AUTHORIZED DOCUMENTATION

Administration Guide

Novell®
Privileged User Manager

2.2

April 9, 2009

www.novell.com

Legal Notices

Novell, Inc., makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc., makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the Novell International Trade Services Web page (http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Novell, Inc., has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed on the Novell Legal Patents Web page (http://www.novell.com/company/legal/patents/) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the Novell Documentation Web page (http://www.novell.com/documentation).

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide

This Administration Guide explains how to use the Framework Manager to control and audit superuser access to Linux and UNIX machines.

Audience

This guide is intended for users who manage the Privileged User Manager product.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

For the most recent version of the *Administration Guide*, visit the Privileged User Manager Web site (http://www.novell.com/documentation/privilegedusermanager22).

Additional Documentation

Privileged User Manager Getting Started Guide (http://www.novell.com/documentation/privilegedusermanager22/npum_install/data/index.html)

Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as Linux or UNIX, should use forward slashes as required by your software.

Welcome to the Framework

1

Welcome to the Novell Privileged User Manager, a complete secure framework to enable immediate central distribution, update and management of the tools and services required to enable a safe and secure operating environment for today's and tomorrow's IT infrastructure.

Privileged User Manager provides complete control of restricted commands on UNIX/Linux platforms, plus fully configurable auditing.

1.1 Introduction to the Framework

Novell Privileged User Manager uses a Framework as its the base layer which provides an easy to use enterprise architecture into which Privileged User Manager modules are added, giving the problem solving functionality required. The key features of the Framework are listed below:

- The Framework provides the core functionality needed to implement secure, enterprise wide services.
- The Framework presents services such as secure and authenticated communications between the components.
- The Framework provides integrated databases and logging.
- The Framework allows the deployment of Privileged User Manager Modules onto Framework Hosts to implement new functionality.
- With each module that is installed, an additional console is added to the main Framework Console to allow access to new administration functionality.

The Framework is made up of three primary components, which are the Framework Manager, the Framework Console and the Framework Agent.

1.1.1 Framework Manager

The Framework Manager is the server component of the Framework. It provides a centralized registry enabling services and administration of the entire Framework from any single point on the enterprise network.

The Framework Manager is administered through the Framework Console, using a suitable Web Browser with the Macromedia Flash plug-in.

Manager Modules

The manager modules are installed on the Framework Manager by default. The modules can also be distributed to other Framework hosts to provide load balancing and fail-over for the Framework. If there are multiple occurrences of the same type of manager installed onto the Framework, these will operate in Primary/Backup roles. Updates to the data controlled by each group of like managers will only be updated at the primary manager.

The default manager modules are

- Registry Manager (registry) - Maintains a database of all Framework hosts and modules. Provides certificate based registration features for the hosts.
- Package Manager (pkgman) - Manages a repository for Framework Packages.
- Administration Agent (admin) - Provides the functionality for the web based user interface. Framework consoles can be installed onto the Administration Agent and used to control product features.
- Access Manager (auth) - Maintains a list of Framework user accounts. Provides authentication services for the Framework. Note: needs to be installed with a local Registry Manager in order to create a secure user authentication token.

1.1.2 Framework Console

The Framework Console is the default user interface for the Framework. It allows configuration and management of the Framework through the intuitive graphical user interface.

1.1.3 Framework Agent

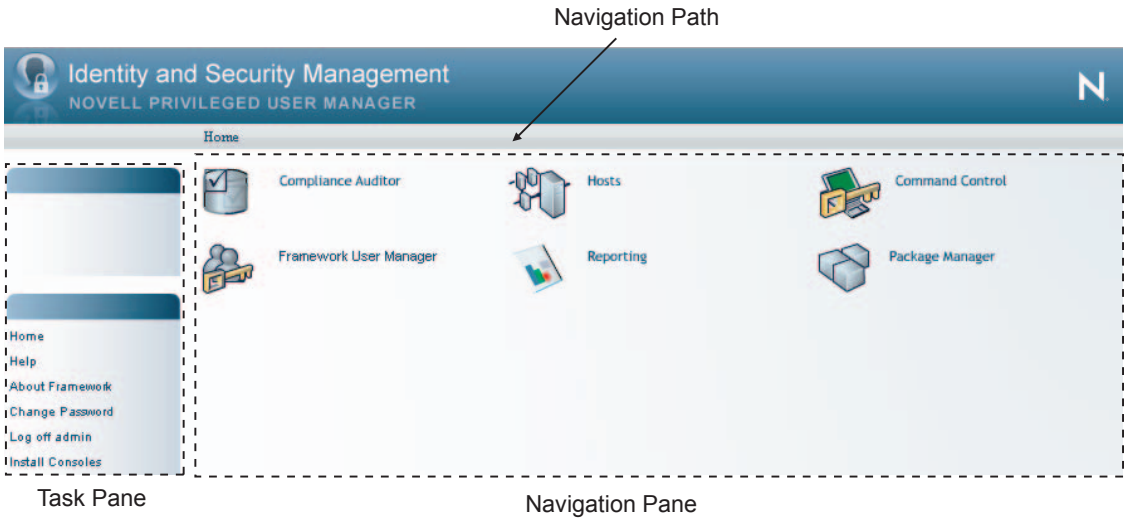
The Framework Agent is the client component of the Framework. It is installed onto all client computers on the enterprise network and is responsible for receiving and carrying out instructions from the Framework Manager on all hosts.

The Framework Agent packages are installed on all Framework hosts. The default agent modules are:

- Registry Agent (regclnt) - Provides a local cached lookup for module locations. The Registry Agent queries the Registry Manager when local cached information is not available or isn't fresh.
- Distribution Agent (distrib) - Provides the interface to control the installation and removal of the packages in the Framework. It has methods to install/remove and list available/ updatable packages. The Distribution Agent retrieves packages from the local Package Managers.
- Store and Forward Agent (strfwd) - Provides a store and forward mechanism for guaranteed delivery of messages. This is used for various core features such as replication of the manager databases.

1.2 The Workspace Layout

The Framework Console consists of three areas:



- Section 1.2.1, “Navigation Path,” on page 13
- Section 1.2.2, “Task Pane,” on page 13
- Section 1.2.3, “Navigation Pane,” on page 14

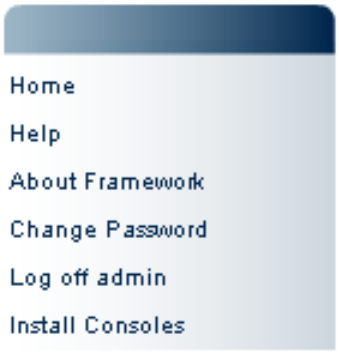
1.2.1 Navigation Path

The *Navigation Path* towards the top centre of the screen shows the current position in the Framework Console.

NOTE: Please note that individual items on the Navigation Path can be used for quick access to a given *Navigation Pane*.

1.2.2 Task Pane

The Task Pane on the left of the screen contains options that are applicable to the current Framework Console display.



1.2.3 Navigation Pane

The Navigation Pane on the right of the screen provides the current administrative facilities, consisting of icons data grids and forms.

1.3 The Privilege User Manager Modules

Each of the Privileged User Manager Modules is a product in its own right and the Framework allows easy deployment, configuration, management, integration and reporting from a central point. The current modules are:

The Command Control Module allows you to manage what, where, when and who may run UNIX privileged commands on any UNIX hosts registered with the Framework. This includes a fully integrated UNIX shell that performs complete session logging.

Setting Up the Framework

2

2.1 Managing Framework Hosts

The Hosts console provides a hierarchical view of all currently defined hosts in your Framework. Each host machine on which you have installed managers and agents must be added to the Framework using the Hosts console. Hosts are identified to the Framework by a unique agent name which is used to register the manager or agent after installation.

The Framework provides load balancing and failover capabilities, based on the hierarchical structure of your Framework. Before building your Framework, refer to Chapter 7, “Load Balancing and Failover,” on page 111 for information on how these features work.

