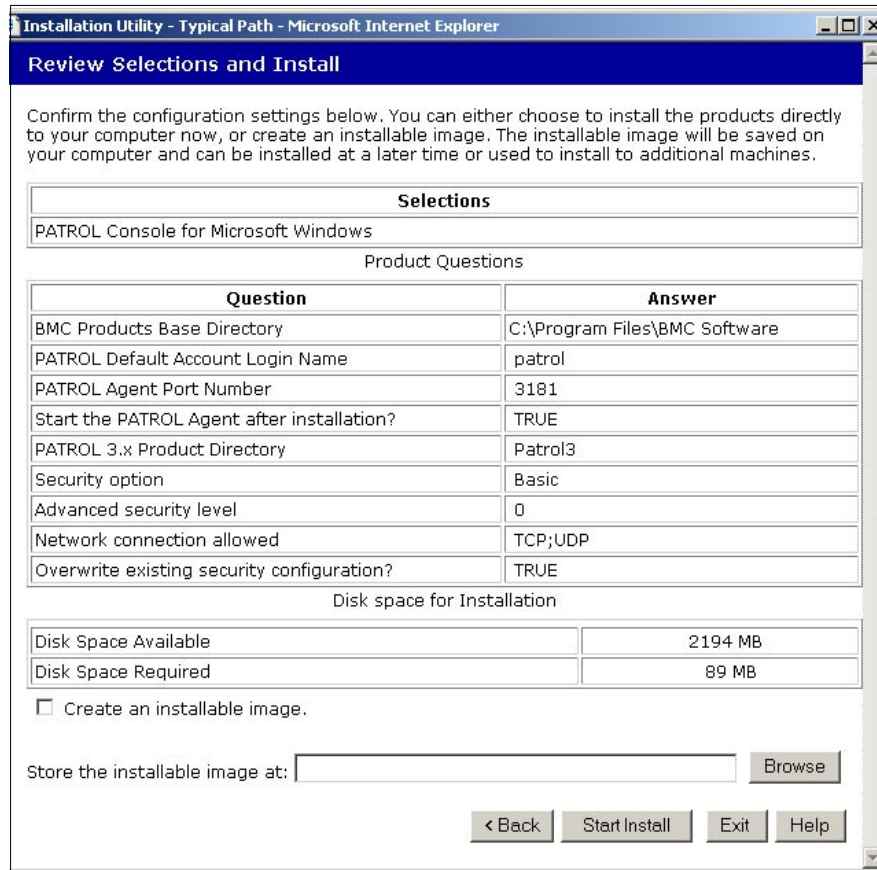


Figure 11-19 Installation Utility: Review Selections and Install dialog

The Review Selections and Install dialog shows the PATROL solution you are installing, the configuration settings that you specified, and the approximate amount of disk space required by the install.

10. Select **Start Install** to begin the installation of the PATROL Classic Console for Microsoft Windows product on your Windows 2000 system. The Installation Status dialog is displayed. This dialog shows the progress of the installation.
11. When the installation is 100% complete, click **Next** to continue. The Success dialog is displayed. The Success dialog shows which PATROL products and components were installed and the location of the installation log file.
12. To view the installation log, click **View Log**. The installation log is displayed in a scrollable window that allows you to review the messages that were generated by the PATROL Installation Utility during the installation of the PATROL Knowledge Module for UNIX on your Windows 2000 system.

When you have finished reviewing the installation log, close the window. You will be returned to the Success dialog.

13. Click **Finish**. You have completed the installation of the PATROL Classic Console for Microsoft Windows on your Windows 2000 system. Now you can install the PATROL Knowledge Module for UNIX (console components) on your Windows 2000 system.

## Installing PATROL Knowledge Module for UNIX on Windows

Use the following steps to install the PATROL Knowledge Module for UNIX v8.3.06 (console components only) on a Microsoft Windows 2000 Advanced Server system.

1. Insert the PATROL Solutions for UNIX Product CD into the CD-ROM drive on your machine. To start the PATROL Installation Utility, do one of the following:
  - Use the Windows Explorer to open the CD directory and double-click **setup.exe**. The Welcome to the Installation Utility dialog is displayed.
  - Open a DOS command window and change to the <CD-ROM\_drive>:\setup.exe directory. Type setup.exe at the prompt, then press Enter. The Welcome to the Installation Utility dialog is displayed.
2. Click **Next** to continue. The Review License Agreement dialog is displayed.
3. Select **Accept** to accept the license agreement, then click **Next** to continue.

**Note:** If you do not accept the license agreement, the installation program will exit when you click **Next**.

The Select Type of Installation dialog is displayed.

4. Select the **Typical** installation path and click **Next** to continue. The Specify Installation Directory dialog is displayed.
5. We used the default installation directory (C:\Program Files\BMC Software) to install PATROL components on our Windows 2000 system. If you want to install PATROL in a different directory, type the directory path in the BMC Products Installation Directory field. Click **Next** to continue. The Select System Roles dialog is displayed.
6. Select **Console System** and click **Next** to continue. We selected a Console System because we want to use this system to view data collected by the PATROL Agent on our SuSE Linux system. The Select Products and Components to Install dialog is displayed.
7. Expand **PATROL Solutions for UNIX -> QuickStart Packages**, select the **Manage UNIX** package, and then click **Next** to continue. The Review Selections and Install dialog is displayed.

The Review Selections and Install dialog shows the PATROL solution you are installing, the configuration settings that you specified, and the approximate amount of disk space required by the install.

8. Select **Start Install** to begin the installation of the PATROL Knowledge Module for UNIX on your Windows 2000 system. The Installation Status dialog is displayed. This dialog shows the progress of the installation.
9. When the installation is 100% complete, click **Next** to continue. The Success dialog is displayed. The Success dialog shows which PATROL products and components were installed and the location of the installation log file.
10. To view the installation log, click **View Log**. The installation log is displayed in a scrollable window that allows you to review the messages that were generated by the PATROL Installation Utility during the installation of the PATROL Knowledge Module for UNIX on your Windows 2000 system.

When you have finished reviewing the installation log, close the window. You will be returned to the Success dialog.

11. Click **Finish**. You have completed the installation of the PATROL Knowledge Module for UNIX console components on your Windows 2000 system. Now you can view the data that the PATROL Agent has been collecting on the SuSE Linux system.

### 11.2.3 Customization

No customization is required to use the PATROL Agent for Linux Enterprise Server version 3.5.00 and the PATROL Knowledge Module for UNIX version 8.3.06 (agent components) that were installed on a SuSE Linux Enterprise Server 7 for S/390 and zSeries image (hostname agent390, Kernel 2.2.16).

No customization is required to use the PATROL Console for Microsoft Windows version 3.5 and the PATROL Knowledge Module for UNIX version 8.3.06 (console components) that were installed on the PC running Microsoft Windows Advanced Server with SP2.

## 11.3 PATROL Internet Server Manager

PATROL Internet Server Manager monitors and manages Internet servers, including servers supporting Secure Socket Layer (SSL) encryption, across a distributed environment from a centralized PATROL console (using a PATROL console is optional). PATROL Internet Server Manager monitors the availability, performance, and integrity of your Internet servers.

The parameters for PATROL Internet Server Manager enable you to analyze Internet server performance quickly and easily, providing a detailed statement of all system activity over time. You can clearly identify peaks, troughs, and trends in the performance of server resources. By enabling you to detect problems, optimize the system, analyze trends, plan capacity, and manage multiple hosts simultaneously, PATROL Internet Server Manager helps to ensure your Internet servers run efficiently 24 hours a day. The key benefits of include the following:

The key benefits of PATROL Internet Server Manager include its ability to:

- ▶ Ensure the quality of Web hosting and e-commerce.
- ▶ Enhance the availability, reliability, and performance of Web applications.
- ▶ Allow you to verify service level agreements with performance reports.
- ▶ Fully integrate with the PATROL family of products.
- ▶ Solve common problems, such as restarting downed Web servers, through automatic corrective actions.

PATROL Internet Server Manager produces these benefits by providing:

- ▶ A broad range of Internet server support
- ▶ Web page validation
- ▶ Server automatic restart and recovery
- ▶ Access and error log monitoring
- ▶ Monitoring of SSL certificates
- ▶ Customized server discovery and monitoring
- ▶ Advanced monitoring capabilities for UNIX servers
- ▶ Monitoring of specific DNS queries
- ▶ Definable server availability alarms and recovery actions
- ▶ Monitoring of dynamic Web pages
- ▶ Integration with PATROL Web Performance Reporter

### 11.3.1 Broad Range of Server Support

The following types of Internet servers can be managed on single, multiple, or virtual hosted systems:

- ▶ HTTP and HTTPS Web servers (for example, Apache, Netscape, iPlanet, or Tomcat)
- ▶ DNS servers and clients
- ▶ FTP servers
- ▶ LDAP servers
- ▶ Mail (SMTP, IMAP, and POP3) servers
- ▶ NNTP news servers
- ▶ HTTP proxy servers
- ▶ IRC chat servers

While PATROL Internet Server Manager can monitor all SMTP mail servers and DNS servers for availability and response time, for UNIX systems it can gather additional important usage data when used to monitor a Sendmail mail server or a BIND DNS server. You can use this additional data to identify and resolve performance problems.

Using the PATROL Classic Console (shown in Figure 11-20), you can click the Internet Server Manager icons and drill down to access detailed information, menu commands, and reports.

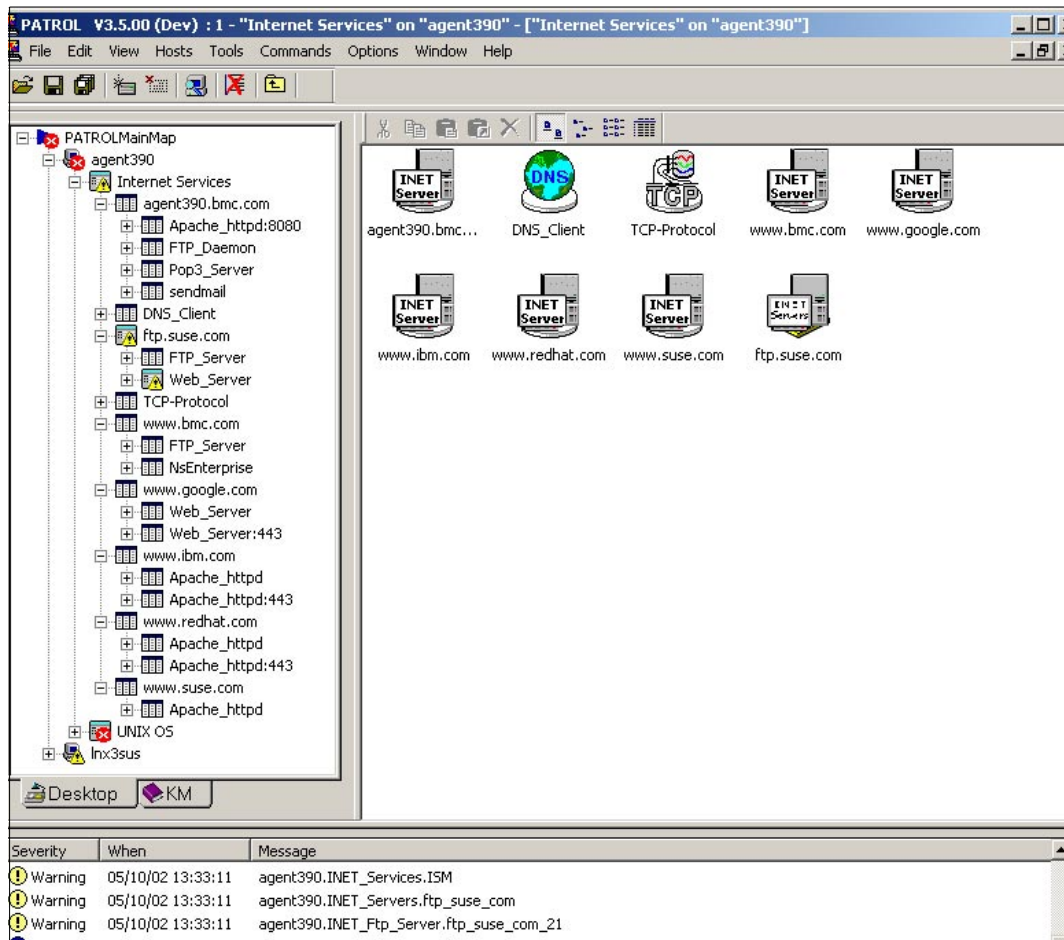


Figure 11-20 PATROL Classic Console: PATROL Internet Server Manager



## 11.3.2 Configuration/layout

To monitor and manage SuSE Linux Enterprise Server 7 for S/390 and zSeries using the PATROL Internet Server Manager product, we must first determine the role of the system.

For the system that we will use to manage Internet server resources, the PATROL Agent for Internet Server Manager version 5.3.00 and the PATROL Knowledge Module for UNIX version 8.3.06 (agent components) are installed into a SuSE Linux Enterprise Server 7 for S/390 and zSeries image (hostname agent390, Kernel 2.2.16) on a z/VM 4.3 system.

For the system that we will use to view data collected by the PATROL Agent, the PATROL Console for Microsoft Windows version 3.5 and the PATROL Knowledge Module for UNIX version 8.3.06 (console components) are installed on a PC running Microsoft Windows Advanced Server with SP2.

## 11.3.3 Installation

We used the PATROL Internet Server Manager Kit (KIT-1032) to install PATROL products on a SuSE Linux system (managed system). This kit contains the PATROL Internet Server Manager v5.3.00 product CD, which we used to install PATROL Internet Server Manager v5.3.00 on a SuSE Linux system. Also included in this kit is the PATROL Internet Server Manager Documentation CD. The documentation CD contains additional information about the products that comprise the kit.

We used the PATROL Solutions for UNIX Kit (KIT-1031) to install PATROL products on a Windows 2000 Advanced Server system (console system). This kit contains the PATROL Solutions for UNIX Product CD and the PATROL Consoles and Migration Tools CD. The PATROL Solutions for UNIX Product CD contains the PATROL Knowledge Module for UNIX v8.3.06 product. The PATROL Consoles and Migration Tools CD contains the PATROL Console for Microsoft Windows v3.5.00 product. Also included in this kit is the PATROL Solutions for UNIX Documentation CD. The documentation CD contains additional information about the products that comprise the kit.

Before you begin installing PATROL, you must ensure that the accounts that you want to use for PATROL meet the following conditions.

### SuSE Linux Installation Account

Following are the BMC Software recommendations for the Linux account that you create for installing PATROL products on your SuSE Linux Enterprise Server 7 for S/390 and zSeries system.

- ▶ The account `.login`, `.profile`, `.cshrc`, and `.kshrc` files should contain as little user customization as possible. Specifically, there should be no aliases, the prompt should be set to the default, and there should be no command in these files to change the `umask` setting. The recommended `umask` setting for the installation account is `022`.
- ▶ Do not use `root` to install PATROL products because this may create security risks.
- ▶ Be sure the account has permission to create directories in the directory where you will install PATROL products. The computers on which you want to install PATROL must have `ftp` and `telnet` enabled.
- ▶ PATROL configuration requires privileges usually reserved by the system administrator. These privileges include access to a `root` account on the hardware where you want to install PATROL.

- ▶ BMC Software recommends that you install PATROL on local partitions, not on NFS-mounted partitions. If you do install PATROL on NFS-mounted partitions, the root account must have been granted root access privileges on the NFS server.
- ▶ The account that you use to install PATROL must have permission to write the installation logs to the \$HOME and /tmp directories on the computer where you are installing products.

The installation account (patqa1) that was used for the installation on the SuSE Linux Enterprise Server 7 for S/390 and zSeries system meets these requirements.

### Windows 2000 Installation Account

The account that you use for installing PATROL products on your Windows 2000 system must be a member of the Administrators group. The account can be either a local account or a domain account. During the installation, PATROL will assign the following user rights to the account you specify:

- ▶ Act as part of operating system
- ▶ Debug programs
- ▶ Increase quotas
- ▶ Log on as a service
- ▶ Log on locally
- ▶ Profile system performance
- ▶ Replace a process level token

### Installing PATROL Internet Server Manager on Linux

Use the following steps to install PATROL Internet Server Manager v5.3.00 on your SuSE Linux Enterprise Server 7 for S/390 and zSeries system.

1. Mount the PATROL Internet Server Manager v5.3.00 CD and type `./setup.sh` at the root of the CD. The Welcome to the Installation Utility dialog is displayed.

**Note:** If you do not have a browser installed on your SuSE Linux system, a message is displayed telling you that a Web server was started and to use a browser on another system. Start the browser and connect to the following URL:

```
http://<IP Address of SuSE Linux System>:50001
```

2. Click **Next** to continue. The Review License Agreement dialog is displayed.
3. Select **Accept** to accept the license agreement, then click **Next** to continue.

**Note:** If you do not accept the license agreement, the installation program will exit when you click **Next**.

The Select Type of Installation dialog is displayed.

4. Select the **Typical** installation path and click **Next** to continue. The Specify Installation Directory dialog is displayed.
5. Specify the base installation directory to use for the PATROL for Linux Enterprise Server components. We used the home directory for user patqa1 (/export/home/patqa1/PATROL3500\_82688). On UNIX systems, the default installation directory is /opt/BMC. Be sure you have write permissions for this directory before you install or the installation will fail.

Once you have specified a directory, click **Next** to continue. The Select System Roles dialog is displayed.

6. Select **Managed System** and click **Next** to continue. We selected a **Managed System** because we want PATROL to manage our SuSE Linux system's resources. The Select Products and Components to Install dialog is displayed.

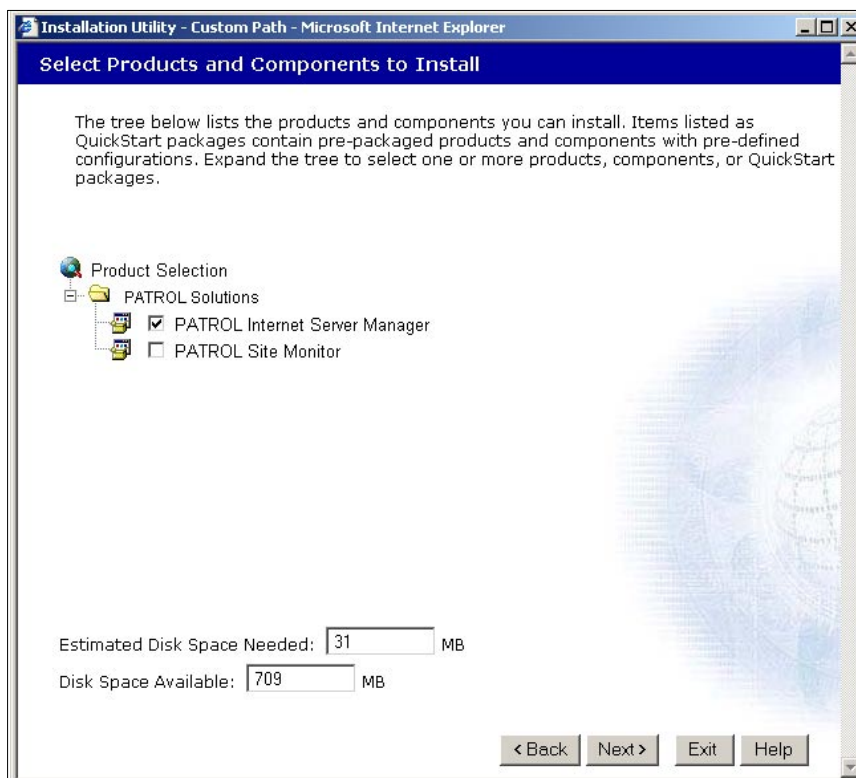


Figure 11-21 Installation Utility: Select Products and Components to Install Dialog

7. Expand **PATROL Solutions**, select the **PATROL Internet Server Manager** package, and then click **Next** to continue. The Identify a Site Monitor Host dialog is displayed.
8. We do not want to configure a Site Monitor Host. Click **Next** to continue. The Information for the PATROL Agent dialog is displayed.
9. Select **Start the PATROL Agent automatically**, then click **Next** to continue. The Review Selections and Install dialog is displayed.

The Review Selections and Install dialog shows the PATROL solution we are installing, the configuration settings that we specified, and the approximate amount of disk space required by the install.

10. Select **Start Install** to begin the installation of the PATROL Internet Server Manager on your SuSE Linux system. The Installation Status dialog is displayed. This dialog shows the progress of the installation.
11. When the installation is 100% complete, click **Next** to continue. The Success dialog is displayed. The Success dialog shows which PATROL products and components were installed and the location of the installation log file.
12. To view the installation log, click **View Log**. The installation log is displayed in a scrollable window that allows you to review the messages that were generated by the PATROL Installation Utility during the installation of PATROL Internet Server Manager on your SuSE Linux system.

When you have finished reviewing the installation log, close the window. You will be returned to the Success dialog.

13. Click **Finish**. You have completed installing PATROL Internet Server Manager v5.3.00 on your SuSE system. Now you need to install a PATROL Console so you can view the data collected by the PATROL Agent on your system.

## Installing PATROL Console for Microsoft Windows

Use the following steps to install the PATROL Console for Microsoft Windows v3.5.00 on a Microsoft Windows 2000 Advanced Server system.

1. Insert the PATROL Console and Migration Tools CD into the CD-ROM drive on your machine. To start the PATROL Installation Utility, perform one of the following:
  - Use the Windows Explorer to open the CD directory and double-click on **setup.exe**. The Welcome to the Installation Utility dialog is displayed.
  - Open a DOS command window and change to the <CD-ROM\_drive>:\setup.exe directory. Type setup.exe at the prompt, then press Enter. The Welcome to the Installation Utility dialog is displayed.
2. Click **Next** to continue. The Review License Agreement dialog is displayed.
3. Select **Accept** to accept the license agreement, then click **Next** to continue.

**Note:** If you do not accept the license agreement, the installation program will exit when you click **Next**.

The Select Type of Installation dialog is displayed.

4. Select the **Typical** installation path and click **Next** to continue. The Specify Installation Directory dialog is displayed.
5. We used the default installation directory (C:\Program Files\BMC Software) to install PATROL components on our Windows 2000 system. If you want to install PATROL in a different directory, type the directory path in the BMC Products Installation Directory field. Click **Next** to continue. The Select System Roles dialog is displayed.
6. Select **Console System** and click **Next** to continue. We selected a Console System because we want to use this system to view data collected by the PATROL Agent on our SuSE Linux system. The Select Products and Components to Install dialog is displayed.
7. Expand **PATROL Classic Consoles** (PATROL 3), select **PATROL Console for Microsoft Windows**, and then click **Next** to continue. The PATROL Default Account Properties dialog is displayed.
8. Enter the PATROL default account name and Password in the appropriate fields. Click **Next** to continue. The Information for the PATROL Agent dialog is displayed.
9. Select **Start the PATROL Agent automatically**, then click **Next** to continue. The Review Selections and Install dialog is displayed.

The Review Selections and Install dialog shows the PATROL solution you are installing, the configuration settings that you specified, and the approximate amount of disk space required by the install.
10. Select **Start Install** to begin the installation of the PATROL Classic Console for Microsoft Windows product on your Windows 2000 system. The Installation Status dialog is displayed. This dialog shows the progress of the installation.

11. When the installation is 100% complete, click **Next** to continue. The Success dialog is displayed. The Success dialog shows which PATROL products and components were installed and the location of the installation log file.
12. To view the installation log, click **View Log**. The installation log is displayed in a scrollable window that allows you to review the messages that were generated by the PATROL Installation Utility during the installation of the PATROL Knowledge Module for UNIX on your Windows 2000 system.  
  
When you have finished reviewing the installation log, close the window. You will be returned to the Success dialog.
13. Click **Finish**. You have completed the installation of the PATROL Classic Console for Microsoft Windows on your Windows 2000 system. Now you can install the PATROL Internet Server Manager (console components) on your Windows 2000 system.

### Installing PATROL Internet Server Manager on Windows

Use the following steps to install the PATROL Internet Server Manager v5.3.00 (console components only) on a Microsoft Windows 2000 Advanced Server system.

1. Insert the PATROL Internet Server Manager CD into the CD-ROM drive on your machine. To start the PATROL Installation Utility, do one of the following:
  - Use the Windows Explorer to open the CD directory and double-click **setup.exe**. The Welcome to the Installation Utility dialog is displayed.
  - Open a DOS command window and change to the <CD-ROM\_drive>:\setup.exe directory. Type setup.exe at the prompt, then press Enter. The Welcome to the Installation Utility dialog is displayed.
2. Click **Next** to continue. The Review License Agreement dialog is displayed.
3. Select **Accept** to accept the license agreement, then click **Next** to continue.

**Note:** If you do not accept the license agreement, the installation program will exit when you click **Next**.

The Select Type of Installation dialog is displayed.

4. Select the **Typical** installation path and click **Next** to continue. The Specify Installation Directory dialog is displayed.
5. We used the default installation directory (C:\Program Files\BMC Software) to install PATROL components on our Windows 2000 system. If you want to install PATROL in a different directory, type the directory path in the BMC Products Installation Directory field.  
  
Click **Next** to continue. The Select System Roles dialog is displayed.
6. Select **Console Systems** and click **Next** to continue. We selected a Console System because we want to use this system to view data collected by the PATROL Agent on our SuSE Linux system. The Select Products and Components to Install dialog is displayed.
7. Expand **PATROL Solutions**, select the **PATROL Internet Server Manager** solution, and then click **Next** to continue. The Information for the PATROL Agent dialog is displayed.
8. Select **Start the PATROL Agent automatically**, then click **Next** to continue. The Review Selections and Install dialog is displayed.

The Review Selections and Install dialog shows the PATROL solution you are installing, the configuration settings that you specified, and the approximate amount of disk space required by the install.

9. Select **Start Install** to begin the installation of the PATROL Internet Server Manager on your Windows 2000 system. The Installation Status dialog is displayed. This dialog shows the progress of the installation.
10. When the installation is 100% complete, click **Next** to continue. The Success dialog is displayed. The Success dialog shows which PATROL products and components were installed and the location of the installation log file.
11. To view the installation log, click **View Log**. The installation log is displayed in a scrollable window that allows you to review the messages that were generated by the PATROL Installation Utility during the installation of the PATROL Internet Server Manager on your Windows 2000 system.  
  
When you have finished reviewing the installation log, close the window. You will be returned to the Success dialog.
12. Click **Finish**. You have completed the installation of the PATROL Internet Server Manager console components on your Windows 2000 system. Now you can view the data that the PATROL Agent has been collecting on the SuSE Linux system.

### 11.3.4 Customization

After a Web server has been discovered, all active parameters are automatically enabled and PATROL Internet Server Manager begins monitoring. Each application instance uses a configuration for monitoring and managing your servers. The configuration uses default values that can be verified or changed according to your system requirements.

When an Internet server is auto-discovered or added, PATROL Internet Server Manager may automatically prompt you to verify configuration information. You can also manually configure a server application instance at any time.

Depending on the type of server, you can configure some or all of the following:

- ▶ Basic settings (used for alarm ranges, instance name, requesting a URL type and location, and authentication of user names)
- ▶ Local settings (used to determine the Server Root directory, administration commands, automatic recovery action)
- ▶ Files settings (for setting the Server Root address, Access Log and Error Log locations, and Access Log format)
- ▶ Remote settings (for PATROL Agent port on the remote system)

After we installed and loaded PATROL Internet Server Manager, we configured the following settings:

- ▶ Basic Log Monitoring
- ▶ Basic settings
- ▶ File settings
- ▶ Local settings

#### Enabling basic log monitoring

In this release of PATROL Internet Server Manager, to monitor your Web server's access logs, you need to load one of the following log monitoring KM list files:

- ▶ ISM\_LOGMON\_BASIC.kml, which provides basic access log monitoring functions and complete error log monitoring functionality

- ▶ ISM\_LOGMON\_ADVANCED.kml, which contains all the functionality of the ISM\_LOGMON\_BASIC.kml file and provides real-time instances of the top 10 clients, URLs, users, or virtual servers accessing the Web site, and complete error log monitoring functionality

We only used the basic log monitoring functionality for this installation. To enable basic log monitoring, perform the following from the PATROL console:

1. From the PATROL main menu choose **Files -> Load KM**. A list of available KMs appears in the Knowledge Module File Selection dialog.
2. Select **ISM\_LOGMON\_BASIC.kml**.
3. Click **OK**. Basic log monitoring capability is now enabled for PATROL Internet Server Manager.

### Configuring basic settings

Since we did not like the default name for the FTP server instance that was automatically discovered by the PATROL Internet Server Manager product, we used the **Configure Basic Settings** menu command to change the name from FTP\_Daemon to agent390\_FTP\_Daemon.

To configure the basic settings for an application instance (such as an FTP server or a POP3 server), perform the following:

1. Right-click the server application instance. The application menu is displayed.
2. Choose **KM Commands -> PATROL Admin -> Configure -> Basic** to display the Basic Configuration dialog for the specified server instance.



Figure 11-22 Basic Configuration dialog

The Basic Configuration dialog displays the name of the server instance as it appears in the PATROL Console. The installation default is FTP\_Daemon.

3. To change the name to be displayed in the PATROL Console, type the new instance name. We changed the name to agent390\_FTP\_Daemon.
4. Click **Accept** to save the entry and close the dialog.

### Configuring file settings

Since we enabled basic log monitoring, we need to configure the location of the log files on our SuSE Linux system.

To configure a Web server's file settings, including the Server Root (Home) Directory and the Access Log and Error Log locations, statistics, and format, perform the following:

1. Access the menu commands for the application instance.

2. Choose **KM Commands** -> **PATROL Admin** -> **Configure** -> **Files** to display the File System Configure dialog.

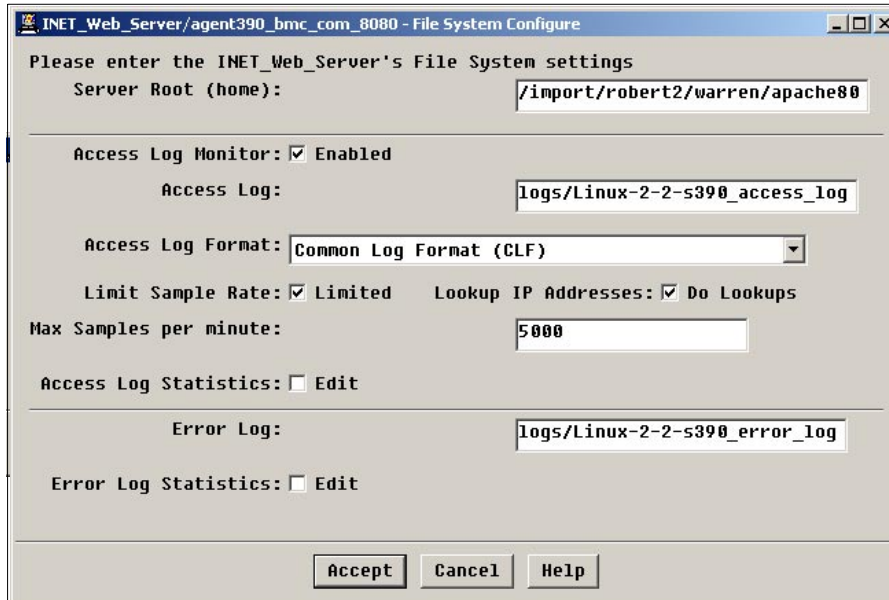


Figure 11-23 File System Configure dialog

3. If needed, enter the proper file system location for your Server Root (Home) directory on your SuSE Linux system.
4. Configure the following properties for access log or error log monitoring as appropriate for your system:
  - Set the Access Log location.
  - Set the Access Log format.
  - Set the limit for the number of samples per minute (the default is 5000).

**Note:** PATROL Internet Server Manager reads only the number of lines indicated on the File System Configure dialog. For example, if you set the limit to 5000 lines, but 50,000 lines were written to the log during the poll interval, PATROL Internet Server Manager would read only the last 5000 lines in the log.

5. Select Lookup IP Addresses if you want to resolve IP addresses to host names in the access log. Do not select Lookup IP Addresses if you do not want PATROL Internet Server Manager to report host names in its access log reports. You may want to turn off DNS lookups in this manner to conserve system resources used in looking up host names.
6. Set the location of the Error Log.
7. Select Edit Error Log Statistic sets.
8. Click **Accept** to save the current entries and close the dialog.

### Configuring local settings

We used the Configure Local Settings menu command to enable Automatic recovery Startup for our Web server. We also configured the following:

- ▶ Restart command
- ▶ Shutdown command
- ▶ Startup command



To configure a server's local settings, perform the following:

1. Access the menu commands for the server's application instance.
2. Choose **KM Commands -> PATROL Admin -> Configure -> Local** to display the Local Configure dialog for the specified server instance.

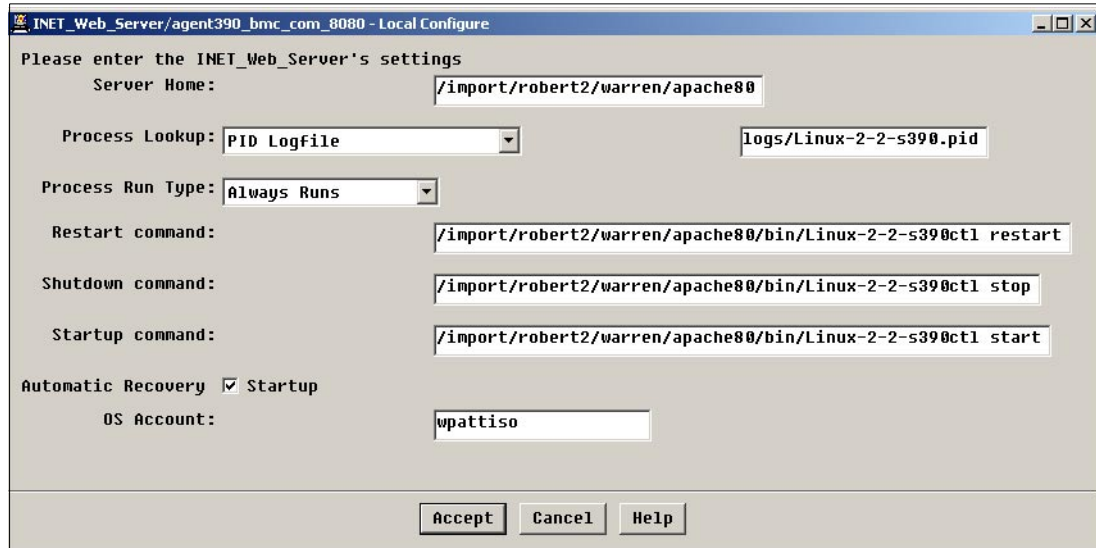


Figure 11-24 Local Configure Dialog

3. Type the full path name of the home directory in the Server Home field. This directory is used as a relative base for other files and commands. Server processes are managed and monitored by looking for the process on the computer.
4. In the Process Lookup selection field, choose the type of process lookup methods, then type the corresponding text in the text field. The following Process Lookup methods are available:
  - PID Logfile
  - Process Regular Expression
  - NT Service Name
5. Configure other local settings for your managed server as needed:
  - Process Run Types: In this field indicate whether the server process should run at all times, as most Web servers do, or if it is launched on demand, as is the case with UNIX FTP servers.
  - Administration Commands: These commands are used to stop, restart, and start a server. For UNIX systems, make sure you have sufficient user privileges using the OS Account field.
  - Automatic Recovery: You can configure the Automatic Recovery function to allow PATROL Internet Server Manager to restart an Internet server that becomes unresponsive. This feature is disabled by default. We selected Select Automatic Recovery Startup for the Web server.
  - OS Account: In this field provide user privileges to the application for Administration Commands (if needed).
6. Click **Accept** to save the current entries and close the dialog box. If you enter an incorrect password when trying to start a Web server with SSL support, you may need to kill the

Web server process involved. You can find a list of all Web server processes using the following command:

```
ps -ef | grep https
```

Using the output of the `ps` command, identify the Web server processes and determine their PIDs (column 2 of the output). Then you can explicitly kill the processes using the following command:

```
kill -9 PIDLIST
```



# System management using **MAINVIEW for Linux Servers**

This chapter describes how to use MAINVIEW for Linux Servers to accomplish a variety of system management tasks.

## 12.1 Operations for MAINVIEW for Linux Servers

MAINVIEW for Linux Servers monitors Linux running on zSeries or Intel platforms. It displays performance metrics via a Web browser or traditional 3270 screen. The agent on each Linux system communicates with a monitor address space running on z/OS or OS/390. The address spaces on z/OS are able to communicate within a Sysplex or across LPARs. Hence, from a single monitor session you are able to see any Linux image across the enterprise.

MAINVIEW separates the monitoring functions from the user presentation of the monitor data. The monitoring functions are handled in the product address space, which then presents monitor records to the user address space. MAINVIEW is the base architecture manipulating the records in the user address space: summarizing, sorting, and selecting the data, and then displaying it in views.

Another important concept within the product is *context*. All views within the product will display metrics from Linux systems based on the context the user has chosen. This may be a single Linux system, all Linux systems in the enterprise, or a user-defined context such as all Linux Web servers.

First we discuss navigation via a 3270 screen. Figure 12-1 is the default main menu when accessing the product from TSO, VTAM session, or an EXCP device. To access MAINVIEW from TSO/ISPF, select the appropriate option from the ISPF panel. For access through VTAM, log on to the appropriate VTAM APPL. EXCP devices are generally set up to automatically bring up MAINVIEW by starting a started task, again usually at IPL time. Either way, the first menu displayed is the overall MAINVIEW Selection Menu.

```
----- MAINVIEW Selection Menu -----
OPTION  ===>  _
                                DATE   -- 02/05/07
                                TIME    -- 07:39:26
                                USERID  -- RDIRDA
                                MODE    -- ISPF 4.8

    0   Parameters and Options
    E   Alerts and Alarms
    P   PLEX Management (PLEXMGR)
    U   Utilities, Tools, and Messages

Solutions for:
    A   Automated Operations
    C   CICS
    D   DB2
    I   IMS
    L   Linux
    N   Network Management
    S   Storage Management
    T   Application Management and Performance Tuning
    W   WebSphere and MQSeries
    Z   OS/390, z/OS, and USS

Enter X to Terminate

                                Copyright BMC Software, Inc. 2002
```

Figure 12-1 MAINVIEW Selection Menu

Select option **L** to display the MAINVIEW for Linux Servers Easy Menu (shown in Figure 12-2). Notice that the following z/OS subsystems are available as well:

- ▶ CICS
- ▶ DB2
- ▶ IMS
- ▶ WebSphere
- ▶ MQSeries
- ▶ UNIX System Services
- ▶ Storage Management

- ▶ VTAM
- ▶ IP
- ▶ The operating system (z/OS or OS/390)

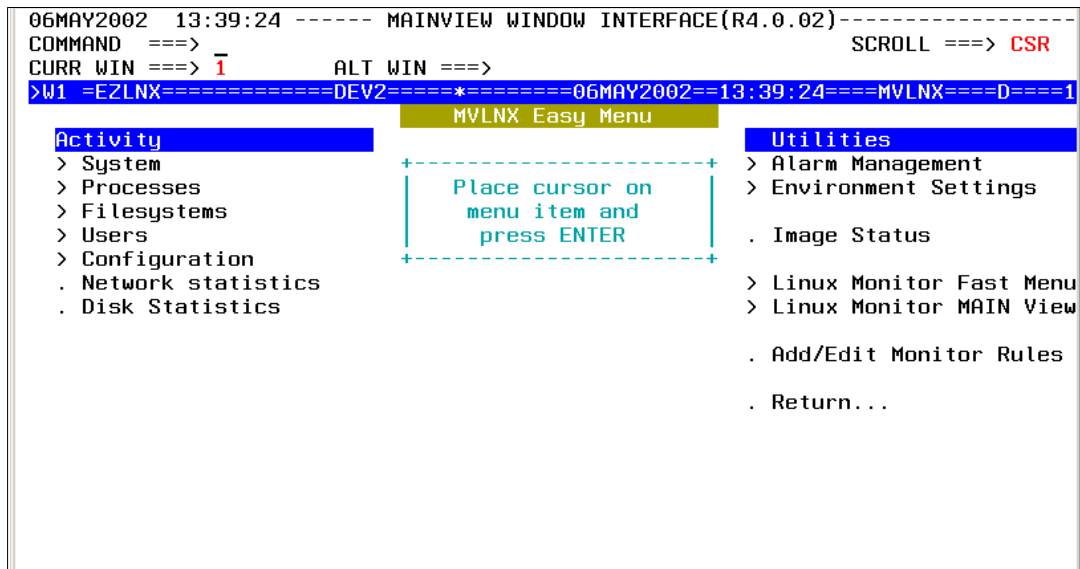


Figure 12-2 MAINVIEW for Linux Servers Easy Menu

The reverse video line displays window information. *W1* is the window status indicator. A user can have up to 20 windows displayed at one time. Windows can be different views in MAINVIEW for Linux Servers or a mixture of views from other MAINVIEW components. The view name in Figure 12-2 is EZLNx and the date and time shows when the data was obtained. Each of the text menu items are hyperlinks to other views. The text preceded by an arrow bracket (>) indicates that it is a hyperlink to another menu. Text preceded with a dot (.) indicates that it is a hyperlink to a data view.

All of the information available in the 3270 interface is also available via MAINVIEW Explorer (a Web browser interface). To access MAINVIEW from the Web, complete the following steps:

1. Point your Internet Explorer to the following URL:

`http://systemname:port`

*systemname* is the DNS name of the z/OS system running the MAINVIEW Explorer Server or its IP address, and *port* is the port number selected for the server to use. The default port number is 3940. The Java cab files required to run the MAINVIEW Explorer will download to your workstation.

2. Enter a user ID and password when you are prompted for a userid and password for the system that the server is running.

MAINVIEW Explorer will RACHECK the userid and password before allowing access. All further authority will be governed by the access levels of your userid. Optionally, you may provide a high-level qualifier for view and configuration files if you do not wish to use your userid as the high-level node for these files. The default main MAINVIEW Explorer menu is shown in Figure 12-3 on page 286.

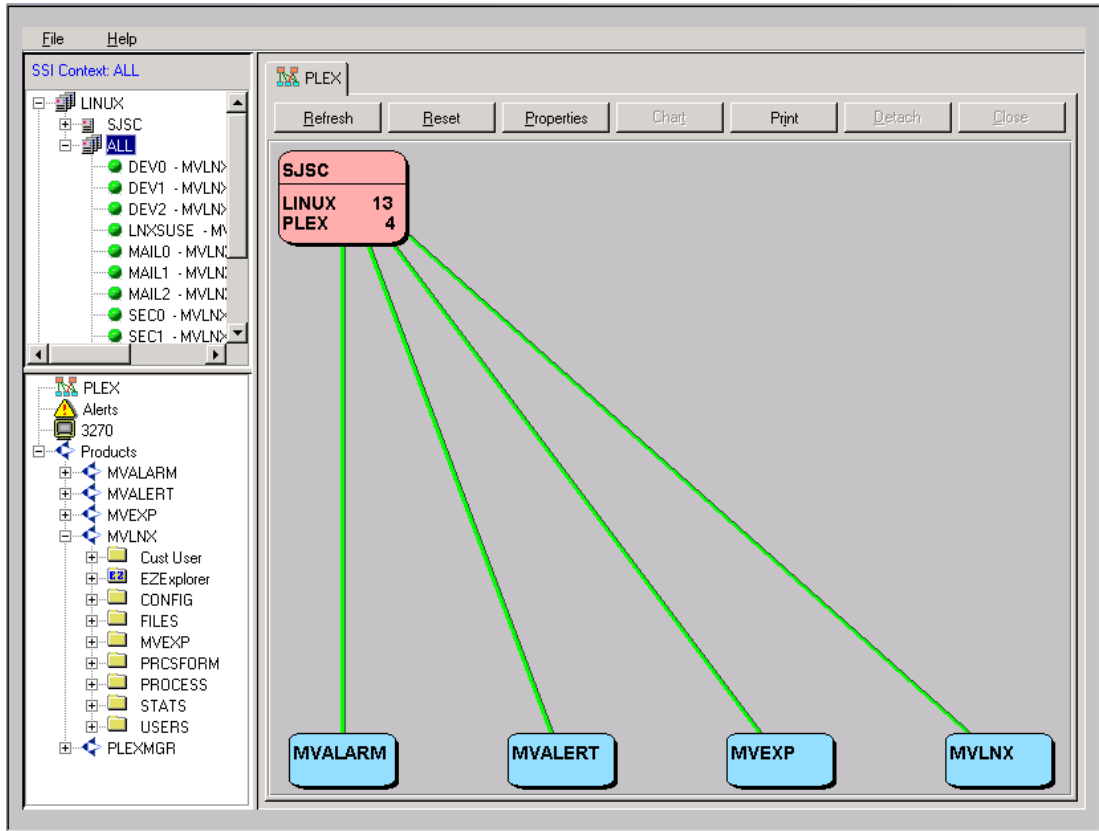


Figure 12-3 MAINVIEW Explorer menu

There are two frames on the MAINVIEW Explorer panel. The frame on the left is the navigation frame. The upper portion displays the systems tree, the lower portion displays the products tree. The frame on the right is the view frame. The same views available on a 3270 are displayed. The context is important to MAINVIEW whether you use 3270 or a Web browser. If you right-click a system in the systems tree, a pop-up menu with set context and display gauges options is displayed. If you double-click a system in the system tree, it will set the context to that system, as well as display the gauges (shown in Figure 12-4 on page 287) with additional graphic features.

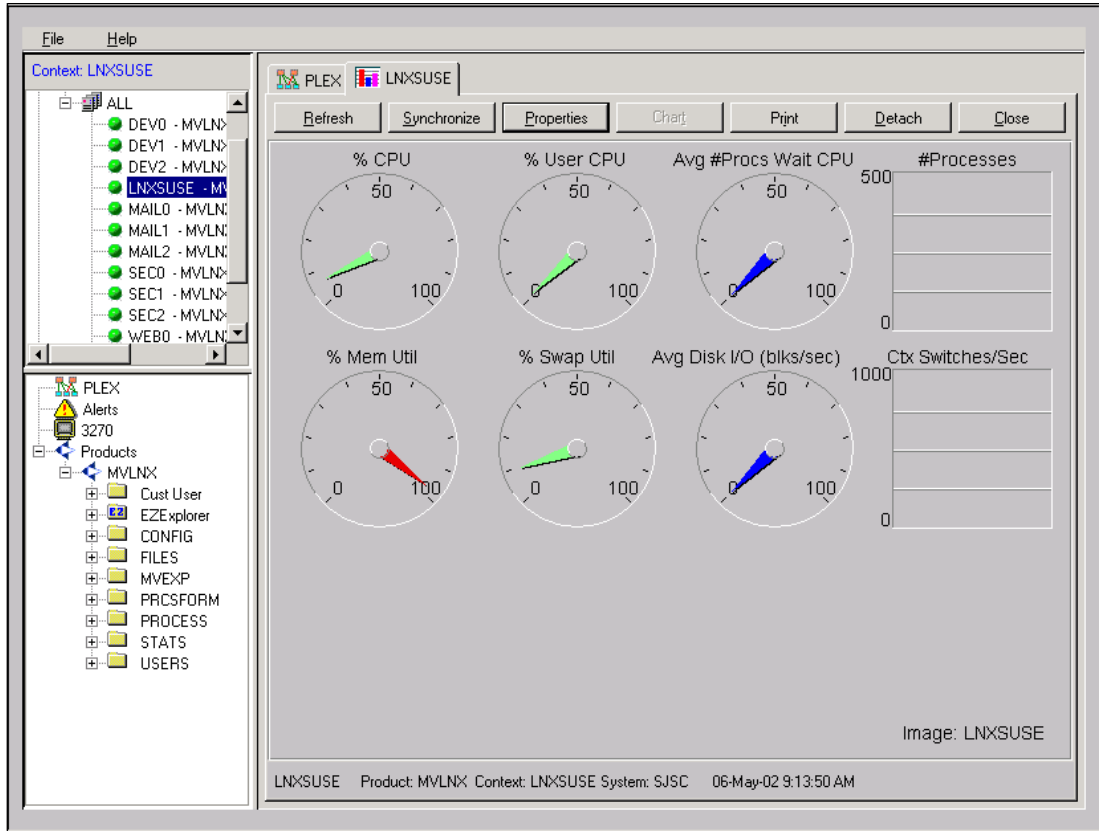


Figure 12-4 LNXSUSE view

The product tree has been built for the current context. Across the top of the view frame are view tabs which allow you to display a previous view.

Click the **Refresh** button to gather new data from the monitor. Click the **Properties** button to customize the view. For example, you can customize the view by changing the gauge type to half circles, histograms, or stoplights.

MAINVIEW Explorer provides graphics features not available with the 3270 interface. By expanding the **Process** folder in the product tree and double-clicking the **PSLUSE** view, you can see a detailed list of all processes running on a Linux system. Each of the column headings in any view can be right-clicked to bring up a customization menu.

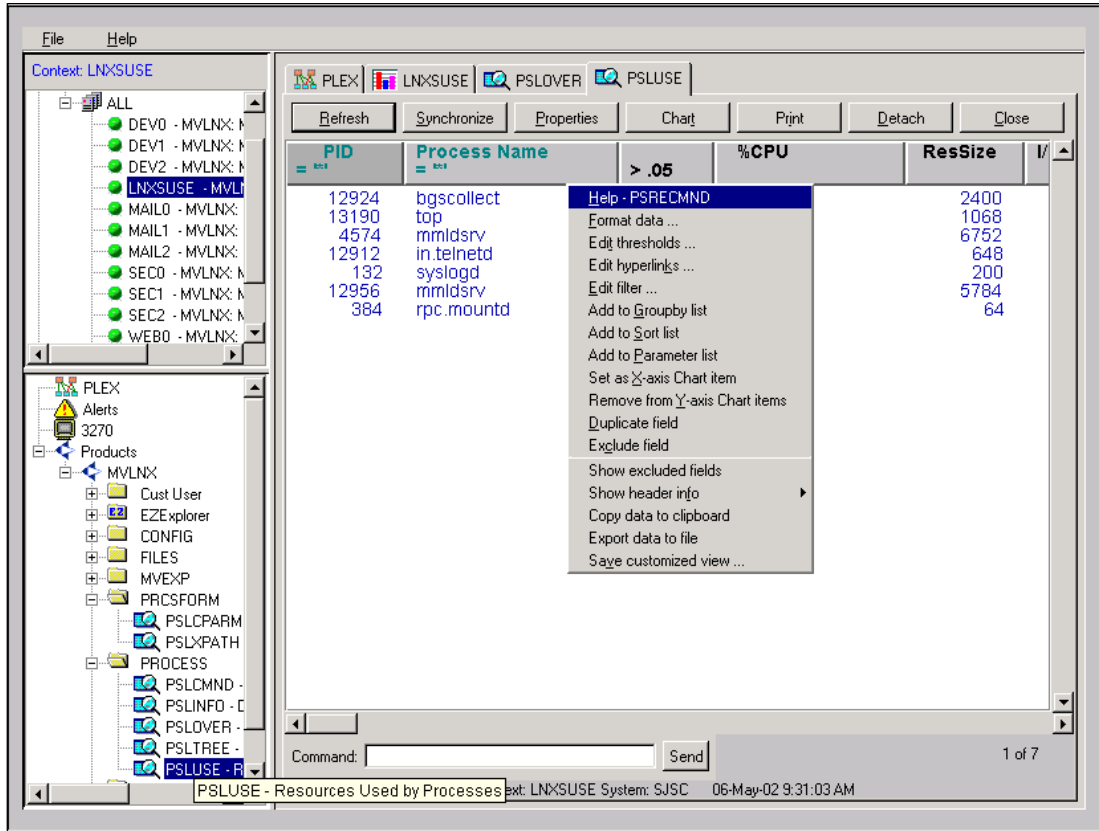


Figure 12-5 PSLUSE view

The MAINVIEW Explorer Web interface also allows you to create graphs. To create a graph, complete the following steps:

1. In the PSLUSE view (shown in Figure 12-5), right-click the **Process Name** header to make it the X-axis.
2. Right-click the **%CPU**, **CPU User**, and **CPU System** headings to select them as Y-axis.
3. Right-click the **%CPU** heading again to filter on processes using more than .05% of the CPU.
4. Click the **Chart** button to graph selected items. This displays the PSLUSE view graph (shown in Figure 12-6 on page 289).



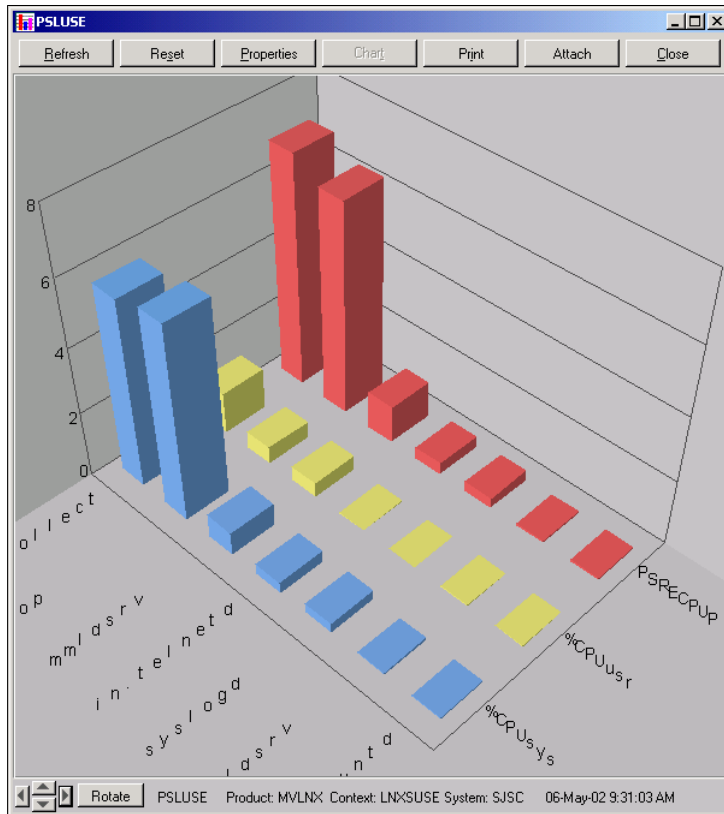


Figure 12-6 PSLUSE view graph window

When you click the **Detach** button, the graph window detaches from the browser window. Clicking on the **Detach** button toggles the function between Attach and Detach. You can click the **Attach** button to put this window back in the view frame. If you want to change the graph type, click the **Properties** button and select the graph tab.

The 3270 interface does not support these graphs, but you can customize the appearance of the views in the 3270 interface by using the **CUST** command. Notice the commands available on the 3270 screen (shown in Figure 12-7 on page 290) correspond to the commands in the Web browser pop-up window (shown in Figure 12-5 on page 288).

```

----- VIEW CUSTOMIZATION - PSLUSE -----
OPTION ==> _
Options: (that require column selection)      Other options:
F - Format      M - Move      I - Include      G - Graph      S - Save view
O - Order      R - Repeat      X - Exclude      P - Parameters E - Show excluded
L - Filter      T - Threshold  H - Hyperlink   Z - Summarize  K - Show template

-----
Some options ask you to select a target column. To do so, either type the
option with the column id on the OPTION line (as in: f e to format column E),
or type just the option, move the cursor to the target column and press ENTER.
Your changes are implemented every time you press ENTER. You can save the
modified view definition with any name you choose and specify where thresholds
-----

```

A	B	D	E	F	G	H
CMD	PID	Process Name	%CPU	ResSize	I/O Rate	VMemSz
			0.....50...100	(KB)	-----	(KB)
	13190	top	3.70	1068	0.00	1764
	12924	bgscollect	2.09	2400	0.00	4148
	132	syslogd	0.29	200	0.00	1300
	4574	mmlsrv	0.25	6784	0.00	9448
	12912	in.telnetd	0.21	648	0.00	1332
	543	nscd	0.05	568	0.00	1508
	13087	mmlsrv	0.03	5784	0.00	8548

Figure 12-7 PSLUSE View Customization

When customization is complete, you can save the customized view to a user library or a system-wide library. You can also save the data that is displayed in a view to a file using the **EXPORT** command.

```

----- Export Open Data Set -----
COMMAND ==>

LIBRARY (PDS):
Project      ==>
Group       ==>
Type        ==>
Member      ==> _      Replace (Y/N)? YES

Other partitioned or sequential data set:
Data Set Name ==>
Volume serial ==>      If not cataloged

Export Options:
Disposition  ==> REPLACE Replace or Append if sequential data set
Output format ==> ASIS   ASIS or CSV
Lines/Page   ==> 0     ASIS format only (NNN)
Sysout Class ==>      If specified, overrides other data set options

Press END to save changes and export report.
Type CANCEL to return to previous panel without saving changes.

```

Figure 12-8 Export Open Data Set

In Figure 12-8, notice the option for the saved file is ASIS, which will save the view just as you see it online. On the other hand, CSV will save the view in comma separated value format. A CSV file may be sent to a workstation and opened with spreadsheet software, thus giving the 3270 user the ability to create graphic reports almost as simply as the Web browser user. Many technicians use the customization facility as an ad-hoc report writer.

## 12.1.1 Availability Management

### Server Availability

The Image Status view (SYLOVERZ), shown in Figure 12-9, displays all Linux servers across the enterprise and their current status.

