XAddHost, XAddHosts, XListHosts, XRemoveHost, XRemoveHosts, XSetAccessControl, XEnableAccessControl, XDisableAccessControl, XHostAddress – control host access and host control structure

XAddHost(display, host)
    Display *display;
    XHostAddress *host;
XAddHosts(display, hosts, num_hosts)
    Display *display;
    XHostAddress *hosts;
    int num_hosts;
XHostAddress *XListHosts(display, nhosts_return, state_return)
    Display *display;
    int *nhosts_return;
    Bool *state_return;
XRemoveHost(display, host)
    Display *display;
    XHostAddress *host;
XRemoveHosts(display, hosts, num_hosts)
    Display *display;
    XHostAddress *hosts;
    int num_hosts;
XSetAccessControl(display, mode)
    Display *display;
    int mode;
XEnableAccessControl(display)
    Display *display;
XDisableAccessControl(display)
    Display *display;

display           Specifies the connection to the X server.
host              Specifies the host that is to be added or removed.
hosts             Specifies each host that is to be added or removed.
mode              Specifies the mode. You can pass EnableAccess or DisableAccess.
nhosts_return     Returns the number of hosts currently in the access control list.
num_hosts         Specifies the number of hosts.
state_return      Returns the state of the access control.

The XAddHost function adds the specified host to the access control list for that display. The server must be on the same host as the client issuing the command, or a BadAccess error results.

XAddHost can generate BadAccess and BadValue errors.

The XAddHosts function adds each specified host to the access control list for that display. The server must be on the same host as the client issuing the command, or a BadAccess error results.

XAddHosts can generate BadAccess and BadValue errors.

The XListHosts function returns the current access control list as well as whether the use of the list at connection setup was enabled or disabled. XListHosts allows a program to find out what machines can make connections. It also returns a pointer to a list of host structures that were allocated by the function. When no longer needed, this memory should be freed by calling XFree.

The XRemoveHost function removes the specified host from the access control list for that display. The server must be on the same host as the client process, or a BadAccess error results. If you remove your
machine from the access list, you can no longer connect to that server, and this operation cannot be reversed unless you reset the server.

XRemoveHost can generate BadAccess and BadValue errors.

The XRemoveHosts function removes each specified host from the access control list for that display. The X server must be on the same host as the client process, or a BadAccess error results. If you remove your machine from the access list, you can no longer connect to that server, and this operation cannot be reversed unless you reset the server.

XRemoveHosts can generate BadAccess and BadValue errors.

The XSetAccessControl function either enables or disables the use of the access control list at each connection setup.

XSetAccessControl can generate BadAccess and BadValue errors.

The XEnableAccessControl function enables the use of the access control list at each connection setup.

XEnableAccessControl can generate a BadAccess error.

The XDDisableAccessControl function disables the use of the access control list at each connection setup.

XDisableAccessControl can generate a BadAccess error.

The XHostAddress structure contains:

typedef struct {
    int family; /* for example FamilyInternet */
    int length; /* length of address, in bytes */
    char *address; /* pointer to where to find the address */
} XHostAddress;

The family member specifies which protocol address family to use (for example, TCP/IP or DECnet) and can be FamilyInternet, FamilyDECnet, or FamilyChaos. The length member specifies the length of the address in bytes. The address member specifies a pointer to the address.

BadAccess A client attempted to modify the access control list from other than the local (or otherwise authorized) host. BadValue Some numeric value falls outside the range of values accepted by the request. Unless a specific range is specified for an argument, the full range defined by the argument’s type is accepted. Any argument defined as a set of alternatives can generate this error.

XFree(3X11)

Xlib – C Language X Interface