From the Hosts console you can perform a number of tasks:

- Add and remove hosts in a domain structure.
- Modify the hosts to configure the host name and port number of the machines they are installed on.
- Update the packages installed on your hosts.
- View and configure the log on each host.
- Configure system status alerts.
- View the status of all hosts or all hosts in a domain.
- Find a specific host or package.
- Restart the agent processes.
- Promote backup managers to primary managers.
- Configure the HTTPS listening address/port.
- Install certificates.
- Change the location of the audit database files.
- View and delete store and forward messages.
- View and clear the registry cache used to enable communication between Framework components.

You should start by adding a domain to the framework, into which you can then add hosts.

2.1.1 Building a Framework

Managing Hosts

Viewing Host Details

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required domain from the *Navigation Pane*.
3. Select the arrow ▶ next to the domain icon to show the hosts on the left side of the *Navigation Pane*.
4. Select the host icon to display the host details and status.
5. Select the arrow ▶ next to the host icon to show the *Packages* icon.
6. Select *Packages* to view details of the packages installed on this host.

Modifying a Host

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required host from the *Navigation Pane*.
3. Select *Modify Host* from the *Task Pane.*
4. You can:
   - Change agent name in the *Agent name* field

     NOTE: The agent name does not need to relate to the existing DNS name used to locate the host on your network.
   - Add a description in the *Description* field which will be displayed next to the agent name in the Console
   - Change the host name in the *Host name* field

     NOTE: The host name must be either the existing DNS name used to locate the host on your network or the IP address for the host.
   - Change the port number in the *Port* field.
5. Select *Finish*.

Moving a Host around the Framework

You can move hosts between domains in the Framework.

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

The *Navigation Pane* displays the current hierarchy for your Framework.

2. Select the required domain from the *Navigation Pane*. The hosts in that domain are displayed on the right side of the *Navigation Pane.*
3. Select the required host.

TIP: To select multiple hosts, hold down the Ctrl key and select the required hosts one at a time, or hold down the Shift key to select a consecutive list of hosts. To select all hosts in a domain, use Ctrl+A.

---

4. Drag the required hosts to the new domain.

Deleting a Host from the Framework

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required domain from the *Navigation Pane*. The hosts in that domain are displayed on the right side of the *Navigation Pane.*
3. Select the required host.

TIP: To select multiple hosts, hold down the Ctrl key and select the required hosts one at a time, or hold down the Shift key to select a consecutive list of hosts. To select all hosts in a domain, use Ctrl+A.

---

4. Select *Delete Host* from the *Task Pane*. The selected hosts are listed.
5. Select *Finish*.

IMPORTANT: This action cannot be undone.

Finding a Host

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select *Hosts*.
3. Select *Find Host* from the *Task Pane*.
4. In the *Agent name* field, enter the name of the host you are looking for. You can use wildcard characters * and ?. For example, entering h* finds all hosts with agent names beginning with h. This field is case sensitive.
5. Select *Find*.
6. To go to a host's details, double-click its agent name. Alternatively, select *Close*.

Finding Packages on Hosts

You can search through all your hosts to find out which hosts a specific package is installed on, or not installed on.

To find a package:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.

2 Select *Hosts*.

3 Select *Find Package* from the *Task Pane*.

4 Select the package you are looking for from the *Package* drop-down list. If you want to find out where this package is not installed, select the *package not installed* checkbox.

5 Select *Find*. A list of agents where the package is or is not installed is displayed.

6 To go to a host's details, double-click the agent name. Alternatively, select *Close*.

Managing Domains

- "Creating a Domain" on page 18
- "Modifying a Domain" on page 18
- "Deleting a Domain from the Framework" on page 18

Creating a Domain

1 To add a new top-level domain, click *Hosts* from the Navigation Pane.

2 (Conditional) If you want to add a new sub-domain to an existing domain, select the required domain.

3 Click *Add Domain* in the *Task Pane*.

4 Enter a domain name.

5 Click *Finish*.

Modifying a Domain

1 Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

The *Navigation Pane* displays the current hierarchy for your Framework.

2 Select the required domain from the *Navigation Pane*.

3 Select *Modify Domain* from the *Task Pane.*

4 Enter the new domain name in the *Domain name* field.

5 Select *Finish*.

Deleting a Domain from the Framework

1 Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

The *Navigation Pane* displays the current hierarchy for your Framework.

2 Select the required domain from the *Navigation Pane*.

NOTE: The Framework does not allow a domain to be deleted if it contains any hosts. You must delete or move the hosts first.

3 Select *Delete Domain* from the *Task Pane.*

4 Select *Finish*.

IMPORTANT: This action cannot be undone.

2.1.2 Monitoring Hosts

Viewing the Host Log

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required host from the *Navigation Pane*.
3. Select *View Host Log* from the *Task Pane*.
4. You can:
   - Set the level of information you want to see on the screen:
     - *Error* displays only Error messages
     - *Warning* displays Warning and Error messages
     - *Information* displays Information, Warning and Error messages
   - Set the interval in seconds between screen refreshes in the *Refresh (secs)* field, from 1 to 60 seconds
   - Set the maximum number of log messages cached to display on the screen in the *Maximum cached log messages* field, from 10 to 1000 messages
   - Pause the screen display using the *Pause* checkbox
   - Clear the screen display using the *Clear* button.
5. Select *Close* to return to the Framework hierarchy view.

Modifying Log Settings

You can modify log settings for all hosts, all hosts in a domain, or an individual host using the Domain Log Settings or Host Log Settings options.

To modify log settings:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. To modify log settings for all hosts, select *Hosts* from the *Navigation pane*.

   To modify log settings for all hosts in a domain, select the required domain from the *Navigation Pane*.

   To modify log settings for an individual host, select the required host from the *Navigation pane*.
3. Select *Domain Log Settings* or *Host Log Settings* from the *Task Pane*.

You can:

- Change the filename and location of the log file in the *Filename* field
- Set the level of information you require from the *Level* drop-down list:
  - *Error* displays only Error messages
  - *Warning* displays Warning and Error messages
  - *Info* displays Information, Warning and Error messages
  - *Debug* displays Debug, Information, Warning and Error messages
  - *Trace* displays Trace, Debug, Information, Warning and Error messages

  NOTE: The *Debug* and *Trace* level settings are primarily for the use of Novell Support.
- Set the log to show all tasks using the *Show all tasks* checkbox

  NOTE: *Show all tasks* is primarily for the use of Novell Support.
- Select the rollover point from the *Rollover* drop-down list at which the log file will be overwritten with new information
- Select the maximum size of the log file from the *Max Size (MB)* drop-down list before it will be overwritten with new information

  NOTE: If the maximum size set for the log file is reached, the log file will be overwritten regardless of the setting of the rollover point.
- Enter a *Rollover Script* in Perl to be executed at the rollover point. An example Perl script is provided in "Example Rollover Script" on page 20.

4 Select *Next* to apply the changes.

If the changes are applied successfully, a green box  will be shown next to the agent name.

If the changes are not applied successfully, for example, if the host is not online, a red box  will be shown next to the agent name.

5 Select *Close*.

Example Rollover Script

This is an example of a Perl script that can be called at the rollover point for the host log file. The script compresses the old unifid.log and then removes any log files that are more than 30 days old.

```
use File::Basename;
system("/usr/bin/gzip $LOG_FILE");
my $log_root = dirname($LOG_FILE);
opendir(LOGDIR, $log_root);
# Find all the compressed log files
my @log_files = map { $_->[1] }
                        map { [ $_, "$log_root/$_" ] }
                        grep { /\.gz$/ }
                        readdir(LOGDIR);
closedir(LOGDIR);
# Delete all log files older than 30 days
my $time = time();
foreach my $log (@log_files) {
    my ($mtime) = (stat($log))[9];
```

```
    my $age  = int((($time - $mtime) / 3600) / 24);
    $ctx->log_info("Checking $log $age");
    next unless $age > 30;

    $ctx->log_info("Deleting $log ($age days old)");
    unlink $log;
}
```

System Alerts

The *System Alerts* screen shows system status alert messages from all hosts in your Framework. The screen shows the time of the alert, the host that originated the alert, the type of alert and information about the alert.

You can define the level of system alerts you require using the *Domain Alert Settings* or *Host Alert Settings* options.