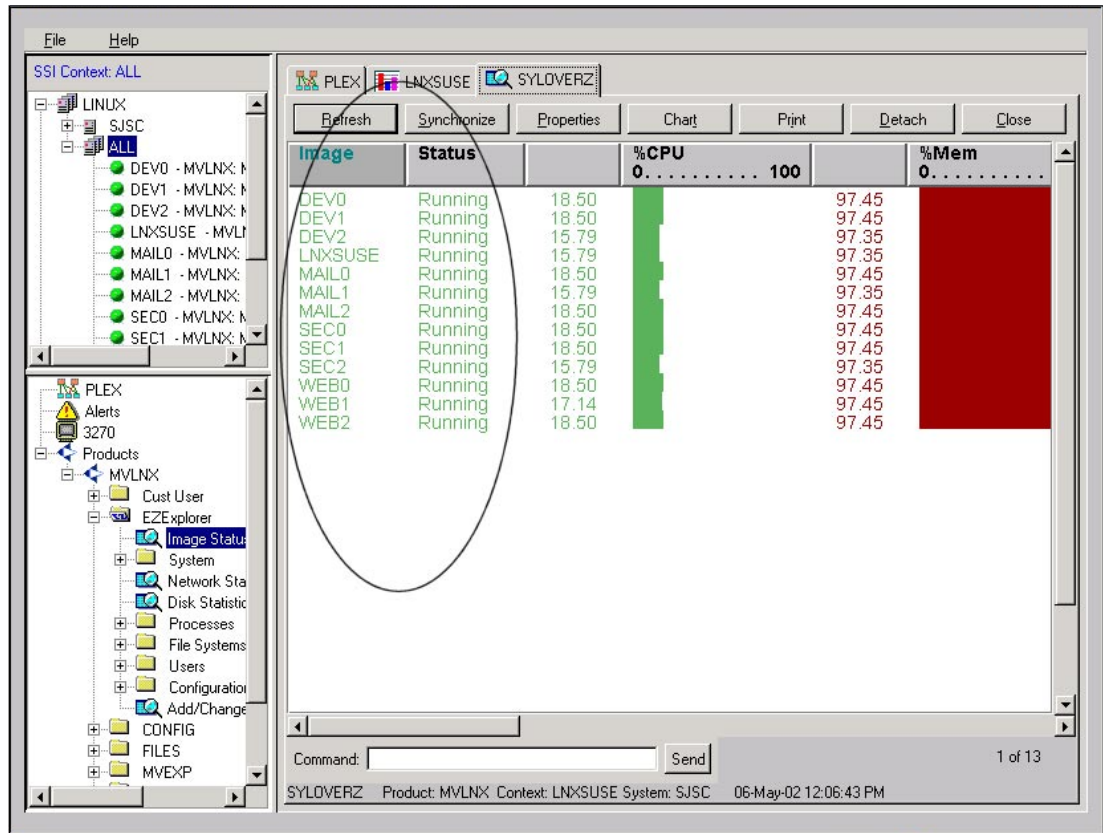


Figure 12-9 SYLOVERZ view

If the MAINVIEW for Linux Servers data collector that runs on Linux is responding with heartbeat data, then MAINVIEW will show that Linux system then Running is displayed in the Status field. If the system is down or not communicating it is shown as Inactive. This view is also helpful in monitoring general Linux system health, which will be discussed in the health monitoring section.

### Required process availability

One problem of availability management is to ensure that required processes are running at the appropriate times. MAINVIEW for Linux Servers can address this problem by customizing a view from the resources used by the process view, PSLUSE. Since MAINVIEW has a built-in SQL-like query language, you can ask for a list of processes by adding a WHERE clause. To do this, you need to know the field name for process name.

1. Right-click the **Process Name** header and the pop-up customization window shows the field name: **PSRECCMD** (shown in Figure 12-10 on page 292).

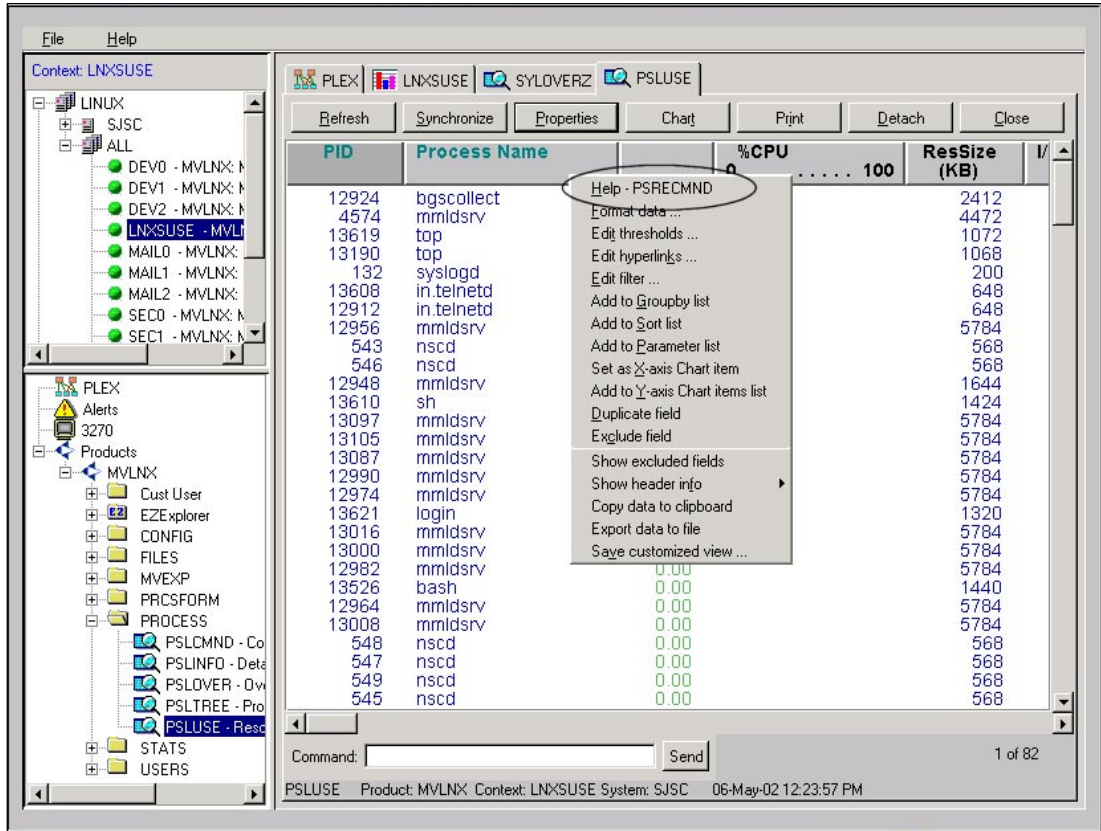


Figure 12-10 PSLUSE view

2. Click the **Properties** button and select the **WHERE** tab to enter a SQL query (shown in Example 12-11).

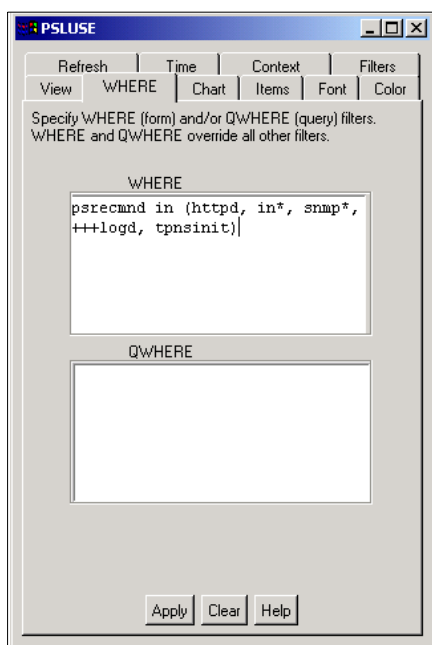


Figure 12-11 PSLUSE query

### 3. Click **Apply**.

Once you click Apply, the view now displays the process names you have requested. Not only can you detect missing processes this way, but you can also catch performance problems for these selected processes. When you exit this customized view, the system prompts whether it should be saved. If you click Yes, the view becomes an option in the product tree under the Cust User folder.

## **Required file system availability**

Another major availability problem is the possibility that a required file system may not be mounted to a Linux system. This can cause applications to fail because a required subdirectory is absent; or it can cause an even more serious problem, such as a full root file system, which could cause the Linux system to crash.

The File System Mount Point view (FSLMNT) shows all mounted file systems, mount points, space available, and related file system metrics.

## **Network availability**

MAINVIEW for Linux Servers displays the IP traffic metrics for each Linux system on the Network Statistics (NTLSTAT) view. The inbound and outbound packets, bytes per second are shown, as well as errors per second and collision rate.

This information is useful for availability management because MAINVIEW for Linux Servers can detect problems not only with extremely high activity rates, but also zero activity rates. Many times, low or zero activity rates can be an indication of network problems or failing applications.

## **12.1.2 Health monitoring: Heartbeat data**

MAINVIEW for Linux Servers stores heartbeat data in the product address space sent from each monitored Linux system. This provides a very quick and low-overhead method of viewing performance metrics for each Linux system. This can be particularly useful if MAINVIEW for Linux Servers is monitoring hundreds (or more) Linux systems. This data is displayed on the Image Status (SYLOVERZ) view shown in Figure 12-9 on page 291.

The Image Status (SYLOVERZ) view displays performance at a glance for each Linux system: % CPU, % Memory, % Swap, I/O Rate, and Page Rate. The data is refreshed as frequently as specified in the monitor rules, which are set up during product customization. These monitor rules can also be used for organizing data. If you are running more Linux systems than can be displayed on your 3270 screen or Web browser session, you may want to take advantage of a powerful MAINVIEW feature, the ability to summarize data.

To summarize data, complete the following steps:

1. Set up a monitor rule for each group of Linux systems to be summarized.

For example, if all the Web server systems have a host name starting with WEB, you can set up a rule for WEB\*.

2. Display the SYLOVERZ view and issue the **CUST** command:, then press PF11 to see the match rule column (shown in Figure 12-12).

```

----- VIEW CUSTOMIZATION - SYLOVERZ -----
OPTION ==> z n_                                SCROLL ==> CSR
Options: (that require column selection)      Other options:
F - Format      M - Move      I - Include      G - Graph      S - Save view
O - Order      R - Repeat     X - Exclude     P - Parameters E - Show excluded
L - Filter     T - Threshold  H - Hyperlink  Z - Summarize  K - Show template

-----
Some options ask you to select a target column. To do so, either type the
option with the column id on the OPTION line (as in: f e to format column E),
or type just the option, move the cursor to the target column and press ENTER.
Your changes are implemented every time you press ENTER. You can save the
modified view definition with any name you choose and specify where thresholds
-----

```

A	K	L	M	N	O
Image	%Swap	I/O-rate	PageRate	Matched	Host Name
	0.....50...100			Rule	
DEV0		0.00	0.00	\$DEFAULT	Dev0
DEV1		0.00	0.00	\$DEFAULT	Dev1
DEV2		0.00	0.00	\$DEFAULT	Dev2
LNXSUSE		0.00	0.00	\$DEFAULT	lnxsuse
MAIL0		0.00	0.00	MAIL*	Mail0
MAIL1		0.00	0.00	MAIL*	Mail1
MAIL2		0.00	0.00	MAIL*	Mail2

Figure 12-12 SYLOVERZ view customization

- Issue the Z N command to summarize on the Matched Rule column.

Now the Matched Rule column is the first column. Use the **Format** command to change its column heading to Linux Appl, then use the **Threshold** command to change the Matched Rule values to what you want to call each group of Linux systems. You should see something like what is displayed in Figure 12-13.

```

----- VIEW CUSTOMIZATION - SYLOVERZ -----
OPTION ==> T                                SCROLL ==> CSR
Options: (that require column selection)      Other options:
F - Format      M - Move      I - Include      G - Graph      S - Save view
O - Order      R - Repeat     X - Exclude     P - Parameters E - Show excluded
L - Filter     T - Threshold  H - Hyperlink  Z - Summarize  K - Show template

-----< Threshold - column: A      element: IMRERLID      >-----
Condition:
1st => A = MAIL*      Attr: Sub:      Inherit from =>      0: GREEN  5: GREEN
2nd => A = WEB*      => 0 => Mail      1: BLUE   6: BLUE
3rd => A = *          => 0 => WebSrv    2: YELLOW 7: YELLOW
4th =>                => 0 => Prod_    3: PINK   8: PINK
5th =>                => =>          4: RED    9: RED
6th =>                => =>
7th =>                => =>
8th =>                => =>

-----

```

A	B	F	G	H	I	J	K
Linux	Image	Status	%CPU		%Mem		
Appl			0.....50...100		0.....50...100		
Prod	*****	Running	15.85		96.23		12.94
Mail	MAIL****	Running	15.85		96.23		12.94
WebSrv	WEB*****	Running	15.85		96.23		12.94

Figure 12-13 SYLOVERZ view customization

- Specify option X to exclude the Image and Hostname columns since they are not useful for a summary of multiple Linux systems.
- Replicate column A, and use the **Format** command again to give the new column B a column header of Linux Count and a summary type of Count. The column is now a count of how many Linux systems are summarized on that line.

**Note:** You may want to change the default formats of the data fields from average to maximum, so the worst value is propagated from a Linux machine up to the summary line.

6. Save the new view, which looks like Figure 12-14.

```

06MAY2002 14:08:18 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
>W1 =RDALOVRZ=====DEV2=====*=====06MAY2002==14:08:04====MVLNX====U====3
Linux Linux Status %CPU %Mem %S
Appl Count ----- 0.....50...100 0.....50...100 0.5
Prod 7 Running 11.18 96.30 12.94
MailSrv 3 Running 11.18 96.30 12.94
WebServ 3 Running 11.18 96.30 12.94

```

Figure 12-14 RDALOVERZ customized view

If you drill down on a Linux Appl you will see the summarized systems (shown in Figure 12-15).

```

06MAY2002 14:12:20 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
>W1 =RDALOVRZ=SYLOVERZ=DEV2=====*=====06MAY2002==14:08:04====MVLNX====D====3
Image Status %CPU %Mem %Swap
----- 0.....50...100 0.....50...100 0..50.100
WEB0 Running 11.18 96.30 12.94
WEB1 Running 11.18 96.30 12.94
WEB2 Running 11.18 96.30 12.94

```

Figure 12-15 RDALOVERZ summarized view

Now you have a mechanism to monitor hundreds of Linux systems running across the enterprise, summarizing multiple systems as you wish, with the ability to drill down to a list of those summarized systems to diagnose problems.

From the Image Status view (SYLOVERZ), you can drill down to the Image Easy Menu (EZLIMAGE). This menu allows you to display detailed metrics for a single Linux system, based on which system you picked when you drilled down from the Image Status view.

The Image Easy Menu is broken into the following four major areas:

- ▶ Activity
- ▶ Information
- ▶ Configuration
- ▶ System parameters

## Activity views

The first health indicator is the System option under the Activity section. This displays the System Resource Usage view, which shows system resource information, including CPU utilization (user, system, idle), run queue, running processes, memory utilization and totals (kernel, swap, buffer), as well as I/O rates. It also shows kernel tables information (process, inode, file), paging (swapping), shared memory and semaphore statistics, and the 1-, 5-, and 15-minute load averages. You can click any CPU name for details about that CPU.

The SYLUSE is a very wide view. Press PF11 to scroll through the many fields. In Figure 12-16, the HS command was used to horizontally split the MAINVIEW window. In this figure the SYLUSE view is in four windows, each window showing part of the view data.

```

07MAY2002 09:40:48 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>>
CURR WIN ==>> 1 ALT WIN ==>>
>W1 -SYLUSE-----LNXSUSE==*-----07MAY2002--09:40:41---MVLNX---D---1
Image          %CPUutil      %IdleCPU %UserCPU %SystemCPU RunQueue RunOcc
-----
0.....50...100
LNXSUSE      53.72 ██████████ 46.29 7.76 45.96 0.00 0.00

+W2 -SYLUSE-----DEVO-----*-----07MAY2002--09:40:41---MVLNX---D---1
Image  RunOcc  1MinAvgLd 5MinAvgLd 15MinAvgLd #Processes %MemUti
-----
DEVO   0.00   0.97   0.67   0.47   78   97.02 ██████████ 0.50.10

+W3 -SYLUSE-----DEVO-----*-----07MAY2002--09:40:41---MVLNX---D---1
Image  MsgOpsRate SemOpsRate %SwpUtil TotVirtMem(KB) FreeVrtM
-----
DEVO   0.00   0.00   15.36 ██████ 0.....50...100 104752

+W4 -SYLUSE-----DEVO-----*-----07MAY2002--09:40:11---MVLNX---D---1
Image          %ProcTblUtil ProcTblSz MaxProcTbSz %InodeTblUtil
-----
0.....50...100
DEVO   7.91 █████ 79 999 0.58

```

Figure 12-16 SYLUSE view (horizontally split)

Window 1 (W1) displays both overall and by user process CPU. Drilling down on the CPU Utilization field, you will see the CPU engine statistics. Drilling down on the User CPU field reveals a list of running processes ordered by CPU use.

Window 2 (W2) displays 1-, 5-, and 15-minute averages of processes ready to run.

Window 3 (W3) displays rates for some Inter process communication events, Messages and Semaphores.

Window 4 (W4) displays the kernel values for process table size, percent of slots in use, and the maximum size. If all process table slots are in use, no more processes may be started in that Linux system.

The CPU Engine Statistics view (SYLCPU) displays per-processor statistics, processor utilization (percent busy) and processor state (user, system, or wait). You can link to this view from the previous CPU use view or from the Image Status menu.

The Swap Statistics (SYLSWAP) view displays swapping (paging) metrics of the system. Swapping (paging) presents a heavy load on the system, so recognizing swap load is vital to managing system performance. This view displays total swap space and utilization, pages scanned, paging rates, reads, writes, faults, and context switches. You can link to this view from the Image Status Menu.



You can also see virtual memory configuration which affects system paging. Under the system parameters section of the Image Status menu is the Virtual Memory option. You can drill down further to see the Virtual Memory System Parameters (CFLVMEM) view, shown in Figure 12-17.

```

09MAY2002 09:11:19 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
SCROLL ==> CSR
>W1 =CFLVMEM=====LNXSUSE==*=====09MAY2002==09:11:18====MVLNX====D====1
Image      nfract      ndirty      nrefill     nref_dirt   age_buffer  age_super   min_perc
----- (bdflush) (bdflush) (bdflush) (bdflush) (bdflush) (bdflush) (bufferm
LNXSUSE    0            0            0            0            0            0            0

```

Figure 12-17 CFLVMEM view

CFLVMEM is a tabular view that displays the values of the system parameters contained in the Linux system's /proc/sys/vm directory. These parameters are related to the kernel's management of virtual memory.

```

07MAY2002 11:59:41 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
SCROLL ==> CSR
>W1 =SYLTBLS=====LNXSUSE==*=====07MAY2002==11:51:29====MVLNX====D====1
Image      %ProcTblUtil ProcTblSz  MaxProcTbSz %InodeTblUtil Ino
----- 0.....50...100 ----- 0.....50...100 ---
LNXSUSE    7.81 █          78          999         0.60

```

Figure 12-18 SYLTBLS view

The Kernel Table Statistics (SYLTBLS) view, shown in Figure 12-18, displays the size and graphically displays the utilization of the process table (processes running), the inode table (files available), and the file table (files in use).

MAINVIEW for Linux Servers also includes views to show how the file system parameters are set. If you drill down on the File System option under the System Parameters heading on the

Image Status Menu, you will see the File System - System Parameters (CFLFSYS) view that is shown in Figure 12-19.

```

09MAY2002 09:19:07 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==> _                               SCROLL ==> CSR
CURR WIN ==> 1                               ALT WIN ==>
>W1 =CFLFSYS=====LNXSUSE=*=====09MAY2002==09:19:07===MVLNX===D===1
Image  nr_dentry nr_unused age_limit want_pages nr_alloc nr_free dquot-max
----- (dentry) (dentry) (dentry) (dentry) (dquot) (dquot) -----
LNXSUSE      0      2706      45      0      0      0      0

```

Figure 12-19 CFLSYS view

CFLFSYS is a tabular view that displays the values of the system parameters contained in the Linux system's /proc/sys/fs directory. These parameters are related to the kernel's management of file systems.

The Network Statistics view is a tabular view showing the inbound and outbound packet count, the inbound and outbound byte count, and collisions and errors encountered for each interface or protocol listed.

MAINVIEW for Linux Servers has an entire submenu of network system parameter displays. These options, under the Network option of System Parameters, are:

- ▶ Core
- ▶ ICMP
- ▶ IP
- ▶ IP Fragmentation
- ▶ TCP
- ▶ Routing
- ▶ Interface Specific
- ▶ Net Neighbor Handling

The Network Core Systems Parameters (CFLNETC) view is shown in Figure 12-20.

```

09MAY2002 09:27:09 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
>W1 =CFLNETC=====LNXSUSE==*=====09MAY2002--09:26:16----MVLNX----D----1
Image      msg-burst msg-cost      netdev optmem_max rmem_default rmem_max wmem_
-----
LNXSUSE          50      5      300      10240      65535      65535

```

Figure 12-20 CFLNETC view

CFLNETC is a tabular view that displays the values of the system parameters contained in the Linux system's /proc/sys/net/core directory. These parameters are related to the kernel's management of core network resources.

### Process views

MAINVIEW for Linux Servers has a number of ways to display running processes. MAINVIEW has columnar displays showing a process overview, resource use, and a graphical representation of process tree information, as well as summaries of processes by User and Group IDs.

```

07MAY2002 12:24:21 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
>W1 =PSLOVER=====LNXSUSE==*=====07MAY2002==12:24:09====MVLNX====D====78
CMD      PID Process Name      Elapsed      %CPU      ResSize I/O Rate
-----
13190 top      28:23:03      7.77      564      0.00
15427 bgscoll 04:31:10      5.05      2412     0.00
132 syslog 167:42:5      1.91      204      0.00
4574 mmlsrv 123:51:1      0.55      7424     0.00
12912 in.telnetd 30:08:20      0.27      136      0.00
548 nsd 167:41:4      0.06      464      0.00
136 klogd 167:42:5      0.06      176      0.00
6 kswapd 167:44:0      0.05      0      0.00
546 nsd 167:41:4      0.03      464      0.00
543 nsd 167:41:4      0.03      464      0.00
15529 in.telnetd 03:17:11      0.03      648      0.00
545 nsd 167:41:4      0.03      464      0.00
13008 mmlsrv 29:58:49      0.03      3208     0.00
13087 mmlsrv 29:18:36      0.02      6320     0.00
549 nsd 167:41:4      0.02      464      0.00
547 nsd 167:41:4      0.02      464      0.00
12982 mmlsrv 30:00:23      0.01      4348     0.00
13000 mmlsrv 29:59:09      0.01      3236     0.00

```

Figure 12-21 PSLOVER view

The Process Overview (PSLOVER) view displays all processes sorted by percent of CPU. It has a line command allowing selected processes to be signaled with a **kill** or a **kill 9**

command. Selecting the Process ID (PID) column displays the Process Menu (shown in Figure 12-22).

```

07MAY2002 12:28:33 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =PSLOVER=EZLPRC==LNXSUSE==*=====07MAY2002==12:24:09====MVLNX====D====1
                                     Process Easy Menu

Current Pid ->                    12912
Start Time ->                     08:14:56
Start Date ->                     06MAY2002

Activity                               General
. Overview                             . Command Name
. Resource Usage                       . Detail

Place cursor on
menu item and
press ENTER

. Return...

```

Figure 12-22 Process menu

The process menu has hyperlinks to Process Overview, Resource Usage, and Command Name views. It also has a hyperlink to the Process Detail view (shown in Figure 12-23).

```

07MAY2002 12:33:37 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =PSLOVER=PSLINFO=LNXSUSE==*=====07MAY2002==12:24:09====MVLNX====D====1
Process Name top                      Image..... LNXSUSE
PID..... 13190                        %CPU..... 7.77      ChildCPU.. 0.00
PPID..... 12914                        %CPU user.. 0.67      ChildUsrTm 0.00
State..... SLEEP                       %CPU system 7.10      ChildSysTm 0.00
Age..... 28:23:03                      I/O rate... 0.00      ChildIOrt. 0.00
Real User... lwetmore                   MajorFlts.. 0.00      ChildMajFt 0.00
Real UID... 59979                       MinorFlts.. 0.00      ChildMinFt 0.00
EffectiveUsr lwetmore                   InBlks/Sec. 0.00      ChildInBlk 0.00
EffectiveUID 59979                      OutBks/Sec. 0.00      ChildOutBk 0.00
Saved User.. lwetmore                   %Total CPU. 5.30      ParntUsrTm 0.00
Saved UID... 59979                       Priority... 9        ParntSysTm 0.00
Real Group.. bmc                        NiceVal.... 0        ParntIOrt. 0.00
Real GID... 1000                        Res Size... 564     ParntMajFt 0.00
EffectiveGrp bmc                        VirtMemSz.. 1772    ParntMinFt 0.00
EffectiveGID 1000                       DataResSz.. 136     ParntInBlk 0.00
Saved Group. bmc                        TextResSz.. 52      ParntOutBk 0.00
Saved GID... 1000                       StackResSz. 12     GroupCPU... 7.77
ShmResSize 1328

```

Figure 12-23 PSLINFO view

PSLINFO is a detailed view that provides data about a single process, including status, resource usage, and statistics for an interval.

```

10MAY2002 08:50:41 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==> █
CURR WIN ==> 1 ALT WIN ==>
>W1 =PSLUSE=====LNXSUSE==*=====10MAY2002==08:50:15====MVLNX====D==65
CMD      PID Process Name          %CPU      ResSize  I/O Rate  VMemSz
-----  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
          (KB) -----
          (KB)
13190 top                7.31      552      0.00     1772
4574 mmldsrv            6.06      6648     0.00     13196
12990 mmldsrv           1.17      2968     0.00     8692
13105 mmldsrv           1.07      3276     0.00     8688
13087 mmldsrv           0.93      4212     0.00     9948
12964 mmldsrv           0.83      2952     0.00     8688
13097 mmldsrv           0.81      3872     0.00     8688
12948 mmldsrv           0.66      4016     0.00     9256
12956 mmldsrv           0.63      3144     0.00     8692
12982 mmldsrv           0.60      2952     0.00     8688
13000 mmldsrv           0.58      3324     0.00     8688
132 syslogd             0.46      436      0.00     1300
13016 mmldsrv           0.44      4240     0.00     8688
13008 mmldsrv           0.37      3380     0.00     8688
12912 in.telnetd        0.22      140      0.00     1332
548 nscd                0.20      488      0.00     1524
546 nscd                0.11      488      0.00     1524

```

Figure 12-24 PSLUSE view

The Process Resource Usage (PSLUSE) view displays processor and storage utilization by each process. If the processes on this Linux system happen to use services running on z/OS (DB2, for example), you could customize a hyperlink from the process to MAINVIEW for DB2 to see performance metrics for the DB2 subsystem used by the Linux process. To define a hyperlink, complete the following steps:

1. Enter the **CUST** command, then pick a column to be used as a hyperlink to MAINVIEW for DB2; for example, the I/O Rate column (G).
2. Type **H G** in the command area to set the hyperlink for that column.

**Note:** A hyperlink needs a condition and a command. For condition, key **G = \*** (always take the hyperlink) and for command key:

**CON db2sys MVDB2;THDACTV**

This means change the context to the specified DB2 subsystem in the MAINVIEW for DB2 product and display the thread active view. Your screen should look like Figure 12-25 on page 302.



```

07MAY2002 12:49:48 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
>W1 =USLOVERZ=====LNXSUSE=*=====07MAY2002==12:47:48===MVLNX===D===5
Real      #Processes Elapsed      %CPU      ResSize  I/O Rate  State
User Name ----- ProcTime      0.....50...100      (KB) -----
bandrese  3 03:40:51  0.00      1083     0.00     SLEEP
bin       1 168:07:2  0.00       96      0.00     SLEEP
http      2 168:06:0  0.00       70      0.00     SLEEP
lwetmore  4 30:32:03  2.81      353     0.00     SLEEP
root      70 168:08:1  0.44      1121    0.00     *****

```

Figure 12-27 USLOVERZ view

The Users Activity (USLOVERZ) view summarizes all processes by Real User ID. The Real User ID column drills down to a process overview of all processes running with that User ID. There is a Group Activity (GRLOVERZ) view, which is very similar except it summarizes all active processes by Group ID.

### Inter-process communication

MAINVIEW for Linux Servers displays Shared Memory, Message Queues, and Semaphores, both the configuration on the Linux system and the key values in use (shown in Figure 12-29).

```

07MAY2002 13:04:20 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
>W1 =SYLSHMI=====LNXSUSE=*=====07MAY2002==13:04:20===MVLNX===D===1
Image     ShrMemSeg Total-Used  Max-Total  Min-Segment  Max-Segment  Maximum
-----  Ids-InUse (KBytes)  (KBytes)  Size(bytes) Size(bytes) Segs/Proc
LNXSUSE   128 1236992 1236992    1 33554432 128
>W2 =SYLSHMS-----LNXSUSE-*-----07MAY2002--13:04:20--MVLNX---D---26
CMD      KeyValue Image      SegmentID  Size(bytes) #Attach Permissions Owner User
-----
4104709F LNXSUSE      62743      1608      0 --rw-rw-rw- root
4104709E LNXSUSE      61592      2476      0 --rw-rw-rw- root
4104709D LNXSUSE      60949      256      0 --rw-rw-rw- root
4104709C LNXSUSE      55051      968      0 --rw-rw-rw- root
4104709B LNXSUSE      55180      964      0 --rw-rw-rw- root
4104709A LNXSUSE      60950      256      0 --rw-rw-rw- root
41047099 LNXSUSE      62353      5920      0 --rw-rw-rw- root
41047097 LNXSUSE      59280      256      0 --rw-rw-rw- root
41047096 LNXSUSE      60947      5100     0 --rw-rw-rw- root
41047095 LNXSUSE      60050      256      0 --rw-rw-rw- root
41047094 LNXSUSE      62861     26960     0 --rw-rw-rw- root
41047093 LNXSUSE      62105     2592      0 --rw-rw-rw- root
41047092 LNXSUSE      59150      812      0 --rw-rw-rw- root
41047091 LNXSUSE      59279      864      0 --rw-rw-rw- root

```

Figure 12-28 Shared Memory Information and Shared Memory Activity views

Figure 12-28 is an example of the Shared Memory Information view with a horizontal split. The Shared Memory Activity is also displayed. The Shared Segment information view is a detailed view showing segment size, identifiers, creator and owner, permissions, and access times (attach, detach, change).

The Message Queue and Semaphore views are similar to the Shared Memory Information view, showing how those facilities have been configured, as well as which key values are in use.

### File system information views

You can switch back to the MAINVIEW Explorer interface to view the Information views. The Information views which monitor the health of the Linux system are the Mounted File Systems and User views.

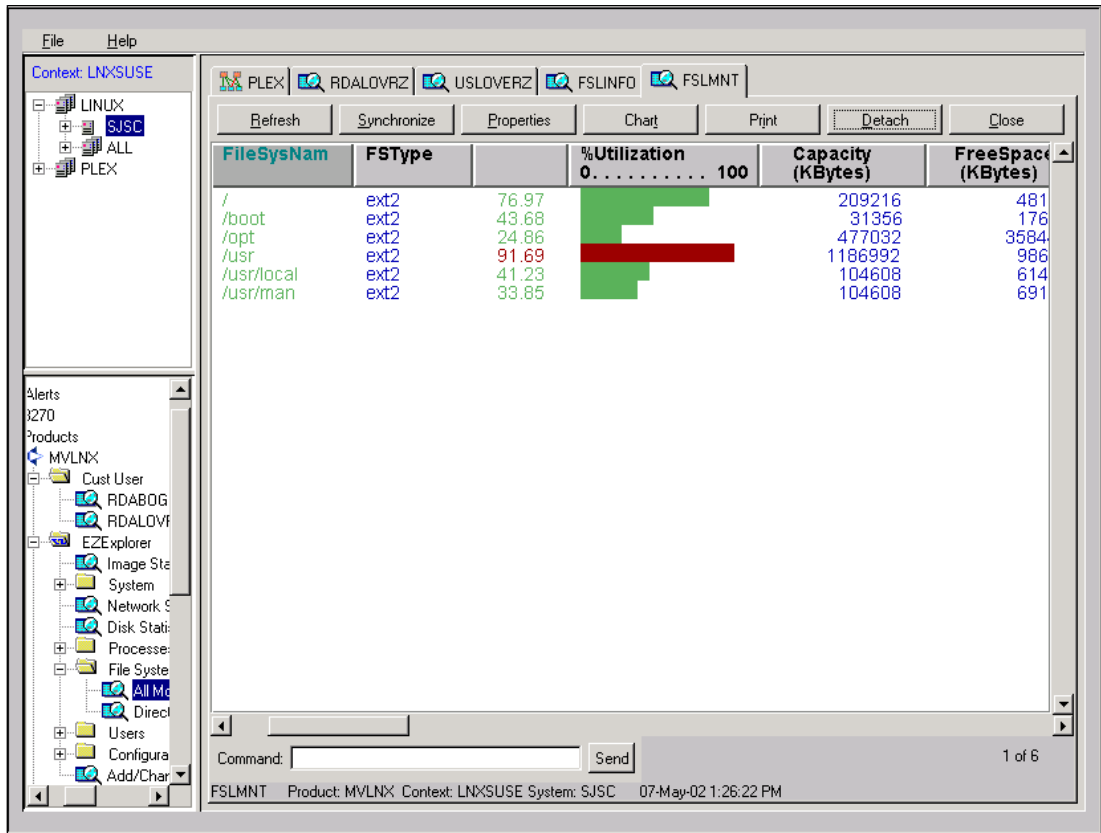


Figure 12-29 FSLMNT view

The All Mounted File Systems (FSLMNT) view displays file system utilization, user utilization, inode utilization and mount points. You can drill down to a subdirectory by right-clicking the file system name in the FSLMNT view, as shown in Figure 12-30.



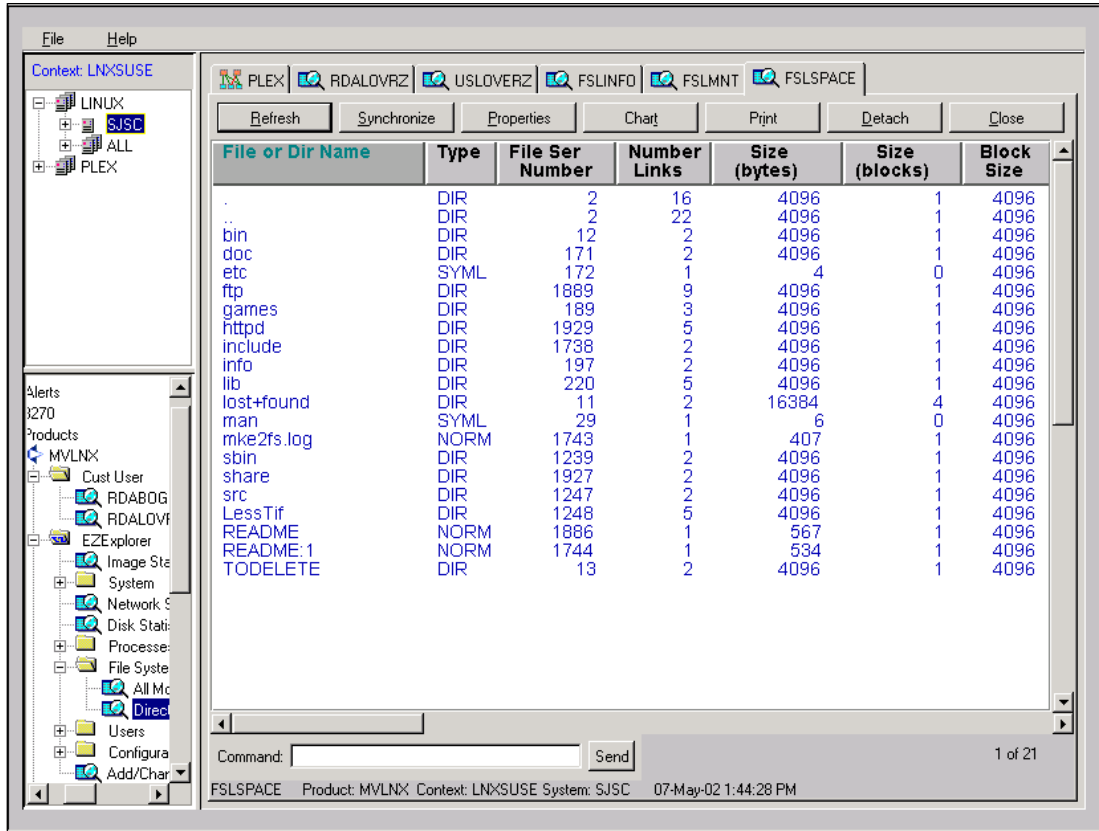


Figure 12-30 FSLSPACE view

The Directory Listing (FSLSPACE) view allows you to see directory information, space allocation, inodes used, group and owner settings, and permissions for files and directories. You can use this view to understand the cause of nearly full file systems or incorrectly set permissions for files and directories.

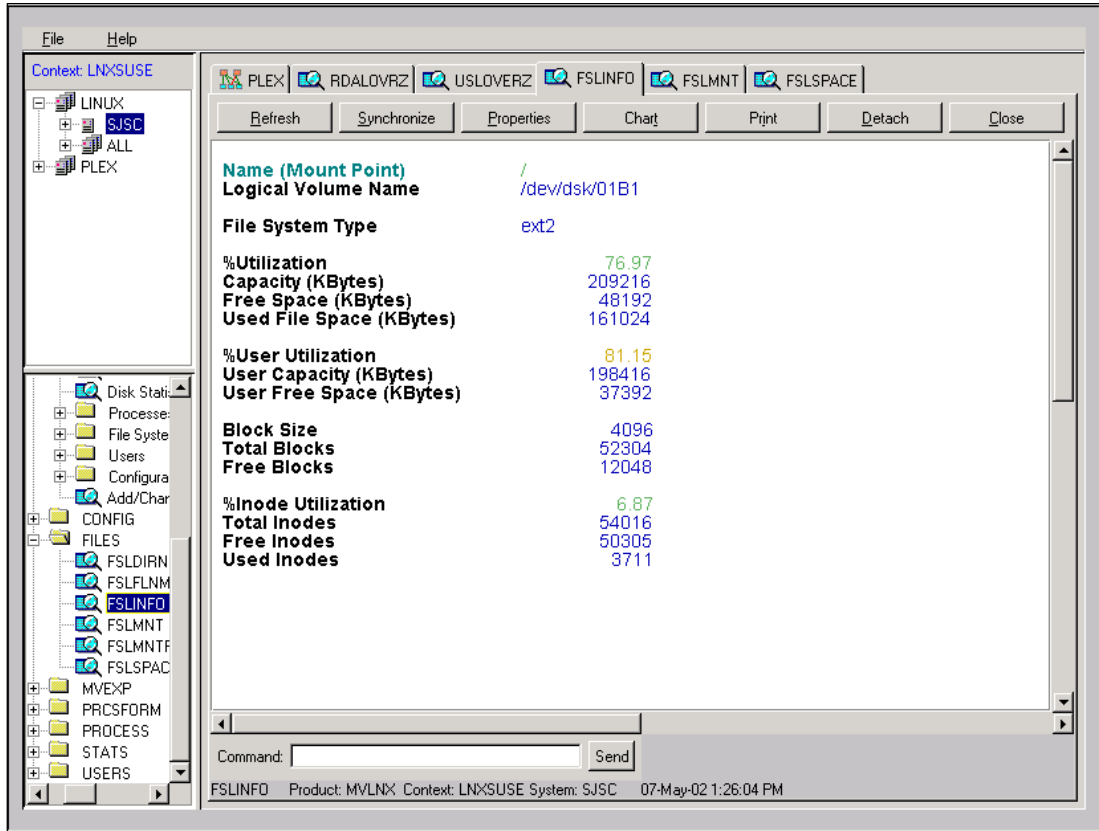


Figure 12-31 FSLINFO view

You can also display the File System Detail (FSLINFO) view. FSLINFO is a detailed view that provides data about a single mounted file system, including status, usage, mount point directory, and mount parameters over an interval.

**User information views**

MAINVIEW for Linux Servers has two informational views showing all users on the system, and a list of all users logged in.

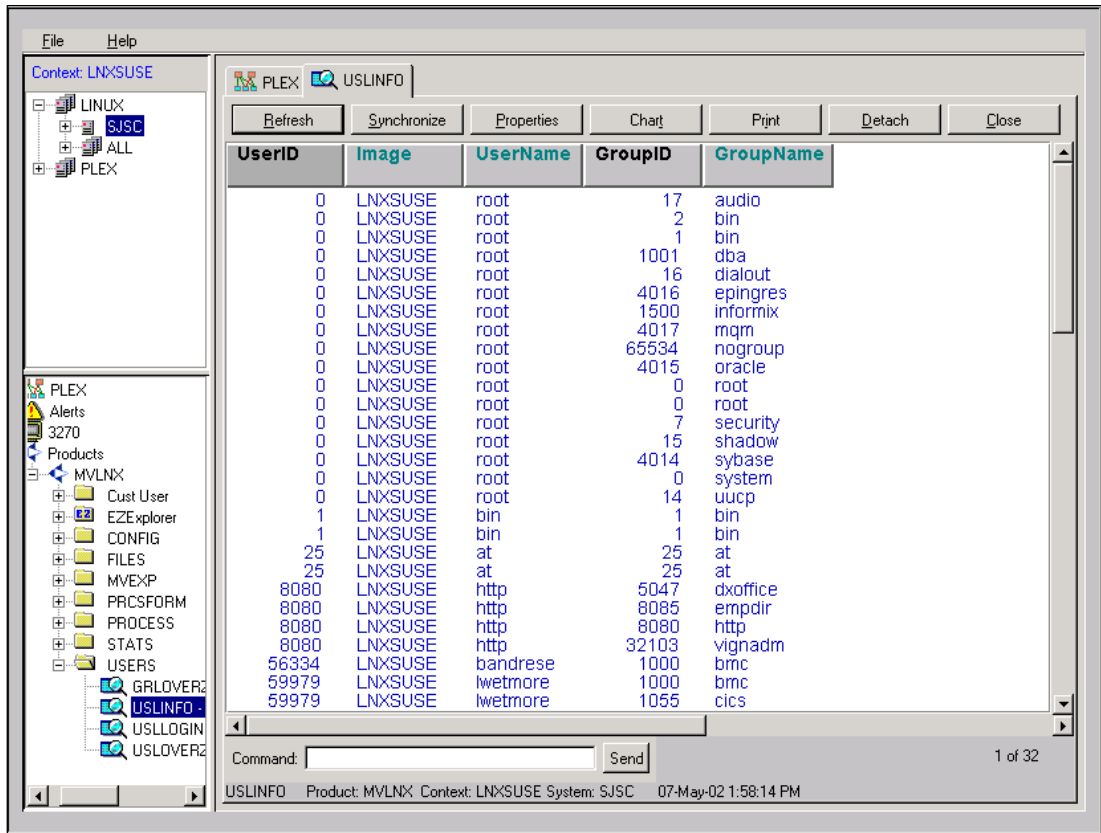


Figure 12-32 USLINFO view

The Userid Info (USLINFO) view displays every user running anything on this Linux system. It includes all groups the user processes run under, as well as the Group ID. From this view, you may want to look for any User ID or Group ID of 0, which gives root authority. You can use MAINVIEW Alarm Manager to automatically monitor for unauthorized root access. MAINVIEW Alarm Manager is discussed in section 12.1.3, “Health monitoring: Automation” on page 308.

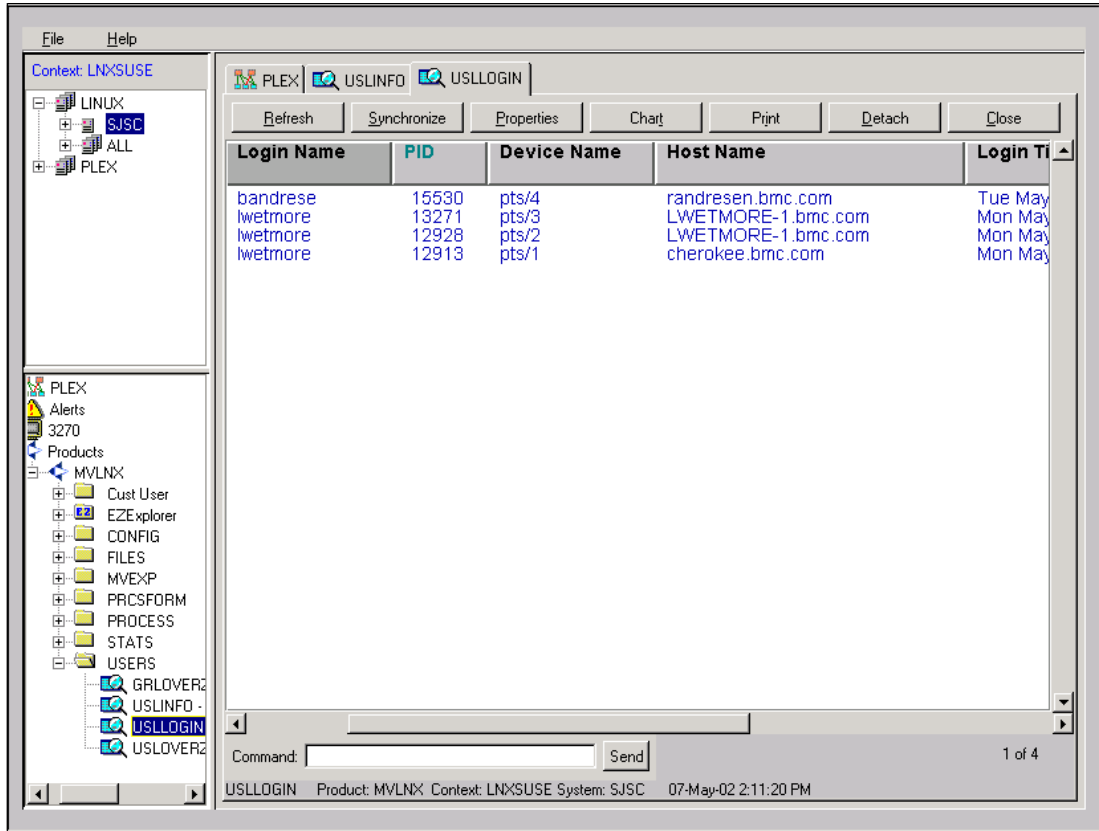


Figure 12-33 USLLOGIN view

The User Login (USLLOGIN) view displays logged-on users. This view correlates login name, process ID (number), controlling terminal device, and login time. USLLOGIN also reports remote host, if applicable. If hackers are detected on your Linux system, you can trace the remote host where the logon came. There are hyperlinks to the processes running for these login sessions.

### 12.1.3 Health monitoring: Automation

Setting up automation with MAINVIEW is essentially a two-step process. First, the event to drive the automation must be identified. This is done with a base technology component MAINVIEW Alarm Manager. Then the automation action needs to be defined through MAINVIEW AutoOPERATOR. If you currently are using another automation package, MAINVIEW Alarm Manager can be set up to route event messages to the z/OS console to allow other automation products to see the event.

MAINVIEW Alarm Manager serves two basic purposes. Besides identifying system events to drive automation, it also collects system warnings from multiple systems to a central display. It externalizes metrics from views from all MAINVIEW components: z/OS, CICS, DB2, IMS, IP, Linux, MQSeries, Unix Systems Services, VTAM and WebSphere. Each MAINVIEW component supplies a sample of alarm definitions. For example, MAINVIEW for Linux Servers comes with the sample alarms shown in Figure 12-34.

```

09MAY2002 11:56:14 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)MVALARM-----
COMMAND ==>
CURR WIN ==> 1 ALT WIN ==>
+W1 =ALMLST01=====SJC=====*(*) /LO EDIT)====MVALARM==D====
C AlarmDef View Product Context Description Update Updat
- Name-----
AVGCPU SYLUSE MVLNX SAMPCTXT MVLNX Average CPU exceptio 13:38:22 24SEP
AVGSWP SYLUSE MVLNX SAMPCTXT MVLNX Average Swap excepti 13:38:33 24SEP
FILTBL SYLUSE MVLNX SAMPCTXT MVLNX File Table exception 13:09:39 24SEP
FSYSIN FSLMNT MVLNX SAMPCTXT MVLNX File System I-Node e 13:17:43 24SEP
FSYSUT FSLMNT MVLNX SAMPCTXT MVLNX File System Util. ex 13:16:11 24SEP
INOTBL SYLUSE MVLNX SAMPCTXT MVLNX I-Node Table excepti 13:06:09 24SEP
PRCCPU PSLOVER MVLNX SAMPCTXT MVLNX Process CPU exceptio 13:13:52 24SEP
PRCTBL SYLUSE MVLNX SAMPCTXT MVLNX Process Table except 13:06:33 24SEP

```

Figure 12-34 ALMLST01 view

Each alarm shows its source view, product component, and a brief description. Every alarm definition belongs to an alarm group; these defaults are in group L0. If you want to automatically kill a process that is using over 90% of the CPU, complete the following steps:

1. Display the Process Resource Utilization (PSLUSE) view, which shows the percentage of CPU used by each process.
2. Enter the **SETALARM** command along with the group name to which the alarm should belong.
3. Position your cursor on the field you want the alarm to check, as shown in Figure 12-35.

```

09MAY2002 12:04:28 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==> setalarm l0
CURR WIN ==> 1 ALT WIN ==>
>W1 =PSLUSE=====LNXSUSE==*=====09MAY2002==12:02:46====MVLNX====D====6E
CMD PID Process Name %CPU ResSize I/O Rate VMemSz
(KB) -----
13190 top 6.82 552 0.00 1772
19847 bgscollect 5.07 2408 0.00 4160
4574 mmlsrv 0.53 5872 0.00 13196
132 syslogd 0.33 436 0.00 1300
12912 in.telnetd 0.25 140 0.00 1332
549 nscd 0.11 488 0.00 1524
12948 mmlsrv 0.06 4016 0.00 9256
547 nscd 0.06 488 0.00 1524
546 nscd 0.05 488 0.00 1524
12956 mmlsrv 0.05 3144 0.00 8692
548 nscd 0.04 488 0.00 1524
13105 mmlsrv 0.04 3276 0.00 8688
12982 mmlsrv 0.04 2952 0.00 8688
13097 mmlsrv 0.03 4252 0.00 8688
545 nscd 0.03 488 0.00 1524
12964 mmlsrv 0.03 2896 0.00 8688
543 nscd 0.02 488 0.00 1524

```

Figure 12-35 PSLUSE view

4. Press Enter to split the MAINVIEW screen, keeping the original view at the top and putting the Alarm Definition view below (shown in Figure 12-35 on page 309).

```

09MAY2002 12:12:40 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)-----
COMMAND ==> █ SCROLL ==> CSR
CURR WIN ==> 2 ALT WIN ==>
>W1 -PSLUSE-----LNXSUSE--*-----09MAY2002--12:02:46---MVLNX---D---68
CMD PID Process Name %CPU ResSize I/O Rate VMemSz
----- (KB) ----- (KB)
13190 top 6.82 552 0.00 1772
W2 =ALMADD01=====SJSC=====*(NEWMON00/L0 A MOD)====MVALARM==D====1
-Identification- -Definitions-
Alarm Def Name.. NEWMON00 Conditions... Defined
Group ID..... L0 Thresholds... 1
Description.... MVLNX PSLUSE exceptions on LNXSUSE t Filters..... NotDefined
Runtime Status.. NotFound Filters.... 0
Admin Status... NotInstall Where Text.. NotDefined
Library Status.. Enabled Expressions.. Defined
Passing PARMs... No Messages.... Default
Actions..... Defined
-----Source----- Frequency.... Defined
Product..... MVLNX
Context..... LNXSUSE -Last Update-
View..... PSLUSE Date..... 09MAY2002
Scope..... * Time..... 12:12:40
UserId..... RDIRDA

```

Figure 12-36 PSLUSE view and ALMADD01 view

5. From the Alarm Definition panel, change the default alarm name to something more meaningful, say *CPUHOG*, and update the description to describe why this alarm has been defined.
6. Down the right side of the view are a series of hyperlinks you can use to set how frequently MAINVIEW Alarm Manager will check this value, the alarm start and end message text, and whether this alarm should be displayed on the operator console.
7. Change the default message text to add the Linux system name, so the automation rule will know to which system to direct the command.
  - a. Click the **Messages** hyperlink.
  - b. Add the Linux system name to the start message.
  - c. The field name is PSRETARG; type **.V** at the end to indicate the value of that field, rather than the heading.

```

10MAY2002 10:08:15 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)MVALARM-----
COMMAND ==> █ SCROLL ==> CSR
CURR WIN ==> 1 ALT WIN ==>
W1 =ALMEDI01=ALMMSG01=SJSC=====*(CPUHOG /L0 E MOD)====MVALARM==D====1
Alarm... Message Id Message Text
Start.. CPUHOGSL0 &PSRETARG.V &PSREPID.H: &PSREPID.V &PSRECPUP.H > &
PSRECPUP.T
End.... CPUHOGEL0 &PSREPID.H: &PSREPID.V End of Alarm Condition
Alarm... Destination Messages ordered by:
View... Yes + (PSRECPUP,D)
Console No

```

Figure 12-37 ALMMSG01 view

- d. Press PF3 to return to the alarm definition view.
8. Enter an **INSTALL** command to tell MAINVIEW Alarm Manager to begin checking this alarm, then a **SAVE** command to save the alarm definition.
9. Once the alarm has been set up, you can select **Alerts and Alarms**, option **E** from the MAINVIEW Selection Menu, then select **2 Alarms**. You will see the Alarm Manager main menu.
10. Click the **Current Alarms** field. The Alarms view (shown in Figure 12-31) is displayed.

```

10MAY2002 11:54:15 ----- MAINVIEW WINDOW INTERFACE(R4.0.02)MVALARM-----
COMMAND ==> █
CURR WIN ==> 1          ALT WIN ==>
>W1 =ALARM=====SJSC=====10MAY2002==11:47:51====MVALARM==D====3
Alarm   Sev Hlp Message Id   Message Text
Time--- Ind -----
11:47:08 MAJ NO CPUHOGSL0   LNXSUSE PID: 4574 %CPU > 80
11:47:08 MAJ NO CPUHOGSL0   LNXSUSE PID: 22257 %CPU > 80
11:43:59 MAJ NO CPUHOGSL0   LNXSUSE PID: 22197 %CPU > 80

```

Figure 12-38 ALARM view

**Note:** The Alarm view has hyperlinks on the Alarm Message ID, which will bring up the original view in the product where the alarm was generated, and on the help indicator. You can define your own help text with your specific standards and procedures for each alarm.

Now that you have defined the alarm, you are ready to define the automation action for the event which triggered the alarm.

11. From the MAINVIEW Selection Menu, select option **A** (Automated Operations). This displays the MAINVIEW AutoOPERATOR menu.
12. From the MAINVIEW AutoOPERATOR menu, select option **8** (Automation).
13. Then select option **2** (Display/Modify Rules and Rule Sets). You will see a panel like the one shown in Figure 12-39.

```

BMC Software ----- Automation Control ----- AutoOPERATOR
COMMAND ==> █ TGT ==> SYSC
Primary commands: Add, Statshow, Cmdshow DATE --- 02/05/09
TIME --- 13:18:22

Automation Status ==> ACTIVE (Active, Inactive)
Automation Strategy ==> INDIVIDUAL (Individual, All, First, Qualified)
Honor MPF Suppression ==> NO (NO/YES)

Automation Library
LC CMDS --- (S)elect, (E)nable, (D)isable, (T)est, (SA)ve
(M)ove, (B)efore or (A)fter, (F)ilter Criteria

LC Rule-Set Status Rules Fired Filtered Date Time Strategy
---
RULJRNL ENABLED 197 84 232 e 09-MAY-02 00:14:50 FIRST
AADRULBC ENABLED 2 0 56 e 09-MAY-02 00:14:51 FIRST
AADRULBS ENABLED 530 0 109,285 09-MAY-02 00:14:56 FIRST
RULINUX ENABLED 0 0 0 09-MAY-02 13:18:22 FIRST
AADRULBA DISABLED N/A N/A N/A N/A N/A
AADRULBB DISABLED N/A N/A N/A N/A N/A
AADRULBD DISABLED N/A N/A N/A N/A N/A
AADRULBE DISABLED N/A N/A N/A N/A N/A
AADRULBF DISABLED N/A N/A N/A N/A N/A
AADRULBG DISABLED N/A N/A N/A N/A N/A
AADRULBH DISABLED N/A N/A N/A N/A N/A
AADRULBP DISABLED N/A N/A N/A N/A N/A

```

Figure 12-39 Automation Control

Rules are organized in Rule Sets, which are members of the MAINVIEW AutoOPERATOR UBBPARAM data set. A Rule Set is a way to organize groups of related rules.

14. Add a new rule set with an **ADD** command, then select that rule set by typing an **S** to display the Rule Set Overview panel.
15. Enter another **ADD** command to access the Rule Definition panel (shown in Figure 12-40).

```

BMC Software ----- Rule Processor Detail Control ----- AutoOPERATOR
COMMAND ==> █ TGT --- SYSC
TIME --- 13:27:02
DATE --- 02/05/09

The following options are displayed in sequence, or may
be selected by entering the two-character code

S1 - Selection Criteria A1 - Action Specification
SV - Variable Dependencies AA - Alert Action(s) I
AD - Alert Action(s) II

Rule ID ==> CPUHOGS
Event Type ==> ALRM Type of event ( ? for list)
Initial Mode ==> ENABLED (ENABLED/DISABLED/TEST)

Criteria match rate threshold:
If matched ==> (Maximum # times matched within INTERVAL, 0-100)
in seconds ==> (Interval length, 1-99999 seconds)
then status ==> (SUSPEND, DISABLE, NOACTION)

Application information:
Group ==> LINUX Function ==> Code ==>
Author ==> RDIRDA Description ==> CATCH PROCESS HOGS
Last Modified by on at

```

Figure 12-40 Rule Processor Detail Control

16. Complete the fields as shown in Figure 12-40. The event type is L for alarm. The application information at the bottom of the panel is mainly to document the rule you are creating.
17. Press Enter to display the Selection Criteria panel (shown in Figure 12-41).



```

BMC Software ----- Selection Criteria - ALRM ----- AutoOPERATOR
COMMAND ==> TGT --- SYSC

      Rule-set == RULINUX           Rule-id == CPUHOGS

Text Description:
Alarm ID ==> CPUHOGSL0           Alarm Identifier
Alarm Text (Enter Below):

Context ==> █                   Alarm Context
Scope ==>                       Alarm Scope
Userid ==>                       Monitor Identifier
Queue ==>                       Alarm Queue
Priority ==> MAJOR               Alarm Priority
Type ==> START                  Alarm Start/Stop Event

```

Figure 12-41 Alarm selection criteria

The Selection Criteria panel identifies the event for which the rule is to be started.