The existence of system alerts is indicated by a flashing Framework icon in the bottom right hand corner of the screen.

1. Select the icon to display the *System Alerts* screen.
2. To clear the existing alerts, select *Finish*.

   To leave the *System Alerts* screen without clearing the existing alerts, select *Cancel*. The Framework icon continues to flash.

Modifying Alert Settings

You can configure your Framework hosts to generate system status alert messages when specific events occur, such as the agent exceeding a specified memory usage. If a system alert has been triggered, the Framework icon in the bottom right hand corner of the screen flashes. To view the system alerts, select the icon (see "System Alerts" on page 21 for details).

You can modify alert settings for all hosts, all hosts in a domain, or an individual host using the Domain Alert Settings or Host Alert Settings options.

To modify alert settings:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. To modify alert settings for all hosts, select *Hosts* from the *Navigation pane*.

   To modify alert settings for all hosts in a domain, select the required domain from the *Navigation Pane*.

   To modify alert settings for an individual host, select the required host from the *Navigation pane*.
3. Select *Domain Alert Settings* or *Host Alert Settings* from the *Task Pane*.

   NOTE: Changes made to a domain's alert settings will override the current settings for individual hosts in that domain. Changes then made to an individual host's alert settings override the current domain alert settings on that host.

   Configure at least one of the options described in the following steps.

4. From the *Alert on log level* drop down list, select the level of log information you require to trigger an alert. For example, if you want alerts to be triggered when error messages occur in the log, select *Error*. The *Warning* option will include *Warning* and *Error* messages. The *Info* option will include *Info*, *Warning* and *Error* messages. Select *Never* to switch this setting off.
5. In the *Alert log filter* field, define a specific message you want to trigger alerts, or part of a message with wildcard symbols *. You can use regular expressions in this field: to do this, select the *Regular expression* checkbox and enter your regular expression.

   NOTE: This setting is independent of the setting in Step 4.
6. From the *Time offset (mins)* drop down list, select or type the time offset in minutes you want to trigger an alert. An alert will be triggered if a host's time setting differs from the time setting of the Primary Registry Manager by this number of minutes. Time offsets may cause problems because certificates are time-based. The UTC (Universal Time Coordinated) value is used.
7. From the *Pending messages (mins)* drop down list, select or type the number of minutes you want to trigger an alert. An alert is triggered if an event has been in the queue of store and forward messages for this number of minutes.
8. From the *Maximum memory (MB)* drop down list, select or type the number of MB of memory you want to trigger an alert. An alert is triggered if a host is using more than this amount of memory.
9. From the *Minimum disk space (MB)* drop down list, select or type the minimum amount in MB of disk space you want to trigger an alert. An alert is triggered if a host has less than this amount of disk space remaining in the default installation location.
10. Select the *Expired certificate* checkbox to configure an alert to be triggered if an agent's certificate has expired.
11. Select *Next*. The settings are applied to the hosts.

    If the settings were applied successfully, the indicator next to the host name will go green  .

    If the settings were not applied successfully, for example, if the host is offline, the indicator will go red  .
12. Select *Close*. If any of your settings have caused an alert to be triggered, the Framework icon will flash.

Viewing the Host Status

The *Host Status* feature allows you to view the current status of all your hosts, or all the hosts in a domain, on one screen.

To view the host status:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select *Hosts* from the *Navigation Pane* if you want to view the status for all hosts, or select the required domain to view the status for all hosts in the domain.
3. Select *Host Status* from the *Task Pane*. The status for each host is displayed, as shown in the following table, with a summary at the bottom of the screen.

	The host is online.

	There is a status problem with the host, for example, the host's time offset exceeds the defined level (see Step 5). Clicking the arrow to the left of the green box to display status messages.
	The host is offline.
	The host is unregistered.

4 If required, select the hosts you want to view using the *Online*, *Offline* and *Unregistered* checkboxes.

If you have a long list of hosts, clear the Auto scroll checkbox to stop the list scrolling past automatically.

5 If required, change the filter settings from the default values and select *Restart* to check the status again. The filters available are:

- **Maximum time offset (minutes):** the difference in system time between the host and the Primary Registry Manager. If the time offset exceeds the value in this field, a warning indicator is displayed.
- **Minimum disk space (MB):** if the available disk space on the host machine goes below the value in this field, a warning indicator is displayed.
- **Maximum Memory (MB):** if the memory used by the host exceeds the value in this field, a warning indicator is displayed.

6 To go to a host's details, double-click it. Alternatively select *Close*.

2.1.3 Host Options

Managing Certificates

Certificates Overview

For added security, you can install a certificate for use when accessing the Framework Console. To access this option, you need to select the specific packages on the host you want to protect. You must then complete a certificate request form, send it to your chosen certification authority, and then install the certificate you are supplied with.

Requesting a Certificate

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required host from the *Navigation Pane.*
3. With the host's packages displayed, select the *Administration Agent (admin)* for the Framework Console.
4. Select *Request Certificate* from the *Task Pane*.
5. Enter the required details as described in your chosen certification authority documentation.
6. Select *Finish*.

   The required text for your certificate request is displayed in the text area.
7. Copy the certificate request into an e-mail and send it to your chosen certification authority.

When you receive the certificate from your certification authority, you must install it.

Installing a Certificate

When you have received the requested certificate from your certification authority:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required host from the *Navigation Pane.*
3. With the host's packages displayed, select the package for which you have requested the certificate.
4. Select *Install Certificate* from the *Task Pane*.
5. Paste the certificate into the text area.
6. Select *Finish*.

Restarting the Agent

If you are having problems, Novell Support might ask you to restart an agent, as described below.

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required host from the *Navigation Pane*.

   TIP: To select multiple hosts in a domain, select the domain then hold down the Ctrl key and select the required hosts from the right hand pane one at a time, or hold down the Shift key to select a consecutive list of hosts. To select all hosts in a domain, use Ctrl+A.

   ---
3. Select *Restart Agent* from the *Task Pane.*

4 Select the type of restart you want to perform as advised by Novell Support.

5 Select *Finish.*

Modifying the Connector

You can modify the way you connect to the Framework Console. You can define which interface card and port to use, and increase the security of the connection using SSL.

To access this option, you need to select the specific packages on the host you want to modify.

1 Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

The *Navigation Pane* displays the current hierarchy for your Framework.

2 Select the required host from the *Navigation Pane.*

3 With the host's packages displayed, select the *Administration Agent (admin)* for the Framework Console.

4 Select *Modify Connector* from the *Task Pane*.

5 You can:

- Define the address of a specific interface card
- Define which port to use
- Check the SSL checkbox if you want to use SSL.

6 Select *Finish*.

Promoting Managers

If you have multiple Framework Managers deployed, the first manager installed is defined as the primary manager by default, and its packages are defined as primary. Manager packages on all other manager hosts act as backups. If your primary manager becomes unavailable, you can select single or multiple manager packages on a host to be promoted to primary status.

1 Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

The *Navigation Pane* displays the current hierarchy for your Framework.

2 Select the required host from the *Navigation Pane.*

3 With the host's packages displayed, select the manager packages you want to promote.

TIP: To select multiple manager packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of manager packages.

---

4 Select *Promote Manager* from the *Task Pane*.

5 Review the list of manager packages you have selected.

6 Select *Finish*.

7 View the host's packages again and verify that the *Status* of the promoted manager packages has changed to primary.

Modifying Audit Settings

If required, you can define an alternative location for the audit databases using the Audit Settings option.

The databases containing audited data from command control (cmndctrl.db) can be placed in an alternative location. The administration audit files (audit.db and audit.ldb) and log.msqs are still stored in the default location /opt/novell/service/local/audit or C:\Program Files\Novell\npum\service\local\audit, but these files are relatively small.

To define an alternative location for the audit databases:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the host with the Audit Manager installed from the *Navigation Pane.*
3. With the host's packages displayed, select the *Audit Manager (audit)* package.
4. Select *Audit Settings* from the *Task Pane*.
5. In the *Audit Path* field, enter the required location for the audit databases.
6. Select *Finish*.