18. Enter CPUHOGSL0 for alarm ID, MAJOR for priority and START for type. This rule will start when a major alarm named CPUHOGSL0 condition starts.
19. Press Enter and you will see a panel specifying more variable checking for this rule. Do not enter anything on that panel; press Enter again to see the Action Specification panel (shown in Figure 12-42).

```

BMC Software ----- Action Specification - ALRM ----- AutoOPERATOR
COMMAND ==> █ TGT --- SYSC

      Rule-set == RULINUX           Rule-id == CPUHOGS

Automation Actions:
EXEC Name/Parms ==> LNXPROC &WORD2 &WORD4
Send (TSO IDs) ==>
Cmd (Type MVS ) ==>

Set Variable ==> ==>
Notify ==> Outboard Pager ID
Info ==>

DOM Id ==> Delete Operator Message
Issue WTO Msg ==>

Press ENTER to continue, END return to Detail Control, CANCEL to cancel changes

```

Figure 12-42 Alarm Action specification

20. On the action specification panel, enter an EXEC name to be initiated when this rule starts. The parameters on the EXEC are WORD2, which is the Linux system name, and WORD4, which is the Process ID. You may also decide to notify a TSO user if you wish.
21. Add the exec called LNXPROC.

```

BMC Software ----- Action Specification - ALRM ----- AutoOPERATOR
COMMAND ==> █ TGT --- SYSC

          Rule-set == RULINUX                Rule-id == CPUHOGS

Automation Actions:
EXEC Name/Parms ==> LNXPROC &WORD2 &WORD4
Send (TSO IDs)   ==>
Cmd (Type MVS ) ==>

Set Variable     ==>                               ==>
Notify          ==>                               Outboard Pager ID
Info           ==>

DOM Id          ==>                               Delete Operator Message
Issue WTO Msg   ==>

Press ENTER to continue, END return to Detail Control, CANCEL to cancel changes

```

Figure 12-43 Alarm Action Specification

This is a CLIST; you may also write Rexx execs. The Linux system and Process ID are passed. The **write** commands will log the actions to the AutoOPERATOR journal. The **rsh** command will require a SYSPRINT DD statement to be present in the AutoOPERATOR started task. The journal messages look like the examples shown in Figure 12-44.

```

BMC Software ----- Log Display ----- General services
COMMAND ==> █ TGT ==> SYSC
Line 477 Log #2 Status INPUT Time 13:43:21 INTV==> 5
13:38:14 AU0013I RULES - RULINUX ENABLED
13:39:01 EM0025I FOLLOWING MSG ISSUED FOR EXEC .. LNXPROC ..
13:39:01 ABOUT TO ISSUE KILL FOR 13190 ON LNXSUSE
13:39:02 KILL COMMAND ISSUED
13:40:55 EM0025I FOLLOWING MSG ISSUED FOR EXEC .. LNXPROC ..
13:40:55 ABOUT TO ISSUE KILL FOR 22516 ON LNXSUSE
13:40:56 KILL COMMAND ISSUED
***** END OF LOG *****

```

Figure 12-44 Log display

The rsh messages can be found in the SYSPRINT DD statement in the AutoOPERATOR address space. The rule you have just built is an example of how to automate Linux commands. You may want to expand the functions automated by passing the severity of the alarm and issuing different Linux commands based on different alarm severities. For example, if a process using 80% of the CPU is a major alarm, you may decide to issue a Linux nice command, this command reduces the priority that a process runs under.

Many system configuration values may be altered on a running Linux system by changing the value of the parameter in the /proc file system (shown in Example 12-1).

*Example 12-1 /proc file system parameter*

```

echo 8192 >
/proc/sys/fs/file-max

```

This will change the value until the Linux system is restarted, when the value will revert to what is defined in the kernel. This gives you another set of possible automation commands, bumping these kernel values as the workload increases in the Linux systems. The goal is to prevent system outages while logging these changes so the kernel can be adjusted to permanently correct inappropriately set values.

The example used the rsh command to interface with the Linux system. This may raise security concerns among Linux systems administrators. If the Linux system is behind the corporate firewall, then the rsh interface may be acceptable, but if the Linux system is exposed to the outside world a more secure method of issuing commands may be required. In this case the PATROL for Linux automation may be a better solution because the automation can be handled by the PATROL agent running on the Linux system.





# System management using PATROL

This chapter describes how to use PATROL for Linux Enterprise Server and PATROL Internet Service Manager to accomplish a variety of system management tasks.

## 13.1 Operations using PATROL for Linux Enterprise Server

PATROL for Linux Enterprise Server automatically discovers and displays a graphic presentation of your Linux operating system components. You can use the PATROL Classic Console on a Microsoft Windows system to quickly check the performance of your critical system resources. For additional information on using the PATROL Console and PATROL for Linux Enterprise Server, see the PATROL for Linux Enterprise Server Documentation CD and the PATROL Console online help.

### 13.1.1 Availability management

The PATROL for Linux Enterprise Server product has over 200 parameters that provide statistical information about system performance and resources. The parameters identified in Table 13-1 monitor the most critical resources on your system. The help topics for these parameters provide more details and recommendations.

**Note:** The Linux kernel on zSeries does not support all the commands or provide all of the statistics that the PATROL KM for UNIX requires. Specifically, on zSeries:

- ▶ The `nfsstat` command is not available.
- ▶ The kernel returns empty proc entries for `disk` and `swap`; thus, the DISK and SWAP parameters in PATROL display values of zero.

Using these parameters, you can monitor critical system resources and take appropriate action before a problem impacts the system workload.

Table 13-1 Critical system resources

Application class (System resource)	Critical system resource parameter	Description
CPU	CPUCpuUtil	Percentage of CPU utilization, which is calculated by subtracting CPU idle time from 100.
	CPULoad	One-minute load average from the uptime command. Load average is the average number of processes in the kernel's run queue during an interval (one minute in this case).
	CPURunQSize	Number of processes in the run queue (RunQ).
	CPUSysTime	Percentage of CPU time spent in system mode doing system tasks, including the CPU resources consumed by calls to kernel routines.
	CPUIo	Percentage of time that the CPU spends waiting for input and output operations.

Application class (System resource)	Critical system resource parameter	Description
DISK	DSKAvgQueue	Average number of disk I/O requests in the queue and is measured only when the queue is occupied. A high number indicates that system throughput is probably slowing down because of the number of I/O requests for this disk.
	DSKBps	Number of blocks read from, or written to, the device per second and indicates the work load for the device. A high number indicates that system throughput is probably slowing down because of the number of I/O requests for this disk.
	DSKMbps	Average disk seek time for the device; indicates the speed of the device. A gradual increase in average disk seek time usually indicates data fragmentation. A heavily fragmented disk can have a negative impact on system throughput.
	DSKPercentBusy	Percentage of time that the device is busy servicing a transfer request; indicates the workload for the device. A high number indicates that system throughput is probably slowing down because of the number of I/O requests for this disk.
	DSKReadWrite	Number of read and write pages read/written to the device per second; indicates the workload for the device. A high number indicates that system throughput is probably slowing down because of the number of I/O requests for this disk.
	DSKSps	Number of disk seeks per second; indicates the work load of the device. A gradual increase in the number disk seeks usually indicates data fragmentation. A heavily fragmented disk can have a negative impact on system throughput.
FILESYSTEM	FSAvailableSpace	Amount of available space for this FILESYSTEM instance. This parameter is critical on the root volume.
	FSFreelnodes	Number of I-nodes available. An I-node is a data structure that maintains information about each file. Measuring I-nodes is critical because once all of your I-nodes have been used, the file system will not accept any more files regardless of how much disk space is available.
KERNEL	KERLgFail	Number of large memory pool requests that were not satisfied. A high number can indicate memory fragmentation or a virtual memory shortage.
	KERLockUsedPercent	Percentage of used kernel lock slots.
	KEROvzFail	Number of requests for oversized memory that could not be satisfied. Oversized memory is allocated dynamically. A high number can indicate memory fragmentation or a virtual memory shortage.
	KERProcUsedPercent	Percentage of used kernel process slots; monitors the process table utilization. Each table entry represents an active process; the number of entries available depends on the number of terminal lines available and the number of processes spawned by each user.
	KERSmlFail	Number of small memory requests that failed. A small memory request is a request for less than 256 bytes. A high number can indicate memory fragmentation or a virtual memory shortage.

Application class (System resource)	Critical system resource parameter	Description
LOG	LOGFileSize	Size of the file you are monitoring. This is an important parameter to monitor because log files usually grow continuously and can often grow large enough to cause problems before you remember to purge old log data.
MEMORY	MEMPageOut	Number of 1-KB pages, or regions, of memory paged out of physical memory to disk. Memory pages are paged out when processes running on the system require more physical memory than is available. Occasional paging is normal; however, frequent paging degrades system performance.
	MEMPFault	Number of detected page protection faults that caused pages to be copied. A page fault occurs when your system is attempting to execute a portion of a process that has been paged out of memory. Excessive page faults usually indicate a memory shortage.
	MEMRegionsOut	Number of 1-KB pages, or regions, of memory paged out of physical memory to disk. Memory pages are paged out when processes running on the system require more physical memory than is available. Occasional paging is normal; however, frequent paging degrades system performance.
NETWORK	NETCollisionPrc	Percentage of output attempts by the selected host that resulted in an Ethernet collision. Collisions are an indicator of network load. A small percentage of collisions are normal, but a high percentage indicates an overloaded network segment.
	NETInErrPrc	Percentage of incoming data packets that contain packet format errors. A large number of incoming data packets with format errors usually indicates faulty hardware on the network. Another possibility is that your system's device driver cannot receive packets fast enough.
	NETOutErrPrc	Percentage of outgoing data packets that contain packet format errors. A large number of outgoing packet format errors indicates a faulty local network interface. The fault could be in the system's network controller, the Ethernet drop, or something else between the main Ethernet cable and your CPU.
NFS	NFSCFsStat	Percentage of all NFS client calls made to retrieve file attributes or file statistics (since the last sample). Programs that check for the existence of a file tend to increase the value of this parameter.
	NFSCGetAttr	Percentage of all NFS client calls that are requests to get file attributes. Programs that check for the existence of a file tend to increase the value of this parameter.
	NFSCLookUp	Percentage of all NFS client calls made to look up directory paths. A high percentage indicates a possible performance bottleneck in traversing NFS mounted directories.
	NFSCReadLink	Percentage of NFS client calls made to read symbolic links. A high percentage indicates that performance could be suffering. A high percentage is often caused by a high number of symbolic links.
	NFSCRpcRetrans	Number of NFS client RPC requests that had to be re-transmitted. Calls are re-transmitted because no response was received from the NFS server within the time-out period. An NFS client experiencing poor server response will have a large number of re-transmitted calls



Application class (System resource)	Critical system resource parameter	Description
PATROLAGENT	PAWorkRateExecsMin	Number of operating system process executions performed per minute by the PATROL Agent. This parameter gets this information from the built-in PATROL Agent namespace variable, /execsPerMin.
PRINTER	PRNQLength	Number of print jobs waiting in the print queue of the selected printer. This parameter can help you keep track of your system's print jobs and printers.
PROCESS	PROCNoZombies	Total number of zombie processes currently running. Zombie processes take up physical memory and process table slots. This can be a critical problem in systems that do not dynamically increase the number of process table slots. Once the process table is full, your system cannot start any new processes, and running processes cannot spawn new processes.
	PROCTopProcs	Processes (up to ten) using the highest percentage of CPU time during the sample period.
SWAP	SWPTotSwapFreeSpace	Total amount of free swap space. Once your system is out of swap space, it cannot start any new processes, and running processes cannot spawn new processes.
	SWPTotSwapUsedPercent	Percentage of the total system-wide swap space in use. Once your system is out of swap space, it cannot start any new processes, and running processes cannot spawn new processes.

### 13.1.2 Health monitoring with PATROL for Linux Enterprise Server

The PATROL for Linux Enterprise Server v1.1.00 product provides the HEALTH AT A GLANCE (HAAG) application. Using this application, you can monitor:

- ▶ Overall CPU usage of your system
- ▶ Overall file system usage of your system
- ▶ Overall virtual memory of your system

The HEALTH AT A GLANCE application class contains parameters on system-wide CPU, file system, and virtual memory usage.

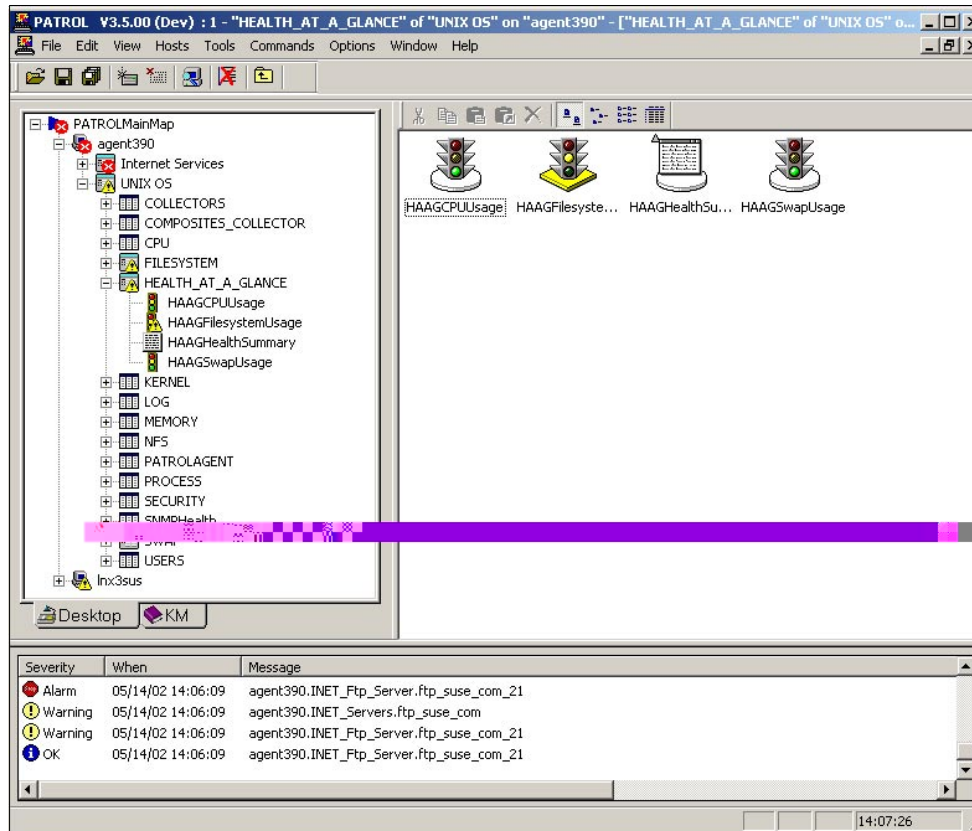


Figure 13-1 HEALTH AT A GLANCE Application parameters

By reviewing the information provided by the HAAG parameters (shown in Figure 13-1), you should be able to determine if your system is healthy or not.

### Viewing the overall CPU, file system, and swap usage

You can use HAAG to view the system's overall statistics on CPU, file system, and swap space usage; and determine if the system is generally healthy.

To do this, use the following steps:

1. Access the HEALTH AT A GLANCE application class so that you can view its parameters.
2. Open the parameter that records the type of system information you want to view:
  - HAAGCpuUsage displays overall CPU usage statistics.
  - HAAGFilesystemUsage displays overall file system usage.
  - HAAGSwapUsage displays overall swap space usage statistics.

PATROL displays a graph that shows the requested information over time.

### Viewing the overall health

You can also view a textual summary of the system's general health. The HAAGHealthSummary parameter reports on the system's overall health based on the other HAAG parameters: HAAGCPUUsage, HAAGFilesystemUsage, and HAAGSwapUsage.

To view the system's overall health, use the following steps:

1. Access the HEALTH AT A GLANCE application class so that you can view its parameters.

- Open the HAAGHealthSummary parameter. PATROL displays the information in a text parameter. Your result should resemble Figure 13-2.

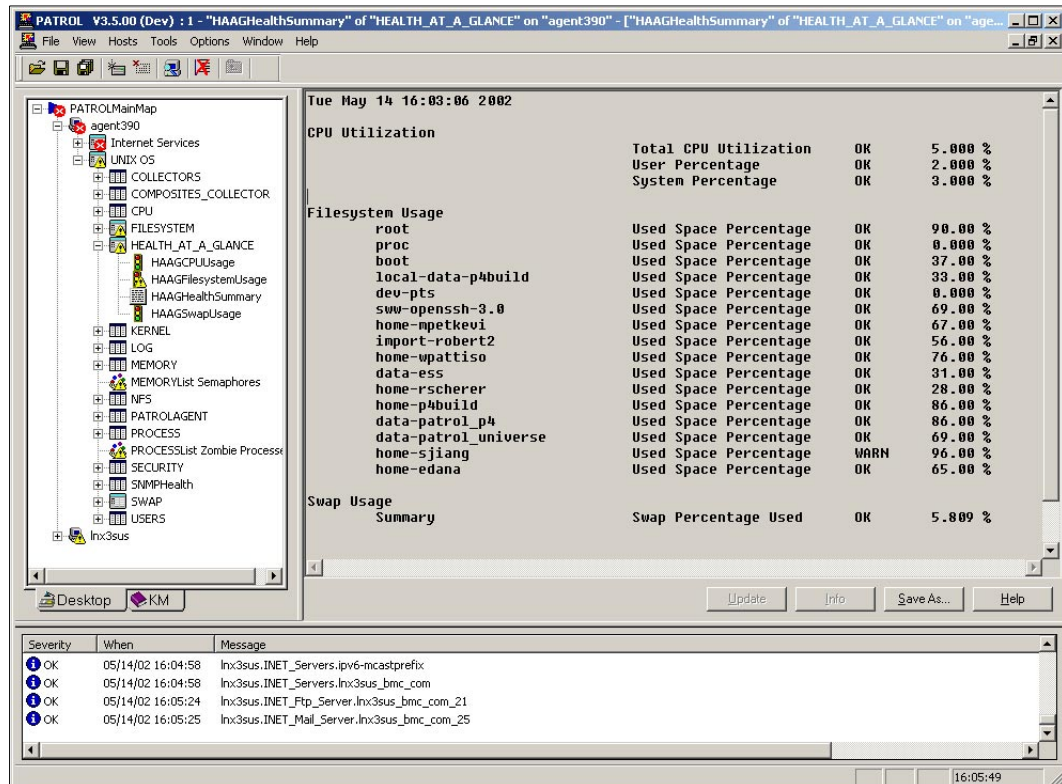


Figure 13-2 HEALTH AT A GLANCE Summary report

The HEALTH AT A GLANCE Summary report shows that the overall health of our SuSE Linux system is OK. The total CPU utilization is five percent and swap usage is not quite six percent. Only one file system has any problems: file system home-sjiang is at 96 percent usage. We may want to contact this user and ask them to erase any unneeded files from their home directory.

### 13.1.3 Automation

#### Automating file system cleanup

This section describes how to set up the File System Cleanup recovery action, which removes files once the percentage of the file system currently in use, measured by the FSCapacity parameter, exceeds the threshold established in either Alarm1 or Alarm2. This recovery action attempts to remove core dump files; if the file system is /tmp, or /tmp is mounted off this file system, the recovery action attempts to remove files from /tmp that are more than 30 days old.

#### Before you begin

There are several prerequisites that must be met before you can automate file system cleanup:

- ▶ You must be root to run this command.
- ▶ For a given FILESYSTEM instance, the FSCapacity parameter must be active.

This recovery action applies to all file system types except NFS, CD-ROM, and proc.

Use the following steps to set up Automatic File System Cleanup.

1. Access the Computer application menu.
2. Select **KM Commands -> Configure Recovery Actions**. PATROL displays the Registered Recovery Actions dialog, which lists all available recovery actions.

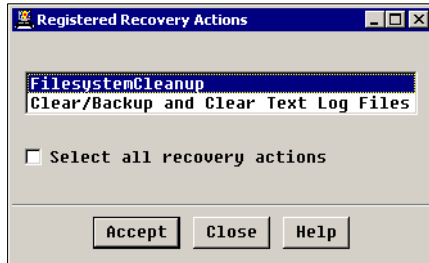


Figure 13-3 Registered Recovery Actions dialog

3. Select the FilesystemCleanup recovery action and click **Accept**. PATROL displays the Recovery Actions Instance dialog. This recovery action applies to all FILESYSTEM instances.

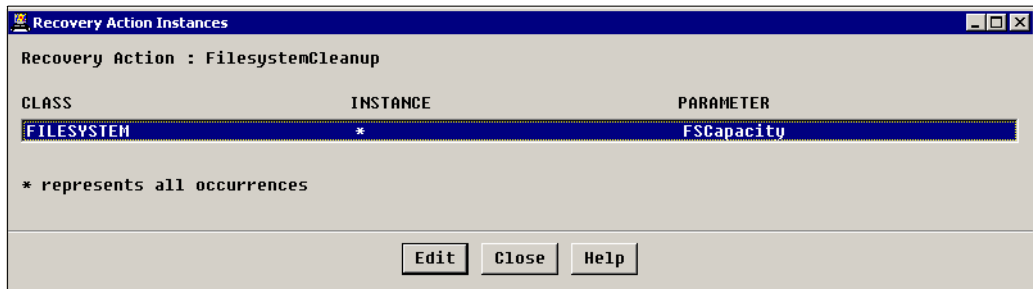


Figure 13-4 Recovery Action Instances dialog

4. Select the only item in the list and click **Edit**. PATROL displays the Edit Recovery Action dialog.

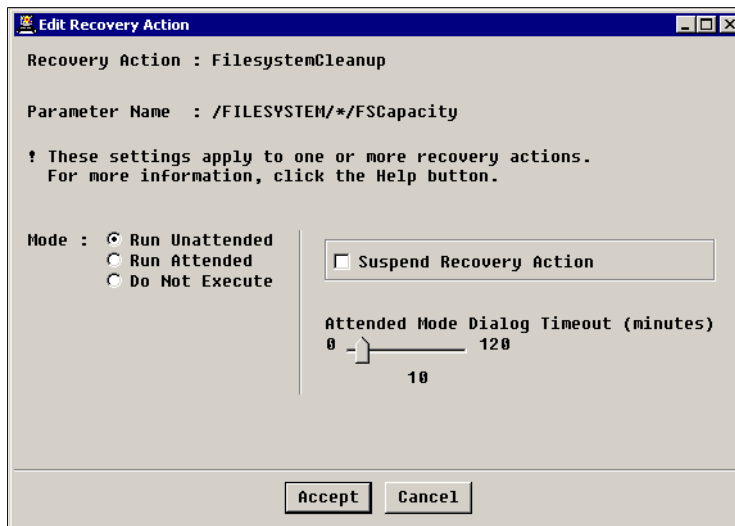


Figure 13-5 Edit Recovery Action dialog

5. Select the mode of the recovery action by clicking the corresponding radio button.

- **Run Unattended** runs the recovery action automatically, without prompting you.
- **Run Attended** prompts you before running the recovery action, and does nothing if you do not respond within the user-defined timeout period.
- **Do Not Execute** does not perform the recovery action; this mode is the default setting.

**Note:** If you select Run Attended, use the slide bar to set the amount of time PATROL waits for user input before it cancels the recovery action.

6. Click **Accept**. PATROL closes the Edit Recovery Actions dialog and displays the Recovery Action Instances dialog.
7. Click **Close**. PATROL closes the Recovery Action Instances dialog and displays the Registered Recovery Actions dialog.
8. Click **Close**. PATROL closes the Registered Recovery Actions dialog and enables the recovery action based on the options that you chose.

### Automating text log file size reduction for individual logs

This section describes how to set up the Clear/Backup and Clear Text Log Files recovery action, which can automatically back up and empty selected log files stored in text format.

To set up automatic text log file size reduction for individual logs, do the following:

1. Access the LOG application menu.
2. Choose **KM Commands -> Edit List of Monitored Files**. PATROL displays the Log Files dialog.

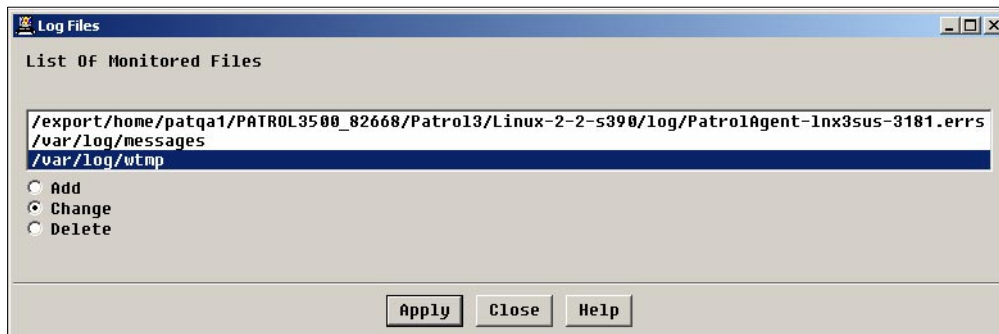


Figure 13-6 Log Files dialog

3. Select the log file you want to change the recovery options for, click the **Change** radio button, then click **Apply** to continue. PATROL displays the Log File dialog for the selected log file.

**Note:** If the log file you want to work with is not already being monitored, select **ADD** and then click **Apply** to continue. PATROL displays the Log File (New) dialog. See “Use the following steps to set up monitoring of a log file:” on page 330 for additional information.

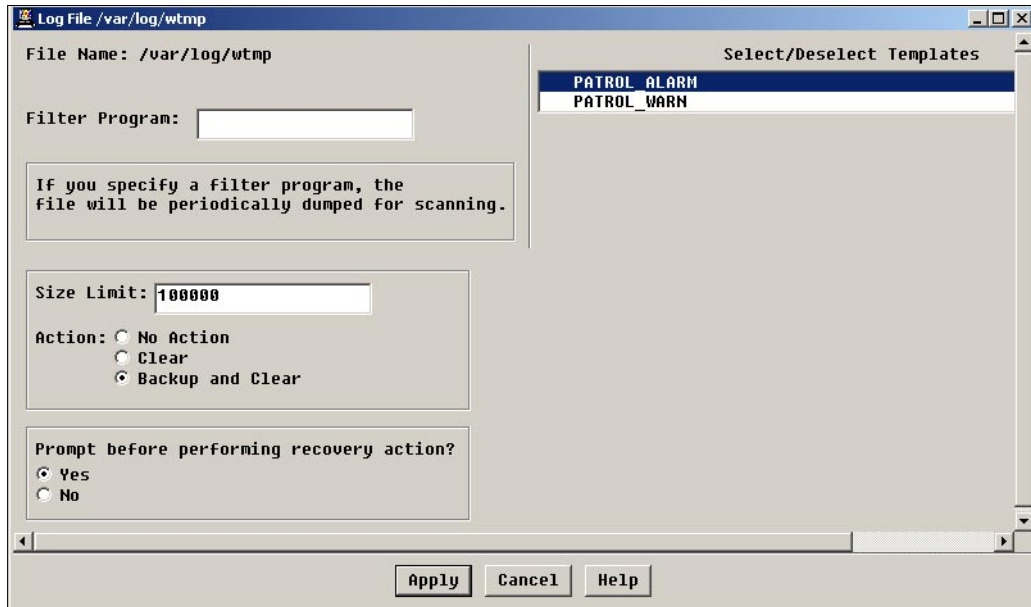


Figure 13-7 Log File dialog

4. To associate a log file search template with the log file, select (or deselect) a template in the Select/Deselect Templates field. Templates marked with an asterisk (\*) are currently applied to the log file entry you are editing.
5. If you are monitoring a binary log file, enter or change the name of the program that translates the file to ASCII in the Filter Program field.
6. Set the file size limit in the Size Limit field.
7. Establish a recovery action for PATROL to take when the log file reaches the specified size limit.
  - **No Action** continues monitoring the log file but does not attempt to reduce its size.
  - **Clear** reduces the log file to 0 MB by deleting all the messages in the log file when the file reaches the size limit.
  - **Clear and Backup** writes all the messages in the log file at the time the file reached the size limit to a backup file, and then reduces the log file to 0 MB. The backup file is written to the same directory, with an incremental number appended to the log file name. For example, the first time that the error\_log.txt reaches its size limit, PATROL creates a backup file named error\_log.txt1. The next time that it reaches its limit, PATROL creates a backup file named error\_log.txt2, and so on.

**Note:** Move the backup files to another location. The PATROL recovery action checks to make sure that the backup file name is not already in use. If hundreds or even thousands of backup files exist in the log directory, PATROL may take some time to complete this recovery action.

8. Choose whether PATROL requires you to approve the LOG recovery action by clicking the appropriate radio button. If you select **No**, PATROL will run the recovery action without any intervention required from you. If you select **Yes**, PATROL displays a dialog similar to Figure 13-8 before running the recovery action.

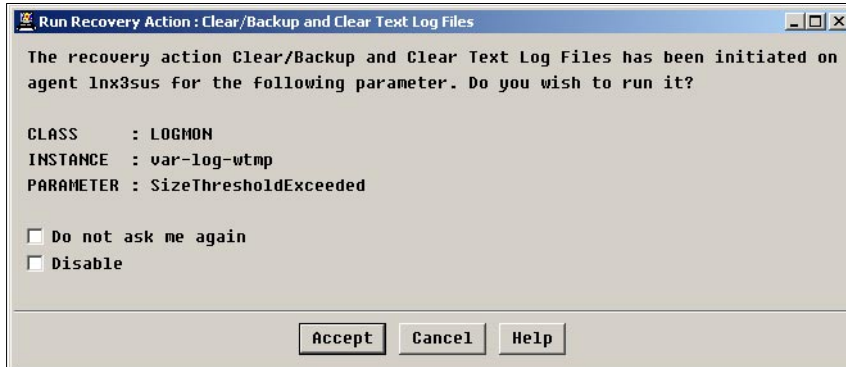


Figure 13-8 Run Recovery Action dialog

- Click **Apply**. PATROL displays a confirmation dialog that lists the log file name and path.

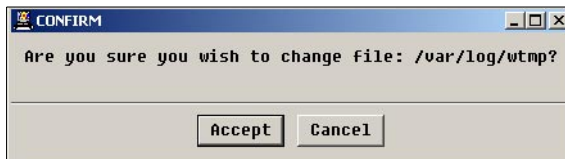


Figure 13-9 Confirm dialog

- Click **Accept** if you want to accept the changes you made to monitoring for the log file. Otherwise click **Cancel**. PATROL displays the Log Files dialog.
- Click **Close** to close the Log Files dialog.

### Changing recovery action mode for individual logs

If you need to change the recovery action mode for a monitored log file, do the following:

- Access the Computer application menu.
- Select **KM Commands -> Configure Recovery Actions**. PATROL displays the Registered Recovery Actions dialog, which lists all available recovery actions.

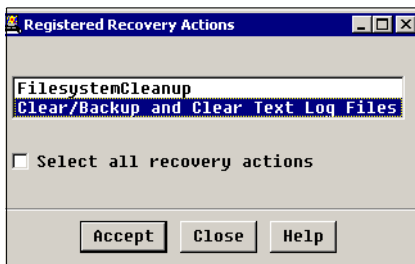


Figure 13-10 Registered Recovery Actions dialog

- Select the Clear/Backup and Clear Text Log Files recovery action and click **Accept**. PATROL displays the Recovery Actions Instance dialog.

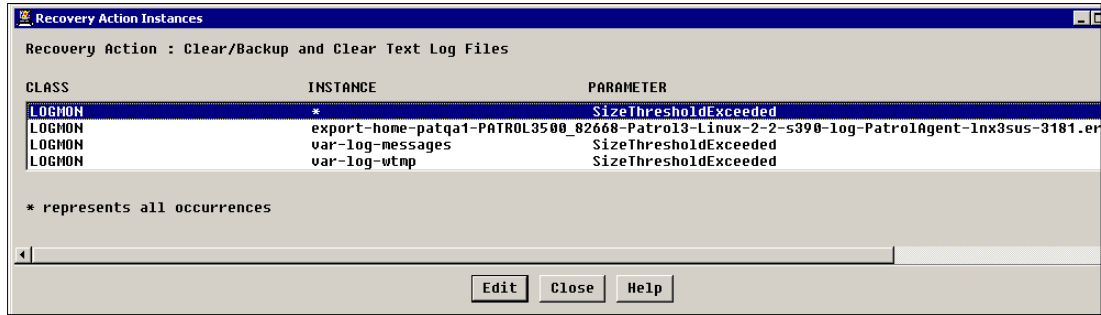


Figure 13-11 Recovery Action Instances dialog

4. Select a log file in the list and click **Edit**. PATROL displays the Edit Recovery Action dialog.

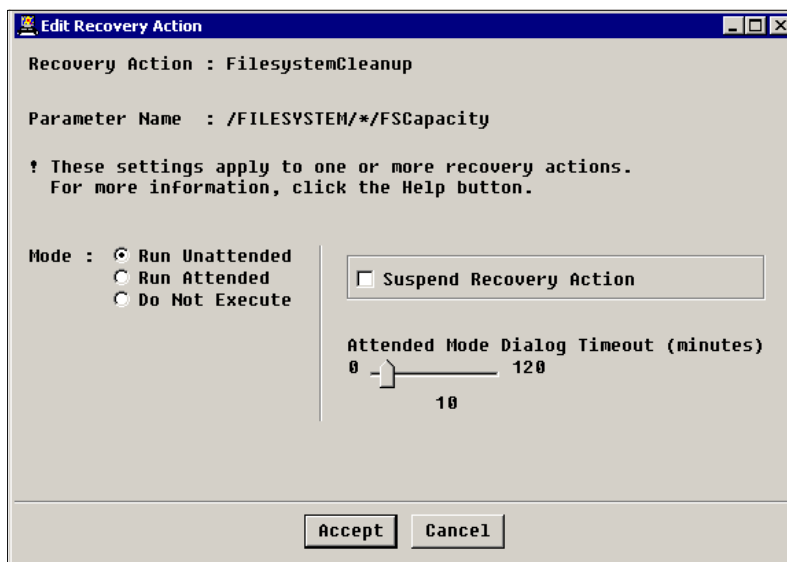


Figure 13-12 Edit Recovery Action dialog

5. Select the mode of the recovery action by clicking the corresponding radio button.
  - **Run Unattended** runs the recovery action automatically, without prompting you.
  - **Run Attended** prompts you before running the recovery action, and does nothing if you do not respond within the user-defined timeout period.
  - **Do Not Execute** does not perform the recovery action; this mode is the default setting.

**Note:** If you select Run Attended, use the slide bar to set the amount of time PATROL waits for user input before it cancels the recovery action.

6. Click **Accept**. PATROL closes the Edit Recovery Actions dialog and displays the Recovery Action Instances dialog.
7. Click **Close**. PATROL closes the Recovery Action Instances dialog and displays the Registered Recovery Actions dialog.
8. Click **Close**. PATROL closes the Registered Recovery Actions dialog and enables the recovery action based on the options that you chose.



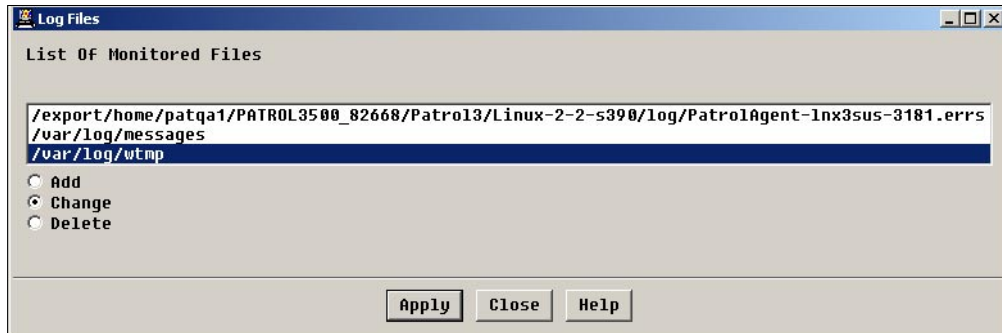


Figure 13-13 Log Files dialog

## 13.2 Data management using PATROL for Linux Enterprise Server

PATROL for Linux Enterprise Server provides applications that monitor and manage file systems, log files, and printers on your SuSE Linux system. For additional information on these applications, see the PATROL Knowledge Module for UNIX User Guide or the PATROL for Linux Enterprise Server online help.

### 13.2.1 Monitoring logs

The PATROL for Linux Enterprise Server product provides the ability to monitor the size, growth rate, and content of log files on your system. It monitors log file size and growth, as well as content, through the LOG, LOGMON, and LOGTEMP applications. LOG application classes allows you to:

- ▶ Monitor the file size and growth rate of selected log files
- ▶ Provide automated recovery actions when a log file exceeds acceptable size or growth rate
- ▶ Define log file searches for certain messages or types of messages
- ▶ Monitor selected log file messages based upon defined searches

The LOG application supports logs in both text and binary format. A field in the Edit List of Monitored Files dialog enables you to select a filter for log files stored in binary format.

The LOG application class contains the instances of logs that are being monitored. Each log instance contains the standard growth rate and file size parameters. If the log is being monitored for certain messages and has a template associated with it, a LOGMON icon is also present. Within the LOGMON icon are several parameters with information about the success of searches for the designated messages.

#### Selecting log files to monitor

This section describes how to set up a log file for monitoring. It allows you to designate any log file on your system.

If you want to monitor specific messages or types of messages sent to a log file, you must define a search. See the *PATROL Knowledge Module for UNIX User Guide* for information on defining a search.

Once you specify a log file, PATROL monitors the log for:

- ▶ File Size - stored in the LOGFileSize parameter
- ▶ Growth Rate - stored in the LOGGrowthRate parameter

The LOG application is set up to automatically monitor the following log files:

- ▶ /etc/wtmp filtered by /usr/lib/acct/fwtmp
- ▶ /etc/utmp filtered by /usr/lib/acct/fwtmp
- ▶ /usr/adm/sulog
- ▶ /usr/adm/syslog
- ▶ /usr/adm/syslog/syslog.log
- ▶ /usr/adm/snmpd.log
- ▶ /usr/lib/cron/log
- ▶ /usr/spool/lp/log
- ▶ /var/adm/sulog
- ▶ /var/log/syslog
- ▶ /var/cron/log
- ▶ /usr/spool/logs/lpsched
- ▶ /usr/spool/lp/logs/requests
- ▶ /var/log/wtmp
- ▶ /var/log/messages
- ▶ /var/log/debug
- ▶ /var/log/spooler
- ▶ /var/log/xferlog
- ▶ /var/log/secure
- ▶ agentlogpath (the PATROL Agent log)

### **Monitoring a log file**

Use the following steps to set up monitoring of a log file:

1. Access the LOG application menu.
2. Choose **KM Commands -> Edit List of Monitored Files**. PATROL displays the Log Files dialog.

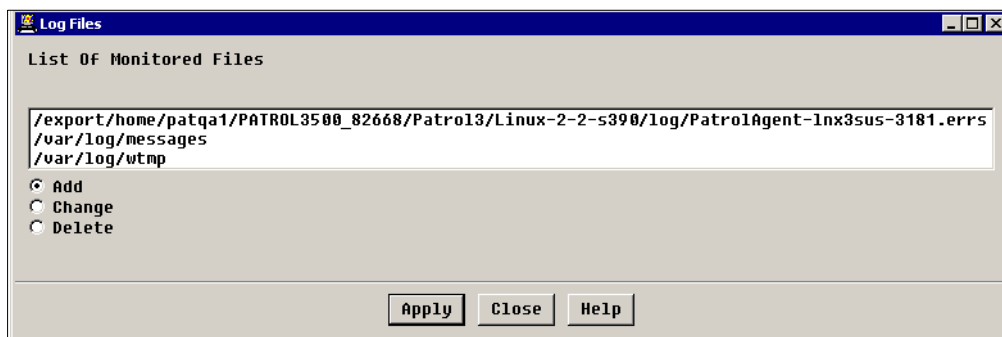


Figure 13-14 Log Files dialog

3. Select the **Add** radio button.
4. Click **Apply**. PATROL displays the Log File (New) dialog.

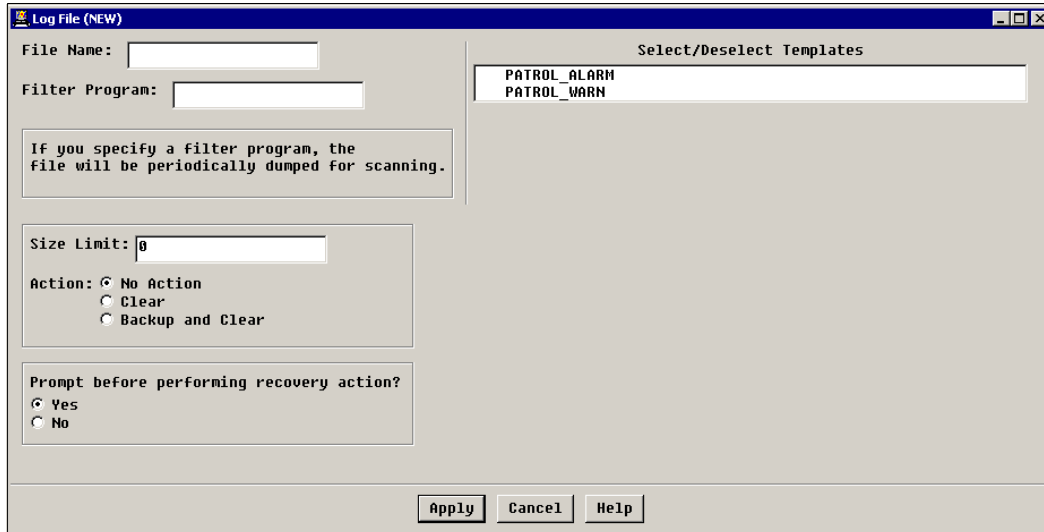


Figure 13-15 Log Files (New) dialog

5. Enter the file name and full path of the log file that you want to monitor in the File Name field.
6. If you want to monitor a binary log file, enter the name of the program that translates the file to ASCII in the Filter Program field.
7. Associate a log file search template with the log file.
  - If you have previously created a template to use for this log file, from the Select/Deselect Templates list, select a log file search template.
  - If you have not created a log file search template that applies to this log file, finish the steps in this task, and then create a search template. See the *PATROL Knowledge Module for UNIX User Guide* for information on defining a search.
8. If you want to set up an automated recovery action that will reduce the log file size when it exceeds a specified limit, see “Automating text log file size reduction for individual logs” on page 325.
 

Click **Apply** to continue. PATROL displays a confirmation dialog that lists the log file name and path. It asks you if you want to add the file to the list of monitored log files.
9. Click **Accept** if you want to monitor the file that you specified. Otherwise click **Cancel**. If you clicked Accept, PATROL displays the Log Files dialog with the new log file name in the List of Monitored Files.

## 13.3 Security monitoring using PATROL for Linux Enterprise Server

PATROL for Linux Enterprise Server monitors various aspects of the operating system that affect its security. The product monitors characteristics of files such as whether or not the file has:

- ▶ Set user or group permissions
- ▶ Global write access

PATROL for Linux Enterprise Server also monitors characteristics of user accounts and user activities that pose a potential threat to security, such as which users:

- ▶ Failed to execute a set user (**su** or **msu**) command
- ▶ Have no passwords, or blank passwords
- ▶ Are running multiple sessions

### 13.3.1 Monitoring files

The SECURITY application enables you to monitor files with potentially dangerous permissions and privileges. Using this feature you can determine if a file has set user or group permissions and if a file can be written to by any user.

To perform any task related to monitoring files, you must provide a user account and password. It does not have to be the root account.

#### Viewing files with SUID or SGID permissions

This section describes how to view a list of files that have the set user ID (SUID) or set group ID (SGID) permission set. This permission allows the file to set its own user ID or group ID, respectively, regardless of the owner or group to which the owner belongs.

To view a list of files with SUID and/or SGID permissions, follow these steps:

1. Access the SECURITY application menu.
2. Select **KM Commands -> Administration -> List SUID and SGID Files**. PATROL prompts you for a user account and password.
3. Type the appropriate user name and password and click **OK**. PATROL displays the Find SUID/SGID Files Like dialog (Figure 13-16).

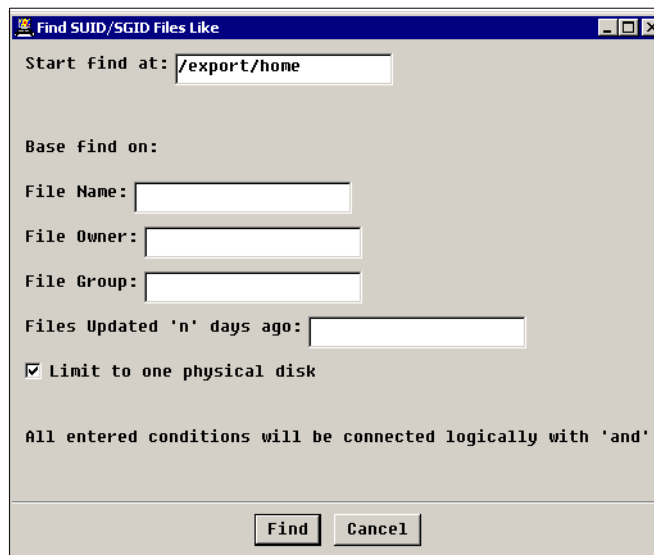


Figure 13-16 Find SUID/SGID Files Like dialog

4. Define the search criteria. The search is a logical AND operation that returns only files that match all the criteria defined by the following fields.
  - **Start find at** is the directory structure from which to begin the search. The default is root (/).

- **File Name** is the regular expression pattern for one or more files. For more information, see the “Regular Expressions” topic in the *PATROL Knowledge Module for Unix User Guide*. Leave this field blank to search for any file name.
  - **File Owner** is the owner of the files.
  - **File Group** is the group to which the owner of the files belongs.
  - **Files updated ‘n’ days ago** is the period in days between the change date and the current date.
  - **Limit to one physical disk** restricts the operation to the disk drive on which the directory defined in the “Start find at” field resides.
5. Click **Find**. PATROL builds a search expression, searches the system for files that match the criteria, and writes the results to a PATROL task object, List SUID and SGID Files, in the UNIX OS container.
  6. Access the List SUID and SGID Files task object. Your results should resemble Figure 13-17.

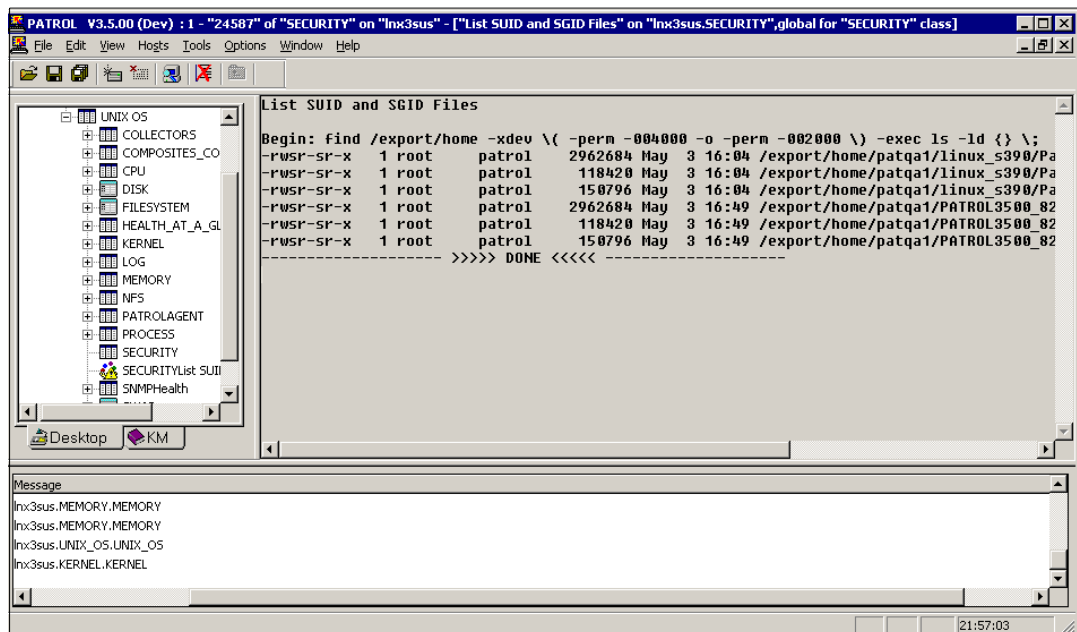


Figure 13-17 Find SUID/SGID Files Like Dialog

Table 13-2 Find Files Like command output format

Search expression	
Begin: find	Find command
/path	Starting point for search
- name <text?*>	File name pattern
-perm -00400 -o -perm -002000	Defines the permissions of files that the find command looks for
-exec ls	Lists the contents of the specified directory
-ld	Options for the ls command: -l long format listing, -d list only the directory names and not its contents
{ }	Argument that substitutes current file

List of files	
-rwsr-sr--	File permissions
#	Number of links to the file
username	Owner of the file
dev	Group of owner
###	Size in bytes
MMM DD hh:mm	Date-time stamp
location	Path and file name
-- >>> DONE <<< --	End of file list

### Viewing files with global write privileges

This section describes how to view all files that can be written to by any user on the system.

To view a list of files that any user can write to, use the following steps:

1. Access the SECURITY application menu.
2. Select **Administration -> List Files With Global Write**. PATROL prompts you for a user account and password.
3. Type the appropriate user name and password and click **OK**. PATROL writes the results to a PATROL task object, List Files with Global Write, in the UNIX OS container.
4. Access the List Files with Global Write task object to view the results of the command. Your results should resemble Example 13-1:

#### *Example 13-1 List Files with global write permission*

---

```
List Files with global write permission
/usr/openwin/lib/locale/libs
/var/ntb/tmp
/var/ntb/appcnfg/appmgr/System_Admin
/etc/PERL
/dev/sty
----- >>>> DONE <<<< -----
```

---

## 13.3.2 Monitoring user activity

The SECURITY application enables you to monitor user activity that could compromise your system's security. This feature allows you to determine which users have no or blank passwords, which users are running multiple sessions, and which users attempted to perform a set user command and failed.

To perform any task related to monitoring user security, you must provide a user account and password. It does not have to be the root account.

### Viewing users that failed to execute su/msu commands

This section describes how to view a list of users who attempted and failed to execute a set user (**su**) or (**msu**) command.

To view a list of users that failed to execute su/msu commands, use the following steps:

1. Access the SECURITY application menu.
2. Select **Administration -> List Failed su/msu logins**. PATROL prompts you for a user account and password.
3. Type the appropriate user name and password and click **OK**. PATROL writes the information to a PATROL task object, List Failed su/msu logins, in the UNIX OS container.
4. Access the List Failed su/msu logins task object to view the results of the command. Your results should resemble Example 13-2.

*Example 13-2 List Failed su/msu logins*

---

```
List Failed su/msu logins
SU 01/02 15:56 - pts/8 dpallet-ssp
SU 01/17 09:39 - pts/8 rabby-root
SU 01/18 17:45 - pts/17 mmoulin-root
SU 01/23 15:13 - pts/13 nedned-root
```

---

The output has the format shown in Example 13-3.

*Example 13-3 List Failed su/msu logins command output format*

---

```
cmd mm/dd hh:mm - pts/# acct_from-acct_to
```

---

*Table 13-3 List Failed su/msu logins command output format*

cmd	The unsuccessful command: <b>su</b> Become super user (root) or other user <b>msu</b> A variant of su that is no longer supported by platforms on which PATROL for Linux Enterprise Server runs.
mm/dd	Month and date on which the failed login attempt occurred.
hh:mm	Time at which the failed login attempt occurred.
pts/#	Pseudo tty subsystem slave terminal name.
acct_from	The user account from which the command was executed.
acct_to	The user account to which the acct_from user attempted to switch.

## Viewing users without passwords

This section describes how to view a list of users who either do not have a password or have a blank password.

To view a list of users that do not have a password or have a blank password, follow these steps:

1. Access the SECURITY application menu.
2. Select **KM Commands -> Administration -> List Users Without Passwords**. PATROL prompts you for a user account and password.
3. Type the appropriate user name and password and click **OK**. PATROL writes the information to a PATROL task object, List Users Without Password, in the UNIX OS container.
4. Access the List Users Without Password task object to view the results. Your results should resemble Figure 13-18.

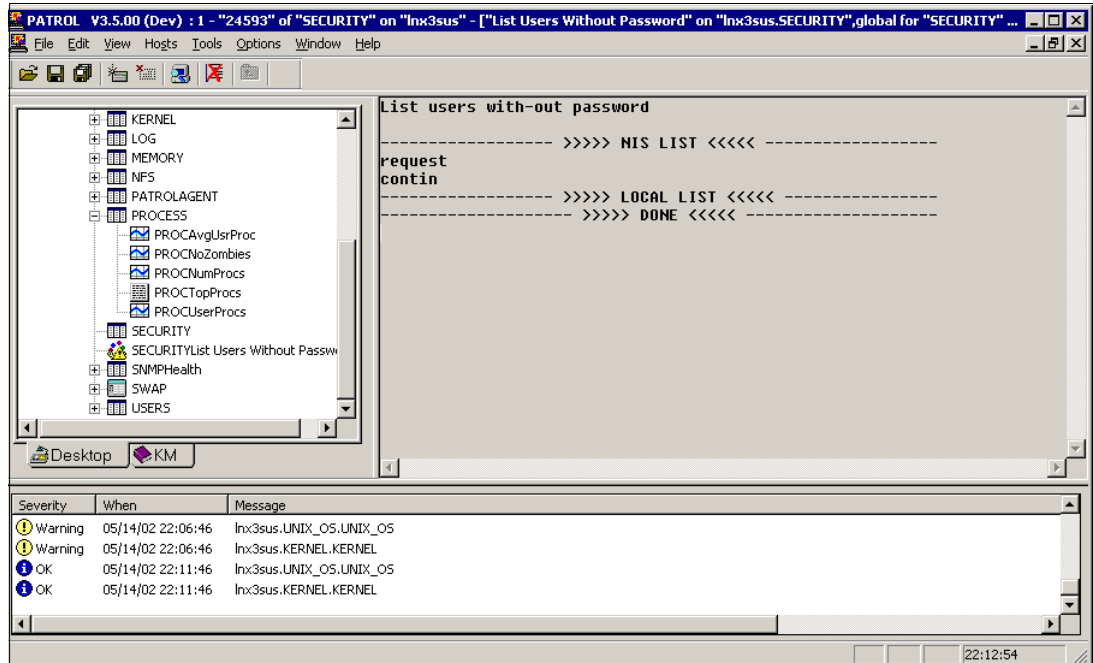


Figure 13-18 List Users Without Passwords dialog

## Viewing users with multiple sessions running

This section describes how to view a list of users who have duplicate user ID entries, which results from running multiple sessions.

To view a list of users with multiple sessions running, use the following steps:

1. Access the SECURITY application menu.
2. Select **KM Commands -> Administration -> List Duplicate User ID Entry**. PATROL prompts you for a user account and password.
3. Type the appropriate user name and password and click **OK**. PATROL writes the information to a PATROL task object, List Duplicate User ID Entry, in the UNIX OS container.
4. Access the List Duplicate User ID Entry task object to view the results. Your results should resemble Example 13-4.

### Example 13-4 List duplicate user ID entry

```
List users with duplicate user id
user id: 9351 -- vdolor, tg1_a, tg1_b, tg2_a, tg1_b, pdolor,
user id: 4262 -- mhartman, ldap,
user id: 923 -- 3Supvr, 4Supvr,
user id: 1092 -- r45, r46,
user id: 2784 -- nots, yesman,
----- >>>> DONE <<<< -----
```

The output has the format shown in Example 13-5.

### Example 13-5 List duplicate user ID entry command output format

```
user id: 4262 -- <username_#>, <username_#+1>,
```



## 13.4 Operations using PATROL Internet Server Manager

PATROL Internet Server Manager automatically discovers and displays a graphic presentation of your internet server. You can use the PATROL Classic Console on a Microsoft Windows system to quickly check the performance of your internet servers. For additional information on using the PATROL Console and PATROL Internet Server Manager, see the PATROL Internet Server Manager Documentation CD and the PATROL Console online help.

### 13.4.1 Availability management

You can use PATROL Internet Server Manager to perform the following advanced tasks to monitor servers on your Linux system:

- ▶ Configure mail queue reporting and mail server usage data
- ▶ Monitor remote mail hubs
- ▶ Configure BIND DNS servers

See the online Help system for more information.

#### Configuring Sendmail servers

While PATROL Internet Server Manager can monitor all SMTP Mail servers for availability and response time, it can also gather additional important usage data when used to monitor a Sendmail mail server, including:

- ▶ Mail queue reporting. You can use PATROL Internet Server Manager to monitor the number of messages in your mail queue. PATROL Internet Server Manager reports active messages and deferred messages as separate totals, allowing you to identify times of the day when your mail server may encounter bottlenecks so you can be prepared when they occur.
- ▶ Mail server usage data. PATROL Internet Server Manager monitors the rate of messages flowing in and out of your mail server. With this data you can analyze increases in usage for your server, discover trends, and anticipate future expansion needs.
- ▶ Remote mail hub monitoring. If you must route mail to mail servers inside your corporation or at other companies or ISPs, you can use PATROL Internet Server Manager to monitor the amount of deferred messages in your queue destined for these servers. You can also set your own thresholds to know when service problems on these systems may cause problems on your mail system.

To configure mail queue reporting and mail server usage data, use the **Configure Local** and **Configure Basic** settings menu commands to activate these features of PATROL Internet Server Manager for your local Sendmail server.