SMTP Settings

The *SMTP Settings* option allows you to provide details of your email server so reports such as the compliance auditor reports can be automatically emailed to the required personnel.

To configure SMTP settings:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the host where the Messaging Component is installed.
3. With the host's packages displayed, select the *Messaging Component (msgagnt)*.
4. Select *SMTP Settings* from the *Task Pane*.
5. In the *SMTP Host* field, enter your email server IP address.
6. In the *SMTP Port* field, select or type the required port number.
7. If you are using a Lotus Notes server, in the *SMTP Domain* field enter the name of your SMTP domain.
8. Select *Finish*.

Viewing Store and Forward Messages

Messages from one host to another are stored if the host cannot communicate with the other host, and forwarded when the communication link is restored. You can view these messages, and delete them if you do not need them.

To view store and forward messages:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the host for which you want to view store and forward messages.
3. Select the host's *Packages* icon (select the arrow next to the host's name to display it).

4. Select *View Messages* from the *Task Pane*. If any stored messages exist, they are displayed. Information about the message is shown, including the time the message was sent, the host the message was being sent to, the module that sent the message, the type of message (method), the number of failed attempts at sending the message, and the next scheduled attempt to send the message, if any.
5. To attempt to send one or more messages again, select the messages and select *Retry*.
6. To delete one or more messages, select the messages and select *Delete*.
7. To refresh the screen, select *Refresh*.
8. Select *Close*.

Viewing the Registry Cache

The registry cache is held by the Registry Agent on each host, and it contains a list of the packages deployed on each host in your Framework. This list is a copy of part of the information held by the Registry Manager, and it enables Framework components to locate and communicate with each other, according to their position in the hierarchy created when you add domains and hosts to your Framework. Agents send requests to managers in their immediate sub domain, and if unsuccessful, try a manager further up in the hierarchy. See Chapter 7, "Load Balancing and Failover," on page 111 for further details.

You can view the registry cache to check which hosts in your Framework a specific manager or agent module is installed on, and their order in the Framework hierarchy according to the hosts they are installed on.

If the registry cache becomes out of date, communication problems can occur. To fix this, you can try clearing the registry cache on the Registry Agent to allow it to be updated by the Registry Manager (see "Clearing the Registry Cache" on page 28 or Step 8 below).

When viewing the registry cache, you can use the stale cache (the default option). The cache is considered stale if it has not been updated by the Registry Manager for 2 hours, and this is usually adequate. If you uncheck the *Use Stale Cache* checkbox, the information is provided by the Registry Manager.

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the host for which you want to view the registry cache.
3. Select the host's *Packages* icon (select the arrow next to the host's name to display it).
4. Select *View Cache* from the *Task Pane*.
5. From the drop down list, select the package you want to look up in the registry cache.
6. If you want to view the latest information from the Registry Manager, uncheck the use stale cache checkbox.
7. Select *Lookup*. Details of the hosts where the module is installed are displayed in order according to their position in the Framework hierarchy. Information shown includes the Framework agent name, IP address, port number, and whether the host has the primary manager component installed (indicated by 1 in the *Primary* column) or not (indicated by 0).

8 If you want to clear the registry cache, select *Clear Cache*. This marks the cache as stale, and it is automatically updated by the Registry Manager. You can also clear the cache using the *Clear Cache* option in the *Task Pane*.

9 Select *Close*.

Clearing the Registry Cache

Novell Support might advise you to try clearing the registry cache if you have communication problems between Privileged User Manager components. The registry cache is held by the Registry Agent and contains a list of manager and agents in your Framework, copied from the Registry Manager. See "Viewing the Registry Cache" on page 27 for more details.

To clear the registry cache:

1 Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

The *Navigation Pane* displays the current hierarchy for your Framework.

2 Select the host for which you want to clear the registry cache.

3 Select the host's *Packages* icon (select the arrow next to the host's name to display it).

4 Select *Clear Cache* from the *Task Pane*. The registry cache is marked as stale and is updated by the Registry Manager.

You can also clear the registry cache using the *View Cache* option.

2.2 Managing Module Distribution

Modules are the individual products that you download from the Web site or an internal server on your network and deploy to your hosts from the Framework Console.

Each module is broken down into individual packages that can be independently updated in the event of a modification.

When viewing the possible modules to download from your chosen server, you see each module broken down into its package components.

Each package can have up to three defining roles that can be deployed across any part of the Framework and these are explained below.

Module Console The user interface into the package that allows configuration and management of the module.

Module Manager Provides the centralized services and administration packages of the module.

Module Agent The client packages of the module, receiving and carrying out instructions from the manager on all hosts.

This section describes the following tasks:

- Section 2.2.1, "Downloading Packages to a Package Manager," on page 29
- Section 2.2.2, "Deploying Packages to Hosts," on page 30
- Section 2.2.3, "Viewing Packages Available to Deploy," on page 33

2.2.1 Downloading Packages to a Package Manager

To download packages to your Framework hosts, you must first download them, add them to a Package Manager, and then add them from the Package Manager to individual hosts.

There are two options for downloading packages to a Package Manager:

1. You can download packages directly from the Update Server.
2. You can download packages from the Update Server onto a local server, and then download packages from this local server onto your Framework hosts.

You must configure the Package Manager to access the server you require.

Configuring the Package Manager

1. Select *Package Manager* from the *Navigation Pane* on the Framework Console home page.
2. Select *Settings* from the *Task Pane*.
3. Select *Local Package Manager*.
4. Specify the *Host name* and *Port* for your Local Package Manager.
5. Select *Finish*.

Downloading Packages from the Update Server

Using this option, the Package Manager contacts the Update Server directly across the internet.

NOTE: The Package Manager requires direct or proxy access to the internet.

---

1. Configure the Package Manager to supply valid customer portal login details to the Update Server, and, if required, configure your proxy host details.
2. Add the required packages to the Package Manager.

Downloading Packages from Local Servers

You can download packages using a local server configured as a Local Package Manager. This option allows you to avoid connecting all your Framework hosts directly to the internet.

1. Install a Framework Manager on a host that has internet access. Use this host to download packages from the Update Server. This host becomes your Local Package Manager.
2. Configure the Package Managers on your main Framework Managers to download from the Local Package Manager.
3. Add the required packages to the Package Managers on your main Framework Managers.

Adding Packages to the Package Manager

1. Select *Package Manager* from the *Navigation Pane* on the Framework Console home page.
2. Select *Add Packages* from the *Task Pane*.
3. Set the *Package Filter* options according to the versions, platforms, types and components you require.
4. Select the required packages from the list of available packages.

   TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.
5. Select *Next* to start downloading.
6. Select *Finish*.

Checking for Updated Packages

1. Select *Package Manager* from the *Navigation Pane* on the Framework Console home page.
2. Select *Check For Updates* from the *Task Pane*.

   The N*avigation Pane* displays updated packages available for download.
3. Select the required packages from the list of available packages.

   TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.
4. Select *Next* to start downloading.
5. Select *Finish*.

Removing Packages

1. Select *Package Manager* from the *Navigation Pane* on the Framework Console home page.
2. Select the packages you want to removed from the list of available packages.

   TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.
3. Select *Remove Packages* from the *Task Pane*.
4. Select *Next* to start removing packages.
5. Select *Finish*.

2.2.2 Deploying Packages to Hosts

- “Installing Packages on a Host” on page 31
- “Updating Packages for a Host or Domain” on page 31
- “Rolling Back Packages for a Host or Domain” on page 32

- "Uninstalling Packages from a Host" on page 32
- "Registering and Unregistering Packages for a Host" on page 32

Installing Packages on a Host

1. Ensure the required packages have been downloaded to the Framework Package Manager by viewing packages available to deploy.
2. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
3. Select the required domain from the *Navigation Pane.*
4. Select the required host.
5. With the host's Packages icon selected, select *Install Packages* from the *Task Pane*.
6. Select the required packages from the list of available packages.

   TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.
7. Select *Next* to start installing the selected packages.
8. Select *Finish*.

Updating Packages for a Host or Domain

When updated packages become available on your local Package Manager, you can update the packages installed on individual hosts or for all hosts in a domain.