1. Access the menu commands for the server's application instance.
2. Choose **KM Commands -> PATROL Admin -> Configure -> Local** to display the Local Configure dialog for the specified server instance.

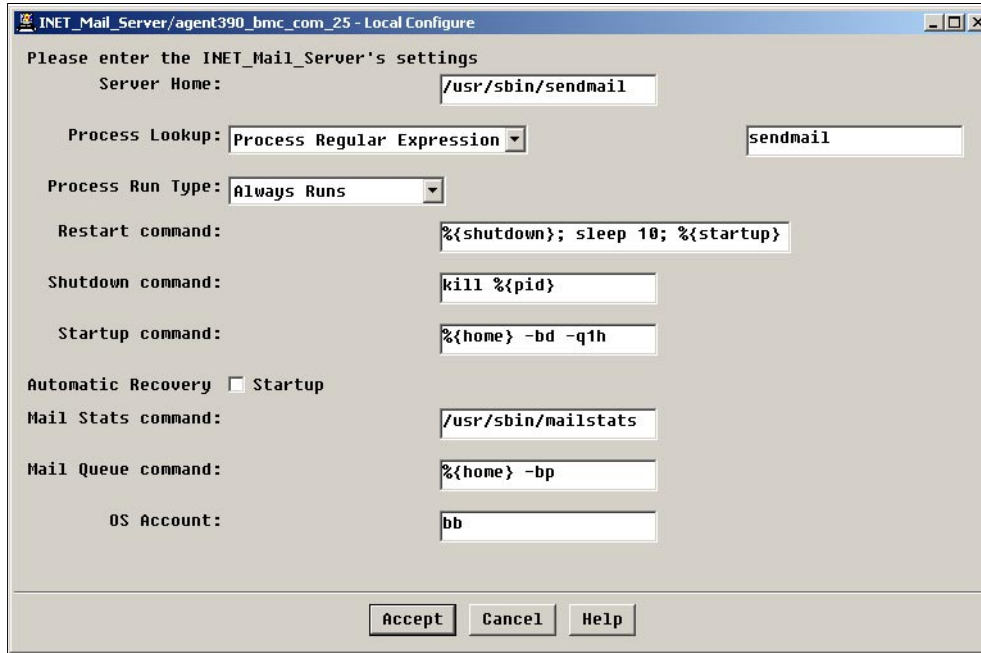


Figure 13-19 Local Configure dialog

3. To take advantage of the Sendmail monitoring capabilities, make sure you properly configure the following three fields in particular from this dialog:
  - **Mail Stats command.** Enter the location of your Sendmail server's MAILSTATS command. This value will vary from system to system, but the default provided is based on your operating system and will generally be acceptable.
  - **Mail Queue command.** Specify the location of your Sendmail server's MAILQ command. Most Sendmail servers accept the path to the actual command, such as /usr/bin/mailq, or you can configure PATROL Internet Server Manager to call the sendmail program by using the **-bp** option. In most cases you should accept the default value provided to you by PATROL Internet Server Manager.
  - **OS Account.** In most cases PATROL Internet Server Manager will need root privileges on your Linux system to analyze Sendmail usage data. This information will be used for the purpose of capturing this usage data only. The username and password you provide will be stored in an encrypted format inside the PATROL Agent configuration database. Enter the username to use (usually root) in the OS Account field.
4. Click **Accept** to confirm your choices. If you have entered a username in the OS Account field, a new dialog will prompt you to enter the password for this account.

If you enter an incorrect password when trying to start a Web server with SSL support, you may need to kill the Web server process involved. You can find a list of all Web server processes using the following command:

```
ps -ef | grep https
```

Using the output of the ps command, identify the Web server processes and determine their PIDs (column 2 of the output). Then you can explicitly kill the processes using the following command:

```
kill -9 PIDLIST
```

## Monitoring remote hubs

If you are monitoring a local Linux Sendmail server, you can also specify remote mail hubs to monitor. For example, your mail server might send many messages to another mail server belonging to ISP XYZ.COM. If mail to XYZ.COM backs up because of problems at that remote mail hub, it may cause your local server to back up. You can monitor the number of deferred messages to that server by defining it as a mail hub.

To monitor a remote mail hub, use the following steps:

1. Access the menu commands for the server's application instance.
2. Choose **KM Commands -> PATROL Admin -> Configure -> Basic** to display the Basic Configure dialog for the specified server instance.

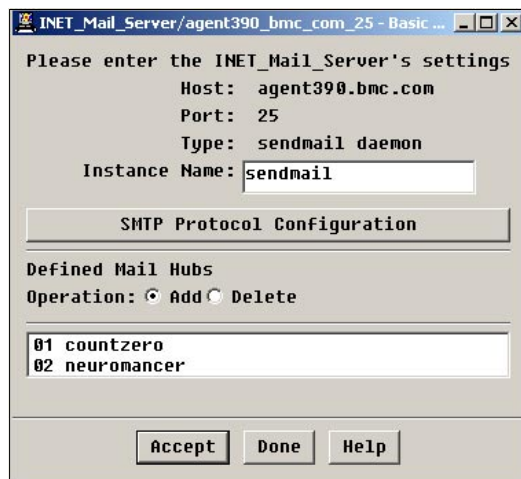


Figure 13-20 Basic Configure dialog

3. Select **Operation: Add** and then click **Accept**. PATROL displays a Basic Configure Confirmation dialog.



Figure 13-21 Basic Configure Confirmation dialog: Add

4. Enter the hostname for the remote mail server (if you know it) or the domain name for the remote server, then click **Add Hub** to continue. PATROL displays the Basic Configure dialog with the name of the remote mail hub you added.
5. If you want to delete a remote mail hub from monitoring by PATROL Internet Server Manager, select the remote mail hub you want to delete and select **Operation: Delete**, then click **Accept**. A confirmation dialog displays asking you to confirm the deletion of the selected remote mail hub.

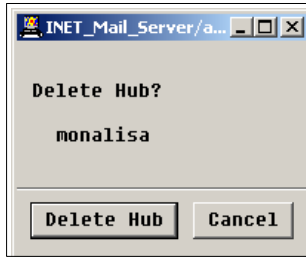


Figure 13-22 Basic Configure Confirmation dialog: Delete

If you want to delete the remote mail hub, click **Delete Hub** to continue. Otherwise click **Cancel**. PATROL displays the Basic Configure dialog with the updated list of remote mail hubs.

6. When the Basic Configure dialog displays, click **Done** to have PATROL update the list of monitored remote mail hubs.

### Monitoring BIND DNS servers

You can use PATROL Internet Server Manager to monitor all types of DNS servers to ensure availability and response time. There are certain additional capabilities available to you when monitoring BIND DNS servers on Linux systems. In these cases, PATROL Internet Server Manager will report statistical data for clients that use your DNS server. PATROL Internet Server Manager will report request rates that you can use to predict when you might encounter bottlenecks on your system. You can use this data to decide when to add additional DNS servers into your environment.

To configure BIND DNS servers, use the following steps:

1. Access the menu commands for the server's application instance.
2. Choose **KM Commands -> PATROL Admin -> Configure -> Local** to display the Local Configure dialog for the specified server instance.

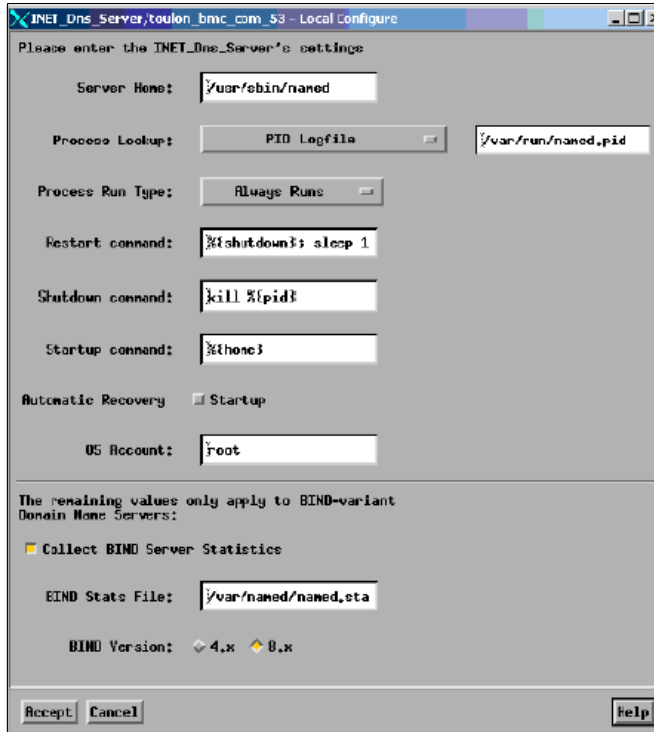


Figure 13-23 Local Configure dialog

3. To configure the advanced DNS settings for BIND DNS servers, enter information in the following three fields:
  - **Collect BIND Server Statistics.** Select this option to enable PATROL Internet Server Manager to gather statistical data. If you do not select this option, the consumer parameters reporting this usage data will not appear in your PATROL Console.
  - **BIND Stats File.** Enter the location of your BIND statistics file (also known as a scoreboard file). Consult your BIND documentation if you are not sure of the location. In most cases, use the default location.
  - **BIND Version.** Select the BIND server version: 8.x or 4.x (PATROL Internet Server Manager has not been tested with BIND version 9.x).
4. Click **Accept** to continue. PATROL Internet Server Manager performs additional monitoring for Bind DNS server based on your configuration changes.

## Validating Web pages

PATROL Internet Server Manager can monitor specific Web pages to verify their availability and response time. PATROL Internet Server Manager can make complex requests to these pages to simulate the transactions a normal Web browser makes. Some of these capabilities include the following:

- ▶ **Accepting dynamic server cookies.** When this feature is enabled, you can verify the results of Web applications that use session data that changes from one user to the next. For example, you might enable dynamic cookies to configure PATROL Internet Server Manager to log on to a shopping site, choose an item for purchase, and purchase the item. Then you could use PATROL Internet Server Manager to compare actual performance and

availability against expected values. You can indicate if you want to store and use dynamic cookies in monitoring the Web server as follows:

- Select **Accept cookies, use with this Web server only** if you want the instance to accept cookies but not share them with other Web server instances. This is the default setting.
  - Select **Accept cookies, make available to all Web servers** if you want the instance to accept cookies that can be shared with other Web server instances.
  - Select **Do not accept cookies** if you do not want this instance to accept cookies.
- ▶ **Authentication.** You can configure PATROL Internet Server Manager to monitor pages or applications using standard authentication methods.
- ▶ **Posting form data.** You can configure PATROL Internet Server Manager to post predefined form data to a Web server. You can use this feature to measure the response time and availability of Web forms, such as customer service complaint pages. Using this capability in conjunction with dynamic cookies, you can establish a unique user session with Web applications, such as storefronts or account status pages.
- ▶ **Page validation.** Using the Content Check feature, you can check to make sure certain data or text is found in your page, such as “Your order is complete.” You can also use this feature to make sure certain data is not found in your page result, such as “Error encountered.”

To configure the basic settings for a Web server instance, follow these steps:

1. Right-click the Web server application instance. The application menu is displayed.
2. Choose **KM Commands -> PATROL Admin -> Configure -> Basic** to display the Basic Configure dialog for the specified server instance.

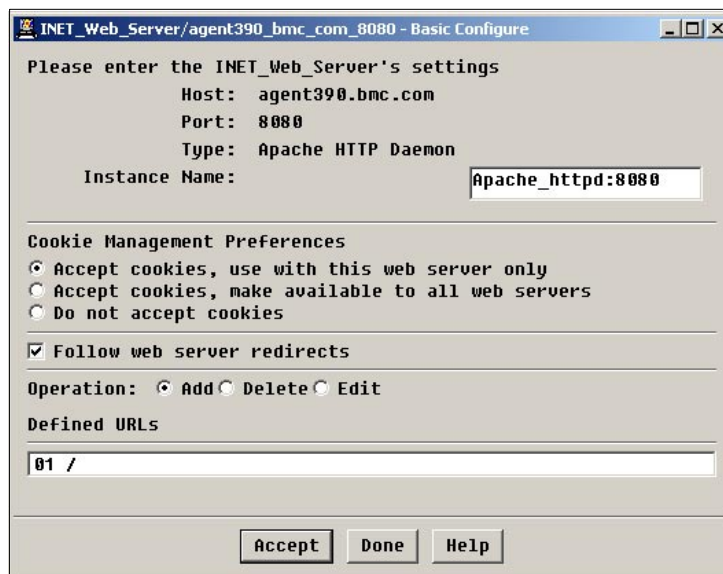


Figure 13-24 Basic Configure dialog

3. Select how you want to handle cookies for this Web server. The default is to not accept cookies.
4. Select **Follow Web server redirects** if you want PATROL Internet Server Manager to follow Web server redirects provided by the Web server in its response on a Location: response header.

5. Add, Delete or Edit URLs.
  - To add a URL, select **Operation: Add**, then click **Accept**.
  - To delete a URL select the URL you want to delete, select **Operation: Delete**, and click **Accept**.
  - To edit a URL, select the URL to edit, select **Operation: Edit**, then click **Accept**.
6. If you select **Operation: Add** or **Operation: Edit**, PATROL displays the Basic Configure URL dialog. You can use the dialog shown in Figure 13-25 to specify the details for a URL to be monitored.

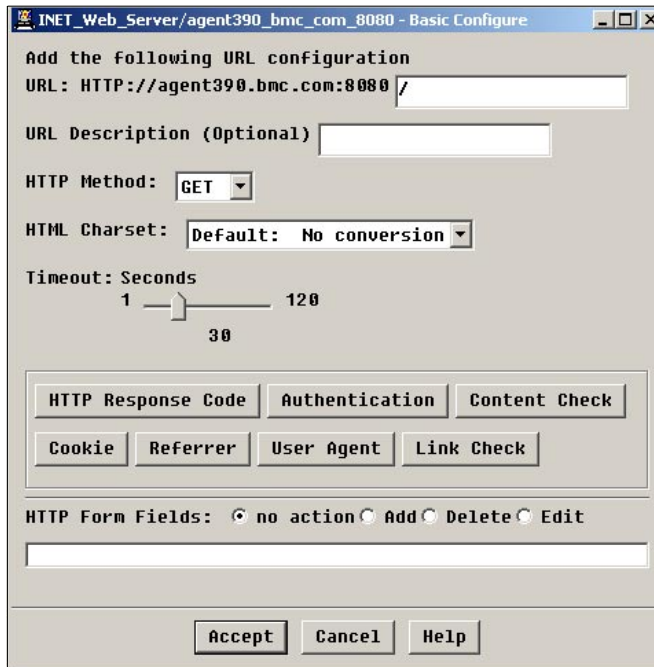


Figure 13-25 Basic Configure URL dialog

7. Configure the following URL settings for each monitored page, as needed:
  - **URL**. Enter the path to reach the page as you would in a standard Web browser. All paths must begin with a “/”.
  - **URL Description (Optional)**. You can enter a description or label for this URL to be displayed in the PATROL Console.
  - **HTTP Method**. Specify whether the request should be a HEAD, GET, or POST (the default method is GET):
    - Use the HEAD method when you do not need to retrieve the page data itself but only want to verify that the Web server is responding to your requests.
    - Use the GET method when you need to check the page content.
    - Use the POST method when you need to submit form data to the Web server. You can also submit form data using the GET method, and the data will be embedded onto the end of the URL to be requested.
  - **HTML Charset**. Unless you are using an international version of PATROL Internet Server Manager, this value must remain Default: No conversion; otherwise, you can select from any supported encoding listed. The page being monitored must use an encoding of the same language as the PATROL Agent and Console.

- **Timeout.** Choose a timeout value between 1 and 120 seconds. When the timeout period expires, ISM considers the page to be unavailable. You can modify this setting if your server needs more time to respond to requests. Once the timeout period has expired, the `httpStatus` parameter will be set to 0 (non-responsive page), and the `httpDownTime` parameter will increase to show how long this page and the server have been down. The `httpDownTime` parameter contains the recovery actions for restarting the server. After the down time value has expired, PATROL Internet Server Manager will try to restart the server if Automatic Recovery has been configured for that server instance.
- **HTTP Response Code.** Specify what HTTP response codes are considered to be OK or WARNINGS. Those not specified in either list are considered to be ALARM conditions.
- **Authentication.** If your Web page requires authentication to view the page, enter the appropriate username and password. On Windows systems, you can enter a domain name to authenticate using a domain controller (NTLM authentication) as follows: `MYNTDOMAIN\username`.
- **Content Check.** Specify what content must be or must not be in the retrieved Web page. Keep in mind that PATROL Internet Server Manager does not render HTML the way a typical Web browser would, so you must enter an HTML match string found in the source page.

**Note:** When using Content Check to check for the results of a Web page, do not use the HEAD HTTP Method. Also, remember to enter data exactly the way it is found in your HTML source when checking for specific content.

- **Cookie.** Specify any cookie data that should be sent with the request. Cookies should be defined as name-value pairs separated by a semicolon and a space. For example:  
`Cookie: Name1=Value1; Name2=Value2; Name3=Value3`  
 If you have directed ISM to accept dynamic cookies as mentioned previously, these cookies will be sent along with any static cookie data defined in this field.
- **Referrer.** If your page requires a referring page, specify the Referrer header, if any, that should be sent with the request. This value is used to tell the Web server what page was loaded before making this request. You can generally leave this field empty if you do not know what to put here.
- **User Agent.** Specify the user-agent header that should be sent with the request. Do not leave this field blank. Web servers use the user agent to determine the type of Web browser that is making the request. This data may be used to deliver custom responses and is generally stored in the Web server logs for later analysis. By default, PATROL Internet Server Manager sends a user agent header with the request that identifies PATROL as the Web browser.  
  
 This feature allows you to discount requests in your server logs that come from PATROL Internet Server Manager when reporting customer use data. To have PATROL Internet Server Manager send a standard user agent similar to a regular Web browser, enter a value similar to the following, depending on the browser:
  - For Microsoft Internet Explorer 5.01 on Windows 2000: `Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)`
  - For Netscape Navigator 4.5 on Sun Solaris 2.6: `Mozilla/4.5 [en] (X11; U; SunOS 5.6 sun4m)`



- **Link Check.** Set up link checking for the page being requested. This feature allows you to search for broken links found in pages that are linked from this page. See the online help for more detailed information.
  - **HTTP Form Fields.** When using form fields, specify a name and a value. If the name you choose matches (case-insensitive match) password or passwd, then the data is stored encrypted in the PATROL Agent's configuration database so that it cannot be viewed by malicious users.
    - To add a new form field, select **Add** and click **Accept**.
    - To delete a form field, select the name-value pair from the list, select **Delete** and click **Accept**.
    - To edit a form field, select the name-value pair from the list, select **Edit**, then click **Accept**.
8. Click **Accept** when you have finished editing the page data to return to the main dialog for Configure Basic Settings. At this point, you can continue to add or edit pages to monitor.
  9. Click **Done** to save your changes.

### Monitoring remote Internet servers

You can monitor Web server application instances remotely. This configuration allows you to report response times that are representative of the times seen by clients across the Internet or intranet. Local monitoring does not require network access time. You can use the information reported to improve server performance.

To monitor remote Internet servers, complete the following:

1. Access the menu commands for the Internet Servers application instance.
2. Choose **KM Commands -> Begin Monitoring**. PATROL displays the Begin Monitoring Internet hosts from... dialog.



Figure 13-26 Begin Monitoring Internet hosts from... dialog

3. Enter the name of the Internet server that you want to monitor remotely in the Enter a host name field. You may also enter the server's IP address if the server's name is not in DNS.
4. Click **Accept** to continue. PATROL displays the dialog shown in Figure 13-27.

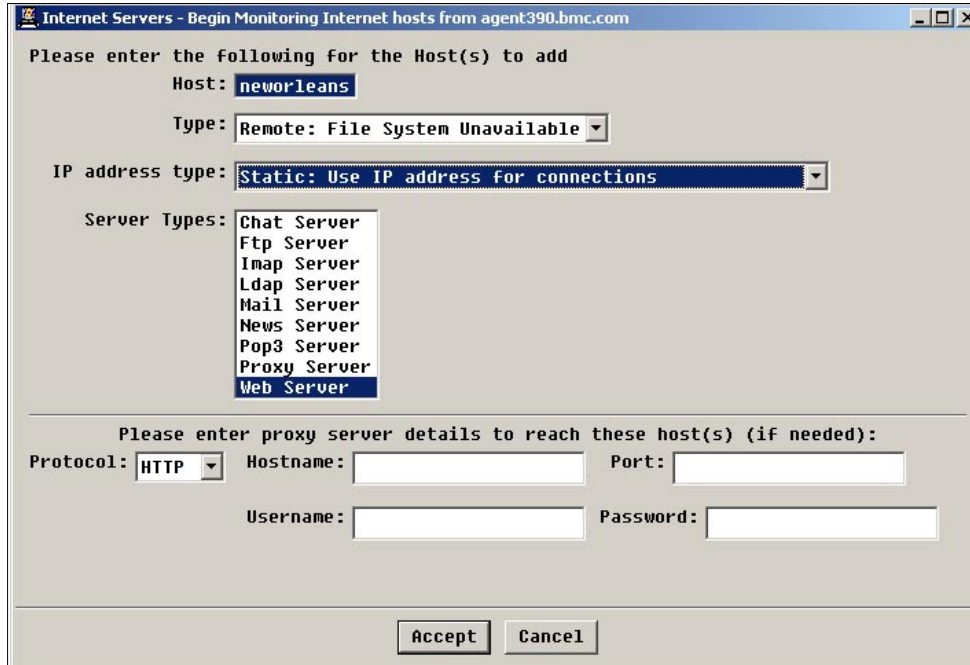


Figure 13-27 *Begin Monitoring Internet hosts from... dialog*

5. From the Type selection list, select one of the following options to specify whether the file system containing the remote Web server's Access Log and Error Log are available to the remote PATROL Agent:
  - Local
  - Remote: File System Available
  - Remote: File System Unavailable
6. Select the IP address type to be used. Valid options are:
  - **Static** to statically store the IP address of the remote Web server system for use during each connection. This method is preferred for optimum performance.
  - **Floating** to perform a lookup of the remote Web server's IP address for each connection. This method is preferred if the IP address for the remote Web server is subject to dynamic changes.
7. In the Server Type field, select the type of remote server that will be monitored.
8. Click **Accept** to save the entry and continue. PATROL displays the dialog shown in Figure 13-28.



Figure 13-28 *Begin Monitoring Internet hosts from... dialog*

9. Click **Accept** to close the dialog. PATROL starts monitoring the remote server.

When you add a host for monitoring, a new INET\_Servers instance is created in your console. When the INET\_Servers application instance is created, PATROL Internet Server Manager automatically discovers Internet servers, such as Web servers, FTP servers, and Mail servers, running on this host. Some types of servers display a response dialog on the PATROL Console asking you to verify the server's configuration information.

## 13.4.2 Health monitoring

The parameters for PATROL Internet Server Manager enable you to analyze Internet server performance quickly and easily by providing a detailed statement of all system activity over time. You can clearly identify peaks, troughs, and trends in the performance of server resources. By enabling you to detect problems, optimize the system, analyze trends, plan capacity, and manage multiple hosts simultaneously, PATROL Internet Server Manager helps to ensure your Internet server installations run efficiently 24 hours a day.

For additional information on the data collected by the PATROL Internet Server Manager product, see the following:

- ▶ “Parameter Summary” topic in the *PATROL Internet Server Manager v5.3.00 User Guide*
- ▶ “Task” and Parameter” topics in the PATROL Internet Server Manager v5.3.00 online Help

## 13.5 Security monitoring using PATROL Internet Server Manager

### 13.5.1 SSL certificate monitoring

The PATROL Internet Server Manager components provide parameters that allow you to monitor SSL certificates on secure Web server instances. These parameters check the certificate and each certificate in its chain every twelve hours to determine the time before a certificate expires. These parameters obtain their values from the certificate retrieved from the Web server.

- ▶ The certChain parameter gives information about each component in the certificate chain. The following information is displayed by the parameter:
  - Subject Issuer
  - Date Issued
  - Date Expires
- ▶ The certDaysLeft parameter indicates the number of days until the server certificate expires. If there is more than one component in the certificate chain, this parameter indicates the minimum number of days until one of the components expires.

By default, the certDaysLeft parameter goes into a WARNING state if there are from 16 to 31 days left until the certificate expires. It will go into an ALARM state if there are 15 or fewer days remaining, or if the certificate information could not be retrieved from the server. The certificate is checked every 12 hours. You can change this interval and the WARNING and ALARM ranges using a PATROL Developer Console to customize your KM installation.



## Part 5



# Computer Associates

Products in the following Computer Associates brands are described in this part:

- ▶ Unicenter
- ▶ BrightStor
- ▶ eTrust
- ▶ Advantage
- ▶ CleverPath





# Computer Associates Linux solutions

In this chapter, we provide an overview of the Computer Associates products used to manage Linux, along with their installation considerations.

## 14.1 Computer Associates solutions for Linux for zSeries and S/390

The following sections provide introductory and installation information for these solutions, as well as additional product feature and usage information. For detailed information on any of these products, see the Computer Associates product documentation. Computer Associates offers additional products (not described here due to space limitations), that can help you manage Linux. For the latest information on all Computer Associates products, visit <http://ca.com>.

### 14.1.1 Unicenter: Enterprise management for Linux for zSeries and S/390

The CA Unicenter solutions provide powerful control of the core IT resources that make or break an organization's e-business responsiveness. These products are part of CA's comprehensive e-business Infrastructure Management focus area, and consist of modular network and systems management solutions that deploy quickly and incorporate changing technologies, from wireless to Voice-over IP (VoIP). Advanced intelligence and visualization make these ideal solutions for forward-looking e-businesses.

Unicenter Enterprise Management solutions for Linux for zSeries and S/390 include the following:

- ▶ Unicenter Network and Systems Management
- ▶ Unicenter NSM Performance Management
- ▶ Unicenter Management for Web Servers (Apache Agent)
- ▶ Unicenter Software Delivery Agent
- ▶ Unicenter Universal Job Management Agent

### 14.1.2 eTrust: Security for Linux for zSeries and S/390

CA provides an industrial-strength suite of eTrust solutions to protect the entire enterprise. These security solutions for Linux for zSeries and S/390 help administrators protect data and applications through policy-based control and user authentication.

eTrust Security solutions for Linux for zSeries and S/390 include the following:

- ▶ eTrust Access Control
- ▶ eTrust Directory
- ▶ eTrust CA-ACF2 Security (Interface to Linux for zSeries and S/390)
- ▶ eTrust CA-Top Secret Security (Interface to Linux for zSeries and S/390)

eTrust Security distributed solutions that integrate with information from Linux include the following:

- ▶ eTrust Admin
- ▶ eTrust Audit

Additional Linux security solutions across the eTrust brand are anticipated.



### 14.1.3 BrightStor: Storage for Linux for zSeries and S/390

The Internet and e-business have changed the way companies pursue new business opportunities. In the time-critical, Web-enabled marketplace, data remains the most important corporate resource. Data loss affects the bottom line, causing missed opportunities and lost revenue. Data corruption affects credibility, because bad data translates into inaccurate transactions. E-business success with Linux for zSeries and S/390 demands a comprehensive data protection solution that can ensure data availability and integrity without compromising performance.

BrightStor Storage solutions for Linux for zSeries and S/390 includes BrightStor Enterprise Backup.

### 14.1.4 Advantage: Data management and application development for Linux for zSeries and S/390

In the Advantage brand family, CA offers industrial-strength databases and the solutions necessary to rapidly integrate applications, databases and business partner systems via powerful XML, transport and messaging services. These application development, deployment and integration solutions are integrated and open-standards-based, covering platforms from mainframes to wireless devices, including Linux for zSeries and S/390.

Advantage solutions for Linux for zSeries and S/390 include the following:

- ▶ Advantage Ingres Enterprise Relational Database
- ▶ Advantage CA-XCOM Data Transport
- ▶ Advantage Data Transport Agent

### 14.1.5 z/VM Solutions for Linux for zSeries and S/390

Mainframes can run thousands of Linux images under z/VM, and CA provides a comprehensive suite of z/VM management solutions to optimize this strategic mainframe environment. The CA portfolio includes solutions for security, console automation, backup/recovery, resource management, job management, performance management, automated operations and interactions with applications.

VM solutions for Linux for zSeries and S/390 are packaged in a single bundle called VM:Manager VM Management Suite for Mainframe Linux, and includes the following:

- ▶ Unicenter VM:Account
- ▶ Unicenter VM:Operator
- ▶ Unicenter VM:Schedule
- ▶ Unicenter VM:Spool
- ▶ BrightStor VM:Backup
- ▶ BrightStor VM:Tape
- ▶ eTrust VM:Director
- ▶ Unicenter CA-Explore Performance Management for VM
- ▶ eTrust CA-Top Secret Security for VM or eTrust CA-ACF2 Security for VM

## 14.1.6 CleverPath: Portal solutions for Linux for zSeries and S/390

CleverPath Portal integrates information into a personalized and intelligent environment that can be accessed from a Web browser, mobile phone or wireless PDA. Virtually any information, regardless of source, can be presented and secured through CleverPath Portal and combined with critical information to provide effective collaboration and decision making. An extensive Portlet Library also provides out-of-the-box integration with a wide variety of applications, systems and components.

## 14.2 Unicenter Network and Systems Management

Unicenter Network and Systems Management manages the health and availability of operating systems and provides basic status management on all infrastructure elements such as network devices, business applications and database systems. Auto Discovery builds a database with information on system elements and populates 2D and 3D dynamic visualizations. The Historian feature keeps systems administrators informed about past events and object status, and predictive management capabilities inform them of possible bottlenecks in their systems and applications so that they can take automated actions to avoid them. Portal technology provides personalized intuitive information for both technical- and business-focused administrators.

The components that provide for monitoring the health of your system include the manager and various agents, including agents that monitor the system, processes, and performance.

For detailed information on the components, see the product documentation.

### 14.2.1 Installing the Manager

Follow these steps to install the Enterprise Management Manager on a Linux machine.

1. Create a temporary directory on the target Linux machine for the Unicenter NSM installation materials.
  - Log in to the Linux machine using the root ID and password.
  - Make a directory using the Linux `mkdir` command. This new directory should **not** be the same as the directory where Unicenter NSM will be installed or located
  - in `/tmp` or `/usr/tmp`.
2. Transfer the installation materials from the distribution media to the target Linux machine. To do this, ftp in binary mode, the NSM30.tar file, setupNSM script and sNSM\_<LANG>.lcl file to the target Linux machine in the directory created in step 1 above. Replace <LANG> with the language of the target Linux machine.

The NSM tar file is:

```
<CD Drive>\enu\Linux-s390\EMFiles\NSM30.tar
```

You must also copy the Unicenter Explorer tar file:

```
<CD Drive>\enu\Linux-s390\ComFiles\ue.yyyymmdd.tar
```

where *yyymmdd* indicates image build date

3. On the target Linux machine, change to the directory created in step 1 above. Issue the following command:

```
cd <dir>
```

4. Issue the following command:

```
chmod 755 setupNSM
```

5. Execute the following script:

```
./setupNSM em
```

The setupNSM script extracts the installation materials, and then executes the setup utility. To restart the installation procedures after setupNSM has extracted the materials, run setup from the directory created in step 1.

6. The Unicenter NSM setup script displays three options:

1. Install Unicenter NSM Enterprise Management
2. Remove Unicenter NSM Enterprise Management
3. Exit the Unicenter NSM Enterprise Management Installation

Select option 1 (Install Unicenter NSM).

The installation prompts you for the information it requires. Follow the instructions provided by the setup script.

7. Cleanup—On the target Linux machine remove the temporary directory (and its contents) created in step 1.

## 14.2.2 Installing Agent Technology and Agents

Follow these steps to install the Agent Technology and one or more agents on a Linux machine.

1. Create a temporary directory on the target Linux machine for the Unicenter NSM installation materials.
  - Log in to the Linux machine using the root ID and password.
  - Make a directory using the Linux `mkdir` command. This new directory should NOT be the same as the directory where any component of Unicenter NSM will be installed or reside in `/tmp` or `/usr/tmp`.
2. Transfer the installation materials from the distribution media to the target Linux machine. To do this, ftp in binary mode, the AGENT tar file, setupNSM script and sNSM\_<LANG>.lcl file to the target Linux machine in the directory created in step 1 above. Replace <LANG> with the language of the target Linux machine.

The AGENT tarfile is:

```
<CD Drive>\enu\Linux-s390\AWFiles\AGENT.tar (supports SuSE 7.0)
```

3. On the target machine, change to the directory created in step 1. Issue the following command:

```
cd <dir>
```

4. Issue the following command:

```
chmod 755 setupNSM
```

5. If Agent Technology is already installed, enter the following commands:

```
awservices stop  
camclose (If CAM already installed)
```

6. Execute the following script:

```
./setupNSM atech (provides menu of agents to install)
```

The installation prompts you for any information it requires. Follow the instructions provided by the script. The setupNSM utility only extracts the product image. Once this is complete, the setup utility executes the install.agents utility. If difficulties are encountered after the extraction of the image, re-run `install.aworks` or `install.agents`.

7. Configuration—For Unicenter NSM Agent Technology to function correctly, the AGENTWORKS\_DIR environment variable must be set to the Agent Technology install path.

The PATH variable should also include:

```
$AGENTWORKS_DIR/services/bin  
$AGENTWORKS_DIR/services/tools  
$AGENTWORKS_DIR/agents/bin
```

The above settings should also be added to /etc/profile or the Agent Technology Users .profile or .cshrc.

8. Kernel Configuration—Kernel recommended values for Unicenter NSM Agent Technology for the Linux platform are:

```
msgmap=258          msgmax=32768  
msgmnb=65535       msgmni=128  
msgseg=7168        msgssz=32  
msgtql=256         ninode=500  
semmap=256         semmni=256  
semmns=512         semmsl=100  
shmmni=512         shmmax=16777216
```

9. Starting Unicenter NSM Agent Technology—Run /etc/profile or the Agent Technology Users .profile/.cshrc to set the environment.

To start the Agent Technology component enter the following command:

```
awservices start
```

10. Cleanup—On the target machine remove the temporary directory created in step 1 and its contents.

## 14.3 Unicenter NSM performance management

Unicenter NSM provides historical, real-time, and proactive performance monitoring for workstations, servers, and devices on which e-business applications depend. With tools for visualization, analysis, and reporting, even the most complex systems can be reliably managed with minimized overhead and without the “fire fighting” that usually characterizes attempts to tackle unexpected performance issues caused by unanticipated business demands on the infrastructure.

The Agent Technology component of Unicenter NSM allows you to closely monitor the critical computing resources in your enterprise by configuring individual agents—such as the Ingres Agent—to instrument these resources.

Management information is standardized according to a management protocol that is understood by both managers and agents and transmitted according to a communications protocol. The Ingres Agent is based upon the Agent Technology architecture. For communications protocols it uses the user datagram protocol (UDP) of the transmission control protocol/Internet protocol (TCP/IP) suite.

As a part of Unicenter NSM, the following agents can be installed and configured using setupNSM with the appropriate parameters:

- ▶ System Agent
- ▶ Performance Agents

- ▶ DB2 Agent
- ▶ Ingres Agent
- ▶ Process Agent
- ▶ Log Agent
- ▶ Informix Database Agent

### 14.3.1 The System Agent

The System Agent examines critical statistics within the host operating system and compares them to thresholds to determine the health of the system. The agent gathers statistics by collecting information for the O/S kernel and related interfaces. The agent compares these statistics to user-specified thresholds that define warning and critical levels.

If a threshold is exceeded, the agent sends an SNMP trap to the Distributed State Machine (DSM). The trap may cause the system status to change, which is reported by changing the color of the System Agent icon in Unicenter tools such as Unicenter Explorer, WorldView and Node View. Within Node View and Agent View, the particular resource category associated with the metric is highlighted (for example, I/O usage). You can view the metric and related statistics through the MIB Browser and Agent View.

The System Agent monitors the following system-related resources:

- ▶ System
- ▶ CPU
- ▶ Load averages
- ▶ Swap space
- ▶ Real memory
- ▶ Network interfaces
- ▶ Inter-process communication

You can also configure the agent to monitor file systems, physical disks, files, processes, and printer queues.

### 14.3.2 The Performance Agents

The Performance Agents collect both real-time and historical performance data from the environment being managed, and make this data available to Unicenter NSM Performance Management graphical applications which allow you to view, analyze, report on and manage this data. For information on these graphical applications and examples of their use, see the section on Unicenter NSM Performance Management in the next chapter.

The Performance Agents ensure network resource availability and performance by providing event and performance management for a wide range of operating system-related resources. In addition, the Performance Agents can collect performance-related data from any devices supported by an SNMP agent, such as routers, printers, hubs, and application/database monitoring agents.

Using a three-dimensional model for data management, the agent stores the collected performance information locally, reducing the load on the machine that monitors and manages resources across the enterprise. Additionally, the agent footprint is small, thereby placing minimal demands on the systems being managed.

The Performance Agents includes two fundamental sub-components:

**PrfAgent:** Responsible for real-time transient data collection. The agent is self-contained and requires no configuration. It responds, as required, to demands for real-time performance information from Performance Management client applications such as Performance Scope.

**HpaAgent:** Responsible for historical long-term data collection and data management. The agent is configured and managed using the Unicenter Performance Configuration component. The data it collects is provided automatically to the Performance Management client applications such as Performance Trend and Performance Scope for the purpose of detailed trend analysis and capacity planning.

The installation process installs both of these sub-components as one entity.

### 14.3.3 The DB2 Agent

The DB2 Agent examines critical statistics within the DB2 system and compares them to thresholds to determine the health of the system. The agent gathers statistics by collecting information through the DB2 snapshot API. The agent compares these statistics to user-specified thresholds that define warning and critical levels, sending an SNMP trap to the Distributed State Machine if a threshold is exceeded. In the event of a status change, the color of the DB2 Agent icon changes in Unicenter Explorer, WorldView, and Node View.

The DB2 Agent monitors the following resources:

**Manager information:** Memory status and sort heap.

**Database information:** Backup, cache, SQL table, and Sort status, connection information, counting, heap log, locking and integrity.

**Application information:** Statistics pertaining to those applications which access specified databases, including sort, SQL table, and cache status, cursor, and locks.

**Table Space information:** Statistics pertaining to tables associated with specified databases, including I/O, buffer and buffer index writes, I/O requests and physical space.

### 14.3.4 The Ingres Agent

By default, the Ingres Agent is configured to monitor database connection health, DBMS servers, communications servers, General Communications Architecture (GCA) layers, cache (both local and shared), logging system, locking system, file systems, and checkpoints. You can configure the agent to also monitor database tables.

The Ingres Agent provides the capability to monitor the connection status of system and all specified user databases. Should a connection fail, an SNMP trap is issued indicating the failure.

The Ingres client-server architecture allows multiple users access to databases through connections to one or more DBMS server processes. The DBMS server is a multi-threaded process that performs asynchronous disk input and output. It can execute queries on behalf of multiple users. These queries execute as multiple sessions inside the DBMS server.

### 14.3.5 The Process Agent

The Process Agent queries the host operating system to obtain real-time data on all running processes. The agent can compare this data to user-defined profiles for specific processes.

This section describes the watchers used by the Process Agent to monitor these critical processes.

The Process Agent can be configured to monitor process watchers. By default, the Process Agent is not configured to monitor any process watchers. You can further configure the agent using the Process Agent View, MIB Browser, or a configuration set.

A watcher is a condition-action set that looks for the specific behavior of a process and matches this behavior against a user-defined profile. For each detected difference from a profile, the agent can issue an SNMP trap, trigger an action, or both.

The status of a process is the result of any of the following criteria:

- ▶ **Process instances:** When the number of process instances exceeds its minimum or maximum instance threshold, the process status is marked as DOWN. This condition can indicate that the required number of process instances are not executing. For example, assume that the network file server (NFS) daemons are monitored. Five processes must be operating at all times. When only four of the daemons are operating, an NFS link to a file system has been lost.
- ▶ **Child processes:** When the number of children for a process exceeds either its minimum or maximum child threshold, the process status is marked as DOWN.
- ▶ **Process CPU utilization:** When CPU utilization for a process exceeds either its permitted minimum or maximum utilization threshold, the process status is marked as DOWN.
- ▶ **Process size:** When the size of a process, in blocks of virtual memory, exceeds either its minimum or maximum size threshold, the process status is marked as DOWN. This condition can indicate that the process has a memory leak.
- ▶ **Process termination:** If a process is stopped at any time during monitoring, however briefly, you can configure the process status to be set to DOWN. This condition can be used to indicate that the execution of a critical process has been terminated and the resource may not function correctly thereafter.
- ▶ **User permission:** When any configured process is executed by a user who does not have executable permission, the process status is marked as DOWN. When the status of a process watcher is marked DOWN, a trap or a call-back action can be initiated.

### 14.3.6 The Log Agent

The Log Agent uses log watchers to monitor log file data. A log watcher is a condition-action set that looks for a pattern in a log file. The agent processes the text contained in the log file in a filter, which compares the text to regular expressions (patterns).

With a filtering condition applied, the log watcher's state can change from UP to DOWN or from DOWN to UP, and/or the agent can send an SNMP trap. A trap contains its name and type, the log watcher's current state, information on the monitored log file, and the text found in the log file. You can configure whether the Log Agent will send state change traps, match traps or both:

- ▶ The Log Agent sends *state change traps* via the SNMP Administrator to the DSM. A state change trap is generated only if the state of the log watcher changes from UP to DOWN or from DOWN to UP.
- ▶ The Log Agent sends *match traps* via the SNMP Administrator to the Unicenter Event Agent on the same system. A match trap is used to forward the text string found in a log file. It is generated independently from the log watcher's state change.

The Log Agent can be configured to monitor ASCII log files and to facilitate the detection of faults in applications running under the operating system. Besides the monitoring of single log files, the agent offers the monitoring of all files in a subdirectory, where the subdirectory name may contain wildcards. Furthermore, the Log Agent can monitor the Windows NT/Windows 2000 Event Log, the Linux Console, and files with ASCII control characters.

The Log Agent also offers a read-from-end-of-file option for Status Policy, support for files containing null characters, and support for extended regular expressions.

### 14.3.7 The Informix Database Agent

The Informix Database Agent provides database health and status monitoring capabilities by gathering data from seven principal areas of the RDBMS:

- ▶ Informix server availability and usage
- ▶ Virtual processor utilization
- ▶ Dbspace capacity and usage
- ▶ Database size
- ▶ Database table size
- ▶ Database server lock overflows, requests and waits
- ▶ Database server log capacity and usage

The Informix Database Agent View lets you perform the following operations:

- ▶ View summary information relating to the Agent and the overall status
- ▶ Monitor resources for properties, such as availability, usage, and size of logical and physical units in the database
- ▶ View the availability of all dbspaces managed by the database server

## 14.4 Unicenter Management for Web Servers (Apache Agent)

Unicenter Management for Web Servers is an integrated management suite that delivers all of the monitoring and management functionality necessary to ensure the availability and performance of critical Web servers. This solution collects and correlates a wide range of information to ensure that the Web server and applications are functioning properly, being used effectively, delivering accurate information and responding appropriately to end users. Based on business policies, Unicenter Management for Web Servers automatically notifies personnel and automates corrective actions in response to problems.

The Web server agents are intelligent, programmable agents that monitor the availability, health, and performance of a Web server. The Web server agent for Linux for zSeries and S/390 is the Apache agent.

### Operating system support

Unicenter Management for Web Servers comprises the following components: console, manager, and agents. Although the console and manager systems require Windows NT/2000, TCP/IP and SNMP v1 or v2c, and Microsoft SQL Server 7.0 or higher, the agents have their individual requirements.

All Web server agent systems must run the following:

- ▶ TCP/IP and SNMP v1 or v2c



- ▶ If Agent Technology is not installed, use the appropriate Unicenter Agent Technology CD to install it.

The Apache HTTPD Server for S390 and z/OS supports Linux/390 from SuSE 7.0

### 14.4.1 Installing the Apache Agent

To install the Apache agent on a Linux/s390 machine:

1. Change user to root using `su`.
2. Create a work directory and untar the installation file using the following commands:

```
mkdir /tmp_UM4WS
tar -xvf UM4WS.LNX22.s390.UM4WS42.AW22.tar -C /tmp_UM4WS
```

3. Install the UM4WS component using the following commands:

```
cd /tmp_UM4WS
./install.setup
```

4. Remove the work directory using the following command:

```
cd /
rm -rf /tmp_UM4WS
```

#### The Apache configuration file

The Apache agent gathers full statistics from the Web server. You must enable the status module by uncommenting the following line in your Apache configuration file (`httpd.conf`):

```
LoadModule status_module modules/mod_status.so
```

Verify that the Apache `access.conf` file is configured to allow the agent to access the server-status URL. See the following URL in the online Apache help for more information:

[http://htdocs/manual/mod/mod\\_status.html](http://htdocs/manual/mod/mod_status.html)

#### Post-installation steps

Perform the following steps after the installation is complete:

1. Exit and log in again.
2. Start `awservices` using the following command:

```
awservices start
```

## 14.5 Unicenter Software Delivery Agent

Unicenter Software Delivery is a flexible tool that allows administrators to build, distribute, install, configure, verify, manage, and remove software packages on target systems anywhere in the enterprise. The Unicenter Software Delivery Agent includes Linux for zSeries and S/390 as a target platform, providing a solution that scales from the smallest network to enterprise installations with multiple networks and locations.

Software Delivery provides:

- ▶ Software Distribution Managers with an easy, intuitive explorer-style GUI
- ▶ Software Distribution Agents with software catalog support
- ▶ Automatic Software Packaging
- ▶ Full scripting language for custom installations

Software Delivery (SD) has an open install model that fits and supports any software installation. SD relies on vendor-provided installation methods, for example, to perform upgrades, installing only new or changed files. When using the most widely accepted installation routines, such as Install Shield, different delivery options allow users to copy only new or updated files over the network. Using SD features like delta packages, provided by the Packagers, can reduce data traffic.

The Software Delivery environment can include two types of sites—Areas and Enterprises.

**Area component**—Each Software Delivery environment includes at least one Area. Areas, and all connected computers, are collectively managed and monitored by a Local Administrator. The computers comprising an Area can include one Local Server, and can optionally contain a Workgroup Server, Staging Servers, one or more Admin Consoles, and a number of Agents. Linux agents are installed as daemons in the Linux operating system. They are continually available for software delivery from the Local Server.

**Enterprise components**—Enterprises allow Enterprise Administrators to centrally manage multiple Areas, and consist of an Enterprise Server and one or more Admin Consoles. Enterprise Administrators can also group two or more Areas with similar needs together into Domains.

For detailed information on the components, see the product documentation.

### Pre-installation considerations

Software Delivery makes a variety of software and hardware configurations available to you. Before you begin the installation procedure for Software Delivery, you should:

- ▶ Choose the network computers to include in the Software Delivery structure.
- ▶ Determine the type of Software Delivery installation required for each of your workstations: Software Delivery Server, Admin Console, or Agent.
- ▶ Determine whether to perform manual or unattended installation of agents.  
**Note:** If you are including a large number of agents in your Software Delivery configuration, unattended installation expedites the process.
- ▶ Determine where the Software Delivery Software Library resides. This library contains the programs available for distribution and installation on Agent computers belonging to the Software Delivery site. Enterprise Servers, Local Servers, and Workgroup Servers contain their own Libraries.

In addition, remember the following operating system-specific considerations:

- ▶ Administration of all Linux-based Local, Workgroup, and Enterprise Servers is performed using a Windows NT/2000 or Windows 9x/Me workstation. Administrators for these servers also need an equivalent user ID on each Linux server before they can interact with that server.
- ▶ Some Linux programs are stored on magnetic tape. If the server on which the program is being registered is not a Linux machine, it must have access to a device capable of reading this type of media.
- ▶ To manage Windows computers from a Linux-based server, you must install TCP/IP on the Admin Console computer.

## 14.5.1 Installing Software Delivery

A Software Delivery manager is installed on a central server. Linux Agent software is installed on each target computer, using either the SDDeploy facility for automatic installation, or

manually from the operating system command line. When the Agent component is installed, it defines the target computer as a registered member of the Software Delivery network, capable of receiving software installations from the Local, Workgroup, or Staging Server.

The Packager allows Administrators to automate and customize the packaging process. It is installed on a packaging computer, and packages software products and data into a form suitable for registration into a Software Delivery library or direct installation onto a target computer.

Software Delivery includes a catalog of install options, the Product Explorer, that lets you choose the components you want to install. Once you have made your selections from the Product Explorer, you are guided through the installation step-by-step.

A Software Delivery manager supports up to 12,000 target computers. The Agent component can be installed at every target computer automatically, using the SDDeploy facility. New versions of Software Delivery are provided with ready-to-install upgrade procedures. This allows all Software Delivery components to be upgraded from a central server.

### **Post-installation tasks**

The following post-installation tasks must be performed to set up Software Delivery:

1. Set up Software Libraries. Software Library directories store registered software packages for distribution, installation, and management on target computers.
2. Register SD Programs for supported operating systems. Register the programs for the components in your environment to the Enterprise Library. For Enterprise Servers, all SD programs for the operating system on which you install the Server are registered automatically. For example, on a Linux Enterprise Server, the SD Server and SD Agent for Linux are registered automatically in the Enterprise Library. SD programs designed for other platforms must be registered manually.

## **14.6 Unicenter Universal Job Management Agent**

The Unicenter Universal Job Management Agent extends the power, scalability and flexibility of CA job management solutions throughout the enterprise, including Linux for zSeries and S/390, while allowing job and schedule definitions and administration to remain centralized. This enables administrators to apply consistent corporate policy to all managed jobs. Through its integration with Unicenter AutoSys Job Management, Unicenter CA-7 Job Management, Unicenter CA-Scheduler Job Management, Unicenter CA-Jobtrac Job Management and Unicenter Network and Systems Management Job Management Option, this technology enables cross-platform scheduling by initiating and tracking units of work such as jobs, tasks and processes that execute on Linux for zSeries and S/390.

### **14.6.1 Installing the Unicenter Universal Job Management Agent**

The following are requirements for installing the Unicenter Universal Job Management Agent:

- ▶ Hardware: 64 MB RAM and 40 MB disk space
- ▶ Software: SuSE Linux 7.0
- ▶ GUI: On the machine that is using Event Management to display messages, the display resolution of your monitor must be set to a minimum of 800x600 with 256 colors for proper viewing.

You must install Unicenter Universal Job Management Agent on each Linux S/390 system that processes scheduling requests from the managing application and platform. You can install directly on your Linux machine or use FTP.

## Direct installation steps

To install directly:

1. Insert the installation CD-ROM into your Linux machine with a CD-ROM drive.
2. Follow the Linux operating system instructions to:
  - a. Mount the installation CD-ROM to the file system. For example, enter:

```
mkdir /CD
mount -r /dev/cd0 /CD
```
  - b. Export the file system for NFS mount on Linux S/390. For example, enter:

```
exportfs -i -o ro,root=<target Linux S/390 machine> /CD
```
3. Create a mount point on the target Linux S/390 machine:

```
mkdir /cdrom
```
4. On the target Linux S/390 machine NFS mount the file system with the installation materials:

```
mount -ro <UNIX/Linux machine>:<filesystem> /cdrom
```
5. Change to the directory containing the installation materials.
6. Execute the following script:

```
./setupNSM wkldagt
```
7. Follow the on-screen instructions to complete the installation.

## FTP installation steps

To install by FTP:

1. Insert the installation CD-ROM into your Linux machine with a CD-ROM drive.
2. Follow the Linux operating system instructions to mount the CD-ROM drive. For example, enter:

```
mkdir /CD
mount -r /dev/cd0 /CD
```
3. Create a temporary directory on the target Linux S/390 machine for the Unicenter NSM installation materials.
4. FTP in binary mode the installation materials from the CD on the Linux machine to the temporary directory you created on the target Linux S/390 machine.

```
<cd drive>/enu/Linux-S390/WkldAgt/WkldAgtUnx.tar
../setupNSM
../sNSM_<LANG>.1c1
```
5. Change to the temporary directory you created on the target Linux S/390 machine.
6. Issue the following command:

```
chmod 755 setupNSM
```
7. Execute the following script:

```
./setupNSM wkldagt
```
8. Follow the on-screen instructions to complete the installation.

## Post-installation tasks

Post-installation tasks comprise the following:

- ▶ Configuring CAICCI on Linux S/390
- ▶ Adding new manager nodes
- ▶ Changing the machine for message viewing
- ▶ Validating passwords for job submission
- ▶ Setting up the syslog daemon
- ▶ Rerouting messages to a remote host
- ▶ Placing changes into effect

See the product documentation for additional details on post-installation tasks. Information on configuring CAICCI on Linux S/390 is included here.

### ***Configuring CAICCI on Linux S/390***

CAICCI provides cross-platform communication. In order for the Computer Associates job management solution on the managing machine and Unicenter Universal Job Management Agent on the target Linux S/390 machine to communicate, you need to configure CAICCI on the Linux S/390 platform.

During post-installation, perform the following steps on every Linux S/390 machine on which you have installed the Unicenter Universal Job Management Agent:

1. Ensure that you can PING the manager node. If you cannot PING the managing system, modify the TCP/IP setup on the Linux S/390 machine as well as on the managing node. This may require a DNS entry or an entry in the Linux S/390 machine's `/etc/hosts` file for the manager node name.
2. Define a connection between nodes on the manager node and/or on the Unicenter Universal Job Management Agent machine. To define it on the Unicenter Universal Job Management Agent machine, use LOCAL and REMOTE statements in `$CAIGLBL0000/cci/config/<local host>/ccirmtd.prf`, the CAICCI configuration file. The LOCAL statement applies to the local computer. REMOTE statements apply to the remote nodes that exchange information with CAICCI.

The following LOCAL statement example tells CAICCI that the local machine's TCP/IP name is UNIX01 and that any remote system wishing to communicate with this system can do so by referencing UNIX01 as the name. Any Unicenter Universal Job Management Agent within the network can send and receive messages to this system independent of hardware or protocols, if it uses the name UNIX01.

```
LOCAL=UNIX01 UNIX01 32768 STARTUP
```

The following REMOTE statement example tells CAICCI to attempt a connection to 172.24.11.12 and to internally register MF01 as the CCI name. Any Unicenter Universal Job Management Agent within the network can send and receive messages to this system independent of hardware or protocols, if it uses the name MF01.

```
REMOTE=172.24.11.12 MF01 32768 STARTUP PORT=1721
```

The following REMOTE statement example tells CAICCI to attempt a connection to the computer whose TCP/IP name is UNIX01 and whose CCI name is UNIXSERVER1, but also to allow any Unicenter Universal Job Management Agent to send and receive messages using the alias UNIX1.

```
REMOTE=UNIX01 UNIXSERVER1 32768 STARTUP ALIAS=UNIX1
```

3. The Remote Server must be running. To determine whether it is running, issue the following command:

```
unifstat
```

A series of messages appears indicating the status of servers. If the Remote Server is running, the following message is included:

```
CA-CCI Remote Server   nnn   running
```

where *nnn* is the PID of the Remote Server process.

4. Start the Remote Server, if necessary. If the Remote Server is not running, issue the following command to start all processes including CAICCI:

```
unistart all
```

If you modified the `ccirmt.d.prf` file in step 2, and the Remote Server is active, issue the following commands:

```
unishutdown all
```

```
unistart all
```

These commands stop and start all processes.

## 14.7 eTrust Access Control

eTrust Access Control enables administrators to protect business-critical data and applications by establishing comprehensive security policies that regulate access to critical business assets. A GUI centralizes control over security policies as well as the administration of users, groups and system resources.

### 14.7.1 Installing eTrust Access Control

1. If you already have eTrust AC installed and it is running, shut it down by logging in as an administrator and then entering the following command:

```
<eTrustACDir>/bin/secons -s
```

where *eTrustACDir* is the directory in which eTrust Access Control is installed.

2. Log in as root.
3. If you are installing eTrust AC for the first time, create a directory to store the eTrust AC files. We recommend `/opt/CA/eTrustAccessControl`. These installation instructions assume you are installing eTrust AC files in `/opt/CA/eTrustAccessControl`.

**Note:** The installation directory should be put on a locally mounted file system.

4. You will be using several files from the eTrust AC CD-ROM:
  - A script that installs the eTrust AC base package from the tar file, named `install_base`.
  - A script that installs the eTrust AC GUI package, named `install_admt`.
  - Compressed tar file containing the eTrust AC base package, named `_LINUX390_<eACVersion>.tar.Z`
  - Compressed tar file containing the eTrust AC Motif GUI package, named `ADMT_LINUX390_<eACVersion>.tar.Z`
5. Run the `install_base` script, by using the `install_base` command. The installation script finds the appropriate tar file, so typing the name of your tar file is optional.

```
root@linux021:~/versions/ac/510/eTrust/media <2>
[root@linux021 media]# ./install_base
```

Figure 14-1 eTrust Access Control install command

**Note:** Full installation instructions are found in the product documentation, in Appendix A of the Administrator Guide.