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* will display the current hierarchy for your Framework.
2. Select the required domain from the *Navigation Pane*. If you want to update packages for a specific host, select the required host.
3. Select Update Domain Packages or Update Packages from the *Task Pane*. The Framework checks for updates on your Package Manager and displays any updated packages available for download.
4. Select the required packages from the list of available packages.

   TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.
5. If you want to create a backup of the current installed packages, check *Create backup*. If required, you can roll back to the backed up packages using the *Rollback Packages* option.

   NOTE: Only the last update is stored. Creating backups requires extra space.
6. Select *Next* to start downloading the selected packages.
7. Select *Finish*.

Rolling Back Packages for a Host or Domain

If you chose to create a backup when updating packages for an individual host or for a domain, you can roll back to the last backed up packages. Only the last backed up packages are saved.

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required domain from the *Navigation Pane*. If you want to roll back packages for a specific host, select the required host.
3. Select *Rollback Domain Packages* or *Rollback Packages* from the *Task Pane*.
4. Select the required packages from the list of available backed up packages.

   TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.
5. Select *Next* to start rolling back to the previously backed up packages.
6. Select *Finish*.

Uninstalling Packages from a Host

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page. The *Navigation Pane* displays the current hierarchy for your Framework.
2. Select the required domain from the *Navigation Pane.*
3. *Select* > *the* required host.
4. With the host's *Packages* icon selected, select the required packages from the list of installed packages.

   TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.
5. Select *Uninstall Packages* from the *Task Pane*.
6. Select *Next* to start uninstalling the selected packages.
7. Select *Finish*.

Registering and Unregistering Packages for a Host

If you want to stop a package functioning without having to remove it completely, you can unregister it. You can then register it again later if required. Packages are automatically registered when you add them, so you only need to register them if you have previously unregistered them.

Registered packages are shown with a green tick: 

Unregistered packages are shown with a red exclamation mark: 

To register or unregister a package for a host:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

   The *Navigation Pane* displays the current hierarchy for your Framework.

2 Select the required domain from the *Navigation Pane.*

3 *Select the* required host.

4 With the host's packages displayed, select the packages you want to register or unregister.

TIP: To select multiple packages, hold down the Ctrl key and select the required packages one at a time, or hold down the Shift key to select a consecutive list of packages. To select all packages, use Ctrl+A.

5 Select *Register Package* or *Unregister Package* from the *Task Pane*, as required.

NOTE: The Framework Console does not refresh automatically. To check whether your packages have been registered or unregistered successfully, you should go to another screen and then return to the list of packages.

2.2.3 Viewing Packages Available to Deploy

- "Viewing All Available Packages" on page 33
- "Viewing Packages for a Host to Deploy" on page 33
- "Viewing Consoles to Deploy" on page 33

Viewing All Available Packages

To view the packages you have available to deploy, including consoles:

1 Select *Package Manager* from the *Navigation Pane* on the Framework Console home page.

The *Navigation Pane* displays the packages that you have downloaded from the Update Server or Local Package Manager.

Viewing Packages for a Host to Deploy

To view the packages available to deploy to individual hosts:

1 View the host details.

2 Select *Install Packages* from the *Task Pane*.

The *Navigation Pane* displays a list of packages available for deployment on the selected host.

Viewing Consoles to Deploy

1 Select Install *Consoles* from the *Task Pane* on the Framework Console home page.

The *Navigation Pane* displays a list of consoles available for deployment on the host.

2.3 Managing the Workspace

- Section 2.3.1, "Managing the Consoles," on page 34
- Section 2.3.2, "Adding Consoles to the Framework Console," on page 34
- Section 2.3.3, "Removing Consoles from the Framework Console," on page 34
- Section 2.3.4, "Updating Consoles in the Framework Console," on page 34

2.3.1 Managing the Consoles

The Framework Console can be extended by installing console packages. Console packages provide the administrative and reporting panes for the Privileged User Manager modules.

Console packages must have been downloaded to the Package Manager before they become available for installation.

2.3.2 Adding Consoles to the Framework Console

1. Ensure the required consoles have been downloaded to the Framework Package Manager by viewing consoles available to deploy.
2. Select *Install Consoles* from the *Navigation Pane* on the Framework Console home page.
3. Select the required consoles from the list of available consoles.

   TIP: To select multiple consoles, hold down the Ctrl key and select the required consoles one at a time, or hold down the Shift key to select a consecutive list of consoles. To select all consoles, use Ctrl+A.
4. Select *Next* to start installing.
5. Review the list of installed consoles.
6. Select *Finish*.

2.3.3 Removing Consoles from the Framework Console

1. Select *Uninstall Consoles* from the *Task Pane* on the Framework Console home page.
2. Select the required consoles from the list of available consoles.

   TIP: To select multiple consoles, hold down the Ctrl key and select the required consoles one at a time, or hold down the Shift key to select a consecutive list of consoles. To select all consoles, use Ctrl+A.
3. Select *Next* to start uninstalling.
4. Review the list of removed consoles.
5. Select *Finish*.

2.3.4 Updating Consoles in the Framework Console

When updated console packages become available on your Package Manager, you can update the console packages installed on your Framework Console.

1. Select *Update Consoles* from the *Task Pane*.

   The *Navigation pane* will display updated consoles available for deployment.
2. Select the required consoles from the list of available consoles.

   TIP: To select multiple consoles, hold down the Ctrl key and select the required consoles one at a time, or hold down the Shift key to select a consecutive list of consoles. To select all consoles, use Ctrl+A.

3. Select *Next* to start installing.
4. Review the list of updated consoles.
5. Select *Finish*.

NOTE: After updating a console you must shut down and re-open the Framework Console to see the changes.

Managing Framework Users

3

Privileged User Manager provides comprehensive user management facilities to control access to the Framework consoles. Role-based authorization is used to determine which user groups can access specific consoles and perform specific tasks.

The admin user created when the Framework is initially installed belongs to the admin group, which has full access to all installed consoles and can perform all tasks. You can use this user account to create additional user accounts and groups using the *Manage Users* console, part of the Access Control module. For information about deploying the Access Control module see Section 3.1, “Deploying the Access Control Module,” on page 37.

When you add a new user, the user will be unable to access any of the Framework consoles until added to a group which contains a role allowing them appropriate access. For example, if you want a user to be able to access only the Compliance Auditor console, you must create a group and configure the appropriate Compliance Auditor roles, then create the user and add the user to the group.

You can create additional users with the same access as the admin user by adding them to the admin group, or create your own group with access to all modules and roles. You can also configure these additional users to be super users. Only users who belong to a group with the 'super' role configured can view and administer super users.

For information about adding and configuring users and groups, refer to the appropriate topics in this section.

3.1 Deploying the Access Control Module

The Access Control modules comprise the following packages:

- Access Manager (auth): holds the Framework user account information and controls access to the Framework modules.
- Access Control Console (displayed in the Framework Console as the *Manage Users* console): installed into the Framework console and required for configuring Framework users and groups.

The Access Manager is only shown as an available package on hosts with the Registry Manager (registry) deployed.

In addition, the Access Control console can only be deployed on hosts with the Administration Manager (admin) deployed.

To deploy the Access Control modules:

1. Download the following packages to your local Package Manager:
   - Access Manager
   - Access Control Console
   - Registry Manager
   - Administration Manager

   See Section 2.2, “Managing Module Distribution,” on page 28 for details.
2. Install the Registry Manager on the host you want to be the Access Manager, then install the Access Manager on the same host. This can be on any operating system including Windows. See Section 2.2.2, “Deploying Packages to Hosts,” on page 30 for details. The packages can be deployed to as many hosts as you wish to build an environment with load balancing and failover.
3. Install the Administration Manager on the same host or a different host. It can be deployed to as many hosts as you wish to build an environment with load balancing and failover.
4. Install the Access Control Console on a host with the Administration Manager installed. See Section 2.3.2, “Adding Consoles to the Framework Console,” on page 34 for details.

The Access Control module is now deployed and ready to use.

3.2 Changing a Framework User's Password

Framework users can change their own passwords using the Change Password option, which is always available in the Task Pane. If a Framework user belongs to a group with the appropriate auth role defined (see Section 3.4.3, “Configuring Roles,” on page 48), they can also change other users' passwords using the Modify User option.