You will need to specify information as the script prompts you. When the installation process is done you will see the following messages:

```
etrust@linux021:~
Restarting inetd daemon, in order to activate the changes.

Interactive setup procedure done.
Protecting eTrust look-aside database files.
Verifying eTrust seosd.under_NIS_server seos.ini token.
Verifying eTrust passwd.nis_env token in seos.ini.
Verifying eTrust seosd.resolve_rebind token.
Setting eTrust seosd.domain_names token in seos.ini.
Setting permissions of bin/sepmd bin/sepmdadm lbin/sepmd lbin/secrepsw bin/selo
grcd.
Setting permissions of /usr/seos/man/man8/ files.

Installing the RSV package ...
RSV package installed successfully in /usr/seos/rsv
To start the RSV daemon, execute /usr/seos/rsv/adm/startrsv

Successfully updated PROGRAM /usr/seos/bin/sebuildla
Successfully updated PROGRAM /usr/seos/bin/sesudo

Installation complete.
Check seos.ini file for the right configuration.
If you upgraded eTrust you must REBOOT the machine.

apropo::~/versions/ac/510/eTrust/media:63>
```

Figure 14-2 Sample eTrust Access Control installation messages

**Tip:** You may want to append /usr/seos/bin directory to your path, and /usr/seos/man to your MANPATH.

### System Requirements

- ▶ Operating system support:
  - SuSE 7.0 (Kernel 2.2.16)
  - SuSE SLES-7 (Kernel 2.4.7) (\*)
  - Red Hat Linux release 7.2 (Kernel 2.4.9) (\*)
- (\*) Needs a special fix on top of version 5.1 SP1
- ▶ Minimum disk space required—60 MB
- ▶ Suggested disk space—100 MB (or more)
- ▶ Memory required (RAM)—64 MB (or more)

## 14.7.2 Invoking eTrust Access Control

1. If you have not rebooted since installing eTrust AC, reboot now.

2. Enter the command `<eTrustDir>/bin/seoload` to start the product main daemons.
3. Wait while the seoload command starts three daemons: database server, agent and watchdog.

Wait for all messages shown in Figure 14-3 to appear.

```

root@linux021 root]# /usr/seos/bin/seoload
eTrust kernel extension is already loaded.
Starting eTrust daemon. (/usr/seos/bin/seosd)
19 May 2002 07:24:47> WAKE_UP : Server going up
19 May 2002 07:24:48> INFO : Filter Mask: "WATCHDOG*" is registered
19 May 2002 07:24:48> INFO : Filter Mask: "INFO : Setting PV*" is registered
19 May 2002 07:24:48> INFO : Filter Mask: "INFO : DB*" is registered
19 May 2002 07:24:48> INFO : Filter Mask: "*seosd.trace*" is registered
19 May 2002 07:24:48> INFO : Filter Mask: "*FILE*secons*(*/log/*)*" is registered
Starting seosd. PID = 24308.
Starting seagent. PID = 24311
[Root@linux021 root]#

```

Figure 14-3 Sample eTrust Access Control start-up messages

Note that this example does not show watchdog starting and does not show the seagent loading the database image.

## 14.8 eTrust Directory

eTrust Directory is a highly scalable solution for large-scale, business-critical directory service applications. The only directory solution to deliver assured performance and reliability through the use of commercial RDBMs, it delivers the customer-proven capability to support over 20 million entries and 1000 searches per second. eTrust Directory enables administrators to manage information on employees, customers and services, and to deliver new e-business services with high levels of confidence and security.

The major components of eTrust Directory are:

- ▶ **DXserver:** a high performance Directory System Agent (DSA) that provides versatility and reliability via a number of features including access (DAP and LDAP) and server-to-server (DSP and DISP) protocols, high integrity security, distributed operation processing, ease of management, and a reliable data store (RDBMS)
- ▶ **DXtools:** a flexible set of utilities for importing, exporting, and synchronizing data that facilitates interaction with external data systems
- ▶ **JXplorer:** a powerful, feature-rich, Java-based LDAP GUI browser
- ▶ **DXweb:** a Web-based GUI browser
- ▶ **DXconfig:** a Web-based GUI configuration editor



## 14.8.1 Installing eTrust Directory

These installation instructions and the description in the next chapter apply to version 4.0. At the time of writing, version 4.0 is expected to become available. The S/390 port should be available shortly after version 4.0 becomes available for Windows, Solaris and Red Hat. The distribution CD-ROM includes versions for Linux Red Hat 7.0, 7.1, and 7.2. The S/390 version will be on a separate CD-ROM.

To install eTrust Directory on a Linux system, log in as root and run the dxsetup installation program:

```
# cd /cdrom/cdrom0/dxserver/unix/install
# ./dxsetup.sh
```

This will detect which operating system you are on and will apply the appropriate files. For example, if you are on Red Hat, the install gets the files from:

```
# cd /cdrom/cdrom0/dxserver/unix/linux_i686
```

For S/390 distribution, the install gets the files from:

```
# cd /cdrom/cdrom0/dxserver/unix/linux_s390
```

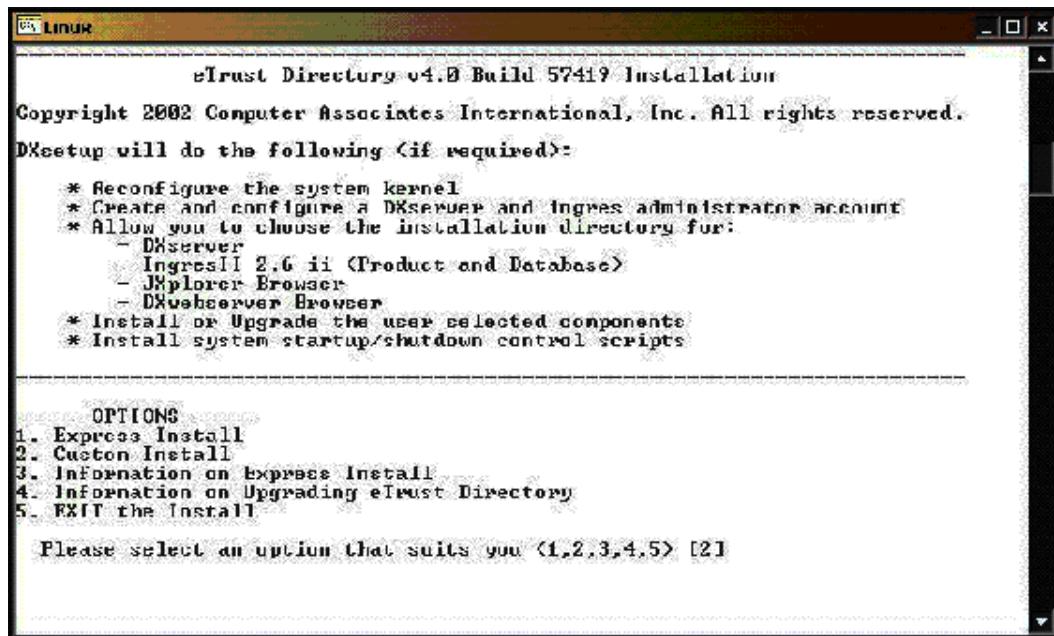


Figure 14-4 Sample installation screen for eTrust Directory

Note that you need to accept the licensing agreement defaults for a successful install.

When you load eTrust Directory, DXserver is started automatically. If you need to, you can stop the server and restart it during a session. The installation process also adds a number of directory samples.

## 14.9 eTrust CA-ACF2 Security (interface to Linux for zSeries and S/390)

Through an open source Pluggable Authentication Module (PAM), users can be authenticated to Linux systems directly through eTrust CA-ACF2 Security, providing enhanced password control and greater auditability, and eliminating the need to maintain user accounts on Linux.

See 14.11, “PAM support for Computer Associates External Security Managers” on page 370 for more information.

## 14.10 eTrust CA-Top Secret Security (interface to Linux for zSeries and S/390)

Through an open source Pluggable Authentication Module (PAM), users can be authenticated to Linux systems directly through eTrust CA-Top Secret Security, providing enhanced password control and greater auditability, and eliminating the need to maintain user accounts on Linux. See the section on PAM Support for Computer Associates External Security Managers for more information.

## 14.11 PAM support for Computer Associates External Security Managers

PAM is a flexible, open-source architecture for authenticating users primarily on Linux systems. Enhancements to eTrust CA-ACF2 Security for z/OS and OS/390 (eTrust CA-ACF2) and eTrust CA-Top Secret Security for z/OS and OS/390 (eTrust CA-Top Secret) allow these products to act as an authentication server for one or more Linux systems, eliminating the need for redundant security administration to define users on a system-by-system basis.

The PAM Module runs on the Linux/390 and allows the normal logon processing to perform the validation of the user ID and password on eTrust CA-ACF2 and eTrust CA-Top Secret. The routing of the validation process to eTrust CA-ACF2 or eTrust CA-Top Secret uses TCP/IP for its communication. This can be performed over a normal or secure TCP/IP connection using SSL/TLS.

The following information describes the Linux components. For detailed information on z/OS and OS/390 components, see the product documentation.

The Linux components are `pam_CA_esm.so` and `CA_esm_proxy`. The `pam_CA_esm.so` component is an interface for Linux PAM facility that uses z/OS and OS/390-based External Security Managers (ESMs) from Computer Associates to perform all the typical security management functions. This component operates as a PAM module and is called through PAM just like any other PAM module. This component was based in part on Luke Howard's `luke@padl.com pam_ldap` package, which is copyrighted and distributed by him under the GNU “LGPL version 2” license.

The component `CA_esm_proxy` component is a proxy server that sits between the PAM module and the ESM and handles the mechanics of getting requests from the Linux system to the ESM and returning responses. This component is not based on any prior code and is copyrighted by Computer Associates and distributed by Computer Associates under the GNU “LGPL version 2” license.

### 14.11.1 Installing a source distribution

Assuming you have received a source RPM file (`pam_CA_esm-*.src.rpm`) or a distribution tar file (`pam_CA_esm-*.tar.gz`), the process to build the package from source and install the executables is relatively simple and should be familiar to most open source developers. All steps should complete with no errors or warnings.

## Installing from a tar file

1. Enter `tar -xzf pam_CA_esm-<level>.tar.gz` to extract all the package files from the compressed tar file.  
  
*<level>* is the package level (for example 1.0). The files will be extracted into a directory named `pam_CA_esm-<level>` in the current directory (for example `pam_CA_esm-1.0`).
2. Enter `cd pam_CA_esm-<level>` to change the current working directory to the directory created by the previous command.
3. Enter `./configure ...` to configure the package. In addition to the options supported by all `configure` programs produced by the GNU `autoconf` tool, this `configure` supports the `--enable-ssl` and `--disable-ssl` options to enable or disable use of SSL/TLS encryption on the connection to the ESM. SSL support requires that you already have the "OpenSSL" package installed on your Linux system—if not, the next step will fail. "`--disable-ssl`" is the default.
4. Enter `make` to compile all the source code for the package and build the executable files.
5. Enter `su` to obtain root access before the next step.
6. Enter `make install` to install the executable and configuration files.

## Installing from a source RPM file

To install from a source RPM file, issue the following commands:

1. Enter `su` to obtain root access before executing the `rpm` command.
2. Enter `rpm --rebuild pam_CA_esm-<level>-<version>.src.rpm` to build the architecture-specific binary RPM file. *<level>* is the package level (for example 1.0) and *<version>* is the packaging level (for example 1). The binary RPM file is placed in `/usr/src/redhat/RPMS/<arch>/pam_CA_esm-<level>-<version>-<arch>.rpm` where *<arch>* is the architecture name for the Linux system (such as `s390` for Linux/390, `i386` for Linux on Intel processors).
3. Type `rpm -i /usr/src/redhat/RPMS/<arch>/pam_CA_esm-<level>-<version>-<arch>.rpm` to install the package from the binary RPM file built in the previous step.

### 14.11.2 Installing a binary distribution

Assuming you have received an architecture-specific binary RPM file (`pam_CA_esm-*.<arch>.rpm`), the process to install the executables is straightforward. All steps should complete with no errors or warnings.

To install from an architecture-specific binary RPM file, issue the following commands:

```
su
rpm -i pam_CA_esm-<level>-<version>.<arch>.rpm
```

## 14.12 eTrust Admin

eTrust Admin is an enterprise manager of user accounts for disparate namespaces within an organization. Some examples of the namespaces are eTrust CA-ACF2 Security, eTrust CA-Top Secret Security, eTrust Access Control, Oracle DBMS, Microsoft Windows NT and 2000, Microsoft Exchange, and many native Linux environments.

Use eTrust Admin to relate user accounts to a global user identity, allowing the organization to easily determine the systems to which a user has access. eTrust Admin provides role-based administration, allowing you to drop a global user on a predefined role by automatically

creating the proper system accounts for which the role is authorized. Likewise, eTrust Admin can suspend and delete accounts as the need arises.

You can use eTrust Admin to manage the native Linux 390 environment through the eTrust Access Control option, as well as by the eTrust Access Control environment itself.

The eTrust Admin server runs on a Microsoft Windows NT or 2000 server. The eTrust Access Control option and/or the eTrust CA-ACF2 Security or eTrust CA-Top Secret Security option is required to manage the LDAP PAM environment. CA Common Services on the mainframe are used as the communications method to eTrust CA-ACF2 Security and eTrust CA-Top Secret Security.

### **14.12.1 Using eTrust Admin in the native Linux 390 environment**

The following three steps are required to use eTrust Admin in the native Linux 390 environment:

1. Install eTrust Access Control on your Linux 390 system.
2. Install the eTrust Access Control option on your eTrust Admin server.
3. Authorize the eTrust Admin server to manage eTrust Access Control. To do this, create the appropriate TERMINAL Class eTrust Access Control rule. In addition, a trusted eTrust Access Control administrator ID must issue the user management commands.

## **14.13 eTrust Audit**

eTrust Audit collects enterprise-wide security and system audit data, filters collected information for consolidated viewing and reporting, and automatically triggers appropriate action upon detecting unusual or malicious activities on the system.

eTrust Audit can collect event information from a wide spectrum of sources, including UNIX and Windows NT and Windows 2000, Web servers, other eTrust products, mainframe systems and multiple RDBMS.

The eTrust Audit component structure of the Recorder, Router, Policy Manager and Collector enables collected audit events to be redirected and filtered throughout the environment. This architecture allows eTrust Audit through eTrust Access Control to collect events on Linux systems.

With eTrust Audit accepting data from any platform or application through its Submit API (SAPI) or eAudit SNMP recorder, virtually any platform, including mainframe, can submit its events to an eTrust Audit router for further handling.

### **14.13.1 Routing audit information from Linux**

You can use eTrust Access Control to collect events on your Linux machine and distribute them to eTrust Audit. To do this, you use the routing daemon SeLogRD on your Linux machine in conjunction with a Log Router on your eTrust Audit machine.

On your Linux machine running eTrust Access Control, perform the following administrative tasks:

1. Define the routing configuration file in the log subdirectory of the machine running eTrust Access Control. Typically, the path to this file is `/opt/CA/eTrustAccessControl/log`. To send all eTrust Access Control events to eTrust Audit, you use the `selogrd.cfg` configuration file.

Edit the contents of `selogrd.cfg` as in the example below:

```
eAudit Section
host <eTrust Audit Router Hostname>
```

where *<eTrust Audit Router Hostname>* is the hostname where the eTrust Audit Router service is running. You can limit the kind of events sent to eTrust Audit. Refer to the eTrust Access Control Utilities Manual for more information.

2. Execute the log routing daemon to start sending eTrust Access Control audit events to eTrust Audit. For example, from a Linux 390 shell prompt you could issue the command:

```
/opt/CA/eTrustAccessControl/bin/selogrd
```

where `/opt/CA/eTrustAccessControl/bin` is the eTrust Access Control installation directory.

Alternatively, you can start the `selogrd` routing daemon automatically when eTrust Access Control is started by using `seload`. To do this, set the `selogrd` token to `yes`. The `selogrd` token is found in the `daemons` section of `seos.ini`, the eTrust Access Control configuration file. This file can be found in the eTrust Access Control installation directory.

See the next chapter for information on using eTrust Audit to set policy for collecting event information.

## 14.14 BrightStor Enterprise Backup

BrightStor Enterprise Backup brings leading-edge data protection technology to businesses deploying Linux on the mainframe. It goes beyond simple data protection, using multiple data verification methods to enable maximum data integrity and recovery capabilities.

### 14.14.1 Installing BrightStor Enterprise Backup

Hardware requirements—G2 to G6 9672, a zSeries 900 (machine type 2064) or later, or a Multiprise 3000, with current versions of the Apache Web Server, `glibc` and `TCP/IP` available.

Disk space—At least 350 MB.

Software requirements—Either `SuSE Linux Enterprise Server 7` for `S/390`, `Red Hat Linux 7.2` for `S/390`, or `Turbolinux server 6` for `zSeries` and `S/390`. You must also have the `Apache` and `pdksh` Linux packages installed on your machine.

#### Installation steps

The BrightStor EB Server (`BEBsvr`) and Manager (`BEBmgr`) packages have to be installed on your system before installing any other BrightStor EB options. All the available options for BrightStor EB should be installed under the same directory where BrightStor EB has been installed.

You can use the `install` script to install BrightStor EB and all the available packages with BrightStor EB for `S/390`.

1. From the Linux system, log in as `root` and mount the CD-ROM to an available or newly created directory on your local system.
2. To start installation to the default directory (`/opt`), change the directory to:

```
/mnt/cdrom/Linux
```

Run the following script from the Linux command prompt:

```
# ./install
```

**Note:** Depending on the file system mount option that you have used during the mounting of the CD-ROM, you might need to provide the package location with this command.

For example:

```
# ./install /mnt/cdrom/linux
```

3. If you want to install the packages in a directory other than default (for example, /disc1), run the following command:

```
# ./install <package location> <destination>
```

For example:

```
# ./install /mnt/cdrom/linux /disc1
```

## Created directories

Once the installation is successfully completed, the following directories are created on your system:

- ▶ For the CA license, directory /ca\_lic
- ▶ For BrightStor Enterprise Backup, directory /opt/BrightStorEB (if you installed BrightStor Enterprise Backup on the default /opt directory)

## Configuration steps

1. Before configuring BrightStor Enterprise Backup, you need to set up the following environment variable.

For ksh (if you have installed BrightStor EB under /opt directory), set the following:

```
# export BEB_HOME=/opt/BrightStorEB
# export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$BEB_HOME/lib
# export PATH=$PATH:$BEB_HOME/bin:$BEB_HOME/sbin
```

Alternately, you can add the above variables in the root .profile file.

After issuing the above commands, the \$BEB\_HOME variable is set to the BrightStor Enterprise Backup home directory. Also, \$BEB\_HOME/bin and \$BEB\_HOME/sbin are included in the \$PATH variable.

2. Run csetup to set up your BrightStor Enterprise Backup domain and initialize the BrightStor Enterprise Backup database. Go to the \$BEB\_HOME directory and issue the following command:

```
# csetup
```

## 14.15 Advantage Ingres Enterprise Relational Database

Advantage Ingres Enterprise Relational Database is a complete solution for e-business information management, providing access to a wide variety of enterprise data, remote access and high availability replication. The e-business capabilities of Advantage Ingres Enterprise Relational Database deliver exceptional performance, giving organizations the speed they need to integrate diverse data sources and applications without sacrificing information integrity. This solution enables Linux users to create applications with GNU development tools (C compilers) and develop Internet commerce applications using the Apache Web Server.

### 14.15.1 Installing Advantage Ingres Enterprise Relational Database

The following is a summary of the steps for installing Advantage Ingres:

- ▶ Installing the software—Install the software from your distribution medium onto your system in the correct locations with the correct permissions.
- ▶ Setting required configuration parameters—Automatically set required configuration parameters to default values for installed Ingres components (ExpressInstall only) or set required configuration parameters for installed Ingres components (Install only).
- ▶ Starting Ingres—Start Ingres on your system so the Ingres system administrator can access it.
- ▶ Customizing your installation—Set optional configuration parameters to allow Ingres to run as desired.
- ▶ Preparing Ingres for general use—Perform additional tasks needed to prepare Ingres for its users, such as creating an automatic boot command, authorizing users, and creating databases.

These procedures assume you will either enter all needed information via the setup screens or use the defaults. You can also define default values for your terminal type, installation code, and distribution medium before running the install program using `setenv`.

### ***Installing the software***

The following procedure describes the first step in the summary above: installing the software. This procedure transfers all pertinent files from the distribution medium to the `II_SYSTEM` location for your installation on this node. You may use this procedure to install files for the stand-alone system installation configuration.

**Note:** Before running the install program, make a complete backup of your system.

To perform the installation interactively, using the forms-based utility:

1. Make sure your default directory is set to the `ingres` account, as follows:

```
% cd $II_SYSTEM/ingres
```

2. Set your path appropriately, according to the following:

For C shell:

```
% set path=($II_SYSTEM/ingres/{bin,utility} $path)
% rehash
```

For Bourne shell:

```
$ PATH=$II_SYSTEM/ingres/utility:$II_SYSTEM/ingres/
  bin: $PATH
$ export PATH
```

3. If your platform uses shared libraries, add the `$II_SYSTEM/ingres/lib` directory to your shared library path to enable Ingres utilities to use these libraries.

For C shell:

```
% setenv LD_LIBRARY_PATH /lib:$II_SYSTEM/ingres/lib:
  $LD_LIBRARY_PATH
```

For Bourne shell:

```
$ LD_LIBRARY_PATH=/lib:$II_SYSTEM/ingres/lib:
  $LD_LIBRARY_PATH
$ export LD_LIBRARY_PATH
```

4. Insert the CD-ROM that contains Advantage Ingres Enterprise Relational Database into the drive and mount the drive from a local host (if it is not mounted automatically).
5. As root, change to the Advantage Ingres Enterprise Relational Database root directory on the CD-ROM and enter the following command to run the install utility:

```
% install.sh
```

6. Select the option for your hardware platform from the list of supported platforms.
7. Select the option to install the software.
8. Press Enter to accept ingres as the installation owner.
9. Enter the path for II\_SYSTEM that you defined as your Ingres system directory (for example, /install).
10. Enter **y** to confirm that the setting for II\_SYSTEM is correct, or **n** to change the setting.

The install utility prompts you for the type of terminal you are using.

11. Enter the Ingres name for the terminal type, or select one from the short or long list of terminal types you can use for the display. The initial install screen, Ingres Installation Utility, appears.
12. Select an installation option, either PackageInstall or CustomInstall.

**Note:** If you choose not to install a particular component at this time and later find that you need it, you can always add that component to your current installation.

13. If you choose the PackageInstall option, the Install from Distribution Medium screen appears, and you can choose from two different packages. If you choose the CustomInstall option, the Custom Install from Distribution Medium screen appears, as shown in Figure 14-5.

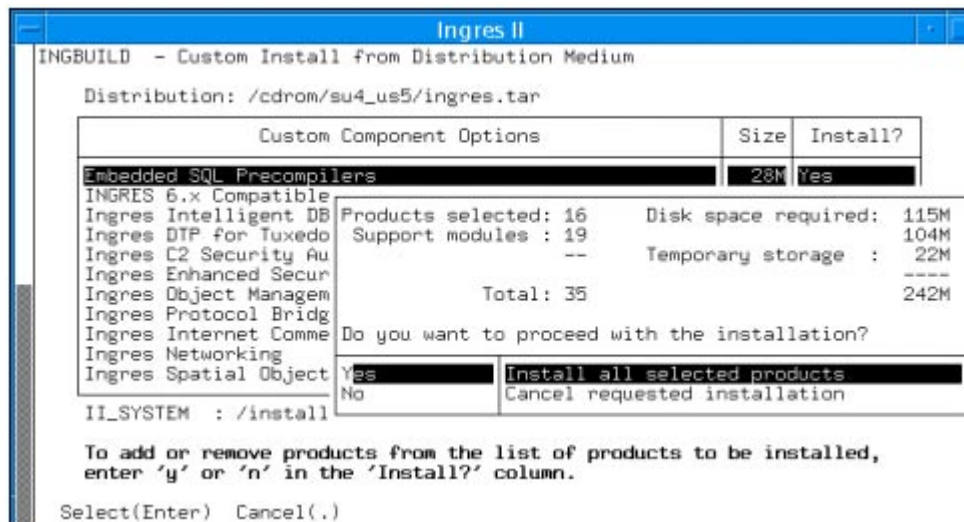


Figure 14-5 Custom installation dialog

You must select each component you want to install.

**Note:** The B1 Security Auditing option appears on this list if your operating system supports it.

In each of these screens, the cursor is in the Install column for the first component. By default, each component is set to Yes in this column in the Custom Install from Distribution Medium screen only.

14. Enter **y** or **n** and press Enter in the Install column for each component or package, as appropriate.

**Note:** To view a brief description of a component, use your cursor to move to the appropriate row, and then select GetInfo. The install utility displays a pop-up screen that describes the component. To return to the install screen, press Enter.

15. Select an installation method option, either Install or ExpressInstall.



- If you choose Install, the standard method, a separate setup procedure is used. If any components you selected are dependent on components you did not select, the install utility automatically installs the needed components. If a licensing or other problem occurs, a pop-up screen describes what to do. If you are not licensed for a needed component, it is recommended that you do not continue the install process, as it may produce unexpected results.
- If you choose ExpressInstall, an automatic setup procedure is used that assigns default values to all configuration parameters.

**Note:** Do not use ExpressInstall if you need to customize your installation. ExpressInstall puts all Ingres files in one location on one disk. After installing, you must change `II_TIMEZONE_NAME` unless you are in the NA-PACIFIC time zone. ExpressInstall upgrades an existing installation, but does not upgrade any existing databases.

16. Verify that you have adequate disk space in the `II_SYSTEM` location by responding to the disk space requirements pop-up screen that displays next.

- To proceed with the installation, press Enter.
- To stop the install process, select No and then press Enter. The install utility stops and returns you to the initial install screen.

The install utility verifies that each component was transferred properly from the distribution medium.

17. If you are using the ExpressInstall option, all selected components are installed and set up. The utility displays messages indicating what is being done. When it is finished, the following message is displayed:

All installed products are now ready for use.

You are ready to start Ingres.

## System requirements

Disk space and memory requirements are dependent upon a number of factors such as number of users, size of the transaction log file, number and size of databases, and so on. Ingres has the following hardware requirements and recommendations:

- ▶ **Number of disks**—At least two separate storage devices for your databases and checkpoint files are strongly recommended.
- ▶ **Disk space**—The amount of space required on any one particular disk is determined by the number of disks in your configuration, the packages you are installing, and the locations you choose for your files. The default file size for your primary and backup transaction log files is 16 MB. However, the recommended size is between 100 and 400 MB, or even larger.
- ▶ **Physical memory**—256 or more megabytes are recommended for a DBMS server installation (at least 64 MB are required).

Advantage Ingres for Linux supports either of the following Web servers:

- ▶ iPlanet Web Server Enterprise Edition version 4.1 on SuSE Linux (Note: iPlanet is not supported on Red Hat 7.x)
- ▶ Apache HTTP Server version 1.3

## 14.16 Advantage CA-XCOM Data Transport

Advantage CA-XCOM Data Transport is a cross-platform, value-added data transport solution offering high performance unattended file transfer with complete audit trails and reporting. It

provides a single solution for sending and receiving files, as well as for sending reports and jobs, to the industry's widest range of platforms by enabling high-speed, unattended data transfers between any two Advantage CA-XCOM Data Transport nodes on the network.

### 14.16.1 Installing Advantage CA-XCOM

Advantage CA-XCOM installation is a simple process using commands under the Linux system. As part of the installation process, you can modify the boot sequence to include automatic startup of the Advantage CA-XCOM daemon.

To install Advantage CA-XCOM, you must understand Linux administration and communication concepts. You will need the following:

- ▶ Superuser (root) privilege
- ▶ The Advantage CA-XCOM distribution media
- ▶ A PC that can read the installation files from the CD and FTP them to the S/390

Before installing Advantage CA-XCOM for the first time, create an xcomadm group by doing the following:

1. Log on as superuser (root).
2. Issue the following command:

```
groupadd xcomadm
```

A group called xcomadm is created in the /etc/group file.

To install Advantage CA-XCOM Data Transport:

1. Insert the Advantage CA-XCOM Linux/S390 CD-ROM into the CD-ROM drive on your PC running, for example, Windows 2000.
2. Open a command prompt on your PC. Change to the CD-ROM drive where the Advantage CA-XCOM Linux/S390 CD-ROM is.
3. At the command line, type the following command and press Enter:

```
ftp <ip address of your Linux system>
```

4. Provide userid and password when requested.
5. Within FTP, execute the following commands:

```
cd /tmp  
binary  
mput CA_XCOM*
```

**Note:** These commands do the following: changes to directory tmp, changes to binary mode, and transfers the Advantage CA-XCOM Linux/S390 installation files.

6. Respond with **y** to the question, then enter **quit** when the transfer is complete. The Advantage CA-XCOM files are copied to the system.
7. Telnet to Linux using userid root. Enter the following command to install Advantage CA-XCOM:

```
rpm -i /tmp/installation_file_name
```

**Note:** Use the following command to find the exact name of the installation file:

```
ls -l /tmp/CA_XCOM*
```

Successful installation messages will be displayed. If there are any problems during the installation, then errors will be displayed to indicate where the problem is.

8. To verify the successful installation, type:

```
rpm -q CA_XCOM
```

The current version of Advantage CA-XCOM Linux/S390 will be displayed.

Note that you can set Advantage CA-XCOM to start automatically each time the system is booted. See the product documentation for complete installation details.

## 14.17 Advantage Data Transport Agent

Advantage Data Transport provides end-to-end management of content delivery securely, reliably and efficiently across multiple platforms, protocols, networks and data formats, helping administrators synchronize, distribute, update, replicate and validate data across the enterprise. The Advantage Data Transport Agent integrates this platform with other platforms managed by Advantage Data Transport.

### 14.17.1 Installation considerations

Before you begin the Advantage Data Transport installation you should:

- ▶ Check that Unicenter NSM is installed on the enterprise.
- ▶ Decide which computers connected to your network will become part of the Advantage Data Transport network.
- ▶ Determine which Advantage Data Transport components are required for each of your computers.
- ▶ Be aware that each computer in the Advantage Data Transport network may require a User Name and Password to verify the user's authority to transfer data to and from that computer.
- ▶ Determine on which machines you wish to perform a typical agent install, or a custom install of any or all Advantage Data Transport components. Note that as you implement or expand your Advantage Data Transport network, you can add agents at any time.
- ▶ Check the readme file supplied with your Advantage Data Transport installation media for updates.

### 14.17.2 Installing Advantage Data Transport

A full unattended installation installs both the Advantage Data Transport Agent and the SDK onto a Linux machine. By default, the agent and SDK are installed into the `/usr/tngdts` directory of the target machine.

**Note:** If you move the installation files from one system to another using FTP, use binary mode to transfer the package file (the one ending in `.Z`) and ASCII mode for the other files.

To install the Advantage Data Transport Agent and the Software Development Kit, do the following:

1. Login as the superuser by entering `su` at the shell prompt.
2. Enter the password for superuser when prompted to do so.
3. From a shell prompt, change to the directory where the installation executable is located and enter:

```
sh ./install.sh
```

Once the installation has begun, a series of messages scrolls across your console, indicating that the installation is in progress. When the installation completes, a message indicates its success.

## 14.18 VM:Manager VM Management Suite for Mainframe Linux

The VM:Manager VM Management Suite for Mainframe Linux is an integrated solution suite designed to provide full VM Systems Management capabilities for customers running Linux on mainframes under VM, including those that utilize IBM's Integrated Facility for Linux (IFL) engines. The initial offering includes interfaces to z/OS, OS/390, and VM Tape Management Systems; z/OS, OS/390, and VM Automated Operations and Message Sharing; and z/OS, OS/390, and VM Resource Management and Accounting. In addition, you can select the security solution of your choice, either:

- ▶ eTrust CA-Top Secret Security for VM
- or
- ▶ eTrust CA-ACF2 Security for VM

### 14.18.1 Installing VM:Manager VM Management Suite for Mainframe Linux

Installation involves up to five phases:

1. Running Installation Phase One
2. Manually performing required tasks for each product component or having Automated Installation and Maintenance (AIM) automatically perform them for you
3. Running Installation Phase Two
4. Manually initializing each component and performing other required tasks
5. Releasing components to users

This section explains the order in which you should install components, then provides an overview of each of the five phases listed above. For complete details about installing all or any of these components, see the product documentation.

#### Installation sequence

You can install one VM:Manager for Mainframe Linux component, any combination of components, or all components in one installation session. The time required to complete Installation Phase One is the same regardless of the number of components you select, so you save time if you install as many components as possible at one time.

If you are going to use the Automated Component Management facility to enable AIM to create userids and allocate minidisks automatically, you must install eTrust VM:Director first in a separate installation session, then activate the facility, then install the other components.

#### Installation Phase One

In Installation Phase One, you select the product components you want AIM to install. AIM then scans either your CP source directory or object directory, whichever you specify, to gather minidisk and directory information. It also examines the CP nucleus on the IPL DASD volume and various CP data areas and control blocks to gather information about your DASD allocation, site size, and site configuration. AIM then:

- ▶ Builds a report called DISKSIZE LISTING that lists the minidisks you must allocate, specifies their sizes in device-dependent units, and gives the recommended block size for formatting each minidisk.
- ▶ Creates work files.
- ▶ Maps unallocated DASD areas.
- ▶ Builds template directory entries for service virtual machine, worker virtual machine, and system administrator or tape librarian userids for each component you are installing.
- ▶ Creates the VMRMANT installation database, INSDBASE VMSI. This database contains information about your site's system configuration and the components you are going to install. Each time AIM runs, it updates the database to reflect the current configuration information of your site. Installation Phase Two uses the INSDBASE VMSI database.

### **Component system administrator and tape librarian userids**

Unicenter VM:Account, eTrust VM:Director, Unicenter VM:Operator, and Unicenter VM:Spool require a system administrator userid. BrightStor VM:Tape requires a tape librarian userid. If you are installing one or more of these components, you can use separate userids or you can have VMRMANT serve as the system administrator or tape librarian userid. AIM allows VMRMANT to serve as the system administrator or tape librarian for more than one component and for more than one copy of a component. Or you can assign a userid other than VMRMANT as system administrator or tape librarian for one or more components.

### **Completing installation tasks for each component**

The instructions for creating userids and allocating minidisks assume you are logged on as the MAINT userid. If you are using another userid, substitute that userid for MAINT in this section.

1. For each component you installed, change the filetype of each template directory entry AIM created from XDIRECT to DIRECT.
2. Print the DISKSIZE LISTING report created by Installation Phase One on the VMRMANT 191 minidisk. This report lists minidisks you must allocate, specifies required sizes, and gives the recommended block size for formatting each minidisk.

### **Installation Phase Two**

In Installation Phase Two, AIM:

- ▶ Links to, accesses, and formats the minidisks allocated to the components you are installing
- ▶ Loads the program material for the components you are installing to the component userids created after Installation Phase One
- ▶ Loads component PUBLIC files to the VMRMANT 193 minidisk
- ▶ Builds the configuration files for the components you are installing
- ▶ Creates MDISKS files for the VMRMANT userid and the userids of the components you are installing
- ▶ If eTrust VM:Director is already installed on your system and the Automated Component Management facility is activated, makes sure component directory entries are correct for your site; creates service virtual machine, system administrator, and worker virtual machine userids for the components you are installing; and allocates minidisks to those userids. If you are installing components that use SFS, and VMRMANT is an SFS administrator of the file pools you are using, enrolls service virtual machine userids in file pools, creates directories required by the components, and allocates DASD for each file space required by the components.

To minimize repositioning of the distribution tape, AIM loads component program material in the order it occurs on the tape. AIM records in the VMRMANT installation database INSDBASE VMSI when it successfully loads a component. If you interrupt AIM and restart it later, AIM loads only those components it did not completely load before you interrupted it.

If you created component userids manually and neglected to create one, the manner in which AIM proceeds depends on the status of the Automated Component Management facility. If the facility is currently activated, AIM automatically creates the userid. If the facility is not currently activated, AIM issues a message when it reaches the component in question on the distribution tape, then skips that userid and continues with the userids that do exist.

If AIM encounters errors while loading a component, it proceeds to the next component. After AIM has loaded all components, it displays a summary of all errors it encountered.

### **Releasing components to users**

VM users communicate with most component service virtual machines through communications modules. The AIM procedure loads these communications modules and other public files to the VMRMANT 193 minidisk. To complete the installation, you must give the VM users access to these public files.

## **14.19 Unicenter VM:Account**

Unicenter VM:Account is a resource accounting, reporting, and capacity management system for the VM environment. Unicenter VM:Account provides more than project accounting, invoicing, workload balancing, and the preservation of data integrity. Because Unicenter VM:Account continuously collects VM accounting information, performing real-time validation and costing as it receives data, data centers have up-to-the-minute information, monitoring and tracking, and tighter control of resource consumption, project costs and budget limits. Unicenter VM:Account interfaces with other Computer Associates VM Software products to complement their use in the data center.

Unicenter VM:Account is a component of the VM:Manager VM Management Suite for Mainframe Linux and is installed as part of that overall Computer Associates product solution installation process or as a separate component.

## **14.20 Unicenter VM:Operator**

Unicenter VM:Operator is an automated console message management system for VM. It allows you to minimize human intervention by automatically recognizing and then responding to messages. Full-screen and extended color control capabilities and simplified online review of the operator console enhance operations productivity and reduce costly errors. Unicenter VM:Operator interfaces with other Computer Associates VM products to complement their use in the VM data center.

Unicenter VM:Operator is a component of the VM:Manager VM Management Suite for Mainframe Linux and is installed as part of that overall Computer Associates product solution installation process, or as a separate component.

## 14.21 Unicenter VM:Schedule

Unicenter VM:Schedule maximizes personnel and computer resources by allowing both end users and data center management to schedule jobs and tasks to run automatically. Unicenter VM:Schedule improves machine utilization through its off-peak scheduling capability, balancing the workload across shifts. Automatic scheduling of operations tasks improves system reliability and reduces operator errors. Unicenter VM:Schedule interacts with other Computer Associates VM software products to complement their use in the VM data center.

Unicenter VM:Schedule is a component of the VM:Manager VM Management Suite for Mainframe Linux and is installed as part of that overall Computer Associates product solution installation process, or as a separate component.

## 14.22 Unicenter VM:Spool

Unicenter VM:Spool is a spool space management system for the VM environment. Unicenter VM:Spool allows you to get the information you need when you need it so you can resolve problems before they become critical. Unicenter VM:Spool provides flexible, decentralized spool management and optimizes spool space while minimizing manual intervention.

Unicenter VM:Spool is a component of VM:Manager VM Management Suite for Mainframe Linux and is installed as part of that overall Computer Associates product solution installation process, or as a separate component.

## 14.23 BrightStor VM:Backup

BrightStor VM:Backup provides the features you need to effectively backup and restore CMS and non-CMS data in your VM environment. Through the use of BrightStor VM:Backup, you get unsurpassed reliability and performance of backup and restore. You also get automated tape management, catalog management, and administration.

BrightStor VM:Backup is a component of VM:Manager VM Management Suite for Mainframe Linux and is installed as part of that overall Computer Associates product solution installation process, or as a separate component.

## 14.24 BrightStor VM:Tape

BrightStor VM:Tape is a comprehensive tape drive and tape volume manager for the VM environment. It replaces potentially error-prone manual tape catalogue procedures with a fully automated control system. Having information on tape volumes and tape drives immediately accessible through full screen panels significantly improves operator productivity. And allowing BrightStor VM:Tape to control the allocation of your tape devices helps you to maximize your hardware investment. Comprehensive reporting and auditing provide you with the evidence that you are protecting your organization's data efficiently.

BrightStor VM:Tape is a component of VM:Manager VM Management Suite for Mainframe Linux and is installed as part of that overall Computer Associates product solution installation process or as a separate component.

## 14.25 eTrust VM:Director

eTrust VM:Director is an efficient and comprehensive directory and DASD management system for VM. It significantly reduces the time required to manage the resources defined in the VM directory and prevents costly errors. The VM directory is the lifeblood of any VM system, but the time-consuming and error prone methods used to update and maintain the directory are very costly. The VM directory is where users and their virtual machines and mini-disks are defined. An error, such as accidentally overlaying two users' minidisks on the same physical storage, is not only possible, but likely.

The VM directory consists of definitions in a file that must be maintained manually. It's up to your system administrator to ensure that each and every user is defined properly, and that disk space is being used efficiently. With the manual systems employed in native VM, it's not so much a question of if a mistake will be made, but when. With eTrust VM:Director, you can safely delegate aspects of directory management to your users without loss of control.

eTrust VM:Director is a component of VM:Manager VM Management Suite for Mainframe Linux and is installed as part of that overall Computer Associates product solution installation process, or as a separate component.

## 14.26 Unicenter CA-Explore Performance Management for VM

Unicenter CA-Explore Performance Management for VM delivers a comprehensive scale of performance management and monitoring information, which offers focused information on the collaborative eBusiness infrastructure. This new technology permits you to track mission-critical applications, baseline system performance, and integrate information across multiple environments. It is a system monitor product designed for use with z/VM and VM/ESA.

The increased demand for VM operating systems has also meant an increase in VM performance problems. Performance tuning is especially important when several operating systems are running on one CPU. Important considerations such as "Why are the CMS users receiving such good response time while my OS/390 guest is running so poorly?" and "How much are my CMS users affecting my production VSE guest?" can turn into problems unless they are addressed quickly. Unicenter CA-Explore for VM transforms performance management by bridging the gap between performance problems and high-tech solutions.

Unicenter CA-Explore for VM provides the ultimate in VM performance management. The product allows VM data centers to create reports and track vital aspects of system activity. The benefits include:

- ▶ Views, in real time, of the performance of the routers
- ▶ CMS and CP interface
- ▶ Split-screen mode
- ▶ Color and highlighting capabilities
- ▶ History reporting to determine exactly when performance problems occur
- ▶ Advanced visualization to reduce training time
- ▶ End-to-end response time collection
- ▶ Scaleable performance data
- ▶ Support for DB2 under VM
- ▶ Threshold monitoring



- ▶ Diagnostic commands
- ▶ Resource Control Facility

### Solve your VM performance problems

By using Unicenter CA-Explore for VM real-time and batch reporting facilities, a window is provided to see what is happening in your system. The various viewing and reporting features of Unicenter CA-Explore for VM make it the best performance management product today. The components include:

**Real-time**—Provides online monitoring of system performance based on events as they occur. The displays are tabular and plot report panels about different kinds of system activity as it occurs. You can also display CP control blocks, counters, and storage.

**Flashback**—Displays information about past performance. Flashback displays are similar in format to real-time displays. The flashback function allows you to display tabular and plot panels about different kinds of recent activity.

**Report writers**—Generate batch reports and plots for performance analysis. Unicenter CA-Explore for VM has the following report writers:

- ▶ EMAP (CA-Explore Monitor Analysis Program). Use for generating customized reports and plots for analyzing system performance. EMAP gives information not available to the Unicenter CA-Explore for VM history report writers, such as detailed data about the dispatcher and scheduler overhead. EMAP uses data from the CP Monitor format log file created by Unicenter CA-Explore for VM or data collected by MONWRITE. You can also use EMAP to archive or list performance data files.
- ▶ DASD reporting (EXPLDASD). Use for generating reports for analyzing DASD seek performance. Data is extracted from the CP Monitor format log file.
- ▶ Unicenter CA-Explore for VM history reporting (EXPLHIST). Use for generating canned reports and plots for analyzing performance data for VM. EXPLHIST uses performance data collected by the Unicenter CA-Explore for VM real-time monitor.

In addition to the main EXPLORE subsystems, Unicenter CA-Explore for VM includes a number of other components, some of which are:

**SQL/DS Data Collector**—Collect SQL/DS performance data for use with the real-time and history reporting subsystems.

**Resource Control Facility**—Monitor and control use of critical operating system resources by VM users.

**Utilities**—Manage and process disk and tape files containing Unicenter CA-Explore for VM and CP Monitor data.

## 14.26.1 Installing Unicenter CA-Explore Performance Management for VM

Unicenter CA-Explore Performance Management for VM is installed using z/VM or VM/ESA running in ESA mode.

Computer Associates provides the following materials for installation and use of Unicenter CA-Explore for VM.

- ▶ Basic Software
- ▶ Precoded procedures and CMS EXEC 2 and REXX files
- ▶ HELP files

- ▶ EMAP
- ▶ Comprehensive documentation that explains how to use, install, customize and maintain Unicenter CA-Explore for VM

### Summary of installation steps

The following steps summarize the installation of Unicenter CA-Explore for VM installation process. Review this list before attempting to install Unicenter CA-Explore for VM. You can also use it as a checklist when installing the product.

1. Define the Unicenter CA-Explore for VM service machine.
2. Load EXPLORE programs from the distribution tape to the EXPLORE service machine.
3. Download the Acrobat PDF file.
4. Link-edit the Unicenter CA-Explore for VM programs.
5. Define access to Unicenter CA-Explore for VM for all or selected users.
6. Define the DCSS (discontiguous saved segment) to be used by the CP Monitor facility.
7. Customize Unicenter CA-Explore for VM.
8. Perform control block mapping. This step is optional.
9. Set up Unicenter CA-Explore for VM security. This step is optional.

## 14.27 CleverPath Portal

CleverPath Portal uses Java 2 technology to allow the product to run on the greatest number of platforms. The product has been tested with Red Hat Linux 7.2, 7.1, 6.2, and Java Runtime Environment (JRE) 1.3.

### Pre-installation considerations

The CleverPath Portal Java process requires dedicated application memory; additional memory will be required by other applications and your operating system. The total amount required will depend on the applications being run and the operating system you are using.

By setting the Java startup parameters, you can change the amount of memory available to CleverPath Portal. For example, to set a minimum memory size of 128 MB and a maximum size of 512 MB, you would set the following parameters:

```
Unix: [INSTALL_DIR]/portal.sh
      TOMCAT_OPTS="-Dportal.dir=[INSTALL_DIR] -ms128M -mx512M"
```

Linux servers require a 300 MHz processor, at least 512 MB dedicated application memory, 100 MB hard disk space for installation with additional disk space for content, and 150 MB database table space with additional space available as needed.

If the Linux environment you are installing on does not have uuencode and uudecode installed, the portal installation will fail. Install these utilities as directed by the Linux environment vendor.

### J2EE and servlet considerations

CleverPath Portal runs as a Java Servlet and conforms to the Java Servlet 2.2 and JSP 1.1 specifications. A web server is not required to install or run CleverPath Portal, since it ships with Tomcat 3.3a (<http://jakarta.apache.org/tomcat/>). CleverPath Portal has been fully tested in the Tomcat environment, and is designed to run in any servlet environment that supports the Java Servlet 2.2 and JSP 1.1 specifications.

## Database considerations

CleverPath Portal uses Java JDBC 2.0 to connect to a relational database. By default, the Intersolv SequeLink 4.5.1 JDBC driver is installed and used, but CleverPath Portal can connect to a database using any JDBC 2.0-compliant driver. Only JDBC drivers with a .jar extension can be used.

## Browser considerations

CleverPath Portal on Linux has been tested with Netscape 6.2 and 6.1 and Netscape Navigator 4.7 and 4.0.

### 14.27.1 Installing CleverPath Portal

The CleverPath Portal CD-ROM contains the software you need to install the CleverPath Portal Server. The CleverPath Portal client requires no installation—client users access the CleverPath Portal Server by entering the Server URL in their Web browser and supplying their user name and password when prompted.

You can install and run CleverPath Portal Server on more than one Linux server to distribute server processing among different systems. To set up a multiple server environment, do the following:

- ▶ Use the same database for all the servers. You must specify the database when you install the CleverPath Portal Server software on each system.
- ▶ Designate one of the servers as the main server.

The CleverPath Portal Server setup program consists of a series of easy-to-use wizards. The wizards use information you supply to configure the CleverPath Portal Server.

To install the CleverPath Portal Server:

1. If you have not already done so, set up your database and create a user with permissions to create and alter tables.
2. Close all other programs you may be running.
3. Start your database. The database must be running to successfully complete the installation.
4. Log in to the server as the administrator or as a member of the administrator group.
5. Insert the CleverPath Portal CD-ROM. Execute the command `./setup.sh` from the CD-ROM's "portal" directory.
6. Follow the instructions for the Installation Wizard.

**Note:** The portal cannot be installed into a directory with a space in the name because of limitations in the way the software handles filenames.

## Starting the CleverPath Portal Server

After the CleverPath Portal Server has been installed, you can log on to the Portal and begin using it immediately.

1. Navigate to a prompt window and use the `cd` command to change to the directory in which you installed the Portal Server. At the shell command line, enter `/portal.sh` and then press Enter.
2. Open the Web browser, then type the URL of the CleverPath Portal Server in the location or address field, and press Enter. For example, the URL might resemble the following:

```
http://portalserver:8080
```

where *portalserver* is the server name (or IP address) of the CleverPath Portal Server and 8080 is the port that CleverPath Portal Server monitors. The port number was specified during setup. The default port is 8080.

The Portal Login dialog opens.

3. Enter your user ID and password in the appropriate fields, then click OK. Initially, the user ID for the Admin user is *admin* and the password is *admin*. You may want to change the password after you log in for the first time.

The CleverPath Portal opens, displaying the workplace for the admin user.



# System management with Computer Associates software

In this chapter we describe the use of Computer Associates software products for systems management on Linux for zSeries and S/390.

# 15.1 Operations

## 15.1.1 Availability management

### Unicenter Management for Web Servers (Apache Agent)

This section describes some of the features and considerations for using the Apache Server agent as part of Unicenter Management for Web Servers.

A Web server agent is an intelligent, programmable agent that monitors the availability, health, and performance of a Web server. The agent is modeled as a collection of components. Each component has its own unique structure for information pertaining to an agent instance plus a common summary status. The agent automatically correlates the status of servers, services, disks, invalid URLs, polled URL counters, server health, polled counters, and events, and it provides a WebCrawler that lets you test the validity of links and references on your Web site.

Features include the following:

#### Status

The agent maintains and reports three levels of status: normal, warning, and critical.

- ▶ Normal status means that the monitored component does not reach or exceed any threshold value. A green check mark indicates normal status.
- ▶ Warning status means that the monitored component reached the warning threshold value, but not the critical threshold value. A yellow exclamation mark indicates warning status.
- ▶ Critical status means that the monitored component reached the critical threshold value. A red X indicates critical status.

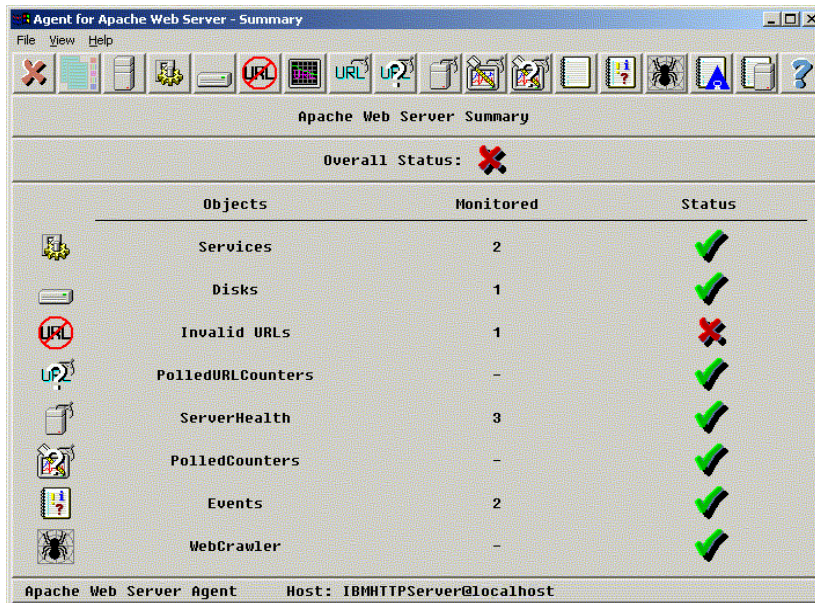


Figure 15-1 Sample Agent for Apache Web Server Summary

#### Thresholds/Actions

You can define warning and critical thresholds for the monitored components.

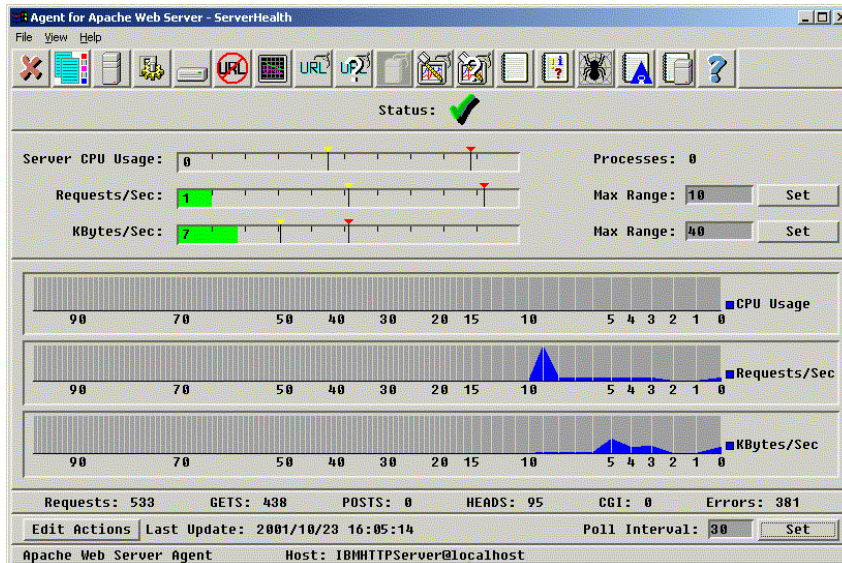


Figure 15-2 Sample of ServerHealth and threshold settings

You can also define local actions to take when warning or critical thresholds are reached. A local action is a command executed on the system where the agent is running.

The agent also maintains an audit log that includes the command line, a time stamp, and the status. If the command fails, an event is reported. More importantly, any time a local action is taken, an event is recorded so that you have an audit trail.

### Counters

You can customize the agent to monitor a set of counters that indicate the performance of the Web server or services (URLs).

### URLs

You determine which services and URLs you want to monitor.

### Statistics

The agents display the current value, as well as its high, low, rolling average, and standard deviation. Also, it displays the counts of warning and critical statuses and the time of the last poll.

### Auto-Restart

The agents can automatically monitor and restart stopped services and daemons, such as HTTP and FTP. They can even reboot the server after a specified number of failed restart attempts.

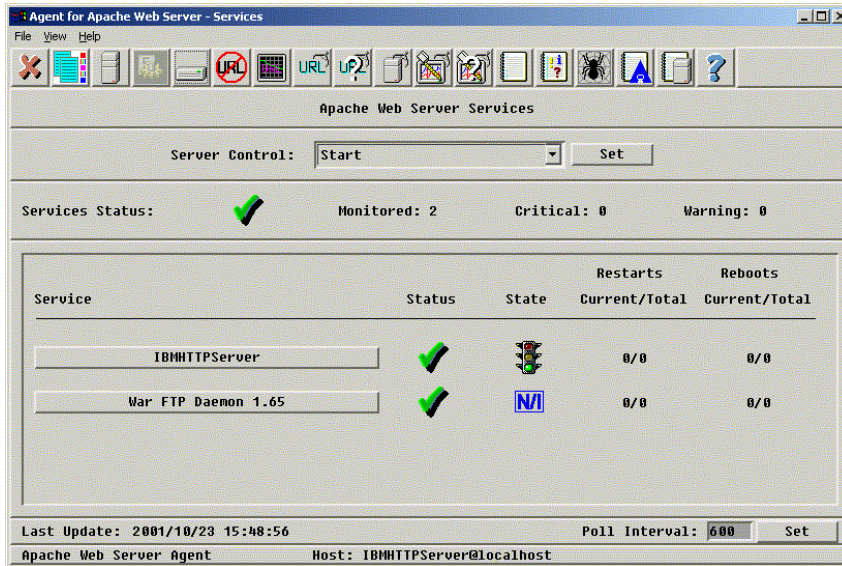


Figure 15-3 Sample dialog for managing Apache Web Server Services

### Disk Space

The agents monitor workspace, log files, and Web content disks for available free space.

### Events

The agents monitor critical events reported by the Web server. When appropriate, the agent sets the component status to warning or critical and reports to the manager for correlation and action. These events are stored locally for later review, providing a convenient place to view only those events important to the maintenance of the Web server. You can flush the store on demand.

### Resets

You can reset the alert counts and statistics on demand. In addition, you can specify an automatic reset interval. The agent resets the values to zero automatically at the specified interval. If you set the automatic reset interval to zero, you disable the automated reset feature.

### Invalid Links

The WebCrawler displays a list of URLs with bad links and references. You can associate a severity level (warning or critical) based on the number of bad links. You can also assign an action to execute when the warning or critical threshold is reached.



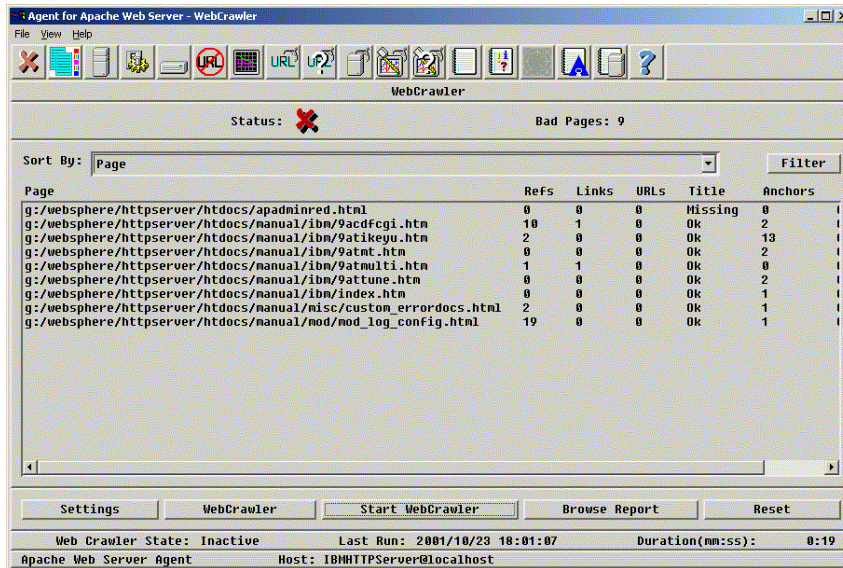


Figure 15-4 Sample WebCrawler display of bad links and references

You can run multiple instances of a Web server agent on a system. Multiple instance support is available for all Web server agents.

## Starting and stopping the agent

After the agent is installed, you can start it, determine its status, and stop it from the command prompt using the following commands.

### Starting

To start the agent, enter the following case-sensitive command:

```
apAgent start
```

The following is displayed if the agent starts without problems:

```
RUNNING apAgent:apAgent <OK>
```

If the agent is already running when you issue the command, the following appears:

```
-1 : Already running
```

### Learning Status

To learn the status of the Web server agent, enter the following command:

```
apAgent status
```

The following is displayed if the agent is running:

```
RUNNING apAgent:apAgent <OK>
```

The following is displayed if the agent is not running:

```
STOPPED apAgent:apAgent <OK>
```

### Stopping

To stop the Web server agent, enter the following command:

```
apAgent stop
```

The following is displayed if the agent is running and stops without problems:

```
STOPPED apAgent:apAgent <OK>
```

The following is displayed if the agent is already stopped:

```
-1 : Resource is not running
```

## Configuring the Apache Agent

The configuration of an agent determines what objects the agent monitors, how the agent gathers resource data, and how the agent assesses the state of managed objects. By using a single file called a configuration set, you can specify default values so that whenever you stop and restart an agent, the same initial settings are used.

For additional information on configuring the Apache Agent, see the Computer Associates product documentation.

## Unicenter NSM Performance Management

This section describes some of the features of Unicenter NSM Performance Management.

As fundamental business operations become more and more dependent on e-business, there is an increasing need for tools that enable the efficient performance management of the servers and devices within the enterprise, thereby assisting IT departments to reliably and economically meet their business goals.

To gain control of these critical, distributed, and in many cases disparate systems, systems administrators need to be able to understand how these systems are loaded, so that they can obtain the best response times, optimize throughput, and ensure long-term reliability. What is needed are tools that can analyze and present this data in a meaningful and consistent form, enabling visualization of the performance of their systems and devices, not just for a given point in time, but over days, weeks, months and even years. In order to achieve an effective Performance Management strategy, it is therefore important to address two fundamental requirements:

- ▶ The day-to-day management of unexpected performance problems across the enterprise
- ▶ The ability to perform long-term planning, trend analysis, and reporting

Unicenter NSM Performance Management is designed to address these two fundamental requirements, and provides a comprehensive and holistic solution to the complex challenge of managing the performance and throughput of the many heterogeneous servers, workstations, and devices on which mission-critical e-business applications depend.

Unicenter NSM Performance Management provides four graphical applications to visualize, analyze, report and configure the Performance and Resource Usage data collected by the Performance Agents from the many disparate servers and devices that are typically present in a large enterprise environment. These applications are:

- ▶ Performance Scope
- ▶ Performance Trend
- ▶ Performance Configuration
- ▶ Chargeback

The Performance Agents collect both real-time and historical performance data from the environment being managed. Historical performance data is stored in a 3-dimensional data model that enables powerful statistical processing and management of the data. The four graphical applications listed provide the ability to view, analyze, report on, and manage this data via intuitive mechanisms, using the same techniques regardless of the platform or environment where the source data originated.

## Performance Scope

Performance Scope is designed to assist systems and network administrators by allowing them to monitor the current performance of the enterprise and perform real-time analysis of any performance problems or outages that may occur.

Fundamentally, Performance Scope provides a real-time view of the performance of the servers and devices across the enterprise using platform-independent techniques, thus allowing disparate operating systems and devices to be visualized and analyzed using the same methods and techniques.

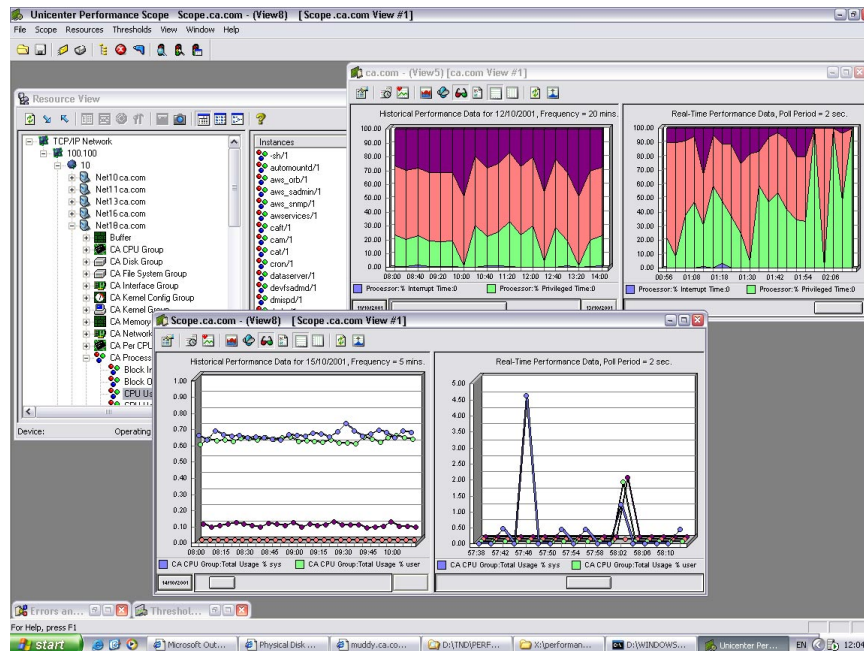


Figure 15-5 Sample Unicenter Performance Scope view showing historical and real-time data

Each Performance Scope view seamlessly joins real-time and historical data, allowing the performance of resources to be viewed both now and from the past. This functionality allows roll-backs in time to determine the point at which a problem first occurred.

Each Performance Scope view can display the performance of several different resources. Multiple concurrent views are also supported, allowing the monitoring of many machines.

One or many resources from one or many servers can be monitored by simply dragging and dropping resources from the Resource View.

Comprehensive tool tips are available to explain the meaning and importance of the many thousands of metrics monitored. Predefined thresholds can also be assigned to resources, causing alarms to be generated and actions to occur in the event of a threshold breach.

Performance Scope can monitor thousands of different resources including UNIX, Windows 2000/NT, NetWare, Linux, and SNMP-based metrics. This enables IT organizations to use consistent techniques and methodologies to measure server performance data, react quickly to unexpected incidents, and efficiently resolve problems regardless of system type.

## Performance Trend

Performance Trend is an easy-to-use mechanism that allows IT organizations to operate preemptively. Performance Trend enables the loading of complex historical performance data

from a wide range of disparate systems and devices into a spreadsheet environment, to support easy reporting and analysis of this data.

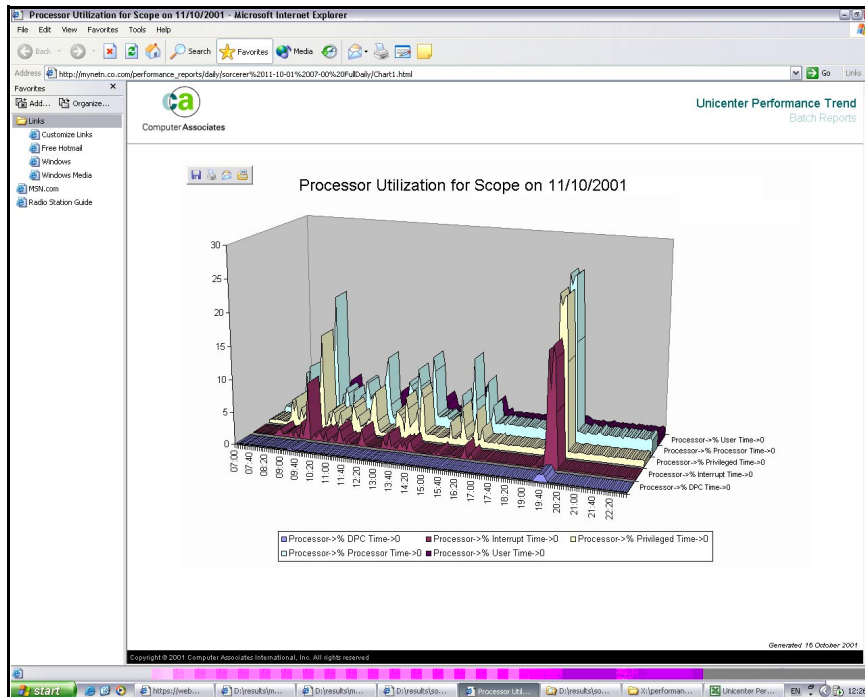


Figure 15-6 Sample Unicenter Performance Trend report

Use Performance Trend on an enterprise-wide level to:

- ▶ Identify which servers and devices are heavily loaded
- ▶ Observe patterns of activity and use of applications and servers
- ▶ Identify problematical trends
- ▶ Investigate the impact of moving applications and users to other servers
- ▶ Determine the effect of running workloads at different times
- ▶ Provide automated, graphical reports

Features include the following:

### **Ad Hoc Chart Creation**

This feature allows the rapid generation of performance charts and is primarily intended for quickly analyzing a few metrics in a one-time scenario.

### **Pre-defined Chart Creation**

Performance Trend provides out-of-the-box charts that offer immediate value, allowing for the analysis of your systems from day one. These predefined chart definitions can be modified and/or supplemented by your own staff.

### **Resource Correlation Engine**

Correlation is a statistical function for determining relationships between two sets of data. These metrics could be real-world metrics (such as the temperature in the office and the air conditioning setting), or computer-related metrics, such as the number of processes and the amount of memory being used.

The Correlation equation attempts to determine a value, which represents the degree of relationship between two sets of data. Searching and locating correlations in performance data is critical, since finding a high correlation between two metrics can often highlight and explain an otherwise unexplained performance problem. However, performing correlations across vast volumes of data from disparate sources has been difficult and time consuming.

Performance Trend provides a solution to this problem. The Resource Correlation Engine is able to search for interesting correlations between metrics gathered from a single machine/device across a single day, a single machine/device across many days or periods, or many machines/devices.

### ***Chart Configuration Wizard***

The Chart Configuration Wizard provides a simple and intuitive interface for maintaining predefined chart definitions. The wizard allows you to create new definitions, to remove obsolete definitions, and to modify existing definitions.

### ***On-chart thresholding***

Performance Trend can analyze performance metrics to determine if any data point exceeds a predefined Warning and/or Critical threshold. Graphical and text-based Threshold Reports provide breach severity and duration information.

### ***Automated batch creation of reports in HTML format***

Charts can be saved as HTML files, both interactively and in batch. This allows Performance Trend to automatically generate performance reports in off-peak periods (overnight) and then publish these reports to the intranet/Internet with no direct human intervention.

### ***Customizable macros***

A high degree of extensibility is afforded by a series of macros that can be used to perform useful data conversion and analysis functions. As an added benefit, the source for these macros is also supplied, allowing them to be modified and tailored to your requirements.

## **Performance Configuration**

The Performance Configuration component enables the configuration, distribution, and management of Performance Management polices across the e-business enterprise.

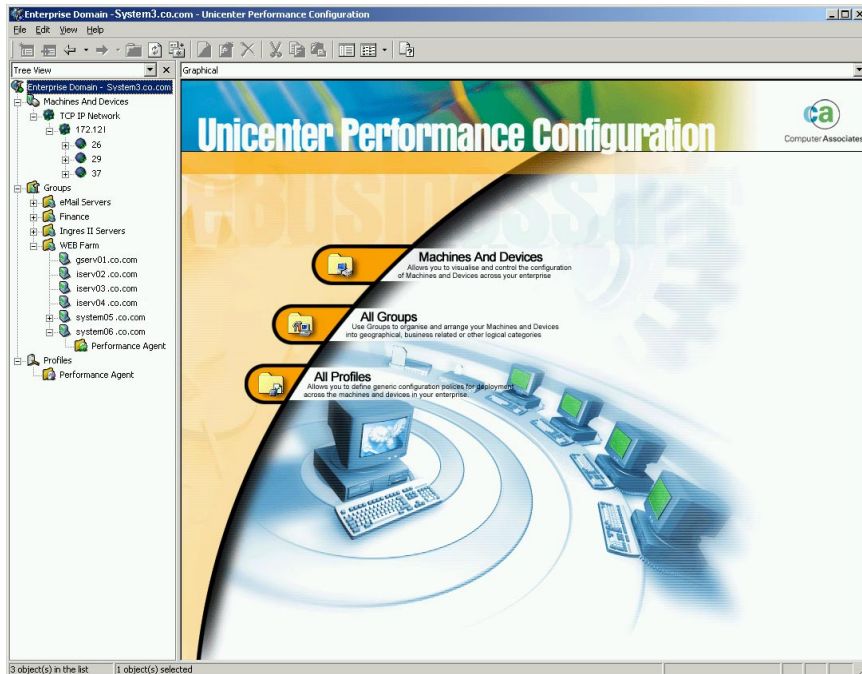


Figure 15-7 Sample Unicenter Performance Configuration initial view

Fundamentally, Performance Configuration uses advanced visualization to report on:

- ▶ The machines and devices that exist across the enterprise
- ▶ Customizable configuration groups
- ▶ Performance management policies

Complex Performance Monitoring/Management Policies can be easily created using advanced visualization. These Rule groups can be dragged and dropped onto machines/devices or configuration groups, causing immediate and dynamic reconfiguration of the agents on the selected remote nodes.

Performance management configuration rules have the additional benefit of being operating system- and device-independent, providing a layer of abstraction from the specifics of each platform. This allows a single performance management policy to be created and deployed to many disparate nodes, so that many machines with different operating systems can be managed equally.

When performance agents are initially installed on a given node, they begin executing in a default profile, ensuring that agents are immediately productive.

## Resource Accounting and Chargeback

Resource accounting and Chargeback capabilities provide a simple yet powerful method of attributing resource accounting data obtained from heterogeneous platforms across the enterprise to real-world charge groups. This enables you to provide better service by identifying how each user in your organization is using distributed systems, allowing you to allocate charges, maintain budgets, and to generate billing invoices and reports.

## 15.1.2 Health Monitoring

### **Unicenter Network and Systems Management**

This section describes some of the features and considerations for using Unicenter Network and Systems Management.

Unicenter Network and Systems Management manages the health and availability of operating systems and provides basic status management on all infrastructure elements such as network devices, business applications and database systems. Powerful Auto Discovery builds a database with information on system elements and populates 2D and 3D dynamic visualizations. The Historian feature keeps systems administrators informed about past events and object status, and predictive management capabilities inform them of possible bottlenecks in their systems.

Features include the following:

#### ***Event Management***

Robust event management capabilities can handle volumes of system-related events, correlate and prioritize them, and invoke actions according to business policies while consolidating events from multiple platforms and sources. Event management policies can suppress events, forward messages to other platforms, and issue commands locally or on other platforms.

Unicenter Event Management integrates with CA's automation solutions Unicenter CA-OPS/MVS Event Management and Automation, Unicenter SOLVE:Operations Automation, and Unicenter Network and Systems Management Automation Point.

#### ***Unicenter Real World Interface***

Manage information on the entire enterprise through the integrated, Web-based user interface.

#### ***Business Process Views***

Unicenter NSM offers the unique ability to define Business Process Views and classify managed objects into them.

#### ***Common Object Repository***

The object repository is a self-managed data warehouse that stores information about managed objects, their properties, and relationships.

#### ***Enterprise Discovery***

Through its automated processes, Unicenter NSM discovers networked objects and all resources within the enterprise, including systems, desktops, networks, applications, and databases.

#### ***Enterprise Management Console***

Integrated with the Real World Interface, the Enterprise Management Console enables users to centrally monitor events as they occur on multiple platforms.

#### ***Agent Technology***

Unicenter NSM uses intelligent and proactive agents to gather information about the IT infrastructure. These localized agents are highly scalable and can automatically initiate corrective measures. They send alerts to Unicenter NSM managers about problems based on user-defined thresholds, which can be centrally or locally set.

## Distributed State Machine (DSM)

The Distributed State Machine monitors and manages resources across the networked enterprise according to your specifications. The DSM instructs agents to collect information about managed resources and can respond according to user-defined DSM policy as resource values or states change.

## Using Unicenter Network and Systems Management

Launch the Unicenter Explorer to access most of the functions. Once you have run Discovery, the Unicenter Explorer provides a number of different ways to view your enterprise.

The Network Topology view, for instance, displays your network in a hierarchical tree structure. At the lowest level you see the agents that are monitoring resources on the Linux system. Agents are available to monitor various databases—including Sybase, Informix, Oracle, and DB2—as well as processes, performance, and logs.

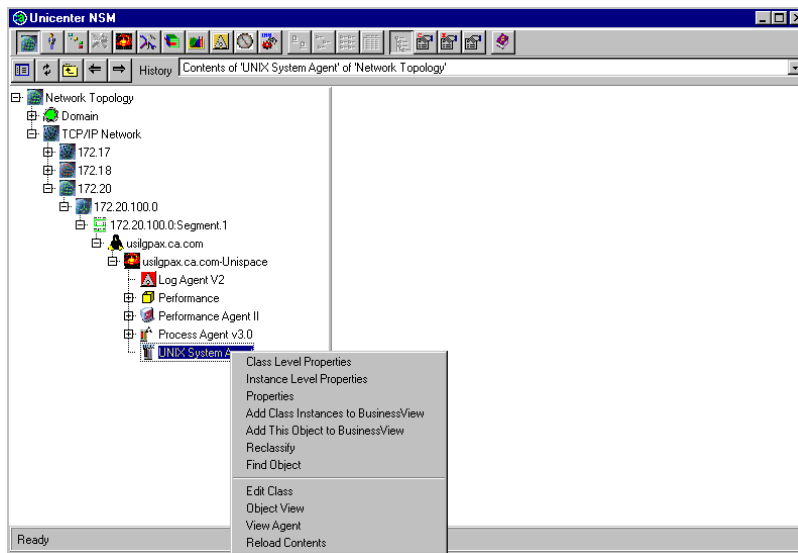


Figure 15-8 Sample Unicenter NSM Network Topology view

The Agent View, accessed by right-clicking on an agent object, provides a graphic representation of the health of any monitored resource in your enterprise, such as CPU usage of a monitored Linux system. You can display a graph showing CPU usage, or choose the detailed display, which presents detailed information related to the CPU being monitored by the agent.

**CPU Status:** The current status of the monitored CPU.

**Lag Value:** The number of consecutive breaches that have occurred.

**Lag Setting:** The number of consecutive breaches that have to occur for the state to change to warning or critical.

**CPU Value:** The total percentage of the CPU time that the processor was busy, that is, 100%, Idle%, Wait%.

**Idle:** The percentage of time that the CPU was idle.

**Wait I/O:** The percentage of time that the CPU spent waiting for blocking I/O requests.

**Total:** The sum of the CPU Value, Idle, and Wait I/O percentages.



**User:** The percentage of time that the CPU spent servicing user requests.

**System:** The percentage of time that the CPU spent servicing system requests.

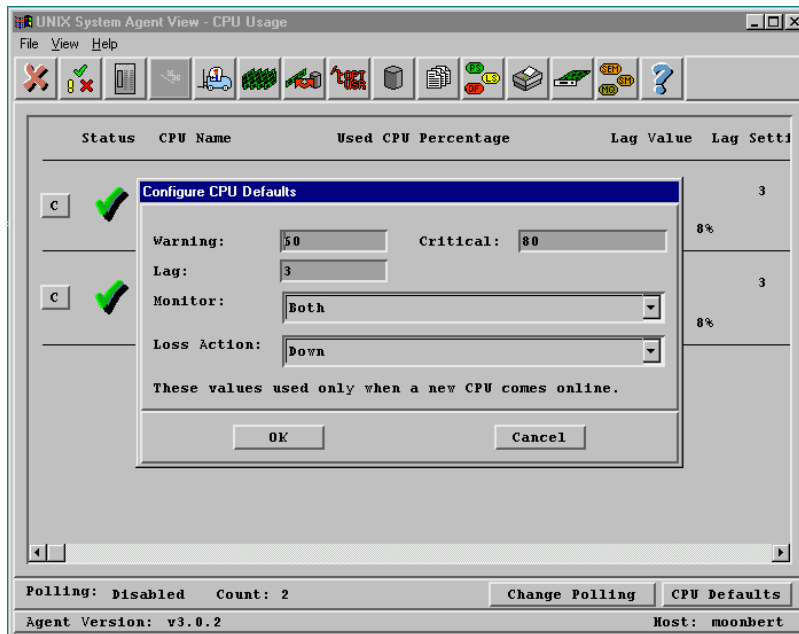


Figure 15-9 Example of configuring CPU defaults through the Agent view

Click CPU Defaults to set the warning and critical thresholds and the lag threshold for each newly discovered CPU. You can also indicate whether your want the warning threshold, critical threshold, neither, or both monitored.

Use Enterprise Management to access the many functions that enable you to manage your enterprise, such as Event Management, Problem Management, or Security Management.

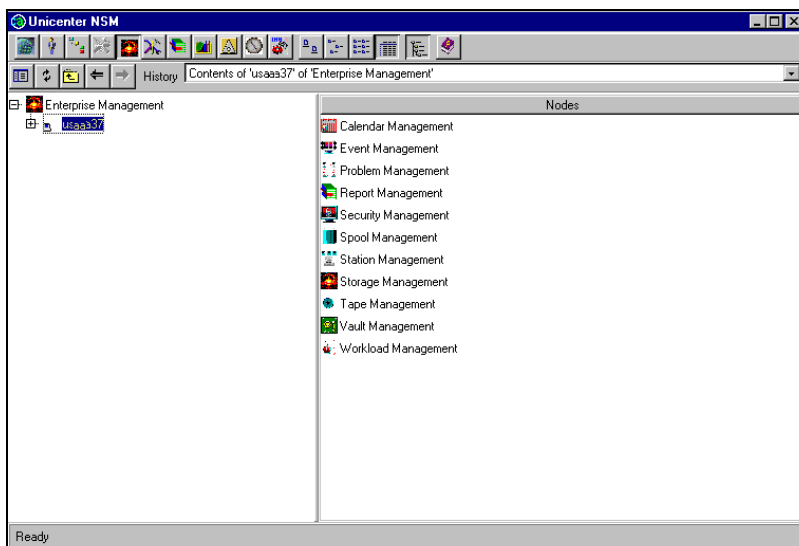


Figure 15-10 Example of the Unicenter NSM Enterprise Management view

Event Management is a powerful tool for handling events that take place anywhere in the enterprise. Event Management collects related network-wide messages for display at a single location, routing them to appropriate locations as needed. Created daily, the console log is a file containing all messages written to the event console on that day.

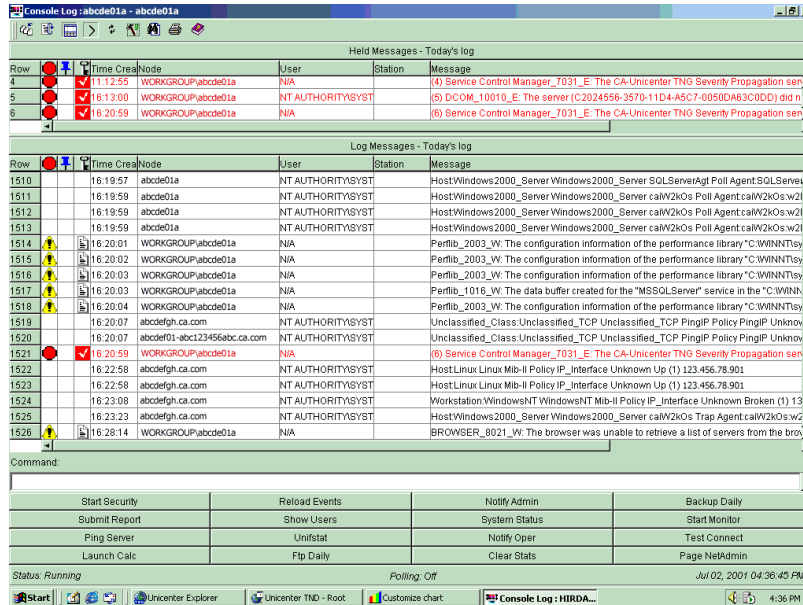


Figure 15-11 Sample Unicenter Event Management console log

Events, such as critical problems detected by an agent, cause messages to be sent to the Event Console. Messages sent to the Event Console can be intercepted and used to launch automated actions, which you define.

You create an automated response using the Messages function of Event Management. Specify the exact text of the message, and specify the action or actions that are to occur when the message occurs. After that, issue the **opreload** command from the Event Console; now the message/action pair is loaded into the database, ready to be invoked automatically when needed.

### 15.1.3 Automation

#### Unicenter Software Delivery Agent

This section describes some of the features and considerations for using Unicenter Software Delivery.

Unicenter Software Delivery controls all aspects of the installation process. In the Software Management architecture, when a product is being installed, the Software Management agent considers all aspects of the installation that have a bearing on other products. For example, it understands which components are shared between applications, so when one application is removed, it knows whether to remove the shared component.

The extended enterprise supported by Unicenter Software Delivery includes desktops, servers, laptops, ATMs, kiosks, and so forth. These platforms are supported on LAN, WAN or wireless networks. In addition, Unicenter Software Delivery fully supports PDAs and other devices built on Palm OS, Windows CE, and Linux with existing infrastructure.

Features include the following:

### ***Software management***

**Discreet Distribution:** Systems with fragile or high-cost connections receive software packages in a cost-effective, user-friendly manner. Distribution is in the background, and is sensitive to both CPU utilization and bandwidth usage.

**Software Catalog:** Users have the ability to choose the software packages to install. Using the Software Catalog, administrators can define catalogs and assign users to each catalog. Users can browse catalogs and choose optional software packages, while Unicenter Software Delivery maintains records of the software installed.

**Software Packager:** The Packager allows administrators to automate and customize the packaging process. It is installed on a packaging computer, and packages software products and data by installing a product on the packaging computer and recording the information required for successful installation of the package. It then generates an installable software image to be registered in the Software Library. For tracking and consistency, the Packager automatically versions new software packages, and can create delta packages, to send complete packages to new installations, while existing users receive only updates or changes.

**Staging Servers:** You can optionally install a Staging Server to improve network communications between Local Servers and Agents. Staging Servers intercept single jobs from the Local Server and distribute these jobs to each intended Agent recipient. Staging Servers can also collect communications from a number of Agents, such as computer attributes, and forward the information to the Local Server as a single request, rather than as multiple requests from many Agents.

**Dependency Management:** Before allowing any installation to proceed, Unicenter Software Delivery ensures that all hardware and software requirements and dependencies are met. It can automatically install and configure prerequisites if needed.

**Sequential Installations:** Unicenter Software Delivery allows sequential jobs to run only if previously dependent jobs are installed successfully. This synchronization ensures that all “pieces” are installed properly, preventing “inoperable” software from existing on a system. In addition, automatic rollback ensures that, if an installation in a sequence fails, all steps are automatically rolled back, leaving the system in its previous working state. Extensive error reporting is available for the administrator.

**Installation Parameters:** Installation parameters and user-specific settings can be resolved locally during installation, enabling easy installation and upgrade of groupware and e-mail systems.

### ***Automation and manageability***

**Computer/User Groups:** Computer/user groups can be created and maintained dynamically or manually. Dynamic groups can be based on hardware or software inventory information, information in Windows NT Domains, Active Directory, NDS, or any LDAP-compliant directory. Computer groups can be further organized into smaller Nested Groups to simplify the targeting of new software releases to the right computers or users. For example, a group can be based on a region, with groups based on cities beneath it, and, within each city, groups representing individual offices.

**Software Groups:** Software packages can be grouped together to reflect organizational standards. For example, a basic package can exist for most users, while special packages can be provided for the sales teams.

**Template groups:** Template groups, a combination of software and computer groups, allows the system to dynamically evaluate all computers and users in the group and ensure

installation of required software. Computers not conforming to a set policy are automatically updated with required software or release levels.

### ***Scalability support***

**Multi-tiered distribution:** Enterprise servers, local servers, staging servers, and workgroup servers provide the scalability to support hundreds of thousands of nodes. Unicenter Software Delivery also enables multi-tiered fan out bulk transfers using broadcast and multicast techniques from managers to individual systems, quickly delivering data and content while limiting network load.

**Tiered management architecture:** In large distributed organizations, the responsibility of delivering software is often regional or local, or a mixture depending on the software packages. Unicenter Software Delivery uniquely maps to these responsibilities, allowing distribution from three levels (Enterprise, Local and Workgroup).

**Powerful security:** Unicenter Software Delivery's powerful abilities to change and re-arrange the IT infrastructure demand an integrated security system to define and enforce access and restrictions for individual administrators. Security can be defined for any object in the system, including software packages, computers, groups, servers, etc. Several levels of security permissions are available for each object or group of objects (Full Control, Change, Manage, Read, View, No Access, and Special Access). Security policies can be defined in as granular a manner as needed, including specifying who can deliver a particular application package to a single machine, and can be inherited from the parent class, allowing both geographical restrictions and functional restrictions.

### ***Unicenter software delivery integration***

**Functional integration:** By default, all software is scanned by virus detection software before it is packaged. In addition, Unicenter Asset Management provides workstation and server hardware and software information, which can set the criteria to initiate software distribution. Unicenter Remote Control gives administrators the ability to remotely investigate problems on target systems if needed. These products share the same user interface, event management, scripting language, and so forth.

**Directory integration:** Directories, including LDAP-compatible directories, are often used to manage access to networked resources. Unicenter Software Delivery can use information from these directories to deliver applications on a dynamic basis.

**Enterprise management integration:** Unicenter solutions, widely recognized as the standard for enterprise management, provide a single point of control for monitoring and managing every resource critical to e-business performance and availability, including systems, networks, applications, databases and more.

### ***Software Libraries***

Software Libraries store software programs to be distributed to target computers. There are several different types of items in the Software Library:

**Programs:** Actual software packages that include either vendor-provided or customized, site-written routines for the installation, activation, configuration, or removal of the package, defined as Item Procedures. Item Procedures identify the startup programs and additional execution parameters required to perform a specific program function, such as installing or removing a software package or recovering a failed job at a target computer. They can also trigger the remote activation of a particular product; for example, an archive program. In addition to the executables provided with the software program, you can write and register additional item procedures to provide more customized or automated functions for your site, to minimize the user intervention required during installation processes. For example, if a program's installation procedure prompts for information common to all of the target

computers, you can create a customized installation procedure containing answers to these prompts.

**SW Detector:** Detects and catalogs software installed on a computer, but not registered in the Software Library.

**Documents:** Documents can be registered in the Software Library. They can also be sent to target machines, provided that an added Item Procedure is defined for the program, such as a batch file defining where the document should be copied.

You can use installation parameters to customize each software package.

### ***Unattended software installations***

The Packager and the Installer are tools that allow you to generate software packages, and to install and uninstall them reliably and without user attention, in an easy and automatic way. SD supports the following types of unattended software installations:

▶ **Setup program with response file**

Most programs support response files. A response file contains a list of pre-programmed replies to setup screens.

▶ **Custom setup program**

If an application does not support response files, you can write a custom setup program.

▶ **Macro program**

Macro programs take control of the setup program. The macro program is a fast and easy method for creating unattended setup procedures.

▶ **Reverse engineering**

In this method, an image of all software and configuration changes made during installation is created and then copied to another computer with the same result as a traditional installation. Reverse Engineering is a fast and reliable way of creating software packages, and reduces network traffic. The AutoScript Generator can create low-impact software packages and track a variety of changes: files, icons, configuration files, services, registry changes.

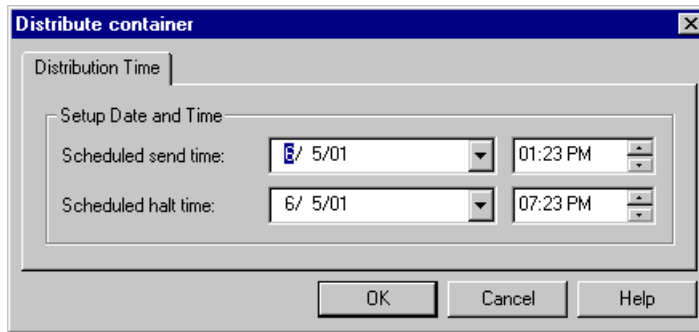
The Packager allows you to create complete, combined, or delta products, as well as delta versions for installation of product updates, reducing data volumes and transmission costs.

### ***Software delivery steps***

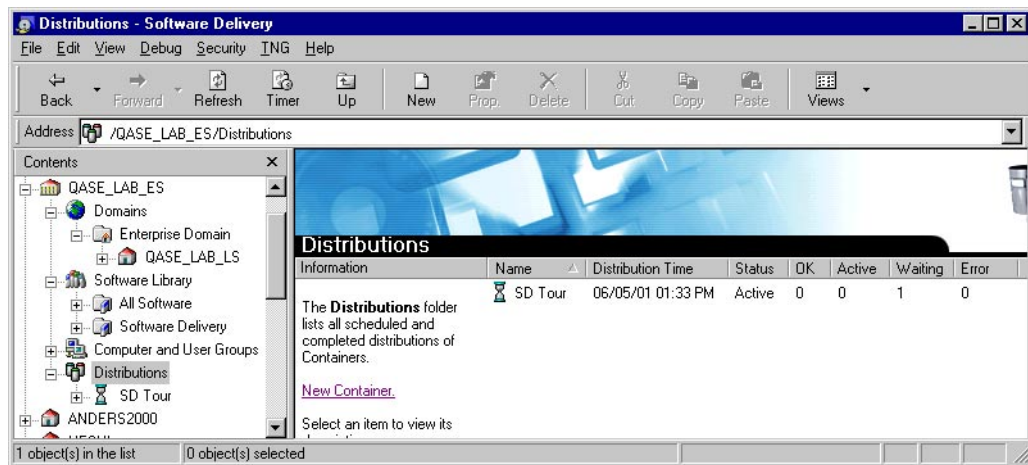
The basic steps to perform software delivery follow a logical sequence:

1. Select a program to install and register it with the Software Library.
2. Define an installation procedure and register it. The installation procedure includes essential instructions such as running the job at shutdown, which is necessary when updating kernel drivers on a Linux server, and defining a query to determine prerequisites.
3. Define a container for distributing the package and register it. Add the software to the container.

- Choose the time you want the package distributed to a Local Library.



- You can monitor the distribution to verify that the package was delivered and the installation successfully completed.



Advanced techniques permit defining groups of users and computers targeted for installation. Groups can be defined according to job function, location, or other criteria.

## Unicenter Universal Job Management Agent

This section describes some of the features of Unicenter Universal Job Management Agent.

Unicenter Universal Job Management Agent enables cross-platform scheduling by initiating and tracking units of work, such as jobs, tasks, and processes, on a wide range of platforms, from NT to Linux. It spans the entire array of Computer Associates job management solutions including Unicenter AutoSys Job Management, Unicenter CA-7 Job Management, Unicenter CA-Scheduler Job Management, Unicenter CA-Jobtrac Job Management and Unicenter Network and Systems Management Job Management Option.

Features include the following:

- ▶ Leverages return on investment in Computer Associates job management solutions and reduces deployment costs. Extends the power, scalability, and flexibility of these solutions throughout the enterprise.
- ▶ Improves productivity by centralizing all aspects of workload management. Job and schedule definitions, along with administration, remain centralized. This enables users to leverage their administrators' skills and apply consistent corporate policy to all job management functions.
- ▶ Provides any Computer Associates job management solution with the ability to schedule and manage jobs on a wide variety of platforms.

- ▶ Provides a uniform set of features for use by all Computer Associates job management solutions.

A Computer Associates job management solution sends a request to an appropriate Unicenter Universal Job Management Agent to initiate a remote job or task. The agent honors this request by initiating the job or task on that remote system. In addition, the Unicenter Universal Job Management Agent monitors the jobs or tasks and sends event information (start, execute and end) to that job management solution. The job management solution then uses this information to keep the job status information up to date. All messages associated with the Unicenter Universal Job Management Agent are recorded in the Unicenter Event Management Console Log.

## 15.2 Data management

### 15.2.1 File backup and restore

#### BrightStor Enterprise Backup

This section describes some of the features and considerations for using BrightStor Enterprise Backup.

BrightStor Enterprise Backup goes beyond simple data protection to provide a comprehensive storage solution. It offers multiple data-verification methods to enable maximum data integrity, and provides robust recovery capabilities across all Linux platforms, including Red Hat and SuSE. In addition to offering a comprehensive solution for the Linux environment, it extends data protection support to include Windows and NetWare workstations and servers, offering complete protection within a heterogeneous environment.

Features and benefits include core technology manageability, data availability, performance, serviceability, integrated client agents, and optional features for extended functionality.

#### Starting BrightStor Enterprise Backup

1. Log in to the selected host as the root user.
2. At the command line, type:

```
# cstart
```

You will see a message indicating that BrightStor Enterprise Backup is starting.

```
Starting BrightStor EB ...
```

#### Stopping BrightStor Enterprise Backup

1. Log in as the root user.
2. At the command line, type:

```
# cstop
```

You will see following messages:

```
Stopping BrightStor EB ...  
BrightStor EB unloaded successfully  
BrightStor EB GUI daemons stopped
```

#### Checking BrightStor Enterprise Backup Status

1. Log in as the root user.

2. At the command line, type:

```
# cstatus
```

If BrightStor Enterprise Backup is running, you will see output such as the following:

```
caservd      921
cadiscovd   923 922
cauthd      924
caloggerd   925
cadbd       931
camediad    979 997 996
caqd        933
cprocess
cacommd     913
httpd       912 905 910 911 908 909
```

## 15.2.2 Managing shared file systems

### Advantage Ingres Enterprise Relational Database

This section describes some of the features and considerations for using Advantage Ingres.

Distributed databases are becoming more and more prevalent. The ability to manage and distribute data across many locations, while still guaranteeing security and integrity, is essential. Advantage Ingres enables users to work in a distributed environment that is simple to manage, easy to customize, and provides fault-tolerant data replication with total integrity.

Advantage Ingres is the foundation on which many mission-critical applications have been built. A fundamental requirement for any e-business venture is to be able to function reliably while ensuring data integrity. This calls for applications of the highest order that can be relied upon to support all aspects of business processing. Advantage Ingres is the foundation of many of such applications.

Advantage Ingres underpins many of the Computer Associates solutions that manage global e-businesses. In addition to being found in the newest breed of Computer Associates applications for use in CRM, Mobile, B2B, B2C, and XSP environments, Advantage Ingres can be found in a number of possibly unexpected places such as X-ray machines, flight deck control systems, air traffic control systems, telephone switches, and military command and control systems. The common denominator in all of these applications is the requirement for a low- cost, resilient, reliable mission-critical database engine that requires minimum maintenance while delivering maximum performance and availability.

### Using Interfaces to Advantage Ingres

Advantage Ingres provides several tools for interfacing with the Advantage Ingres database. They include:

- ▶ CA ODBC Driver for Ingres
- ▶ JDBC Driver
- ▶ Ingres/OpenAPI
- ▶ Embedded SQL (ESQL)
- ▶ Ingres/ICE (Internet Commerce Enabled)

This section provides an introduction to each tool.

#### ***The CA ODBC Driver for Ingres***

The CA ODBC driver for Ingres is compliant with Microsoft Open Database Connectivity (ODBC) interface specifications. ODBC is a specification for an application programming



interface (API) that enables applications to access multiple database management systems using Structured Query Language (SQL).

ODBC permits maximum interoperability—a single application can access many different database management systems. This enables an ODBC developer to develop, compile, and deploy an application without targeting a specific type of data source. Users can then add the database drivers that link the application to the database management systems of their choice.

### ***Ingres JDBC Driver***

The Ingres JDBC driver is a pure Java implementation of the JDBC 2.1 interface. The driver supports both application and applet access to Ingres data sources through a middle-ware Ingres JDBC server.

The driver supports the full JDBC 2.1 interface with the exception of the CallableStatement interface for executing database procedures. Database procedures may be executed through the driver using the Statement.executeUpdate() method and the Ingres execute procedure statement. The following restrictions are imposed when executing database procedures in this way:

- ▶ Byref (OUT) parameters are not supported.
- ▶ The procedure return value is not accessible.
- ▶ Transaction control statements are not permitted in the procedure.

The database procedure can send messages to the application that can be retrieved as SQLWarning objects using the Statement.getWarnings() method.

### ***Ingres/OpenAPI***

Ingres/OpenAPI is a C programming language interface for accessing an Ingres database. It enables you to develop applications using a set of functions that are called directly with normal function call facilities. This interface provides an alternative to ESQL, which requires a preprocessor in addition to a C compiler.

**Header files**—Each application source file that invokes an Ingres/OpenAPI function must include the Ingres/OpenAPI header file (iiapi.h). This header file includes a platform-dependent header file, iiapidep.h, which configures the API for a particular platform.

**Library**—When using Ingres/OpenAPI, an application must link with the Ingres/OpenAPI library. Applications may also need to be linked with the standard Ingres runtime library.

**Environment variables**—The following environment variables are used by Ingres/OpenAPI:

- ▶ II\_API\_TRACE—The II\_API\_TRACE environment variable specifies the desired trace level of the Ingres/OpenAPI module. When it is not defined, it has a value of zero, which results in no tracing being performed. The II\_API\_TRACE environment variable can be set to display fatal error messages, non-fatal error messages, warning messages, checkpoint messages (such as the Ingres/OpenAPI function that is being executed), or detail information (such as values of input and output parameters).
- ▶ II\_API\_LOG—The II\_API\_LOG environment variable specifies the desired output file for all Ingres/OpenAPI tracing.

### ***Embedded SQL***

The term embedded SQL (ESQL) refers to SQL statements embedded in a host language such as C or FORTRAN. The ESQL statements include most interactive SQL statements, plus statements that fulfill the additional requirements of an embedded program.

All ESQL statements must be processed by the ESQL preprocessor, which converts the ESQL statements into host language source code statements. The resulting statements are calls to a runtime library that provides the interface to Ingres. After the program has been preprocessed, you must compile and link it according to the requirements of the host language.

### ***Ingres/ICE***

Ingres/ICE provides the foundation for Internet-based electronic commerce. It enables a Web client to retrieve data from or update an Ingres database.

You can specify actions to perform in your Web application using special HTML variables defined by Ingres/ICE. For example, there are variables to:

- ▶ Execute dynamic SQL statements
- ▶ Run Ingres database procedures
- ▶ Run Report-Writer reports
- ▶ Run client applications

A page generated by setting HTML variables can contain data from only a single SQL statement. When you need data from more than one statement, you can create Ingres/ICE Macro XML documents. A macro document contains one or more special tags that define an SQL statement to be executed. Macro XML allows you to create Web pages that include data from several database tables and provides many formatting options that control the way the data is presented to the Web client.

Additionally, Ingres/ICE provides macro tag extensions for use with an XML-aware editor, so that adding Ingres/ICE XML elements to your documents is easier than ever. You can simply point-and-click to automatically generate macro syntax within your working environment.

### **Advantage CA-XCOM Data Transport**

This section describes some of the considerations for using Advantage CA-XCOM Data Transport.

Advantage CA-XCOM is a family of software products that provide high-speed data transfer among diverse platforms. It performs file, job, and report transfers over all major data link types through TCP/IP networks. Advantage CA-XCOM allows data centers in various locations worldwide to interact with each other for the purposes of sharing data and automating data and report distribution.

Generally, a file transfer from a local system to a partner system takes place according to the following steps:

1. When the user starts the menu, command line, or application programming interface to initiate the transfer, Advantage CA-XCOM verifies the information contained in the request. For example, when requesting a file transfer, Advantage CA-XCOM checks to see whether the file exists on the local system. If the information is confirmed, Advantage CA-XCOM attempts to transfer the file.
2. When the transfer is in progress, Advantage CA-XCOM receives and verifies information from the partner system. For example, Advantage CA-XCOM can check whether a file exists on the partner system. If you are creating a new file that does not exist on the partner system, then the new file is transferred.
3. When the transfer completes, Advantage CA-XCOM logs the details of the transfer in a log.

## Configuring Advantage CA-XCOM

To perform a transfer, you need to configure Advantage CA-XCOM parameters. To complete the configuration, you need to have superuser (root) privilege.

**Note:** Before performing transfers, you must have TCP/IP configured for Advantage CA-XCOM.

The configuration file contains parameters and values that Advantage CA-XCOM uses to perform the transfer. The file `/usr/spool/xcom/config/xcom.cnf` can be used as a model for your own configuration files. For detailed information about using Advantage CA-XCOM, see the product documentation.

## Advantage Data Transport Agent

This section describes some of the features and considerations for using Advantage Data Transport.

An enterprise-wide data transfer solution that provides the broadest platform and protocol coverage available today, Advantage Data Transport delivers critical e-business data to enterprise-wide systems, providing the ideal solution for environments where file transfer protocol (FTP) fails to deliver production-level security, scalability, and guaranteed delivery.

To be effective, data transport management must monitor and administer the entire data transfer from a central location, as well as manage network performance. This includes obtaining network knowledge, providing network management tools, and supporting the protocols that maximize bandwidth usage. Advantage Data Transport provides centralized data transport management and administration capabilities while automatically optimizing network usage and controlling network traffic to meet today's fast-paced business demands. This powerful solution securely transports all the data across a wide set of platforms, protocols, and data formats. It does this in a flexible and easy-to-use manner, insulating all users from the need to know about topology, protocols, hops, connections, or networking.

Upon installing the ADT Agent, the comprehensive range of features offered by ADT are made available to users of Linux on S/390 and zSeries. The agent integrates seamlessly into your ADT network, allowing robust, convenient data transfer between the whole range of computers in your enterprise.

Features include the following:

- ▶ **Broadcast, Multicast and Fanout:** Advantage Data Transport supports broadcast, multicast, and fanout mechanisms for point-to-many transfers, minimizing network traffic and machine workload. It automatically determines which of these data transfer methods to use without requiring user involvement.
- ▶ **Checkpoint Restart and Session Retry:** Advantage Data Transport guarantees delivery of critical data. Transfers can be automatically resumed from the point of failure, eliminating the need to resend entire files. Transfers can be configured to automatically retry connections, avoiding the need for operator intervention if a remote computer cannot be contacted.
- ▶ **Discreet Mode Transfer for Network Efficiency:** Agents can perform transfers in discreet mode to minimize the impact of the transfer. (Discreet mode is when the transfer makes use of idle network time.) When Advantage Data Transport detects that the system CPU or network utilization is high, the amount of data transferred is progressively reduced to avoid network congestion. Then the amount of data transferred is increased during low network traffic, making efficient use of idle network time.
- ▶ **DHCP Support:** When Advantage Data Transport is used on a computer whose IP address changes frequently, such as on a dial-up line or a DHCP LAN, Advantage Data

Transport contains technology that still recognizes the computer, thereby allowing transfers to be resumed or restarted.

- ▶ **HTTP Transfers:** Advantage Data Transport supports HTTP protocol for fetching data.
- ▶ **Multi-Level Security:** User and password authentication can be activated at multiple levels. Data can also be encrypted for further protection. Advantage Data Transport comes equipped with its own encryption algorithms as well as interfaces to popular encryption packages, plus letting you use your own encryption algorithms.
- ▶ **Domain Security:** Administrators can configure Advantage Data Transport to recognize a domain of agents. This allows Advantage Data Transport users to log in to the domain and perform transfers between agents in the domain without having to give their usernames and passwords for each agent. Advantage Data Transport administrators can add users and agents to a domain, as well as remove them. They can also grant other users “domain administrator” privileges.
- ▶ **Ownership and Security:** Transfers and transfer jobs can be set up so they are accessible for use and modification only by the user who created them.
- ▶ **Alternative Routes:** Advantage Data Transport has routing technology that enables alternative network paths to be established between agents. When a transfer fails, Advantage Data Transport can try a number of alternative routes, in descending order of efficiency, to achieve a successful transfer.
- ▶ **Audit Message Customization:** The rich set of audit messages that Advantage Data Transport generates lets you customize their format and contents at several levels. For example, one class of message might be given a format that makes them easy to process in an external program, whereas another class of message might be made very verbose and descriptive for human reading.
- ▶ **Transfer Macros:** The transfer and transfer-job properties can be set with macros, so variables have values that are placed in the properties at runtime. For example, a user may add the current time of day to an output file name.
- ▶ **Advantage Data Transport Shares:** Administrators can specify folders on a target machine that are accessible by the Data Transport browsing mechanism. This effectively controls access to a machine's file system by Advantage Data Transport users and transfers.
- ▶ **Pre- and Post-Transfer Processing:** The ability to invoke processes before and after sending and receiving files means you can compress or encrypt data, execute commands on remote machines, and invoke other processes, such as a database extract or load. Processing may occur at two levels: on the entire transmission or on individual parcels.
- ▶ **Transfer Job Conditional Logic:** A sequential transfer job (that is, a job whose transfers are executed one after the other, rather than in parallel) can be given conditional logic that allows different actions to be taken if a transfer fails or succeeds. Thus the same job may perform different transfers or skip transfers, depending on the status of the preceding transfers.
- ▶ **Single Point of Control:** Progress and status of all data transfers occurring within an enterprise are monitored from a central location. Data transfers can be aborted, suspended, or resumed at any time. The configuration and runtime settings of any Advantage Data Transport Agent or Manager on the network can be managed from a single point of control, subject to the security policy you establish.
- ▶ **Java Administration Interface:** A Java component permits configuration and management of the Advantage Data Transport environment in a Web browser.
- ▶ **Software Development Kit (SDK):** A fully functional Software Development Kit (SDK) provides access to all Data Transport functions through client-written APIs.

- ▶ **Data Transport Architecture:** Advantage Data Transport consists of two distinct client graphical interfaces:
  - **Transfer Client** creates and administers transfers, transfer jobs, and schedules. It sits at the top of the Advantage Data Transport architecture, as a client application linking into the Data Transport service managers. This is, in essence, a client/server relationship, using communication links to request and receive information about transfers and associated objects.
  - **Administration Client** performs the administrative tasks on individual agents and managers. The Administration Client is used to change the properties of agents and managers that are then used by Advantage Data Transport in creating and executing a transfer, such as security level, packet size, protocol, and throttle factor.

The Data Transport Manager consists of three server processes:

- **Network Object Server**—This server process should be installed on the system that contains the Unicenter Common Object Repository. The Network Object Server records and maintains the description of the Advantage Data Transport network: the computers in it and the properties of the links that connect them. It processes this information to calculate the fastest, most efficient route between Advantage Data Transport systems, and effectively maintains a map of the network.
- **Transfer Object Server**—This server process can be installed on any system in the Advantage Data Transport network. It does not necessarily have to be on the same computer as the Network Object Server. The Transfer Object Server records and maintains the definition of all the individual data transfers in the system. It gathers transfers into groups called *transfer jobs*. It activates, controls, and monitors transfers.
- **Schedule Object Server**—This server can be installed on any systems in the Advantage Data Transport network, but to take full advantage of the transport scheduling capabilities, the system on which it is installed should also have Unicenter Enterprise Management or the Unicenter Calendar option. The Schedule Object Server maintains a list of schedules on which data transport activities are to take place, and records the transfers that are to be activated when their schedules become due.

The modules that actually perform the data transference are called the Data Transport Agents. These are lightweight agents that contain specific knowledge about the different protocols supported by Advantage Data Transport and know how to connect to remote systems. Several of these processes can be running on a system at any given time, depending upon the number of data transport jobs in operation at that time.

Two other important elements in the Advantage Data Transport architecture are the Data Object Agent and the Data Transport Service Monitoring Agent:

- A Data Object Agent is normally located on each agent machine. The Transfer Client communicates with the Data Object Agent to obtain details of files located on remote systems.
- The Data Transport Service Monitoring Agent integrates with Unicenter NSM to provide monitoring of data transport services, transfer status, and key system resources such as disk space. Alarm thresholds can be set to notify administrators of potential problems.

## 15.3 Security

### 15.3.1 User definition and administration

#### eTrust Directory

This section describes some of the features and considerations for using eTrust Directory.

The heart of the eTrust Directory family is the DXserver, which is packaged with powerful implementation and management utilities, including bulk data loaders, load balancers and a powerful Java browser incorporating LDIF and schema management facilities. The DXlink feature enables any LDAP-compliant server to be incorporated into the eTrust Directory backbone. While delivering all the benefits of LDAP compliance, eTrust Directory also provides the turbo-charged performance essential for the next generation of e-business services. Independent testing proves its combined LDAP/X.500 technology significantly outperforms LDAP-only implementations through its DXcache feature.

Features include the following:

- ▶ Customer authentication, authorization and consolidated management
- ▶ Highly distributed with virtually unlimited scalability
- ▶ Support for over 20,000,000 entries and 1000 searches per second
- ▶ True multi-protocol solution combines strengths of X.500 and LDAP V3
- ▶ Successfully interoperates with SAP (Note: version 3.6 SP2 is certified)

Here are some of the major features of eTrust Directory:

**DXlink:** This DXserver feature enables you to incorporate LDAP servers into a unified directory backbone. DXlink provides the power of X.500's distributed searching, security, and management to LDAP server administrators and users.

**Routing:** DXserver routing lets you perform full X.500 distributed searches. It also supports the use of alternate DSAs for availability, query streaming, and load sharing. Shared configuration and automatic knowledge management make a network of DXservers easy to configure and manage. DXserver can also be configured without a database for use as a router for load sharing, proxies, and guard and firewall applications.

**Security:** DXserver implements the powerful X.500 security model. This includes mutual authentication of DSAs to provide security between directory servers, rules-based access controls, routing subject to access controls, automatic alias management, and credit-based controls to stop denial of service.

**Dynamic Configuration:** All aspects of eTrust Directory management can be performed dynamically. This includes the seamless swapping of databases, changes to tracing, access controls, and knowledge references while the DSA remains online.

**Schema:** DXserver's X.500 schema are fully configurable, including attribute syntaxes (basic directory information types), allowing you to easily define your own custom schema. A comprehensive set of schema files is also provided with eTrust Directory.

**Reliability:** The RDBMS provides a high level of data reliability. In addition, X.500 replication can provide backup copies of data. The multiwrite capability of DXserver enables groups of peer DSAs to be kept synchronized while online. If one DSA fails, then a router DSA can forward requests to another DSA in the peer group. DXservers can also be configured with multiple network addresses to allow for network failover.

## Using eTrust Directory

A number of graphical user interfaces are provided to let you browse, search, configure, and update the directory. We provide a brief overview of using JXplorer here. With the JXplorer browser, you can:

- ▶ Connect to any directory that supports LDAP, and browse, search, and update the directory
- ▶ Read the directory's schema directly
- ▶ Cut, paste, and edit subtrees within the directory visually, allowing large-scale manipulation of directory entries
- ▶ Import and export LDIF files from a directory and even manipulate them offline
- ▶ Configure the browser in a large number of ways, including visual appearance and logging information
- ▶ Display directory data within configurable HTML templates
- ▶ Optionally use SSL to communicate securely, and SASL for secure certificate-based authentication

For best performance, we recommend that you use JXplorer installed on the machine you are connecting from (for example, a PC) rather than running the mainframe version.

To start the PC version of JXplorer, ensure that it is installed on your PC, then select **Start -> Programs -> eTrust Directory -> JXplorer**. A connection can then be made to the Directory Server on the host mainframe.

If the JXplorer client software is not installed locally, then JXplorer can be run on the mainframe as you would on any UNIX/Linux system.

For running the mainframe JXplorer from a PC, Xterm software on the client machine is required, and you also need xhost access to the mainframe. To start JXplorer on Linux, issue the following command from the JXplorer directory:

```
./jxstart.sh
```

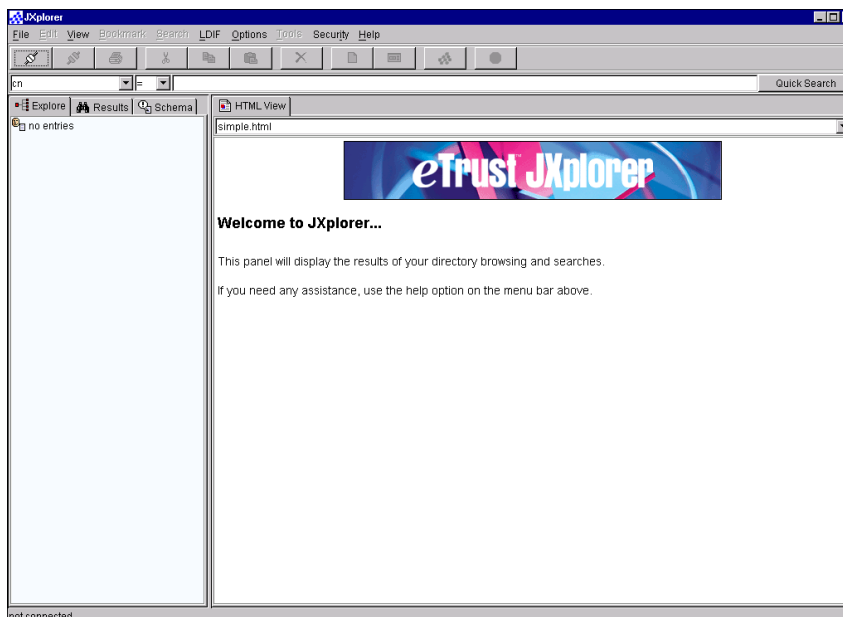


Figure 15-12 eTrust Directory startup window using JXplorer browser

From the File menu, choose **Connect** (or click the Connect button) to display the connection dialog.

The following dialog shows JXplorer set to connect to the Router DSA anonymously. Click **OK** to connect.