To change your own password:

1. Select *Change Password* from the *Task Pane*.
2. In the *Old Password* field, enter your current password.
3. In the *New Password* field, enter your new password and confirm it in the *Confirm Password* field.

   NOTE: Your password must comply with the default *Account Settings* > *f*or the Framework, and/or individual user settings defined using the *Modify User* option.
4. Select *Finish*.

3.3 Users

- Section 3.3.1, “Account Settings,” on page 39
- Section 3.3.2, “Adding a Framework User,” on page 39
- Section 3.3.3, “Modifying a Framework User,” on page 40
- Section 3.3.4, “Removing a Framework User Group from a User,” on page 45
- Section 3.3.5, “Deleting a Framework User,” on page 46

3.3.1 Account Settings

The *Account Settings* option allows you to set your own default values for user settings such as minimum password length. When you add a new user, these default settings will apply and can be overridden for individual users by modifying their individual account settings.

To set your own default account settings:

1. Select *Manage Users* from the *Navigation Pane* on the Framework Console home page.
2. Select *Users* from the *Navigation Pane*.
3. Select *Account Settings* from the *Task Pane*.
4. The following options are available:
   - **Inactivity timeout (minutes) -** enter the number of minutes you will allow users to be inactive before logging them out of the Framework console.
   - **Account lockout -** enter the number of times you will allow a user to enter the wrong password before being locked out. You can re-enable the user's account using the Modify User option, by clearing the Disabled checkbox. You can reset the user's password using the Modify User option.
   - **Inactive days (disable) -** enter the number of days you will allow a user's account to be inactive before it will be disabled. You can reactivate the user's account using the Modify User option, using the Reactivate account checkbox in the Account section.
   - **Inactive days (delete) -** enter the number of days you will allow a user's account to be inactive before it will be deleted.
   - **Password lifetime (days) -** enter the number of days you will allow a user's password to be used before it expires, prompting the user to change the password.
   - **Minimum password length -** enter the minimum number of characters you require in a user's password.
   - **Password history -** enter the number of unique passwords that a user must use before being allowed to reuse an old password.
   - **Minimum alpha -** enter the minimum number of alpha characters you require in a user's password.
   - **Minimum numerics -** enter the minimum number of numeric characters you require in a user's password.
5. Select *Finish*.

3.3.2 Adding a Framework User

When you add a new Framework user:

- The user's account will be set up according to the default values defined in the *Account Settings* option. You can change these settings for individual users using the *Modify User* option.
- The user's password will be set to expire immediately so they will be prompted to change it when they first log on to the Framework Console. You can change this setting for individual users using the *Modify User* option.
- The user will be unable to access any of the Framework consoles until you have added the user to a group with the required roles defined. See Section 3.4.3, “Configuring Roles,” on page 48 for details.

To add a new Framework user:

1. Select *Manage Users* from the *Navigation Pane* on the Framework Console home page.
2. Select *Users* from the *Navigation Pane*.
3. Select *Add User* from the *Task Pane*.
4. Enter a name for the user in the *Username* field.
5. Enter a password for the user in the *Password* field.

   NOTE: The password must comply with the default Account Settings for the Framework.

6. Select *Finish*. You can now configure additional settings for the user's account using the *Modify User* option.

3.3.3 Modifying a Framework User

The *Modify User* option allows you to override the default *Account Settings* for an individual user, and also provides a number of additional configuration settings and tasks, including resetting a user's password and assigning a user to a group.

To modify a Framework user account:

1. Select *Manage Users* from the *Navigation Pane* on the Framework Console home page.
2. Select *Users* from the *Navigation Pane*.
3. Select the user account you want to modify from the left hand pane.
4. Select *Modify User* from the *Task Pane*.
5. You can:
   - Disable the user by checking the *Disabled* checkbox.
   - Add a short comment in the *Comment* field.
   - Add a more detailed description in the *Description* field.
6. To configure additional options, select the section you require as follows:
   - **Password -** allows you to reset the user's password and configure other password settings.
   - **Password validation -** allows you to define the minimum number of alpha and numeric characters required in the user's password.
   - **Account -** allows you to configure the user as a super user, provides information about the user's account, and provides other account configuration options.
   - **Account Details -** allows you to enter personal information for the user, including Staff ID and contact details.
   - **Host Access Control -** allows you to control where the user can access the console from.
   - **Native Maps -** allows you to map the Framework user account to a user account on a UNIX platform to enable command line interaction with Framework modules.
   - **Logon Script -** allows you to define a Perl logon script for the user.
   - **Groups -** allows you to add the user to one or more groups.
7. When you have completed your changes, select *Finish*.

Modify User Options

Modify User: Password

To set password options for a Framework user:

1. From the *Manage Users* console, select the user account you want to modify and select *Modify User*.
2. Select *Password*.
3. The following options and information are available:
   - **Password -** to reset the user's password, enter the new password and re-enter it in the *Confirm* field.

     NOTE: The password must comply with the default *Account Settings* > *f*or the Framework, and/or individual user settings defined using this option and the Password Validation option.

   - **Expired -** check the *Expired* checkbox to expire the user's current password immediately, forcing the user to change it on next logon.
   - **Last changed -** indicates when the password was last changed by the user, or, if not yet changed by the user, when the user and password were created.
   - **Reset password age -** check the *Reset password age* checkbox to reset the age of the password to zero. The user can use the password for the full number of days defined in P*assword lifetime (days)* (see Section 3.3.1, "Account Settings," on page 39), or in *Maximum age*, below, if configured.
   - To override the default account settings for this user, check the appropriate checkbox and set the required values as follows:
     - **Minimum length -** enter the minimum number of characters you require in a user's password.
     - **Maximum age -** enter the number of days you will allow a user's password to be used before it expires, prompting the user to change the password.
     - **History -** enter the number of unique passwords that a user must use before being allowed to reuse an old password.
4. Select *Finish* or select another option.

Modify User: Password Validation

To set password validation options for a Framework user:

1. From the *Manage Users* console, select the user account you want to modify and select *Modify User*.
2. Select *Password validation*.
3. To override the default account settings for this user, check the appropriate checkbox and set the required values as follows:
   - **Min alpha characters -** enter the minimum number of alpha characters you require in the user's password.
   - **Min numeric characters -** enter the minimum number of numeric characters you require in the user's password.
4. Select *Finish* or select another option.

Modify User: Account

To set account options for a Framework user:

1. From the *Manage Users* console, select the user account you want to modify and select *Modify User*.
2. Select *Account*.
3. The following options and information are available:
   - **Super user -** check the *Super user* checkbox to make this user a super user.

     NOTE: The Super user option is only available if you are logged on as a super user. Super users can be viewed and administered only by users belonging to a group with the 'super' role defined for the 'auth' module.
   - **Last logon -** indicates when the user last logged on to the Framework Console.
   - **Reactivate account -** check the *Reactivate account* checkbox to re-enable a user's account which has become locked through bad logons.
   - **Last bad logon -** indicates the last time the user failed to log on successfully.
   - **Bad logons -** indicates the number of times the user has failed to log on successfully since the last successful logon.
   - **Reset bad logon count -** reset the number of unsuccessful logons to zero.
   - To override the default account settings for this user, check the appropriate checkbox and set the required values as follows:
     - **Lockout -** enter the number of times you will allow the user to enter the wrong password before being locked out. You can re-enable the user's account by clearing the *Disabled* checkbox in the main *Modify User* section. You can reset the user's password in the *Password* section.

       Disable inactive days - enter the number of days you will allow the user's account to be inactive before it will be disabled. You can reactivate the user's account using the *Reactivate* account option described above.

       Delete inactive days - enter the number of days you will allow the user's account to be inactive before it will be deleted.

Inactivity logout mins - enter the number of minutes you will allow the user to be inactive before logging the user out of the Framework console.

- **Message of the day -** enter a message to be displayed to the user after a successful logon.

4 Select *Finish* or select another option.

Modify User: Account Details

To set personal account details for a Framework user:

1 From the *Manage Users* console, select the user account you want to modify and select *Modify User*.

2 Select *Account Details*.

3 To set the following options, check the appropriate checkbox and enter the required text:

- **Staff ID -** enter the user's staff ID, for example, the user's unique company identifier.

  Display name - enter a display name for the user, for example, the user's full name. If a name is defined here it can be automatically entered as the *Manager Name* in Account Group and User Group definitions for Command Control, by selecting the manager's Framework user name (see "Modifying an Account Group" on page 72 and "Modifying a User Group" on page 74). It can also be used in Compliance Auditor reports (see Section 5.6.2, "Adding or Modifying an Audit Report," on page 98).

  Email address - enter the user's email address. If an email address is defined here it can be used in Command Control and the Compliance Auditor as described above.

  Telephone number - enter the user's telephone number. If a telephone number is defined here it can be used in Command Control and the Compliance Auditor as described above.

4 Select *Finish* or select another option.

Modify User: Host Access Control

You can control where the user can access the Framework console from by defining a list of ports and hosts from where access is allowed, or a list of ports and hosts from where access is denied.

If you make no entries for this option, access is allowed from any location.

To control where the user can access the Framework console from:

1 From the *Manage Users* console, select the user account you want to modify and select *Modify User*.

2 Select *Host Access Control*.

3 If you want to define a list of locations from where the user is allowed to access the console (and deny access from all other locations):

3a Select the *Access Order* checkbox. Leave the access order as *Allow Deny*.

3b If auditing is required, select the *Auditing* checkbox and select the events you want to be audited from the drop-down list.

3c Select the *Access Allow* checkbox.

3d Select the *Add* button below the Allow table.

3e In the *Port Range* column, enter the required port number or range of port numbers. The following entries are allowed:

*	all ports
port	a single port, e.g. 80
port-port	a range of ports, e.g. 20-30
svcname	resolves a service name to its port, e.g. HTTP

3f In the *Host/IP Subnet* column, enter the required host definition. The following entries are allowed:

*	all hosts
ip address	a full IP address, e.g. 192.168.1.1
ip address-ip address	a range of IP addresses, e.g. 192.168.1.1-192.168.1.12
part ip address	part of an IP address, e.g. 192.168.1
network/netmask	a network/netmask pair, e.g. 192.168.1.0/255.255.255.0
network/nnn CIDR	a network/nnn CIDR, e.g. 192.168.11.0/24
hostname	a hostname, e.g. dellsrv1.novell.com
domain	a domain name, e.g. *.novell.com

3g Repeat Step 3d through Step 3f for any other required location definitions.

4 If you want to define a list of locations from where the user is denied access to the console (and allow access from all other locations):

4a Select the *Access Order* checkbox.

4b Change the order to *Deny Allow*.

4c If auditing is required, select the *Auditing* checkbox and select the events you want to be audited from the drop-down list.

4d Select the *Access Deny* checkbox.

4e Select the *Add* button below the *Deny* list.

4f Enter the required locations as described in steps Step 3e and Step 3f above.

4g Repeat steps Step 4e and Step 4f for any other required location definitions.

5 Select *Finish* or select another option.

Modify User: Native Maps

Users can interact with Privileged User Manager using either the Framework Console or a command line interface on supported UNIX platforms. To provide authentication and access to the components, you must map a user account on the UNIX platform to a Framework user account by creating a native map.

For information about the command line interface, contact PrivilegedUserSupport@novell.com.

To add a new native map for a Framework user:

1. From the *Manage Users* console, select the user account you want to modify and select *Modify User*.
2. Select *Native Maps*.
3. Select *Add*.
4. In the *User* column, type the user's name on the UNIX platform.
5. In the *Host* column, select the hostname for the UNIX platform.
6. Repeat Step 3 through Step 5 for any additional maps you require.
7. Select *Finish* or select another option.

To edit a native map, select it and make the required changes.

To remove a native map, select it and select *Remove*.

Modify User: Logon Script

You can assign a Perl script to a user to be run when the user logs on to the Framework Console. For example, you could assign a script that causes an email to be sent to a manager when the user logs on.

To assign a logon script to a Framework user:

1. From the *Manage Users* console, select the user account you want to modify and select *Modify User*.
2. Select *Logon Script*.
3. Enter the logon script you require for this user. You can type the script or paste it from another document.
4. Select *Finish* or select another option.

Modify User: Groups

To assign a Framework user to one or more groups:

1. From the *Manage Users* console, select the user account you want to modify and select *Modify User*.
2. Select *Groups*.
3. Select the checkboxes for the groups you want this user to belong to.
4. Select *Finish* or select another option.

You can also assign a user to a group using the *Modify Group* option, or by dragging the user onto the group, or the group onto the user.

You can remove a user from a group by unchecking the checkbox for the required group. See Section 3.3.4, “Removing a Framework User Group from a User,” on page 45 for other methods.

3.3.4 Removing a Framework User Group from a User

There are several ways of removing a Framework user group from a Framework user's account, as described below.

From the Navigation Pane

1. From the *Manage Users* console, select the group you want to remove from the user's account.
2. In the right hand pane, select the required user.
3. Select *Remove User* from the *Task Pane*. The user is removed.

Or:

1. From the *Manage Users* console, select the user you want to remove from a group.
2. In the right hand pane, select the group you want to remove from the user's account.
3. Select *Remove Group* from the *Task Pane*. The group is removed.

Using the Modify Group option

1. From the *Manage Users* console, select the group you want to modify and select *Modify Group*.
2. Select *Members*.
3. Uncheck the checkboxes for the user accounts you want to remove from the group.
4. Select *Finish*.

Using the Modify User option

1. From the *Manage Users* console, select the user account you want to modify and select *Modify User*.
2. Select *Groups*.
3. Uncheck the checkboxes for the groups you want to remove from the user's account.
4. Select *Finish*.

3.3.5 Deleting a Framework User

1. Select *Manage Users* from the *Navigation Pane* on the Framework Console home page.
2. Select *Users* from the *Navigation Pane*.
3. Select the user you want to delete from the left hand pane.
4. Select *Delete User* from the *Task Pane*.
5. Select *Finish* to confirm the deletion.

3.4 Groups

3.4.1 Adding a Framework User Group

Framework users must be assigned to one or more groups with the appropriate roles defined before they can access any Framework consoles or perform any tasks.

To add a new Framework user group:

1. Select *Manage Users* from the *Navigation Pane* on the Framework Console home page.
2. Select *Groups* from the *Navigation Pane*.
3. Select *Add Group* from the *Task Pane*.
4. Enter a name for the group in the *Group name* field.
5. Select *Finish*. You can now configure the group, including defining roles, using the *Modify Group* option.

3.4.2 Modifying a Framework User Group

The Modify Group option allows you to:

- Add a comment describing the group
- add users and subgroups to the group
- define administrative roles for the group
- specify an audit manager for the group.

To modify a Framework user group:

1. Select *Manage Users* from the *Navigation Pane* on the Framework Console home page.
2. Select *Groups* from the *Navigation Pane*.
3. Select the group you want to modify from the left hand pane.
4. Select *Modify Group* from the *Task Pane*.
5. In the *Comment* field, enter a comment if required.
6. In the *Members* section select the users you want to be members of this group.

   You can also add a user to groups in the *Groups* section of the *Modify User* option, or by dragging the user onto the group, or the group onto the user.

   You can remove users from the group by deselecting them here. See Section 3.3.4, “Removing a Framework User Group from a User,” on page 45 for other methods.
7. In the *Sub Groups* section, select the groups you want to be sub groups of this group. You can also add sub groups to groups by dragging the group onto the main group.
8. In the *Roles* section, configure the roles you require for this group of users according to the consoles you want them to be able to access and the tasks you want them to be able to perform. You must assign at least one role. See Section 3.4.3, “Configuring Roles,” on page 48 for further details.
9. In the *Audit Manager* section, enter the details of the group's manager if required.
10. Select *Finish*.

3.4.3 Configuring Roles

When you create a new Framework user group, you must assign at least one role to the group to allow the users belonging to the group to access one or more Framework modules and perform tasks.