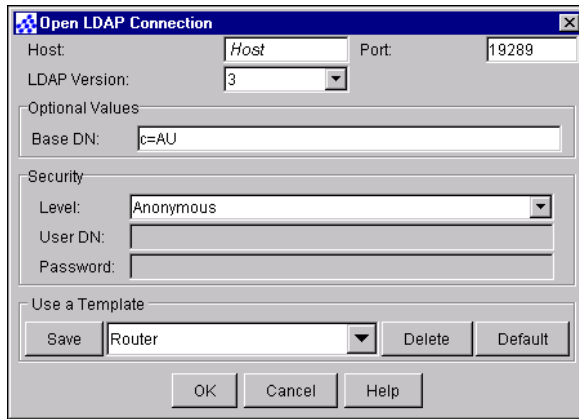


Figure 15-13 eTrust Directory connection dialog

You can execute simple, single-attribute-value searches quickly using the quick search bar, which contains a pull-down list of common attribute types and operators. The operators include:

- ▶ Equals (=)
- ▶ Starting from (>=)
- ▶ Up to (<=)
- ▶ Not equal to (!=)
- ▶ Approximate match (~=)

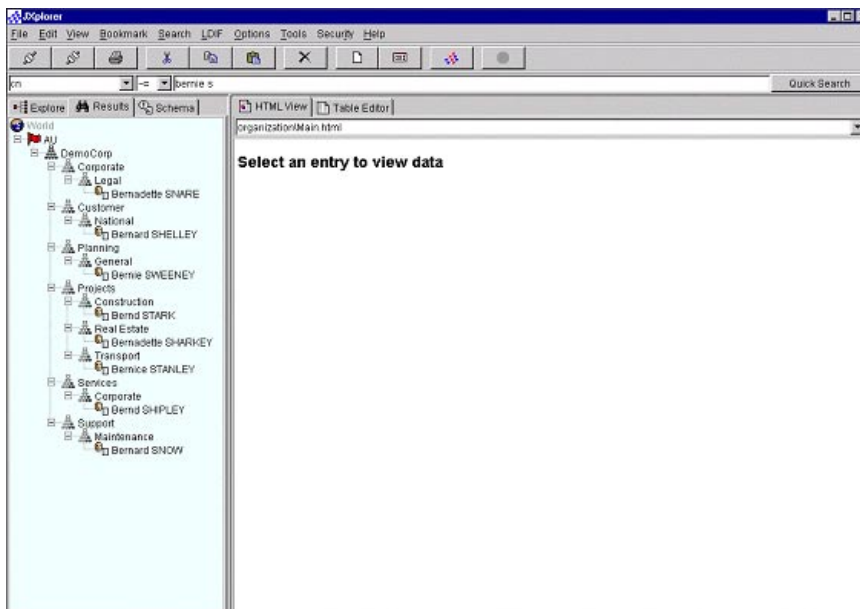


Figure 15-14 eTrust Directory window showing a search for entries with a common name that sounds like "Bernie S"



To display the contents of an entry, double-click on either a highlighted entry in the Results pane, or on any entry in the Explorer window.



Figure 15-15 eTrust Directory example of the results of a search, with an individual record displayed

**Tip:** You can view a photo of a person in a template if you have an attribute type of 'jpegPhoto'. The photo must be in a jpeg format.

You can also easily display all the attribute types and values contained in an entry, and update them.

## eTrust Admin

This section describes some of the features and considerations for using eTrust Admin.

eTrust Admin delivers a robust and scalable solution for enterprise-wide system administration of security systems. It provides end-to-end management across multiple, geographically dispersed directories through a single user interface. eTrust Admin enables administrators to quickly and easily create, modify, and remove users and related objects within directories. Administration is policy-based to help ensure consistent privileges through an IT environment. An intuitive, graphical user interface reduces the time and cost usually associated with the administration of use accounts. eTrust Admin virtually eliminates the steep learning curve common to administrators just beginning to manage security systems and directories on unfamiliar platforms and applications. It simplifies the often complex series of tasks that must be performed when personnel change jobs or leave the company, or when reorganizations require global changes.

Features include the following:

**Role-based User Administration:** Through role-based administration, eTrust Admin manages users according to their job function. Administering users in this manner controls the access that users need across diverse systems.

**Directory Integration:** eTrust Admin is a directory-enabled application. It uses eTrust Directory, a fully standard compliant X.500 directory, to store its information.

**Wide Range of Environments Supported:** eTrust Admin can administer a wide range of environments, such as security and network operating systems, groupware applications, databases, and Enterprise Resource Planning (ERP) applications. This includes an LDAP option for interfacing to Lightweight Directory Access Protocol (LDAP) directories, enabling integration with any LDAP-enabled directory, application, or ERP.

**Integration with Other eTrust Products:** eTrust Admin can be integrated with a wide range of eTrust products, including eTrust Directory, eTrust Web Access Control, eTrust Access Control, eTrust Single Sign-On, eTrust PKI, and eTrust Audit to provide the total, integrated security solution for today's and tomorrow's e-businesses.

**Distributed Administration:** eTrust Admin lets administrators work within a scope so they have access to only the objects they manage, thereby improving productivity and security among all administrators.

**Password Synchronization:** eTrust Admin lets users change their passwords on any managed Windows machine. After the password is changed, eTrust Admin automatically updates the passwords for all other accounts that a user has.

**Intuitive Manager Interface:** Administrators can use the feature-rich graphical user interface (GUI) called the Manager. The task-based Manager conceals system differences and complexities by providing a consistent way of performing tasks on these systems while presenting information in an intuitive manner that accommodates managing large-scale environments, which reduces training costs and improves productivity. The Manager is the most commonly used interface in your organization because you can use it to perform all your user administration tasks.

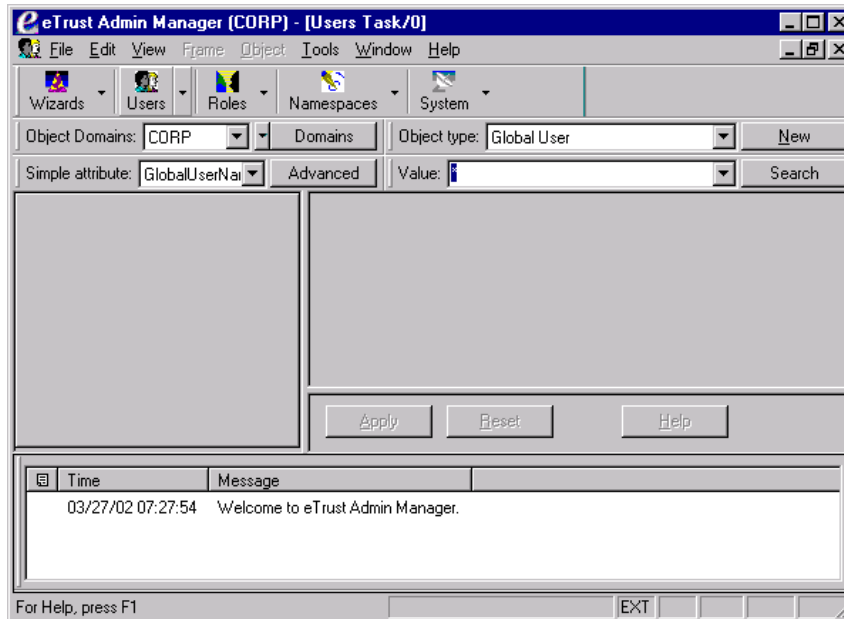


Figure 15-16 eTrust Admin Manager Interface

**Web-based Administration:** Administrators and users can perform selected tasks using the Web Interface or the User Provisioning Workflow Interface.

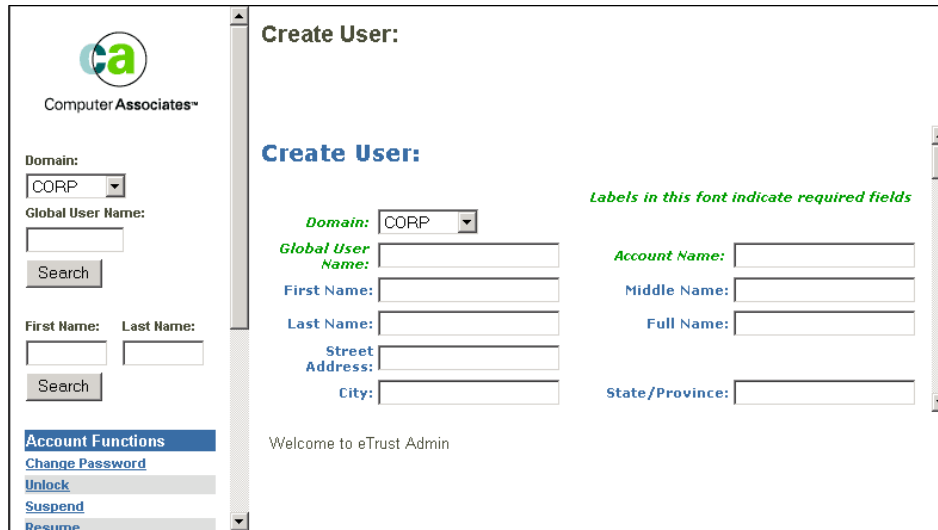


Figure 15-17 eTrust Admin Web Interface

**Auditing and Reporting:** Administrators can log all changes to a central point, and then generate reports and distribute them to management.

**Command Line Interface:** Administrators can use the Batch Utility to perform the same tasks in the Manager. This command line interface is useful when you need to perform repetitive and time-consuming tasks, because you can perform the tasks offline or in the evening hours.

**Software Development Kit:** The eTrust Admin SDK contains development tools for building agents for any Computer Associates component, or any other third-party software, so you can manage these environments also.

## PAM support for Computer Associates External Security Managers

This section describes some of the features and configuration considerations for using the PAM modules with eTrust CA-ACF2 Security and eTrust CA-Top Secret Security. See also the following sections to configure the PAM components for eTrust CA-ACF2 Security and eTrust CA-Top Secret Security.

Features include the following:

- ▶ Password Verification: Allows PAM-enabled applications to use CA ESMs to validate access control
- ▶ Password Changes: Allows PAM-enabled applications to instruct CA ESMs to change a user's password
- ▶ Access Authorization: Allows PAM-enabled applications to use CA ESMs to validate access control based on criteria other than the user's password
- ▶ Session Management: Allows PAM-enabled applications to use CA ESMs to log the start and end of the controlled sessions
- ▶ Case-sensitive and long user names: Translates case-sensitive and long user names (more than 8 characters)

## Configuring the PAM ESM Component

The PAM ESM component is configured like other PAM modules, by placing lines requesting its use in the `/etc/pam.conf` file (for monolithic configurations), or in the files in `/etc/pam.d` (for

multi-file configurations). Some examples are included in the distribution, in `pam_CA_esm-<level>/pam.conf` and `pam_CA_esm-<level>/pam.d/*`. The `pam_CA_esm` module accepts the common PAM module options (`debug`, `no_warn`, `try_first_pass`, `use_authtok`, and `use_first_pass`), plus `config=filename` (to override the default location of the `pam_CA_esm.conf` file).

There are two configuration files, one for each component, named (by default) `/etc/pam_CA_esm.conf` and `/etc/CA_esm_proxy.conf`. Each contains several types of records, most of which are optional and default to some value if omitted. The files in the distribution contain full explanations of each record and commented-out records that specify the default values. Only one record *must* be changed after installation: the `esm-host` record in the `/etc/CA_esm_proxy.conf` file must be set to the hostname (or IPv4 address) of the z/OS and OS/390 system where the Computer Associates ESM is running.

## **eTrust CA-ACF2 Security (interface to Linux for zSeries and S/390)**

This section describes some of the features and configuration considerations for using eTrust CA-ACF2 Security with the PAM component.

eTrust CA-ACF2 Security helps ensure the integrity and security of your critical information assets. Basic and advanced eTrust CA-ACF2 Security mechanisms provide flexibility and control through a unique security configuration that accommodates all organizational structures. Administrative tools, reporting options, and automatic logging capabilities accompany eTrust CA-ACF2 Security, securing your environment while enabling comprehensive auditing and controlled sharing of data and resources.

eTrust CA-ACF2 Security is a component in CA's family of integrated security solutions that protect your information systems and the data they manage from unauthorized disclosure, modification, and destruction. Since its introduction in 1978, eTrust CA-ACF2 Security has maintained leadership in the OS/390, z/OS, and VM marketplace through aggressive delivery of new releases, high-quality technical support services, and adapting to the ever-changing mainframe marketplace.

Features include the following:

- ▶ Supports all OS/390 security extensions
- ▶ Protects data and resources by default
- ▶ Automatic synchronization of security information across networks

eTrust CA-ACF2 Security provides industry-leading technology to help you synchronize and share security information among disparate platforms. eTrust CA-ACF2 Security includes full interoperability between the Unicenter NSM and eTrust CA-ACF2 Security for OS/390 and z/OS security engines, enabling automatic propagation of security changes across security images, including password synchronization and userid suspension propagation. eTrust CA-ACF2 has also deployed SNMP, or Simple Network Management Protocol, to permit end-to-end management and real-time monitoring of security events. Additional capabilities will be provided through CA eTrust solutions including eTrust Admin, eTrust Directory, eTrust Access Control, and eTrust Audit.

## **Configuring eTrust CA-ACF2 for Linux PAM**

For detailed information on configuring eTrust CA-ACF2 for use with the CA PAM ESMs, see the product documentation. A summary is provided here.

Use the following commands to define the user ID and started task information in the eTrust CA-ACF2 database:

```
INSERT PAMSRV NON-CNCL STC GROUP(OPENMVS)
```

```
SET PROFILE(USER) DIV(OMVS)
INSERT PAMD
UID(0) HOME(/usr/lpp/capam)
```

The following steps enable eTrust CA-ACF2 to process system entry validation requests from Linux systems.

1. Define Linux machines.

For each Linux machine, insert a GSO LINUX record. The GSO record defines the Linux machines to eTrust CA-ACF2. It specifies the machine name, IP address, and indicates if the machine is active. (See the eTrust CA-ACF2 Administrator Guide for more information on GSO records.) Use the eTrust CA-ACF2 Panel interface or type the following ACF commands:

```
SET CONTROL(GSO) SYSID(xxxxxxxx)
INSERT LINUX.qual MACHNAME(machine name) IPADDR(IP address) ACTIVE
```

Once the GSO LINUX records are inserted, issue the following command to activate the records:

```
F ACF2,REFRESH(LINUX)
```

2. Define Linux user profiles.

For each Linux user, insert a LINUX USER PROFILE record. The user profile maps a Linux application user identity to an eTrust CA-ACF2 logonid. (See the eTrust CA-ACF2 Administrator Guide for more information on user profiles.) Use the eTrust CA-ACF2 Panel interface or type the following ACF commands:

```
SET P(user) DIV(LINUX)
INSERT logonid LINUXNAME(userid)
```

Where *logonid* is the eTrust CA-ACF2 logonid and *userid* is the Linux user application id.

3. View the Linux machine.

Display the Linux machine definitions currently in use by typing the following ACF command:

```
SHOW LINUX
```

A list appears displaying the machine name and IP address, and indicates if the machine is active.

## **eTrust CA-Top Secret Security (interface to Linux for zSeries and S/390)**

This section describes some of the features and configuration considerations for using eTrust CA-Top Secret Security with the PAM component.

eTrust CA-Top Secret Security is the flagship component in the CA family of integrated security solutions that protect your information assets. It enables controlled sharing of your computers and data, with features that prevent accidental or deliberate destruction, modification, disclosure, and/or misuse of computer resources. It allows you to control who uses these resources, and provides you with the facts you'll need to monitor your security policy effectively. Unauthorized attempts to access resources are automatically denied and logged. Any authorized use of sensitive resources can also be logged for subsequent review.

Features include the following:

- ▶ Flexible and powerful administrative tools
- ▶ Automatic logging facilities
- ▶ Extensive reporting and online monitoring capabilities
- ▶ Analyze and evaluate computer access activities and trends

- ▶ Simplifies security administration

eTrust CA-Top Secret Security is delivered complete with flexible and powerful administrative tools, automatic logging facilities, and extensive reporting and online monitoring capabilities. Authorized individuals are provided a wide range of opportunities to analyze and evaluate computer access activities and trends. Administrators can quickly and easily set and adjust security policies to respond to rapidly changing business needs.

This solution may be combined with Unicenter NSM and eTrust CA-Top Secret WorkStation to provide your security manager with end-to-end control of the distributed environment. The Computer Associates security solutions strengthen security, streamline administration, enable single-point user sign-on, and provide both platform and network-level security and auditing capabilities.

## Configuring eTrust CA-Top Secret for Linux PAM

For detailed information on configuring eTrust CA-Top Secret for use with the CA PAM ESMs, see the product documentation. A summary is provided here.

Use the following steps to define the user ID and started task information in the eTrust CA-Top Secret Security database:

1. Create the group definition:

```
TSS CREATE(PAMGRP) TYPE(GROUP) NAME('PAM SERVER GROUP') DEPT(OMVSDEPT)

TSS ADD(PAMGRP) GID(nn)
```

2. Define the PAM startup address space ID:

```
TSS CREATE(PAMSRV) TYPE(USER) NAME('PAM STARTUP ID')
DEPT(OMVSDEPT) FACILITY(STC) PASSWORD(password,0)

TSS ADD(PAMSRV) GROUP(PAMGRP)
```

3. The STC ACID needs access to the following resources:

```
TSS ADD(anydept) IBMFAC(BPX.FILE) ACCESS(READ)

TSS ADD(anydept) IBMFAC(BPX.SERVER) ACCESS(UPDATE)

TSS ADD(anydept) IBMFAC(BPX.DAEMON) ACCESS(READ)

TSS ADD(anydept) DATASET(TCPIP.)

TSS PERMIT(PAMSRV) IBMFAC(BPX.FILE) ACCESS(READ)

TSS PERMIT(PAMSRV) IBMFAC(BPX.SERVER) ACCESS(UPDATE)

TSS PERMIT(PAMSRV) IBMFAC(BPX.DAEMON) ACCESS(READ)

TSS PERMIT(PAMSRV) DATASET(TCPIP.)
```

## 15.3.2 Access control

### eTrust Access Control

This section describes some of the features and considerations for using eTrust Access Control.

eTrust Access Control provides an essential e-business element—regulating access to critical business assets. In a world where business systems are all too accessible, eTrust Access

Control provides policy-based control of who can access specific systems, what they can do within them, and when they are allowed access. Policies can be created, managed, and distributed on an enterprise-wide basis, or customized to meet the security requirements of specific applications.

This solution can be deployed in individual departments, such as payroll, or to the largest enterprises, and everything in between. Its hardened operating system security, complete auditability, and cross-platform access control secure everything from LANs and Web servers to mainframes. eTrust Access Control's built-in baseline policies give organizations immediate results right out of the box. Open and extensible, this powerful solution supports all industry-standard platforms, databases, and applications; and includes published interfaces allowing it to secure any resource. Ease of use, combined with centralized user and access administration, enables organizations to confidently exploit e-business.

As a part of the eTrust security solution, eTrust Access Control is built on the CA Common Services, providing a powerful, comprehensive solution for building, deploying, and managing security as part of the larger task of enterprise management.

Features include the following:

- ▶ Safeguarding confidential information and critical servers
- ▶ Flexible access control and enforced password quality
- ▶ Cross-domain user management and policy distribution
- ▶ Server protection
- ▶ Immediate security deployment

### **Securing your environment with eTrust Access Control**

Security of the native Linux environment is difficult to manage because there is no separation of duties. Anyone who has superuser privileges can make changes to application data beyond the scope of the application without accountability, and likely without detection.

With eTrust Access Control (eTrust AC), customers can centrally manage user privileges and quickly deploy security policies. Duties can be separated for system administration and security administration. This separation can be extended to other job functions, such as network administration and help desk management. An audit record is written for resource accesses of interest, for example, someone's assuming the superuser's identity while maintaining accountability. eTrust AC ensures the right people have access to the right information. It proactively secures access to data, applications, and resources located on the native operating system servers throughout an organization.

Figure 15-18 is an example of the eTrust AC seam X Windows GUI showing some of the types of resources that are protected with eTrust AC.

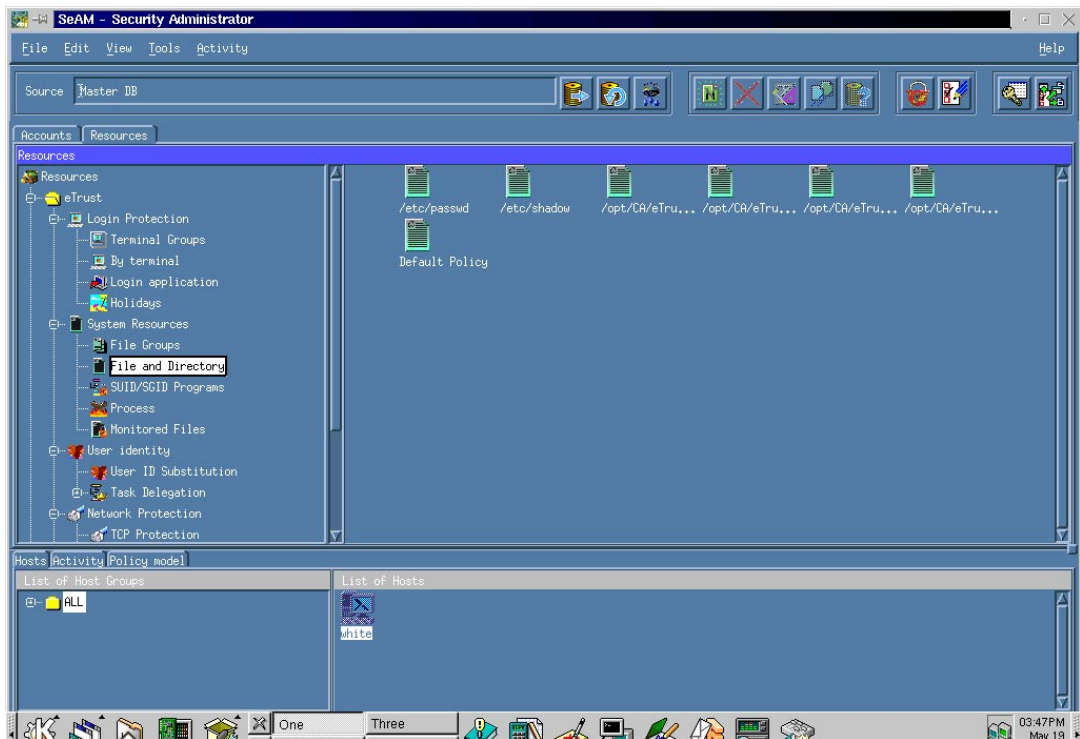


Figure 15-18 The eTrust Access Control seam X Windows GUI

eTrust AC provides reliable, non-intrusive protection through its patent-pending Dynamic Security Extension (DSX) technology. DSX dynamically intercepts security-sensitive requests in real time, without requiring any permanent change to the operating system kernel. eTrust AC typically has minimal impact upon system performance.

### Centralize security management

Admin for eTrust Access Control is the eTrust AC GUI for Microsoft Windows environments. It is similar to the seam X Windows GUI, and enables centralized management of users and access privileges on native operating environments. In addition to user management, eTrust AC provides a robust system for creating, distributing, and managing access control policy via a Policy Model Database (PMDB) hierarchy.

### Auditing actions of specific individuals

Comprehensive security must include a complete and reliable record of activity by individuals. Administrators can configure eTrust AC to audit all security-sensitive events. eTrust AC audit information can be consolidated centrally across multiple systems, and quickly filtered to speed inquiries and analysis. Figure 15-19 shows a detailed audit event as displayed by the Admin for eTrust Access Control GUI.



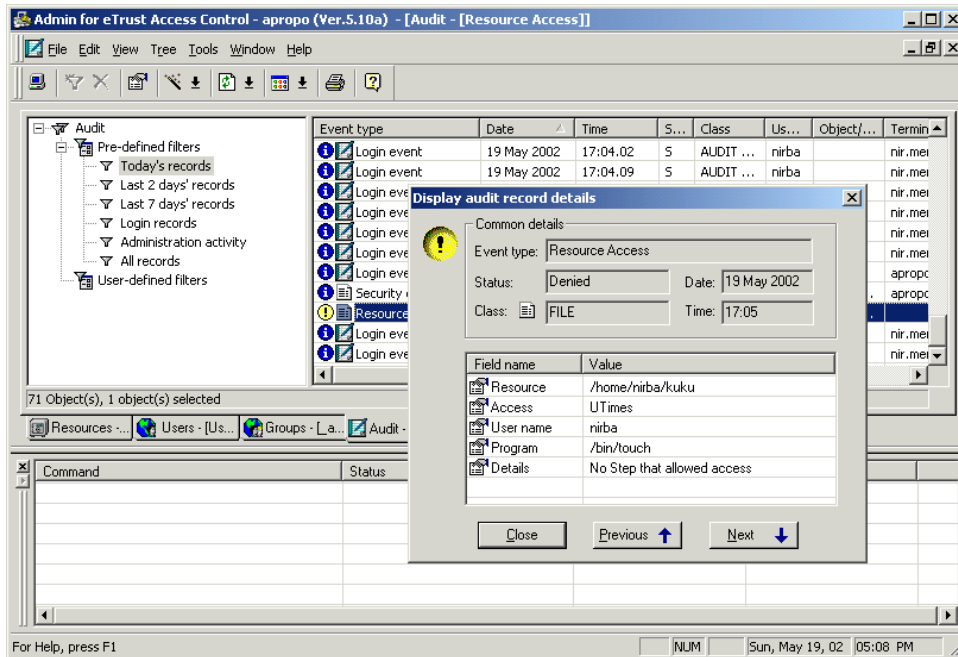


Figure 15-19 Detailed audit event as displayed by the Admin for eTrust Access Control GUI

As already mentioned, eTrust AC policies can be managed with a UNIX X Windows GUI and a Microsoft Windows-based GUI. A powerful command line interface called *selang* is also provided.

The following is an example of using *selang* to create a file access control rule to protect all files in the `/etc` directory.

---

```
eTrust> newfile /etc/* owner(nobody) defaccess(read chdir) \
? comment('Protect the system configuration files in the etc directory')
Successfully created FILE /etc/*
eTrust> auth file /etc/* uid(root) gid(sysadm) access(all)
Successfully added root to /etc/*'s ACL
Successfully added sysadm to /etc/*'s ACL
eTrust> showfile /etc/*
Data for FILE '/etc/*'
-----
Defaccess      : R, Chdir
Acls           :
  Accessor      Access
  sysadm        (GROUP ) R, W, X, Cre, Del, Chown, Chmod, Utime, Sec, Rename,
  Chdir
  root          (USER  ) R, W, X, Cre, Del, Chown, Chmod, Utime, Sec, Rename,
  Chdir
Audit mode     : Failure
Owner          : nobody      (USER  )
Create time    : 19-May-2002 21:14
Update time    : 19-May-2002 21:14
Updated by     : joeadmin
Comment        : Protect the system configuration files in the etc directory
```

---

## Protecting sensitive accounts

In addition to eTrust Access Control's special ability to restrict the superuser (root) account, eTrust AC monitors user surrogate requests (su) and includes a powerful Surrogate DO (SUDO) facility called `sesudo` to further reduce superuser account usage. Even such routine tasks as mounting a CD-ROM drive require superuser authority. `sesudo` allows providing such capabilities to operations staff without the need of providing full access to superuser authority. Even the parameters to such functions can be limited.

Often there is a desire to limit the ability to assume the identity of the superuser. The following example shows how to provide this capability only to members of the `sysadm` group.

---

```
eTrust> newres SURROGATE USER.root owner(nobody) defaccess(none) \  
? comment('Limit which users can assume the superuser identity.')
```

Successfully created SURROGATE USER.root

```
eTrust> auth SURROGATE USER.root gid(sysadm) access(read)
```

Successfully added `sysadm` to USER.root's ACL

```
eTrust> showres SURROGATE USER.root
```

Data for SURROGATE 'USER.root'

---

```
-----  
Defaccess      : None  
Acls           :  
  Accessor      Access  
  sysadm        (GROUP ) R  
Audit mode     : Failure  
Owner          : nobody      (USER  )  
Create time    : 19-May-2002 21:34  
Update time    : 19-May-2002 21:34  
Updated by     : joeadmin  
Comment        : Limit which users can assume the superuser identity.
```

---

In conjunction with protecting who can assume the superuser's identity, it is often desirable to limit the terminals, workstations, and hosts from which the superuser can directly log in. There is a `_default` object for each type of system resource protected by eTrust AC. This object typically protects all objects not explicitly defined to eTrust AC. The `TERMINAL` class in eTrust Access Control provides such protection. Assuming the `_default` object in the `TERMINAL` class is the only `TERMINAL` object, the following example shows how to keep the superuser from directly logging into the system.

---

```
eTrust> editres TERMINAL _default defaccess(read) \  
? comment ('Protect all login stations not explicitly defined to eTrust AC')
```

Successfully updated TERMINAL `_default`

```
eTrust> auth TERMINAL _default uid(root) access(none)
```

Successfully added `root` to `_default`'s ACL

```
eTrust> showres TERMINAL _default
```

Data for TERMINAL '`_default`'

---

```
-----  
Defaccess      : R  
Acls           :  
  Accessor      Access  
  root          (USER  ) None  
Audit mode     : Failure  
Update time    : 19-May-2002 21:46  
Updated by     : joeadmin  
Comment        : Protect all login stations not explicitly defined to eTrust
```

---

Usually the superuser is allowed to log in directly from the system console in case there is a system problem. The system console typically has higher priority over any other device, and it should have limited physical access.

As an extension to the above scenario limiting direct superuser login, eTrust Access Control can also control any login based on the user ID, group membership, the login station, time of day, and even the program used to log in.

## Password and Account Policy

eTrust AC includes a powerful and flexible policy on user activity, including inactivity checking, password expiration, password quality, password history, and more. The Admin for eTrust Access Control screenshot shown in Figure 15-20 illustrates some of the password enforcement available.

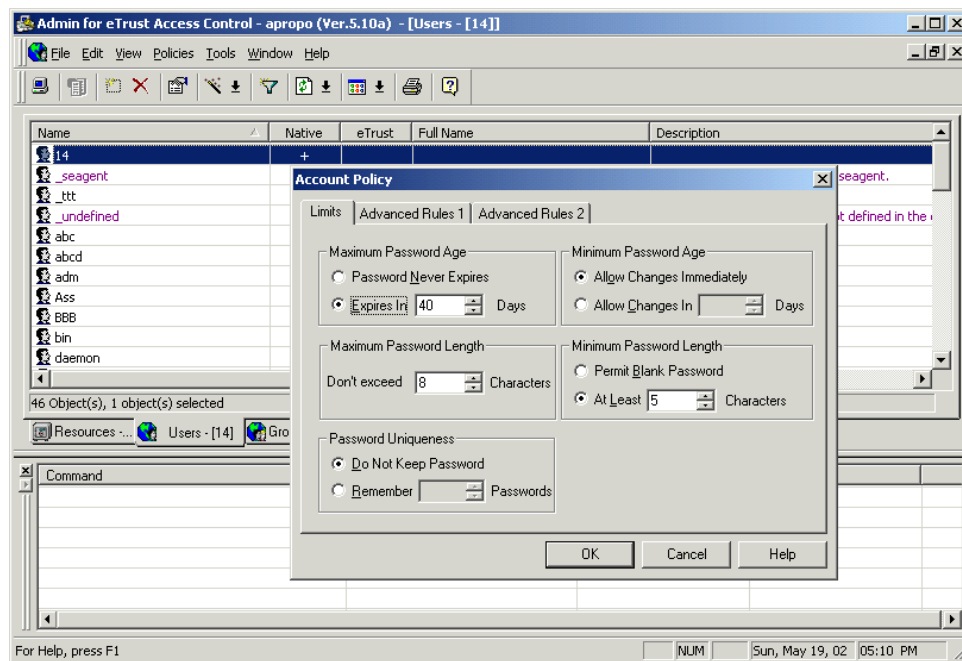


Figure 15-20 Some of the password enforcement available

eTrust Access Control has many other important features, like network protection (both incoming and outgoing connections), the ability to limit who can start and stop system processes, password synchronization across the organization, and much more.

## 15.3.3 Audit

### eTrust Audit

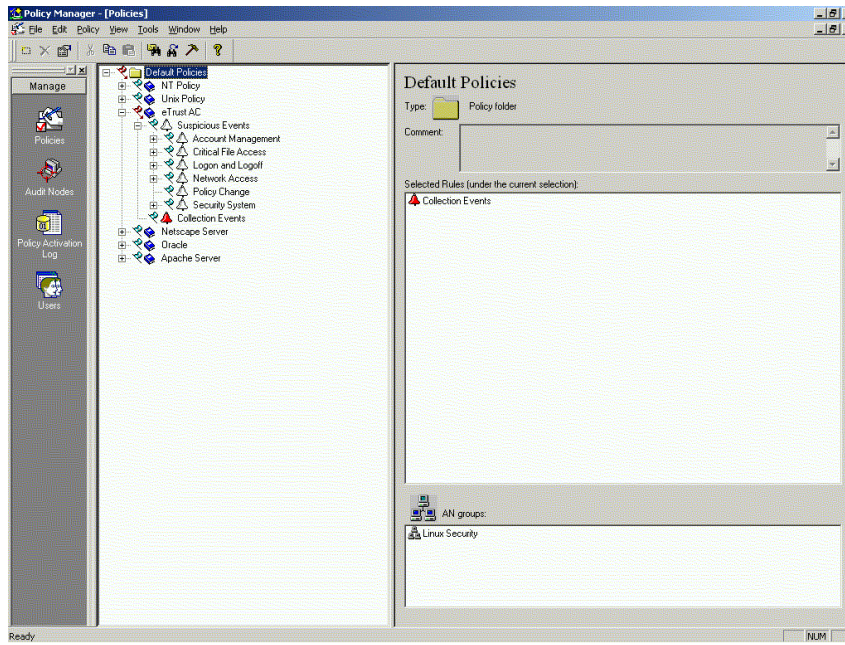
This section describes some of the considerations for using eTrust Audit.

Use eTrust Audit in conjunction with eTrust Access Control to collect event information from a Linux machine. The following section describes how to set up the eTrust Audit Policy Manager to do this. Also refer to the information on setting up eTrust Access Control on the Linux side in the previous chapter.

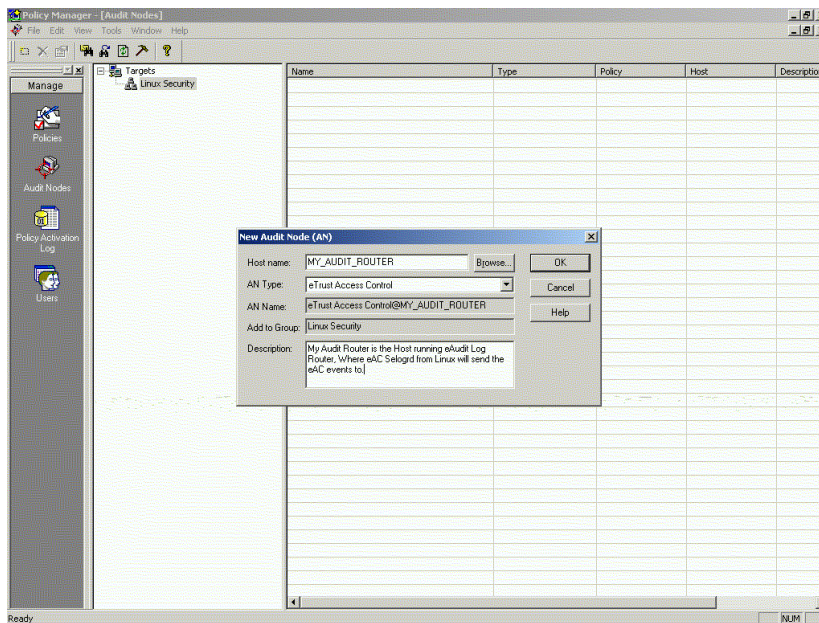
## Using eTrust Audit Policy Manager

On your NT or Windows 2000 machine running eTrust Audit Policy Manager, perform the following administrative tasks:

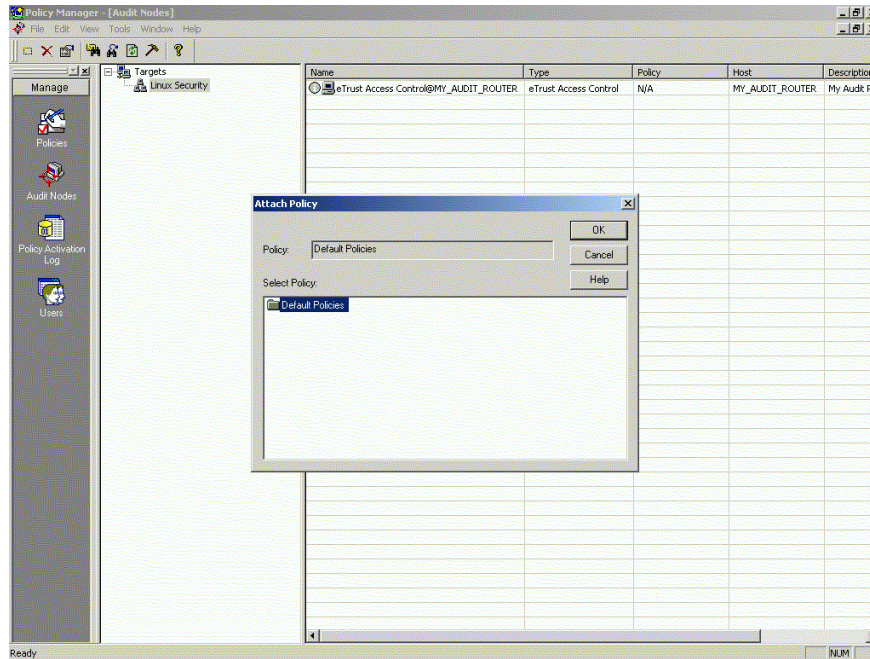
1. Set a Policy for eTrust Audit. In the following figure, the rule Collection Events is selected.



2. Create a new Audit Node with eTrust Access Control selected as the AN Type. The following figure illustrates sample entries from the New Audit Node dialog.



3. Attach the Policy Folder with the eTrust Access Control policy created in step 1 to the AN group with the Audit Node created in step 2.



4. Finally, activate the policy. Events from your Linux machine will be sent using eTrust Access Control SeLogRD to your eTrust Audit Log Router on your Windows NT or Windows 2000 machine.

## 15.4 Managing VM

### 15.4.1 VM:Manager VM Management for Mainframe Linux

This section describes some of the features of VM:Manager VM Management for Mainframe Linux.

VM:Manager VM Management for Mainframe Linux integrates, automates, and supplements its components in the management of your z/VM system. It eliminates many manual system management tasks and gives responsibility for the functions to the software.

#### **How it makes VM system management easier**

The Automated Installation and Maintenance (AIM) component allows you to automate tasks that you may currently be performing manually. Using the AIM component, you can:

- ▶ Configure the other components of VM:Manager for Mainframe Linux
- ▶ Set up the component interfaces

#### **Advantages**

The VM:Manager VM Management for Mainframe Linux configured components significantly reduce the time you spend:

- ▶ Determining the appropriate configuration of your system
- ▶ Setting up component configuration files
- ▶ Activating component interfaces
- ▶ Giving command authorizations to userids

By installing VM:Manager VM Management for Mainframe Linux using the AIM procedures, you can reduce the time and resources your data center invests in component setup, implementation, administration, and daily use.

Thus, using VM:Manager for Mainframe Linux to meet your VM system management needs saves you time, resources, personnel, and headaches.

## **VM:Manager VM Management for Mainframe Linux features**

VM:Manager VM Management for Mainframe Linux contains a set of configured components. The following sections briefly describe them.

### ***Configured components***

Each component is supplied already configured. The records in each component configuration file implement the recommended operation of the component. Once installed, VM:Manager VM Management for Mainframe Linux quickly becomes fully operational.

Some component configuration records are not defined because the specifications of your specific data center cannot be accurately anticipated. However, you can configure these records and change the recommended component configurations to meet the demands of your VM system. When changing a configuration record, make sure you read the documentation on that record.

VM:Manager VM Management for Mainframe Linux is composed of the following products, described in subsequent sections of this chapter.

- ▶ Unicenter VM:Account: Resource accounting, reporting, and capacity management
- ▶ Unicenter VM:Operator: Automated console message management
- ▶ Unicenter VM:Schedule: Personnel and computer resource management
- ▶ Unicenter VM:Spool: Spool information and management
- ▶ BrightStor VM:Backup: CMS and non-CMS data backup with high-speed disaster recovery
- ▶ BrightStor VM:Tape: Drive and tape volume management
- ▶ eTrust VM:Director: Security and directory management
- ▶ Unicenter CA-Explore Performance Management for VM: Performance management

CA has introduced this special suite of solutions to manage the VM environment when Linux runs under VM, to provide comprehensive VM management in support of multiple Linux guests.

There are two VM:Manager Suite for Linux bundles: one includes eTrust CA-ACF2 Security for VM, and the other includes eTrust CA-Top Secret Security for VM. Otherwise, they are identical.

### **Unicenter VM:Account**

This section describes some of the features and considerations for using Unicenter VM:Account.

Unicenter VM:Account is a resource accounting, reporting, and capacity management system that provides a complete picture of your VM resource usage. You can use it for resource tracking and costing, capacity planning, and workload balancing. Unicenter VM:Account provides the information you need to understand and get the most benefit from your VM systems.

Features include the following:

- ▶ Account for all VM resources

- ▶ Timely, protected information
- ▶ Associate costs with resources
- ▶ Track resources and enforce budget limits
- ▶ Track DB2/VM resource/Shared File System usage

## Using Unicenter VM:Account

Unicenter VM:Account provides accounting information in the form of:

- ▶ Full-screen queries with the Query Usage Selection Screen
- ▶ Reporting—10 COBOL reports provide information about your accounting system
- ▶ Trend files collect data over time so you can analyze long-term trends in resource usage

### *Choosing an accounting method*

Decide whether to base your accounting method on account numbers or projects. First, consider the different elements that make up an accounting system.

**Customers:** Customers are persons or organizations who use resources; for example, a department or division in a company. Customers receive invoices charging them for the resources consumed by the userids under their control. Invoices reflect charges in terms of the account numbers assigned to the users under their control or the projects users were working on. Unicenter VM:Account uses a 12-character field, Customer ID, to identify customers.

**Account Numbers:** An account number of up to eight characters exists in the VM user directory for each userid. An account number can have multiple userids associated with it. A userid is assigned to only one account number. A Customer ID can be associated with multiple account numbers, but an account number can belong to only one Customer ID. For example, users in the Accounting department can be identified by one account number, and users in the Data Processing department can be identified by another.

**Projects:** Unicenter VM:Account also maintains project information. You can track projects even if you do not choose to charge customers by project. Project names are 1 to 24 characters long and provide an additional way to track charges. Examples of projects include tasks performed on a specific product or service, contracts, or grants. A project can have one or more account numbers, and an account number can belong to one or more projects. Users can use only one project at a time, but they can use the PROJECT command to change their project at any time during a terminal session.

A Customer ID can have multiple projects associated with it. A project belongs to only one Customer ID.

**COSTABLE Files:** COSTABLE files (also referred to as cost tables) contain rates and other identifying information. Each COSTABLE file corresponds to a different Unicenter VM:Account accounting record. For example, the minidisk (MDISK) COSTABLE file maintains the rates charged for owning minidisk space. Use the UPDATE command to create and maintain COSTABLE files.

### *Accounting structure*

The accounting structure determines how customers are charged for resource usage. It defines the relationship between userids, account numbers, projects, and Customer IDs. You can use one of two structures:

- ▶ Charging by Account Number
- ▶ Charging by Project

If you want to change from one accounting method to the other, make the change at the end of an accounting period.

## **Unicenter VM:Operator**

This section describes some of the features and considerations for using Unicenter VM:Operator.

Unicenter VM:Operator automates VM systems operations by managing message traffic for VM system operators and making it easier to respond to pending messages. Without Unicenter VM:Operator, VM treats all operator messages the same, regardless of importance. All messages go to one operator console, making it difficult to divide tasks among operators working at different physical locations. Further, when the screen clears, the messages are lost because VM does not provide a way to recall, repeat, or save messages.

Features include the following:

- ▶ Special treatment for critical messages
- ▶ Reduction of nonessential message traffic
- ▶ Multiple operator windows and consoles
- ▶ Support for message-triggered action routines
- ▶ Session support
- ▶ Online review of operator console activity
- ▶ Hardcopy logging and spooling
- ▶ System programmer support
- ▶ BrightStor VM:Tape support
- ▶ An interface with VTAM
- ▶ Security features
- ▶ Remote system support

All of these functions can be tailored to meet your site's needs when Unicenter VM:Operator is installed.

### ***Using Unicenter VM:Operator***

**Processes:** Each task Unicenter VM:Operator performs is referred to as a *process*. These tasks include:

- ▶ Processing commands a user enters from the command line
- ▶ Handling tape managers
- ▶ Handling window managers
- ▶ Handling VMYIAMOP connections
- ▶ Executing macros and dialogs

The Unicenter VM:Operator system is multithreaded; that is, processes in the system take turns running. A process runs until it either completes or gives up control. At that point, another process runs. It is a basic assumption that processes function well and give up control voluntarily. To isolate processes from one another and still maintain CMS accessibility, separate CMS SVC save area chains and program stack buffers are maintained for each process. From the programmer's point of view, one process at a time executes in the system.

Unicenter VM:Operator creates an internal control block called a process control block (PCB) for each process in the system. The PCB contains information about the process. Unicenter



VM:Operator maintains the PCB on a variety of linked lists. All PCBs are chained together on a master linked list referred to as the SYSLIST that is anchored in the VMYSYS MODULE.

Use the QPCB command to list all processes currently active in the system. At any point in time, a process can be in one of the following states:

- ▶ Running
- ▶ Eligible to run
- ▶ Waiting to run for an event

Depending on the state of a process, Unicenter VM:Operator places the PCB on a different linked list. The PCB for the currently running process and the PCBs for processes that are eligible to run are linked on the SYSREADY linked list. This is referred to as the *active process queue*. The PCB at the head of the active process queue is the currently running process.

When the currently running process gives up control because it has to wait for an event to take place, the PCB is removed from the active process queue and placed on the SYSWAIT linked list, which is referred to as the *process wait queue*. The next process on the active process queue becomes the running process.

When the event on which the process is waiting occurs, the PCB is removed from the process wait queue and returned to the end of the active process queue; that is, it becomes eligible to run again.

**Control Files:** Unicenter VM:Operator control files identify the general attributes of your system and define how it is configured. There are various types of Unicenter VM:Operator control files. They are the Control file, the Configuration file, CONSOLE files, USERID files, Routing table files, User exits, HOLDMSG files, SESSIONS files, and INCLUDE files.

**Consoles:** Unicenter VM:Operator allows concurrent access from multiple 3270 display consoles. Each console is managed by its own process and can be started, stopped, or restarted as necessary without having to restart Unicenter VM:Operator. There are four types of consoles: Logon, Dedicated, Attached, and VMYIAMOP, which are consoles that can belong to any userid.

**Windows:** Processes can be defined in CONSOLE or USERID files. Processes defined in CONSOLE files begin running during the console initialization and continue running until the console stops. Processes defined in USERID files start when the VMYIAMOP program connects and end when the console is canceled.

All Unicenter VM:Operator console interaction is done in logical display spaces called *windows*. A CONSOLE or USERID file can identify one or more INCLUDE files that describe the windows that can be displayed on a console. INCLUDE files define the characteristics of VM:Operator windows, including processes that execute in each window, program function key settings for each window, color settings, and reserved window text line settings. Windows can be updated without actually being displayed. In fact, processes interacting with windows are unaware of whether the window is visible or covered by another window.

## Unicenter VM:Schedule

This section describes some of the features and considerations for using Unicenter VM:Schedule.

Unicenter VM:Schedule is a component of the VM:Manager VM Management Suite for Mainframe Linux. It allows users to schedule jobs to process in the future instead of right now. A scheduled request consists of an EXEC file, or command plus the scheduling instructions.

Requests can print SCRIPT files, compile COBOL programs, issue CP and CMS commands, and run other types of CMS programs.

Users can schedule requests to run just once or to repeat at regular intervals; for example, at 1:00 p.m. this coming Friday, 6:00 a.m. every business day, or on the last day of every month. Users can schedule requests to run on their own or other users' userids, or EXECs to run on the Unicenter VM:Schedule service virtual machine.

Most scheduled requests run on the user's userid. When it is time for the request to run, Unicenter VM:Schedule checks to see if the user is logged on and reminds the user to log off. Once the userid is logged off, Unicenter VM:Schedule autologs it—that is, logs it on in disconnected mode. The request then runs as if the user entered the command directly from the terminal. Unicenter VM:Schedule monitors and records what happens during request processing. When the request runs for the last time, and whenever an error occurs within the request, Unicenter VM:Schedule sends a file to the user's reader as notification.

Your site can use Unicenter VM:Schedule to set up frequently repeated work in advance, such as monthly accounting reports. Users can schedule time-consuming tasks to run when they are not in the office so that the requests do not interrupt other work.

Users can automatically print their OfficeVision/VM note logs every Friday, run DB2 reports each morning before they get in, and remind colleagues about meetings. Programmers can compile and test programs overnight. Data center staff can automate system backups, performance reporting, and system control utilities.

Features include the following:

- ▶ Enable end users to schedule jobs
- ▶ Improve machine utilization
- ▶ Run concurrent operations
- ▶ Monitor resource consumption and job requests
- ▶ Ensure security within the CP directory by requiring and verifying passwords
- ▶ Schedule maintenance
- ▶ Schedule operations tasks
- ▶ Set up optional class structures
- ▶ Control scheduled requests
- ▶ Automate initiation of common operations

## **Using Unicenter VM:Schedule**

If you are authorized as a user or operator, there are two ways to work with Unicenter VM:Schedule: full-screen menus and line-mode commands. The screens are easy to use, especially if you are unfamiliar with Unicenter VM:Schedule or do not schedule requests often. The line-mode commands let you use Unicenter VM:Schedule within EXECs.

To work with Unicenter VM:Schedule through full-screen menus, type `vmshed` on the CMS command line and press Enter. Unicenter VM:Schedule displays the main menu for your user level.

Type the number of the task you want to perform and press Enter. (If you prefer, you can enter the command name or its abbreviation instead.) Unicenter VM:Schedule displays a fill-in-the-blank screen for that task. Where appropriate, default values for fields are shown. When you complete the task, press PF12 to submit your instructions to Unicenter

VM:Schedule. If you forget to fill in a required field, Unicenter VM:Schedule prompts you to enter the information.

If you want further explanation for any item on a screen, press Tab to put your cursor on the item and press PF1. Unicenter VM:Schedule displays an explanation. Press PF3 to return to the task screen.

To use Unicenter VM:Schedule line-mode commands, use the command VMSCHED (which is the default name of the Unicenter VM:Schedule service virtual machine) as the first word of each command line. If the Unicenter VM:Schedule service virtual machine at your site has a different name, use that name instead.

For example, to transfer the ACCOUNTS request from user ALICE to user LOUISE through a line-mode command, enter from CMS:

```
vmsched transfer accounts alice louise
```

All responses are edited according to your virtual machine's current EMSG setting.

While using Unicenter VM:Schedule, you will receive messages that do one of the following:

- ▶ Inform you of the status and progress of the function or task
- ▶ Prompt you for information about your system
- ▶ Indicate if an error occurred
- ▶ Provide instructions for resolving errors

## Unicenter VM:Spool

This section describes some of the features and considerations for using Unicenter VM:Spool.

Unicenter VM:Spool is a spool-space management system for the z/VM environment. Unicenter VM:Spool lets you:

- ▶ Monitor the use of spool space to prevent problems from occurring
- ▶ Charge users for the use of spool space
- ▶ Back up and restore all or selected spool files to and from tapes
- ▶ Allow users to browse through all of their open and closed spool files; Unicenter VM:Spool does this by serving as a personal spool file manager

Unicenter VM:Spool helps you manage files such as CMS notes, OfficeVision notes, console logs, and other files in your virtual reader, printer, and punch. These files are called *spool files*. If you have any spool files, you receive a message like this when you log on to z/VM:

```
FILES: 002 RDR, 001 PRT, 001 PUN
```

Features include the following:

- ▶ An OVERVIEW screen that shows which users are using the most spool space and which users have the largest spool files
- ▶ A GRAPH screen that shows the percentage of spool space that is in use at specified intervals
- ▶ A SYSUSE screen that shows how much system SPOL, DUMP, and PAGE space is defined, how much is being used, and how it is being used
- ▶ An automatic spool-space monitoring and warning facility that lets you define a usage threshold and receive a message when spool usage approaches that threshold

- ▶ A SPOOLIST screen that lets users manipulate their own spool files
- ▶ A SPOOLALL screen that allows the system administrator to manipulate or obtain summary information about all spool files on the system
- ▶ A purge facility that can remove selected spool files from the system
- ▶ A backup and restore facility that can back up and restore all or selected files to and from tapes
- ▶ A user macro capability, similar to the XEDIT macro capability in CMS, that lets you programmatically manipulate your spool files
- ▶ Utilities that create a database of information about past spool-file activity
- ▶ An accounting facility that lets you charge for the use of spool space
- ▶ An audit facility that tracks the use of Unicenter VM:Spool commands
- ▶ The ability to see what is printing on the system printers

### Using Unicenter VM:Spool

**SPOOLIST screen:** The SPOOLIST screen initially displays only the first eight fields: –Q–, Owner, File, Date, Time, Creator, Filename, and Filetype. The SPOOLIST screen is divided into rows. Each row consists of a line of information about a single spool file. A row contains one entry in each field for a particular file, with several exceptions:

- ▶ The Filename and Filetype fields for an unnamed console file remain blank
- ▶ The Distcode and Destcode fields may be blank
- ▶ The T field is blank if the file has never been transferred

**Moving through screens and menus:** You can perform most Unicenter VM:Spool functions by using the PF (program function) keys, described near the bottom of the SPOOLIST screen. To see the alternate PF key panel, press PF2 (Alt PF). Press PF2 again to see the initial PF key panel.

**Moving around the SPOOLIST screen:** Move up and down the SPOOLIST screen one screen at a time by pressing PF7 (Back) and PF8 (Forw). Use PF5 (Right) and PF4 (Left) to view additional fields on the SPOOLIST screen.

**Looking for a particular string:** To scan the SPOOLIST screen for a particular string, use the / (locate) command. If Unicenter VM:Spool finds the string, the line on which it is located becomes the current line and displays at the top of the screen. The search starts with the line following the current line. The / command is similar to the XEDIT LOCATE command.

For example, to have the current line display the spool file with a filename of SAS, enter:

```
/sas
```

**Saving the list:** Without leaving the SPOOLIST screen, you can save the list of your spool files on your A-disk by entering the SAVE command. For example, to save the list as SPOOL LIST A, enter:

```
save spool list a
```

If you do not assign a name to the file, it defaults to VMSPOOL DISPLAY A.

**Updating the SPOOLIST display:** To update the display with information about new spool files and any changes to your existing spool files, press PF10 (Refresh). The updated display shows spool files that were sent to you since you entered the SPOOLIST screen. Files that you deleted or transferred to another user are not shown.

**Customizing Unicenter VM:Spool functions:** You can customize many of the functions with Unicenter VM:Spool user macros. These macros are similar to XEDIT macros in CMS. For example, on the SPOOLIST screen, you see the spool files in the following order: PRT, PUN, RDR, and OPN. Within each queue, they display as they appear in the CP queue. You can change the order with a user macro that sorts the spool files and displays them the way you want to see them.

**Unicenter VM:Spool commands:** The quickest way to enter SPOOLIST commands is by using the PF keys, but you can also enter SPOOLIST, CP, and CMS commands on the SPOOLIST command line.

## **BrightStor VM:Backup**

This section describes some of the features and considerations for using BrightStor VM:Backup.

BrightStor VM:Backup is a full-screen backup and restore system for z/VM. It performs backup jobs that copy data from minidisks, SFS file spaces, and BFS file spaces to tape. If the data on your minidisk or SFS file space is lost or damaged, BrightStor VM:Backup can restore the data from tape to a virtual reader, minidisk, or SFS file space. Additionally, if the data in your BFS file space is lost or damaged, BrightStor VM:Backup can restore the data from tape to a BFS file space. Through the use of BrightStor VM:Backup, you get excellent reliability and performance of backup and restore. You also get automated tape management, catalog management, and administration.

Features include the following:

### ***Backup capabilities***

- ▶ Define your backup strategy to configure BrightStor VM:Backup to run on a schedule that best suits the needs of your site
- ▶ Back up Shared File System data with complete support of the Shared File System (SFS)
- ▶ Ensure that no CMS minidisk file is missed
- ▶ Back up non-CMS minidisks
- ▶ Handle special minidisk types:
  - CMS and non-CMS data on a recomputed minidisk
  - Data on reserved minidisks, which have a special CMS format
  - Handle active minidisks
  - Backup minidisks without passwords
- ▶ Detect online directory changes automatically
- ▶ Back up to DASD or tape
- ▶ Pack and encrypt data
- ▶ Perform fast physical backups

### ***Restore capabilities***

- ▶ Robust and flexible backups make your restore process quick and simple
- ▶ Use integrated minidisk and SFS facilities
- ▶ Simplify end-user restores
- ▶ Restore files without first restoring minidisks
- ▶ Back up CMS files from one type of DASD and restore them to another type of DASD, regardless of configuration differences between the DASD types

- ▶ Restore an entire DASD volume
- ▶ Restore to an alternate DASD volume
- ▶ Handle a major disaster using the high-speed disaster recovery component
- ▶ Efficiently test your disaster recovery plan

### ***Tape and catalog management***

- ▶ Use built-in or external tape management:
  - Manage all BrightStor VM:Backup tapes
  - Provide pools of scratch tapes for use by different types of backup jobs
  - Provide tape drive selection and allocation for backups and restores
  - Verify the standard label whenever a backup tape is mounted
- ▶ Create duplicate tapes automatically
- ▶ Rely on full standard labels
- ▶ Reduce tape remounts
- ▶ Depend on an online catalog of current data
- ▶ Save catalog disk space

### ***Implementation and administration***

- ▶ Administer the system from full screens
- ▶ Process multiple tasks
- ▶ Reduce problems through checkpoint/restart
- ▶ Protect against unauthorized data access
- ▶ Run without operating system modifications
- ▶ Implement user exits in REXX

### ***Starting BrightStor VM:Backup***

1. Log on to VMBACKUP. AIM verifies the accuracy of the VMBACKUP MDISKS file on the VMRMAINT 192 minidisk. VMISTART verifies that BrightStor VM:Backup is properly configured and asks whether you want to start BrightStor VM:Backup.
2. Enter yes. BrightStor VM:Backup initializes, then displays this message:
 

```
VMBACKUP IS READY TO USE: ddmmmyy hh:mm:ss
```
3. Disconnect the BrightStor VM:Backup service virtual machine.

To have BrightStor VM:Backup start automatically when your VM system is IPLed, include VMBACKUP in your AUTOLOG userid's PROFILE EXEC.

### ***Running a full test backup job***

**Note:** These procedures may vary, depending on the tape management product you are using.

1. Log on to VMRMAINT.
2. Make sure there are two standard label tapes labeled B00001 and B00002 available for your BrightStor VM:Backup test. If not, use the CMS TAPE WVOL1 command to initialize the tapes with standard labels.

You will be running a backup job using templates loaded down for you during installation. If you want to look at these templates, follow these steps:

- a. Enter `vmbackup` to display the System Administrator Main Menu.
  - b. Select **Manage Job Templates**.
  - c. Type `upd` next to the DAILY or WEEKLY template name you want to view.
  - d. When you finish reviewing the template, press PF3 three times to return to CMS.
3. To submit the full backup of the VMRMANT 191 minidisk, enter:
 

```
vmbackup submit weekly
```
  4. You are prompted for the volser of the tape to use for the backup job. Reply by entering:
 

```
vmbackup reply n b00001
```

 where *n* is the request number of the BrightStor VM:Backup prompt.
  5. BrightStor VM:Backup confirms the volser by repeating the request. Reply by entering:
 

```
vmbackup reply m b00001
```

 where *m* is the new BrightStor VM:Backup request number.
  6. After BrightStor VM:Backup confirms the volser, it sends a tape mount request to the tape operator's console:
 

```
VMBMNT623I MOUNT REEL B00001 AT 310...
```
  7. Mount tape B00001 on a drive.
  8. Attach the tape drive to VMBACKUP as 310 by entering:
 

```
attach raddr to vmbackup as 310
```

 where *raddr* is the real address of the tape device and 310 is the virtual address at which BrightStor VM:Backup has requested the drive.

When BrightStor VM:Backup completes the job, it sends output listings and a NOTE file to the VMRMANT virtual reader. The NOTE file indicates whether any problems were encountered during the backup job.

For additional procedures for running an incremental test backup job, restoring a file, and putting the DAILY and WEEKLY templates into production, see the product documentation.

## BrightStor VM:Tape

This section describes some of the features and considerations for using BrightStor VM:Tape.

BrightStor VM:Tape is a comprehensive tape drive and tape volume manager for the z/VM environment. It replaces potentially error-prone manual tape catalog procedures with a fully automated control system. Having information on tape volumes and tape drives immediately accessible through full screen panels significantly improves operator productivity. Allowing BrightStor VM:Tape to control the allocation of your tape devices helps you maximize your hardware investment. Comprehensive reporting and auditing provide you with the evidence that you are protecting your organization's data efficiently.

BrightStor VM:Tape decreases operator tasks, controls access to tape data, and provides comprehensive reporting and auditing of tape use. It provides centralized control of tape drive and volume management for multiple systems with shared DASD. These can be both z/VM or OS/390 systems and can be running native or as guests. BrightStor VM:Tape uses shared DASD to dynamically read and update the CA-1 Tape Management Catalog (all versions) on the MVS system.

Features include the following:

### ***Controls Tape Drives***

BrightStor VM:Tape manages tape drive assignments for the z/VM system operator.

- ▶ Shared drive control
- ▶ Drive sharing with OS/390 and z/VM
- ▶ Tape drive allocation
- ▶ Automated tape mounting

### ***Controls Tape Volumes***

BrightStor VM:Tape uses a Tape Management Catalog (TMC) to keep track of tape usage and authorizations.

- ▶ Single tape volume library for Multiple z/VM systems
- ▶ Single tape volume library for z/VM and OS/390 systems
- ▶ Flexible volume numbering
- ▶ Tape volume verification
- ▶ Advance volume setup
- ▶ “Foreign” tape control
- ▶ Scratch tape pools
- ▶ Automatic scratch selection
- ▶ Retention control
- ▶ Bin/Slot system
- ▶ Processing multiple events concurrently
- ▶ You can customize and optimize BrightStor VM:Tape for your unique requirements with comprehensive user exits

### ***Enables end-users***

- ▶ Request tape mounts
- ▶ Manage tape volumes
- ▶ Control shared tape volumes

### ***Supports tape librarians***

- ▶ Online inquiry/update to TMC
- ▶ Audit tape usage

### ***Supports operations***

- ▶ Operators can display drive and volume information with the BrightStor VM:Tape query command. Operators can also query outstanding requests and the status of any library tape volume.
- ▶ BrightStor VM:Tape significantly reduces operator tasks by providing multiple-system tape drive allocation, tape drive management, and tape volume control.

### ***Easy installation***

No CP or CMS modifications are required to install BrightStor VM:Tape. The Automated Installation and Maintenance system can quickly set up a working BrightStor VM:Tape system with the default configuration file. You can customize BrightStor VM:Tape by modifying the configuration file.



### ***Flexible configuration***

The BrightStor VM:Tape configuration file specifies local system parameters and authorizations.

### ***User exit facility***

Comprehensive user exits for customizing BrightStor VM:Tape are provided.

### ***Phased implementation***

BrightStor VM:Tape supports gradual implementation for sites with unique requirements by allowing the operator to mount tapes on behalf of users. Users can ask the operator for mounts by using traditional methods until full implementation is accomplished. BrightStor VM:Tape also provides read-only access to the TMC and AUDIT files or data sets. This feature allows sites to test the link to previously existing BrightStor VM:Tape or CA-1 systems without affecting them.

### ***Before initializing BrightStor VM:Tape***

Before you initialize BrightStor VM:Tape for the first time, review the predefined SERIES records and records that identify default POOLNAME and POOLASGN files in the VMTAPE CONFIG file. When you use BrightStor VM:Tape with a CMS TMC, AIM defines the SERIES records for reels, or cartridges, or enhanced capacity cartridges. If your data center uses both reel and cartridge tape drives, you must add the appropriate SERIES records to the VMTAPE CONFIG file.

If you want to change the BrightStor VM:Tape configuration, it is better to do so before initialization because AIM builds the TMC when you initialize BrightStor VM:Tape. If you decide to change the BrightStor VM:Tape configuration after initializing BrightStor VM:Tape, you have to erase the TMC and then run VMTBUILD to rebuild the TMC.

### ***Starting BrightStor VM:Tape <CMS>***

Log on to VMTAPE. AIM verifies the accuracy of the VMTAPE MDISKS file on the VMRMANT 192 minidisk. VMISTART verifies that BrightStor VM:Tape is properly configured and asks whether you want to start BrightStor VM:Tape. Enter yes. BrightStor VM:Tape initializes; then it displays this message:

```
VMTAPE IS READY TO USE: ddmmmyy hh:mm:ss
```

### ***Starting BrightStor VM:Tape<OS>***

The VMISTART command in the BrightStor VM:Tape<OS> PROFILE EXEC has been installed with the NOSTART option. This option prevents BrightStor VM:Tape<OS> from starting so that you can set up the shared TMC environment.

1. Add OSTMC and OSAUDIT records to the VMTAPE CONFIG file. These records identify the virtual address and data set names of OS format TMC and audit files.
2. Verify that the TMC data set and audit data set reside on separate, real DASD volumes.
3. Prepare the BrightStor VM:Tape USEREDIT user exit to provide the identical translation for VMTAPE that the USEREDIT macros provide for CA-1.
4. Use the XEDIT command to edit the BrightStor VM:Tape<OS> PROFILE EXEC. Remove the NOSTART option from the VMISTART command.
5. Initialize VMTAPE<OS> by entering profile.

### ***eTrust VM:Director***

This section describes some of the features and considerations for using eTrust VM:Director.

eTrust VM:Director is an efficient and comprehensive directory and DASD management system for z/VM. It significantly reduces the time required to manage the resources defined in the VM directory and prevents costly errors.

## **Manage the VM directory**

### ***Reduce the burden on systems staff***

With eTrust VM:Director, your site can assign users without special VM privileges to be directory managers for any specific group of users on the system. Directory managers can perform routine functions for their user groups such as defining minidisks, authorizing resource sharing, and managing disk space. This significantly reduces demands for centralized system support.

### ***Empowers end users***

End-user screens minimize training needs. The full-screen menus provided by eTrust VM:Director also allows users to change the configuration of their own virtual machines. For example, from the user selection menu, users can change their logon passwords; change their minidisk link modes and passwords; define directory links to other users' minidisks; and remove other users' directory links to their minidisks.

### ***Save time for directory managers***

Utilizing eTrust VM:Director, directory managers can remain interactive while managing disk space. The eTrust VM:Director WAIT/NOWAIT option permits directory managers to regain access to their terminals without waiting for requests to complete. The time-consuming process of adding, deleting, holding, or activating multiple user IDs is automated with eTrust VM:Director. Managers can perform each of these functions for numerous user IDs simultaneously, saving hours of systems staff time.

When new user IDs are added to the directory, eTrust VM:Director saves time with its directory profile and SKELETON file features. A directory profile is a file that contains a set of frequently used directory control statements. These statements can be added to any user ID directory entry through use of a single command. A SKELETON file is a template directory entry file that defines the basic virtual machine configuration for a new user ID in a specific group. For example, when a new user ID is added to the accounts department, the site-defined ACCOUNTS SKELETON file will automatically become the template for the new user. It includes statements such as account numbers valid for that group. Thus, systems staff time is free for other tasks.

eTrust VM:Director lets you update the VM directory significantly faster than traditional methods. In addition, multi-tasking capabilities allow many users to make directory changes simultaneously.

## **SFS support**

### ***Create SFS file pools***

To create a file pool, you must define the file pool's server and its storage groups, which are collections of minidisks. eTrust VM:Director's directory management capabilities include:

- ▶ Full screens to walk you through the process
- ▶ Automatic location of DASD space for new minidisks
- ▶ Fast, reliable directory updates

All directory changes are audited, and you can generate a report of those changes to assist you in completing the configuration of the file pool server.

### ***Enroll users in SFS***

Once a file pool is created, you enroll users in that file pool. eTrust VM:Director provides full screens and linemode commands that make it easy to enroll users and perform other SFS user administration tasks. In addition, eTrust VM:Director provides a special MOVE2SFS command that automatically enrolls users, creates appropriate user file spaces, and moves them from the user's minidisks to file spaces.

### ***Move minidisks to SFS***

Using eTrust VM:Director's MOVE2SFS command greatly simplifies the process of moving user minidisks to SFS. When you issue the command, you specify the user ID, the minidisks to be moved, the file pool, and other parameters. eTrust VM:Director enrolls the user in the file pool, creates SFS file spaces and directories, and copies files from minidisks to the directories, and sets or adjusts the file space allocation limit. eTrust VM:Director even calls a user exit that can perform special functions like notifying affected users and updating applications.

### ***Simplify SFS administration***

With native z/VM, file pool administrators do SFS administration. These administrators have the ability to control all file pool resources and even read from or write to any file in the file pool. No full screens are provided for file pool administrators, and some of their activities are not audited. eTrust VM:Director enables you to reduce the number of file pool administrators and improve security. The first step is to make eTrust VM:Director a file pool administrator. Next, you authorize a limited set of users as eTrust VM:Director SFS administrators. By using eTrust VM:Director full screens and linemode commands, these SFS administrators work through eTrust VM:Director to define and administer the SFS environment to the level of authority that you define. All tasks are fully audited.

### ***Decentralize SFS user administration***

By authorizing certain users as SFS managers, you reduce the workload for administrators. Each SFS manager is assigned a group of users and can perform tasks like enrolling users in file pools, setting and changing file space allocation limits, and deleting file spaces. Tasks can be performed using full screens and linemode commands that are designed for ease of use. SFS administrators never lose control because they set the limits under which SFS managers operate.

## **Manage disk space**

### ***Delegate disk space management***

With eTrust VM:Director, you can divide real disk volumes into sub-pools. The sub-pool resources can then be delegated to directory managers. For example, the system administrator might give 50 or 100 cylinders to an engineering group. The directory manager in that group can then allocate and deallocate minidisks from these cylinders without involving system personnel, again reducing demands on your central system support staff.

### ***Automate DASD relocation***

eTrust VM:Director automates the tedious and error-prone process of CMS minidisk migration. You can also identify which DASD contains fragmented free space. A simple menu selection is used to move minidisks and maximize the amount of contiguous free space available.

### ***Implementation and use***

With eTrust VM:Director, you have the choice of decentralized or centralized directory management, or some combination of both. This means you retain control over all functions while delegating as many or as few as you see fit.

### ***Easy to implement***

eTrust VM:Director is easily installed, requiring no modifications to CP or CMS. A simple utility copies and converts your existing directory to the eTrust VM:Director format.

### ***Change DASD information dynamically***

Adding DASD and changing configuration information is a common system administrator task. You can modify the active configuration while eTrust VM:Director remains operational, reducing the risk of a security violation. eTrust VM:Director tracks all minidisks on your system and validates dynamic changes to the DASD and other configuration information.

### ***Customize user exits***

Numerous user exits are available with eTrust VM:Director.

## **Unicenter CA-Explore Performance Management for VM**

This section describes some of the considerations for using Unicenter CA-Explore Performance Management for VM. We explain how to start and stop Unicenter CA-Explore Performance Management for VM data collection and SQL/DS data collection by Unicenter CA-Explore for VM.

### ***Starting data collection***

Data collection, except for the collection of SQL/DS data, is performed by the EXPLRVM program which resides on the Unicenter CA-Explore for VM service machine's A-disk. To start Unicenter CA-Explore for VM automatically whenever z/VM is IPLed, add the VM service machine to the list of AUTOLOGed user Ids. The Unicenter CA-Explore for VM service machine should normally be left running disconnected.

To manually start data collection, do one of the following:

- ▶ IPL CMS in the service machine.
- ▶ If the service machine's PROFILE EXEC does not contain commands that automatically start Unicenter CA-Explore for VM data collection at IPL, proceed to the next step. The PROFILE EXEC placed on the service machine's A-disk during the install procedure contains commands that automatically start Unicenter CA-Explore data collection at IPL.
- ▶ Issue the EXPLRVM command while logged on to the service machine.

### ***Reconnecting to the Unicenter CA-Explore for VM service machine***

The Unicenter CA-Explore for VM service machine and the SQL data collector are normally run in a disconnected state. We recommend that they be left disconnected for normal operation. If it is necessary to issue commands directly on the service machine, you may use the following procedure to reconnect it:

1. Log on to the Unicenter CA-Explore for VM service machine.
2. If a CP READ status message is displayed (at the lower right corner of the screen) after the RECONNECTED message (which is displayed in the upper left part of the screen, a few lines from the top), then type B and press Enter.
3. Press Enter when the RUNNING status message is displayed.
4. Press PA2 when the MORE status message is displayed.
5. When the Data Collection And Control screen is displayed, you can shut down the service machine, or issue other commands, as necessary.

### ***Stopping data collection***

**Resource Control:** If you are using the Resource Control Facility, you must terminate resource control before stopping Unicenter CA-Explore data collection.

Data collection should normally be left running all of the time, except when it is necessary to perform maintenance, or to shut down the VM system itself.

Use any of these three methods to terminate data collection:

- ▶ From the Data Collection And Control screen on the Unicenter CA-Explore for VM service machine, issue the command **QUIT**.
- ▶ From the command line of the Real-time CMS interface, issue the command **>QUIT**.
- ▶ From the CMS Ready; prompt, or within a user-written exec on a user ID other than the CA-Explore for VM service machine, with the EXPLCMND command line interface, issue the command **EXPLCMND >QUIT**.

## **Starting and stopping SQL/DS data collection**

### ***Starting SQL/DS data collection***

Automatic Start: AUTOLOG the SQL/DS data collection machine. The commands necessary to automatically start SQL/DS data collection were placed in the PROFILE EXEC by the successful execution of the ESQINST exec.

Manual Start: Do one of the following to manually start the SQL/DS data collector:

- ▶ IPL CMS in the SQL/DS data collection machine.
- ▶ Enter EXPSQLDC on the CMS command line while logged on to the SQL/DS data collection machine.

The EXPSQLDC command has the following format:

```
EXPSQLDC [SCAN nn] [DBUSERS nnnnn] [ACTIVE]
```

The command options are described below.

- SCAN *nn* is the data sampling interval, in minutes. The default is 2 minutes. Specify a value from 1 minute to 60 minutes.
- DBUSERS *nnnnn* is the maximum number of user connections (pseudo agents) to monitor. The default is to calculate this number dynamically at initialization time from SQL/DS connection data. Specify a value from 1 user to 99,999 users.
- ACTIVE specifies that SQL/DS data collection is to be performed whether or not a CA-Explore for VM data collector is connected with IUCV. Use of this parameter will cause SQL/DS data collection to impose unnecessary overhead on the database server if the SQL/DS data collection machine does not have an IUCV connection to the CA-Explore for VM service machine.

### ***Reconnecting to the SQL/DS data collection machine***

The SQL/DS data collection machine is normally run in a disconnected state. It is recommended that it be left disconnected for normal operation. The SQL/DS data collector does not support any user interface. If it is necessary to issue commands directly from the SQL/DS data collection machine, use the following procedure to reconnect it.

1. Log on to the SQL/DS data collection machine.
2. If a CP READ status message is displayed (at the lower right corner of the screen) after the RECONNECTED message (which is displayed in the upper left part of the screen, a few lines from the top), type B and press Enter.
3. Press Enter when the RUNNING status message is displayed.
4. Press Clear, or PA2, if the MORE... status message is displayed.

### **Stopping SQL/DS data collection**

To stop SQL/DS data collection, enter the following command while logged on to the SQL/DS data collection machine:

```
EXPQUIT
```

## **15.5 Portals**

### **15.5.1 Portal management**

#### **CleverPath Portal**

This section describes some of the features and considerations for using CleverPath Portal.

CleverPath Portal is a single entry into personalized delivery of Web content. It provides personalized access to all corporate and personal information.

As a CleverPath Portal user, you can locate the information you need from a variety of sources, organize and monitor information that is of special interest or importance to you, and share and collaborate with others in your workgroup as well as throughout your enterprise.

CleverPath Portal can support more than 2 million users, and its rapid deployment tools enable most organizations to perform a successful initial portal installation in two weeks or less. CleverPath Portal can reduce administration and training costs through centralized management and integration with existing systems while supporting zero administration to the desktop, and it provides a truly secure portal by ensuring that only authorized users obtain sensitive information.

Features include the following:

**Workplace:** Your workplace automatically generates a personalized view of the content you've subscribed to. All content from the Knowledge repository (either Channel or Library content) can be organized in your workplace.

When you use CleverPath Portal for the first time, a workplace exists by default, but you can customize it by adding data and creating portlets. You can configure your workplace from the Workplace bar on a Workplace page, or from the Manage Workplaces dialog.

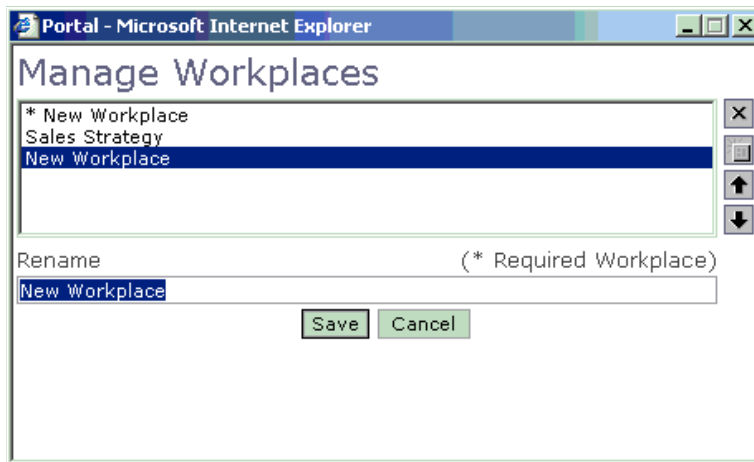


Figure 15-21 Sample CleverPath Manage Workplaces dialog

You can define any number of personal Workplaces, and you can organize and name them however you like. A *default* workplace can be preconfigured by an administrator for new users based on workgroups they establish.

**Knowledge:** The Knowledge repository contains the CleverPath Portal information that you and other people in your organization want to make available throughout your workgroup or enterprise. Knowledge is composed of a Library of content objects like Internet and intranet links, Word documents, or links to other applications, and Channels, which are logical groupings of this information that contain links to objects in the CleverPath Portal Library.

The CleverPath Portal Library is organized using folders and subfolders within a Workgroup. When you create a folder, you specify which users and workgroups have permission to view and modify its content.

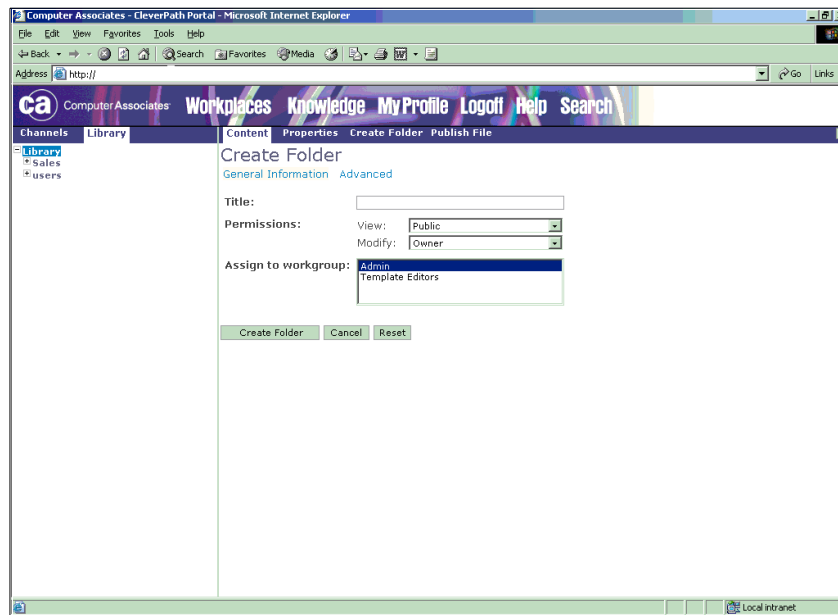


Figure 15-22 Example of creating a CleverPath Portal Library entry

Using Advanced options, you can also define how a folder is displayed when a user selects it, and whether it will be deleted automatically from the Library at a specified time. Add content to the library one object at a time, or use the Archive Extractor Publisher to add multiple files simultaneously.

**Profiles:** Your user profile controls the way CleverPath Portal looks and how it organizes your information. CleverPath Portal provides various templates to personalize the Interface. In addition to the out-of-the-box templates, the CleverPath Portal SDK enables you to customize the Portal interface with your company logo or other visual cues to make your Portal look the way you want it.

**Search:** Using familiar Internet search techniques, you can use the CleverPath Portal Search tool to simplify the process of finding content in the library. CleverPath Portal also provides Spidering to enhance your search capabilities, and Automatic Categories to help organize search content and populate the channels so that current, relevant information is available to the appropriate users.

**Advanced visualization:** Visualization enables you to bring corporate data to life by creating 2D and 3D charts and graphs that can be published to the Portal, modified within the Portal, and refreshed dynamically based on specified time intervals.

**Web Services:** The Web Services Publisher enables you to register remote Web services and native applications with CleverPath Portal. You can create customized portlets for convenient instant access to these registered services.

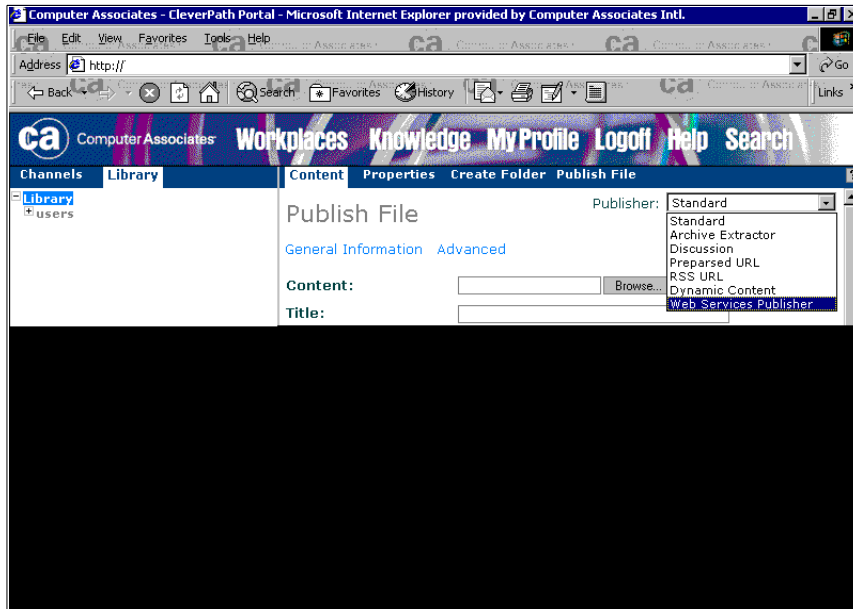


Figure 15-23 Sample screen showing CleverPath Web Services Publisher

**Wireless Services**—You can now access business-critical information directly from your portal, in real time, with CleverPath Portal's wireless capabilities. You can manage your Knowledge content and channels, or view workplaces from your Palm, Pocket PC, cellular phone, or any other wireless device.

To access CleverPath Portal from a wireless device:

1. On your wireless device, enter the following to connect to CleverPath Portal:  
`http://servername:serverport/mobile/`
2. The Portal Login page appears. Enter your user name and password, and then click **OK**.
3. The Portal Menu page appears, indicating options for you to select a function.

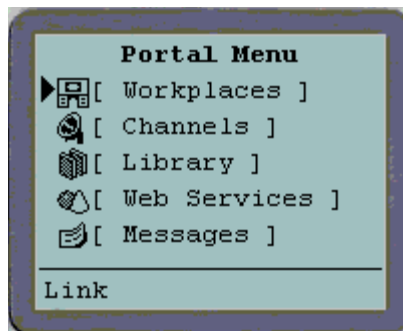


Figure 15-24 Sample CleverPath Portal menu on a wireless device

To manage the Library, for instance, select Library to display a list of existing library content. The Manage option allows you to create a folder, create a Web page link, or perform other library management functions.



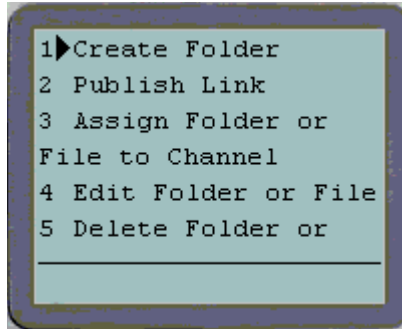


Figure 15-25 Sample CleverPath Portal menu for Library management from a wireless device

If you select Messages from the Portal menu page, you can send an SMS message to a cellular phone using CleverPath Portal's Wireless Services.

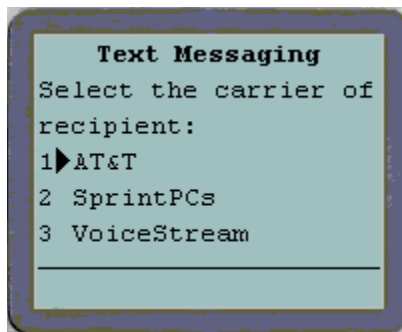


Figure 15-26 Sample CleverPath Portal menu for managing messages on a wireless device

Just select the carrier of the recipient's cell phone from the Text Messaging page, enter the phone number and message text, and click **Send**.





## Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

### IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 452.

- ▶ *Linux on IBM eServer zSeries and S/390: ISP/ASP Solutions*, SG24-6299-00
- ▶ *Linux for IBM eServer zSeries and S/390 : Distributions*, SG24-6264-00
- ▶ *Linux on IBM eServer zSeries and S/390: Application Development*, SG24-6807-00

### Other resources

These publications are also relevant as further information sources:

- ▶ *z/VM V4R3 CP Command and Utility Reference*, SC24-5967
- ▶ *Tivoli Distributed Monitoring (Advanced Edition): Resource Model*, SH19-4564
- ▶ eTrust Access Control Utilities Manual
- ▶ *IBM Secureway Directory Version 3.2.2 for Linux: Installation, Configuration, and Administration Guide ( sec. 9.2.1)*
- ▶ Tivoli Framework Users Guide (sec 9.5.2)
- ▶ *Tivoli Distributed Monitoring (Advanced Edition): Resource Model Reference. ( sec 9.7.1), (10.1.1, )*
- ▶ *Tivoli Distributed Monitoring (Advanced Edition) Users Guide 4.1, (10.1)*
- ▶ *Tivoli Secureway User Administration Supplement for Policy Director (10.3.3)*

## Referenced Web sites

These Web sites are also relevant as further information sources:

- ▶ Tripwire Open Source Software home page:  
<http://www.tripwire.com> (commercial version)  
<http://sourceforge.net/projects/tripwire/> (open source version 2)
- ▶ Moodss Open Source Software home page:  
<http://jfontain.free.fr/>
- ▶ Big Brother Open Source Software home page:  
<http://www.bb4.com>
- ▶ Amanda Open Source Software home page:  
<http://sourceforge.net/projects/amanda/>
- ▶ OpenLDAP Open Source Software home page:  
<http://www.openldap.org/>
- ▶ Device Drivers and Installation Commands Linux for zSeries manual:  
<http://www10.software.ibm.com/developerworks/opensource/linux390/docu/lzsdd08.pdf>
- ▶ LVM How to webpages :  
[http://www.sistina.com/products\\_lvm.htm](http://www.sistina.com/products_lvm.htm)  
[http://www.sistina.com/lvm\\_howtos](http://www.sistina.com/lvm_howtos)  
<http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/LVM-HOWTO.pdf>
- ▶ RAID Howto webpage:  
<http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/Software-RAID-HOWTO.pdf>
- ▶ SIS Open Source Software home page:  
<http://www.sisuite.org>
- ▶ VM Solutions for system performance:  
<http://www.vm.ibm.com/perf/perfprod.html>
- ▶ LDAP How to webpages:  
<http://www.skills-1st.co.uk/papers/security-with-ldap-jan-2002/security-with-ldap.html>  
[http://staff.pisoftware.com/bmarshal/publications/system\\_auth/sage-au/system\\_auth.html](http://staff.pisoftware.com/bmarshal/publications/system_auth/sage-au/system_auth.html)  
<http://www.tldp.org/HOWTO/LDAP-Implementation-HOWTO/index.html>

## How to get IBM Redbooks

You can order hardcopy Redbooks, as well as view, download, or search for Redbooks at the following Web site:

[ibm.com/redbooks](http://ibm.com/redbooks)

You can also download additional materials (code samples or diskette/CD-ROM images) from that site.

## IBM Redbooks collections

Redbooks are also available on CD-ROMs. Click the CD-ROMs button on the Redbooks Web site for information about all the CD-ROMs offered, as well as updates and formats.

# Index

## A

- access control
  - Computer Associates 366, 422
- Access Manager for e-business login prompt 242
- ACL 132, 136, 146
- Activating the LDAP Servers 146
- Advantage
  - Advantage CA-XCOM Configuration 411
  - Advantage CA-XCOM Data Transport 377, 410
  - Advantage Data Transport Agent 379, 411
  - Advantage Data Transport Installation 379
  - Advantage Ingres 374, 408
  - Advantage Ingres Enterprise Relational Database 374
  - Advantage Ingres Enterprise Relational Database Installation 374
  - Data management Tool from CA 353
  - Using Interfaces to Advantage Ingres 408
  - XCOM Data Transport 377, 410
  - XCOM Installation 378
- Advantage Data Transport Agent 379
- Amanda 113
  - AMANDA (Advanced Maryland Automatic Network Disk Archiver) 113
  - amanda.conf parameters 117
  - amcheck command 120
  - Backup process 114
  - Client Setup 120
  - Client System recovery 124
  - Configuration details 115
  - Data Management 114
  - Data Recovery 121
  - dump header 127
  - GNUTAR backup option 114
  - Limitations 114
  - Normal operations 121
  - Recovery of individual files or directories 121
  - Server Recovery 126
- Amanda Normal operations 121
- anaconda 10
- Apache
  - Apache Agent Configuration 394
  - Unicenter Apache agent 360, 390
- Apache Agent Configuration 394
- Apache Agent installation 361
- Apache Configuration File 361
- audit
  - routing information from Linux 372
- Automating File System Cleanup 323
- Automation 308, 402
- Automation Unicenter Software 402
- Automation with MAINVIEW 308
- Availability Management 318, 337
  - Using CA Unicenter 390

- Availability Management with Tivoli Distributed Monitoring (Advanced Edition) 208

## B

- backup
  - Computer Associates 373, 407
  - Computer Associates and VM 383, 437
- Backup process using Amanda 114
- BMC Product Authorization Primary Menu 255
- BrightStor
  - BrighStor Enterprise Backup
    - Status 407
  - BrightStor Enterprise Backup 373, 407
    - Installation 373
    - Startup 407
  - BrightStor VM
    - Backup 383, 437
    - Tape 383, 439
  - Computer Associates 353
  - Enterprise Backup 407
  - Storage for Linux for zSeries and S/390 Overview 353
- BrightStor Enterprise Backup 373
- BrightStor VM
  - Backup 383
  - Backup Overview 437
  - Tape 383
  - Tape Overview 439

## C

- Certification Authority (CA) 140
- Changing passwords 144
- CleverPath
  - Portal 446
  - Portal Solutions for Linux for zSeries and S/390 354
- CleverPath Portal 386, 446
- Cloning Systems 15
  - DDR command 15
  - Gold Image 15
  - silo 16
- CMS boot 8
- CMS Command 8
- Computer Associates 351
  - Advantage Data Transport 379, 411
  - Advantage Ingres 374, 408
  - Advantage XCOM Data Transport 377, 410
  - Apache agent 360, 390
  - Binary Distribution Installation 371
  - BrightStor 353
  - BrightStor Enterprise Backup 373, 407
  - BrightStor VM:Backup 383, 437
  - BrightStor VM:Tape 383, 439
  - CleverPath 354
  - CleverPath Portal 386, 446

- CleverPath Portal Installation 387
- eTrust 352
  - Access Control 366, 422
  - Admin 371, 417
  - Audit 372, 427
  - CA-ACF2 Security 369, 419
  - CA-Top Secret Security 370, 419, 421
  - Directory 368, 414
- eTrust VM:Director 384, 442
- job management 363
- list of Linux solutions 352
- Operations 390
- Unicenter CA-Explore 384
- Unicenter NSM 354, 399
- Unicenter Software Delivery 361
- Unicenter VM:
  - Account 382, 430
  - Operator 382, 432
  - Schedule 383, 433
  - Spool 383, 435
- VM:Manager VM Management Suite 380, 429
- web address 352
- z/VM solutions 353
- Configuration details 115
- Configuring Basic Settings 279
- Configuring eTrust CA-Top Secret for Linux PAM 422
- Configuring Name Service Switch 137
- Configuring Sendmail Servers 337
- Configuring the TSM client 158
- CP FORMAT utility 6
- CPU Configuration 30
- CPU configuration 30
- Creating a Distributed Monitoring Profile 208
- Creating users 142
- crontab 121
- CTC 21
- CTC (Channel-to-Channel) 15
  - Real CTCs 18
- CTC links 10
- Customization 254

## D

- DASD
  - 3390 Model 3 8
  - 3390-3s 37
  - DASD Type 8
  - dasdfmt 124
  - DISKMAP 9
  - ECKD DASD 37
  - Format and label 6
  - Format and label DASD 6
- Data Collection 444
- data management
  - Computer Associates 407
- data recovery
  - Computer Associates 373, 407
- data transport
  - Computer Associates 377, 379, 410–411
- database management
  - Computer Associates 374, 408

- DHCP 149
- directory management
  - Computer Associates and VM 384, 442
  - eTrust Directory 368, 414
- DIRMAINT 9
- Disk Format 9
- Disk setup panel 39
- Disk Space management 443
- Distributed Data Server (DDS) 56
- Distributing the DM Profile 213
- DMZ 3–5
- DNS 271

## E

- Edit Recovery Action Dialog 324
- EMVS
  - Enterprise Volume Management System (EVMS) 43
    - How to get and install 43
    - How to use 48
    - Terminology 44
- Enabling DM Heart Beat 214
- Enabling TLS 141
- Enforcing TLS 141
- environment variables for LDAP DB2 instance 161
- eTrust
  - Access Control 422
    - Install Command 367
  - Access Control overview 422
  - Access Control seam X Windows GUI. 424
  - Admin 371, 417
  - Admin for eTrust Access Control GUI. 425
  - Audit 372, 427
  - Audit information 372
  - Auditing Actions of Specific Individuals 424
  - CA-ACF2 for Linux PAM configuration 420
  - CA-ACF2 Security 369, 419
  - CA-Top Secret for Linux PAM 422
  - CA-Top Secret Security 370, 420–421
  - Centralize Security Management 424
  - Directory 368, 414
  - eTrust Access Control 366, 372, 422
    - installation 366
    - installation messages 367
    - invocation 367
    - start-up 368
  - eTrust VM:Director 384, 442
  - Password and Account Policy 427
  - Protecting Sensitive Accounts 426
  - Security for Linux for zSeries and S/390 352
  - security solutions 352
- eTrust Access Control 366, 422
- eTrust Access Control Password and Account Policy 427
- eTrust Admin 417
  - Manager Interface. 418
  - Native Linux 390 Environment usage 372
  - Overview 371
  - Role-based User Administration 417
  - Web Interface. 419
- eTrust Audit 427
  - and eTrust Access Control 427

- Overview 372
- eTrust Audit Overview 427
- eTrust Audit Policy Manager usage 428
- eTrust CA-ACF2 Security 420
- eTrust CA-ACF2 Security (Interface to Linux for zSeries and S/390) 369, 420
- eTrust CA-Top Secret Security 421
- eTrust CA-Top Secret Security (Interface to Linux for zSeries) 421
- eTrust CA-Top Secret Security (Interface to Linux for zSeries and S/390) 370, 421
- eTrust Directory 368, 414–415
  - Common name search 416
  - Installation 369
  - Installation Screen 369
  - Search results 417
  - Startup window using JXplorer browser. 415
  - User Definition and Administration 414
- eTrust Directory connection dialog. 416
- eTrust Directory Installation 369
- eTrust Directory Overview 414
- eTrust VM
  - Director 384
  - Director Overview 441
- Event Management 399
- EVMS 37, 43–45, 47, 53
  - EVMS back-end 43
  - EVMS modules 43
  - EVMS S/390 segment manager 45
  - evms\_devnode\_fixup 53
  - evms\_gather\_info 53
  - evmsgui 43
  - interface options 43
  - kernel 53
  - utilities 43

## F

- FCON/ESA 34, 74
  - FCON/ESA V.3.2.03 34
- File Backup and Restore
  - using BrightStor 407
- File backup and restore using Tivoli Storage Manager 228
- File/Print servers 5
- Firewall/router systems 5
- Firewalls 3, 240
- Format the A-DISK 9
- FTP 257

## G

- Guest LAN 18, 25, 29

## H

- Health Monitoring 399
  - Unicenter 399
- Health monitoring 293
- Health Monitoring for PATROL 321
- Health Monitoring for PATROL for Linux Enterprise Serv-

- er 321
- Heartbeat Monitoring 293
- Heartbeat Monitoring with MAINVIEW 293
- HiperSocket 29
- HiperSockets 26, 28
- HiperSockets differences 16
- HiperSockets Guest LAN 26
- Home directories 143
- HSI (HiperSockets) 18

## I

- IBM Directory Server 160
  - Configuration 162
  - Installation 161
  - Management Tool (DMT) Add Attribute 165
  - shadowAccount object class 168
- IBM Tivoli Access Manager for e-business 169
- IBM Tivoli Access Manager for e-business layout 169
- IBM Tivoli Enterprise Console 201
  - Component Layout 202
- IBM Tivoli Enterprise Console component layout 202
- IBM Tivoli Identity Manager Create LDAP Connection Object 237
- IBM Tivoli Identity Manager Profile Selection 236
- IBM Tivoli Identity Manager regular user login page 235
- Identity Manager managing files 236
- Identity Manager managing LDAP 237
- indicate command 31
- indicate load 32
- Ingres JDBC Driver 409
- Ingres/ICE 410
- Ingres/OpenAPI 409
- Installation dialog 226
- Installation Notes 191
- Installation of a Tivoli Endpoint Gateway 185
- Installation of other Tivoli Framework Products 183
- Installation of the Tivoli Endpoints 186
- Installation Utility
  - Select Products and Components to Install Dialog 275
- Installing the TSM client 158
- Integrated Facility for Linux
  - VM and Computer Associates 380
- Interfaces to Advantage Ingres 408
- Inter-User Communications Vehicle (IUCV) 18
- IPL 00C CLEAR 12
- ISO Images 14
- ISO9660 14

## J

- J2EE and Servlet Considerations 386
- Java 1.3 57
- Java client 75
- Java Runtime Environment (JRE) 198
- job management
  - Computer Associates 363, 406
- JRE 198–200, 211, 213
- JXplorer
  - and eTrust Directory 415

## K

Kernel  
2.2.16 263  
Kernel Build 13

## L

Labelling the tapes 120  
LDAP 76, 160, 163, 170, 233, 236  
    Connection object 241  
    eTrust Directory 414  
LDAP (Lightweight Directory Access Protocol) 74  
ldap.conf for OpenLDAP 132  
LDIF 133  
LINIPL 12  
Linux 9  
    Boot Linux using the z/VM READER 12  
    CAICCI configuration on Linux S/390 365  
    Configuration 19, 23, 26, 28  
    Creating LINIPL EXEC 12  
    CTC definition 23  
    Debugging 24  
    IPL Linux 9  
    kernel configuration menu 14  
    Linux drivers 19  
    Linux Kernel 2.2 and 2.4 version differences 58  
    Linux z/VM guest 9  
    Linux z/VM guest first time log on 9  
    Linux/390 3  
    Setting up for Linux 10  
Linux kernel configuration menu 14  
List Users Without Passwords Dialog 336  
LNx12 CTC definitions 15  
Local Configure Dialog 338  
Log File Dialog 325  
Log File Dialog screen 326  
Logical Volume Manager (LVM) 37  
LVM 43, 49  
    Region Manager 48

## M

MAINVIEW 252  
    Automation 308  
    Availability Management 291  
    Configuration/Layout 252  
    Customization 254  
    Health monitoring 293, 308  
    Heartbeat monitoring 293  
    Installation 253  
    MAINVIEW for Linux – Servers 252  
    Operations for MAINVIEW for Linux – Servers 284  
MAINVIEW for Linux – Servers customization - product authorization step 254  
MAINVIEW for Linux – Servers Easy Menu 285  
MAINVIEW Selection Menu 284  
Managing Shared File Systems 408  
MDISK 8–9  
Memory configuration 29  
Memory resource 60  
message management

    Computer Associates on VM 382, 432  
Monitoring BIND DNS Servers 340  
Monitoring Logs 329  
Monitoring Remote Hubs 339  
Moodss 96  
    BLT library 98  
    Build RPM-packages 98  
    graphical example. 107  
    Initial screen 100  
    Installation 96, 98–99  
    memstats 102  
    minimal 103  
    myhealth 104  
    myvars 104  
    ping 104  
    Poll time 110  
    snmp 106  
    Source Code 98  
    Startup 99  
    Threshold 108  
        window 109  
    Tktable 96  
    Usage 99

## N

Network availability 293  
Network availability with MAINVIEW 293  
Network configuration 18  
Network Management 354  
NFS 114, 126, 130, 143, 178, 323  
    NFS mounted home directories 143  
NSM 356  
    DB2 Agent 358  
    Informix Database Agent 360  
    Ingres Agent 358  
    Log Agent 359  
    Process Agent 358

## O

OCO (Object Code Only) 13  
Open Systems Adapter (OSA) 18  
OpenLDAP 129, 162, 170  
    Activating the servers 146  
    Adding an initial user 133  
    Adding LDAP base entries 133  
    Changing passwords 144  
    Common Name 140  
    Configuration 131  
    Configuring the master 144  
    Configuring the slave 145  
    Copying the data from master to slave 145  
    Creating an Administrator 134  
    LDAP ACL 141  
    LDAP directory 130  
    Listing entries in the directory 135  
    Managing UID numbers 143  
    OpenLDAP and pam\_ldap user management 130  
    Overview 130  
    Preparing for replication 144



- Server Replication using slurpd 144
- OpenLDAP problem diagnosis 136
- OpenLDAP User administration 142
- Operating System Support 360
- Operations for PATROL for Linux Enterprise Server 318
- Operations for PATROL Internet Server Manager 337
- OS/390 and Linux/390 components 253
- OSA Connections 18
- OSA connections 18
- OSA-Express 19
- OSA-Express adapter 19

## P

- Package Installation commands and attributes 224
- PAM 130, 137, 139, 370
  - Configuration 137
  - Configuring the PAM ESM Component 419
  - Introduction 137
  - PAM and pam\_ldap overview 130
  - PAM API 130
  - PAM Support for Computer Associates External Security Managers 370
  - Pam\_ldap 131, 139
  - pam\_pwcheck 139
  - pam\_unix 139
  - Pluggable Authentication Modules
    - eTrust CA-ACF2 Security 370
  - Support for Computer Associates External Security Managers 419
- PAM and pam\_ldap Overview 130
- PAM Configuration 137
- PAM Support 419
- pam\_ldap 130
- Password and Account Policy 427
- PATROL 262
  - Automation 323
  - Availability Management 318, 337
  - Configuration/Layout 263
  - Customization required to use PATROL Agent for Linux 271
  - Data Management 329
  - Enabling Basic Log Monitoring 278
  - File System Configure Dialog 280
  - Health Monitoring 321, 347
  - Installation 263, 277
  - Installation Console for Microsoft Windows 267
  - Installation for Linux Enterprise Server on Linux 265
  - Installation for Microsoft Windows 276
  - Installation Utility 266–269
  - Installing Knowledge Module for Unix on Windows 270
  - Installing PATROL Internet Server Manager on Linux 274
  - Local Configure Dialog 281
  - Monitoring
    - BIND DNS Servers 340
    - Files 332
    - Logs 329
    - Remote Hubs 339
    - Remote Internet Servers 345

- Operations 337
- PATROL for Linux Enterprise Server 262
- PATROL Internet Server Manager 271
- Security 331
- Security for PATROL Internet Server Manager 347
- Selecting Log Files to Monitor 329
- SSL Certificate Monitoring 347
- User Activity 334
- Viewing Users with Multiple Sessions Running 336
- Viewing Users without Passwords 335
- PATROL Classic Console
  - PATROL for Linux Enterprise Server 263
  - PATROL Internet Server Manager 272
- PATROL for Linux Enterprise Server 262
- PATROL Internet Server Manager 271
- Penguin Farm 3, 5
- penguin farm
  - Tips 29
- Performance Agents 357
- Performance Configuration 397
- performance management 356
  - Computer Associates and VM 384, 444
  - Unicenter NSM Performance Management 394
- Performance Management for VM 444
- Pluggable Authentication Modules
  - eTrust CA-ACF2 Security 370, 419–420
  - eTrust CA-Top Secret Security 370, 419, 421–422
- Policy Server 171
- portal management
  - CleverPath 386, 446
  - Overview 446
- posixAccount and posixGroup object classes 168
- Preparing to Distributed the DM Profile Manager 211
- Protecting Sensitive Accounts using eTrust Access Control 426

## R

- RACHECK 285
- RAID 38, 42, 49
  - Level 0 (striping) 38
  - Level 1 (mirroring) 38
  - Level 4
    - uses large stripes 38
  - Level 5 38
  - mkraid command 42
  - nr-raid-disks 41
  - RAID 4 38
  - RAID array 49
  - RAID device 42
  - RAID levels
    - 0, 1, 4, 5 38
  - RAID support 39
  - RAID-0 39
  - RAID-1 39
  - raiddev 41
  - RAID-tools
    - Setting up Level 0 RAID using Red Hat installation tool 38
    - Setting up Level 1 RAID manually 41
  - Redundant Array of Independent Disks (RAID) 38

- tools 37
- Raid setup panel 40
- RAID-tools 38
- RAMdisk 39
- Recovering a client system with Amanda 124
- Recovering data 121
- Recovering the server using Amanda 126
- Recovery Action Instances Dialog 324
- Red Hat 2.4 64
- Red Hat 7.2 5
- Redbooks Web site 452
  - Contact us xiii
- RedHat 7.2 39
- Registered Recovery Actions Dialog 324
- Resource Management 18
- RMF LDAP interface 76
- RMF PMS 55
  - Configurations files 58
  - CPU resource 60
  - File system resource 60
  - HTTP/XML interface in an RMF PMS 71
  - Logging 59
  - Metrics 60
  - Network resource 60
  - Overview 56
  - Planning 57
  - RMF mix environment 74
  - System resource 60
  - Testing 64
- RMF PMS logging 59
- RMF PMS Planning 57
- RPM 96
- RTM VM/ESA 32

## S

- S/390 37, 57, 98
  - S/390 architecture 106
  - S/390 patches 43
- scheduling
  - Computer Associates and VM 383, 433
- Secure Socket Layer (SSL) 271
- Securing Environment with eTrust Access Control 423
- Securing the connection and verifying the server 140
- security
  - Computer Associates 369–372, 417, 419–422, 427
- Security for PATROL for Linux Enterprise Server 331
- Security for PATROL Internet Server Manager 347
- Security using eTrust Directory 414
- Security using IBM Tivoli Identity Manager 233
- Sendmail Servers 337
- SFS Support 442
- Shared File Systems Management 408
- SIS Architecture Independence 148
- SIS usage 149
- Slapd 136
- slapd.conf for OpenLDAP 131
- SNMP (Simple Network Management Protocol) 106
- software delivery
  - Unicenter Software Delivery 361, 402
- Software Deployment with Tivoli Software Distribution

- 218
- Software Package Editor
  - Building a Package 220
- Software package editor - package properties 221
- spool management
  - Computer Associates and VM 383, 435
- SQL/DS Data Collection 445
- SQL/DS data collection
  - starting for Unicenter CA-Explore 445
- SSL 140, 170
  - Certificate Monitoring 347
  - Generate keys and certificates 140
- Starting BrightStor Enterprise Backup 407
- Starting the TSM client 159
- SuSE 2.2 64
- SuSE 2.4 64
- SuSE 7.0 5, 13
- SuSE 7.0SLES 5
- SuSEconfig utility 16
- system audit
  - eTrust Audit 372, 427
- System Installation Suite Diagram 150
- System Installation Suite(SIS) 147, 150
  - Architecture Independent 148
  - Introduction 148
  - Overview 148–149
  - Source code 151
- System Layout Diagram 4

## T

- tape management
  - Computer Associates and VM 383, 439
- TCP/IP 6, 34, 164, 195, 240
  - port 239
- TCPIP
  - address 255
- TCPMAINT 11
- Tivoli Access Management
  - User Management 241
  - Web Server Management 242
- Tivoli Access Manager 169
  - Access Control 239
  - Installation 170
  - Managing for Audit 244
  - Policy Server 171
  - Product Layout 169
- Tivoli Distributed Monitoring (Advanced Edition) 194–195
  - Availability Management 208
  - Distributing the DM Profile 213
  - Enabling DM Heart Beat 214
  - Health Console 200
  - Health Monitoring 216
  - Installation 197
- Tivoli Distributed Monitoring (Advanced Edition)Distributing the DM Profile 213
- Tivoli Distributed Monitoring Profile Manager. 211
- Tivoli Distribution Status Monitor 227
- Tivoli Endpoints Installation 186
- Tivoli Identity Manager 174

- Component Layout 175
- Component Layout Diagram 175
- Managing files 236
- Managing LDAP 237
- Security 233
- User definition and administration 233
- Tivoli Managed Environment 188
- Tivoli Management Framework 177
  - Installation 178
- Tivoli Management Infrastructure 156
- Tivoli Management Infrastructure Diagram 156
- Tivoli Software Distribution 188
  - Installing the Software Package Editor 219
- Tivoli Software Distribution Installation 190
- Tivoli Storage Manager 157
- TLS 141, 146, 170
  - tls\_ssf parameter 141
  - TLSCertificateFile 145
- TLS enablement 141
- Token Ring OSA card 19
- Transport Layer Security (TLS) 140
- Tripwire 81
  - Creating initial configuration file 83
  - Database Creation 84
  - Database update example 85
  - Generation 83
  - how it works 82
  - Integrity check 85
  - Overview 82
  - Planning 83
  - Usage 83
- Tripwire Tips 93
- Tripwire, how to test 86
- TSM 157
  - Client command line dsmc 229
  - File backup and restore 228
  - Install and Configure TSM Web Client 160
  - Web Client Interface 231
  - Web GUI restore complete 232
- TSM Backup/Archive Client 158
- TSM server web administration 160
- TSM Web Client 231
- TSM Web Client Interface 231

## U

- Unicenter
  - Apache Web Server Agent Summary screen. 390
  - CA-Explore Perf Management for VM 384, 444
  - Computer Associates 352
  - Configuring CPU defaults via Agent View 401
  - Event Management Console Log 402
  - FTP Installation Steps 364
  - Network and Systems Management Usage 400
  - NSM
    - Enterprise Management View 401
    - Network Topology View 400
    - Performance Management 394
  - Performance Configuration 397
  - Performance Configuration initial view 398
  - Performance Scope
    - historical and real-time data 395
  - Performance Trend report 396
  - Server Health and threshold settings 391
  - Software Delivery Agent 402
  - Unicenter CA-Explore 384, 444
  - Unicenter CA-Explore Perf Management for VM 385
  - Unicenter Chargeback 398
  - Unicenter Management for Web Servers 360, 390
  - Unicenter Network & Systems Management 354, 399
  - Unicenter NSM agents 356, 394
  - Unicenter NSM Performance Management 356, 394
  - Unicenter Software Delivery 361, 402
  - Universal Job Management Agent 363, 406
    - Installation 363
    - WebCrawler display of bad links and references. 393
- Unicenter Apache agent
  - installing 360
  - using 390
- Unicenter CA-Explore Performance Management for VM 444
- Unicenter Management for Web Servers (Apache Agent) 360, 390
- Unicenter Network and Systems Management 354, 399
- Unicenter NSM Performance Management 356
- Unicenter Software Delivery Integration 404
- Unicenter Universal Job Management Agent 363, 406
- Unicenter VM
  - Account 382, 430
  - Operator 382, 432
  - Operator Overview 432
  - Schedule 383
  - Schedule Overview 433
  - Spool 383
  - Spool Overview 435
- user account management
  - eTrust Admin 371, 417
- User Activity 334
- User administration 142
- User Definitions 6
- User Management 241
- User Management with OpenLDAP 130
- Using eTrust Audit Policy Manager 428

## V

- VCTC 18, 23
- VHSI (Virtual HiperSockets) 18
- Viewing Users with Multiple Sessions via PATROL 336
- Viewing Users without Passwords using PATROL 335
- Virtual Channel-to-Channel (VCTC) 20
- Virtual Channel-to-Channel connections 20
- VM
  - Account 382, 430
  - Backup 126, 383, 437
  - DEFINE STORAGE 8
  - Director 384, 441
  - ESA mode 8
  - Manager VM Management for Mainframe Linux 380, 429
    - Installation 380
  - Operator 382, 432

- PROFILE EXEC 21
- Schedule 383, 433
- Spool 383, 435
- Tape 383, 439
- VM management using VM Manager 429
- VM management
  - Computer Associates 380
- VM Management for Mainframe Linux 429
- VM resource accounting
  - Computer Associates 382, 430

## W

- Web server
  - management 242
  - Unicenter Apache agent 360, 390
- WebCrawler 392
- WebSEAL functionality 241
- WebSEAL Server 173
- Windows 2000 57
- Windows 2000 Installation Account 264, 274
- Windows 9x 57
- Windows NT 57

## X

- XML 57–58, 71

## Y

- YaST 10

## Z

- z/OS 56, 75
- z/OS RMF PM Java Client 75
- z/VM 5
  - configuration 5, 18, 20, 25, 28
  - guest 7
    - A disk 9
    - LAN (Emulated HiperSockets) 25
  - Linux z/VM guests 8
  - Solutions for Linux for zSeries and S/390 353
  - Upload Linux IPL material to z/VM 10
  - User definitions 6
  - z/VM Guest LAN (Emulated HiperSockets) 25
  - z/VM Reader 12
- z/VM configuration 5, 20, 25, 28
- z/VM guest 5–6, 8
- z/VM v4.30 5
- z900 4
- zSeries 37, 81



Redbooks

# Linux on IBM @server ZSeries and S/390: System Management

(1.0" spine)

0.875" x 1.498"

460 <-> 788 pages








# Linux on IBM server zSeries and S/390: System Management



**Linux system management using solutions from Tivoli, BMC, and Computer Associates**

**Open source software for Linux system management**

**Security, networking, capacity planning, operations, and backup and recovery tools**

This IBM Redbook describes system management for Linux on IBM server zSeries and S/390. It will help you understand the primary management processes for a set of Linux images on zSeries and S/390, and how to identify appropriate tools for your environment.

First we describe the setup of an S/390 environment with multiple Linux images. We discuss how to install and configure several of the most commonly used Linux distributions and the system management tools that are included with those distributions.

Next we discuss the installation, configuration, and use of open source software for both general and special purpose system management, highlighting the offerings from Tripwire, Moodss, Amanda, OpenLDAP, and System Installation Suite.

We then discuss the proprietary Linux system management solutions provided by IBM Tivoli, BMC, and Computer Associates. For each company's offerings, we present an overview of the products, details for installing and configuring them, and information on how to use the solutions to manage your Linux systems.

Many additional tools are available for Linux system management, both from open source projects and from commercial vendors. Because of the rapid growth in this area, it is impossible to provide comprehensive coverage of all the available products. However, the overview of system management issues for Linux presented in this book, along with details for a broad cross-section of products in the marketplace, is useful even for those considering solutions not specifically covered here.

**INTERNATIONAL  
TECHNICAL  
SUPPORT  
ORGANIZATION**

**BUILDING TECHNICAL  
INFORMATION BASED ON  
PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:  
[ibm.com/redbooks](http://ibm.com/redbooks)**

SG24-6820-00

ISBN 0738426105