To allow access to all modules and tasks, you can define a role with Module set to * and Role set to *. This is how the default admin group containing the default admin user is initially configured.

To allow access only to specific modules and tasks, use the Modify Group option (see Section 3.4.2, "Modifying a Framework User Group," on page 47) and define one or more roles according to the tables below:

Manage Users

Module	Role	Allows users to
auth	read	Read the auth database.
		This role must be used with all other auth roles.
	admin	Add/delete users.
		Add/delete groups.
		Assign users to groups.
	act_settings	Modify Account Settings.
	role_admin	Add/remove roles.
	super	View and modify super users.
		View and modify groups with the super role defined.
	*	Perform all roles.

Reporting

Module	Role	Allows users to
audit	read	Read the audit database.
		This role must be used with all other audit roles.
	console	View Reporting Console.

Module	Role	Allows users to
	admin	Modify Reporting settings.
	command	View Command Control reports.
	logon	View Account Logon reports.
	*	Perform all roles.

Compliance Auditor

Module	Role	Allows users to
secaudit	console	View the Compliance Auditor console.
	audit	View and edit records.
	admin	Add and modify audit rules.
	*	Perform all secaudit roles above.
	<audit role name>	Access the records collected by audit rules with this role defined in the Audit Role field on the Modify Audit Rule screen (you can choose your own name for the role). See Section 5.5, “Audit Rules,” on page 96 for details of configuring audit rules.
audit	read	View keystroke replay.
auth	read	Extract user credentials, including name and email address, from the auth database for use with reports.

Hosts

Module	Role	Allows users to
unifi	info	Run the host status check using the command line interface. You must type the word 'info' as it is not available in the drop-down list.
	admin	View the hosts console and perform administrative actions.

Package Manager

Module	Role	Allows users to
pkgman	admin	View/add/update/remove packages.

Command Control

Module	Role	Allows users to
cmdctrl	read	View the Command Control console and run test suites.
	write	Modify the Command Control database. They cannot cancel other users' transactions or modify audit or transaction settings. Must be used in conjunction with the cmdctrl read role.
	admin	Modify the Command Control database, including canceling other users' transactions and modifying audit and transaction settings.
	*	Perform all roles.
auth	read	Extract user credentials, including name and email address, from the auth database into the account and user group definitions. Used in conjunction with the cmdctrl write (with read) and admin roles.

Removing a Role

Select the role you want to remove and select Remove.

3.4.4 Removing a Framework User from a Group

There are several ways of removing a Framework user from a Framework user group, as described below.

From the Navigation Pane

1. From the *Manage Users* console, select the user you want to remove from a group.
2. In the right hand pane, select the group you want to remove the user from.
3. Select *Remove Group* from the *Task Pane*. The group is removed from the user's account.

Or:

1. From the *Manage Users* console, select the group you want to remove the user from.
2. In the right hand pane, select the user you want to remove from the group.
3. Select *Remove User* from the *Task Pane*. The user is removed from the group.

Using the Modify User option

From the Manage Users console, select the user you want to remove from one or more groups and select Modify User.

1. Select *Groups*.
2. Uncheck the checkboxes for the groups you want to remove the user from.
3. Select *Finish*.

Using the Modify Group option

1. From the *Manage Users* console, select the group you want to remove the users from and select *Modify Group*.
2. Select *Members*.
3. Uncheck the checkboxes for the users you want to remove from the group.
4. Select *Finish*.

3.4.5 Deleting a Framework User Group

1. Select *Manage Users* from the *Navigation Pane* on the Framework Console home page.
2. Select *Groups* from the *Navigation Pane*.
3. Select the group you want to delete from the left hand pane.
4. Select *Delete Group* from the *Task Pane*.
5. Select *Finish* to confirm the deletion.

Command Control

4

The command control feature provides UNIX and Linux users with controlled access to privileged commands in a secure manner across the enterprise. Command control enables the complete lock-down of user privilege by providing rules to determine the commands that are authorized to be run, and a powerful account delegation feature that removes the need for common access to the root account.

Command control provides centralized logging of activity across all platforms, and enables the selective capture of session activity for any user, to the keystroke level, which can be viewed using the Compliance Auditor and reporting features.

Additional features include external scripting which provides the facility to authenticate via third party security databases or applications, and comprehensive test suite tools which allow the administrator to model and test new rule combinations before committing to production use.

4.1 How Does It Work?

UNIX/Linux commands are passed to the command control using scripts, commands, or replacement shells, using the methods described under Section 4.3, “Integrating Command Control,” on page 55.

The commands are then validated against configured rule criteria such as the submit user, the submit host, the run host requested, the date/time, and the command name itself.

If authorized to run, the command is executed on the requested run host together with any optional rule components, for example, scripts to force additional authentication, or the user account to run the command as, such as root.

4.2 Deploying Command Control

The following modules are required for the command control feature with auditing functionality:

- Section 4.2.1, “Command Control Modules,” on page 53
- Section 4.2.2, “Auditing Modules,” on page 54
- Section 4.2.3, “Compliance Auditor Modules,” on page 54

For installation instructions, see Section 4.2.4, “Steps Required,” on page 54.

4.2.1 Command Control Modules

The command control feature comprises the following packages:

- Command Control Manager: holds the rule configuration and is responsible for validating user command requests.

- Command Control Agents: installed on machines where user commands are to be controlled/audited.
- Command Control Console: installed into the Framework Console and required to configure command control rules.

4.2.2 Auditing Modules

The auditing modules comprise the following packages:

- Audit Manager: acts as the repository for auditing information collected by the Framework.
- Reporting Console: installed into the Framework Console and required for viewing audit information.
- Command Reporting Console: installed into the main Reporting Console and required for viewing command control audit information.

4.2.3 Compliance Auditor Modules

The Compliance Auditor modules comprise the following packages:

- Compliance Auditor: holds the compliance auditor rules and audit information.
- Compliance Auditor Console: installed into the Framework Console and required for configuring compliance auditor rules and for viewing audit information.

4.2.4 Steps Required

To deploy command control:

1. Download the required packages to your local Package Manager. See Section 4.2, “Deploying Command Control,” on page 53 for details.
2. Install the Command Control Manager package on the host you want to be the Command Control Manager. This can be on any operating system including Windows. See Section 2.2.2, “Deploying Packages to Hosts,” on page 30 for details. Command Control Managers can be deployed to as many hosts as you need to build an environment with load balancing and failover.
3. Install the Command Control Agent package on all UNIX hosts on which you want to implement command control.
4. Install the Audit Manager package on the host you want to be the Audit Manager, and then install the Compliance Auditor package on the same host. This can be on any operating system including Windows, and can be a different host from your Command Control Manager. The auditing packages can be deployed to as many hosts as you need to build an environment with load balancing and failover.
5. Install the following consoles:
   - Command Control Console
   - Reporting Console
   - Command Reporting Console
   - Compliance Auditor Console

   See Section 2.3.2, “Adding Consoles to the Framework Console,” on page 34 for details.

Command control is now deployed and ready to use.

4.3 Integrating Command Control

Privileged User Manager provides a number of shells and functions which are installed as part of the Command Control Agent, including the command control shells rush and crush, based on the ksh shell structure. These shells and functions provide the following options for integrating command control into the UNIX and Linux user environments:

4.3.1 Simple Scripts, Aliases, and Functions

The simplest way of integrating the command control client rush is to implement shell scripts that are executed by the user to carry out the specific function. For example:

```
#!/ usr/bin/rush
set – o remote
ls – l
```

These scripts execute the rush client, set it to use command control, and execute the command. Users can then call these scripts to carry out their privileged tasks, in this case the execution of the `ls – l` command.

4.3.2 Using usrun before a Command

Type usrun before any command to pass the command to the Command Control Manager for authorization. You must define command control rules to control authorization of the commands for the appropriate users.

4.3.3 Complete Session Command Control Using rush

There are two ways of providing complete session command control using the rush client:

- Users can type:

  ```
  usrun rush
  ```

  at the start of the session. An authorization request will be sent to the Command Control Manager. You must define a Command Control rule to authorize `rush` for the appropriate users, and enable session capture. If required, you can define illegal commands, including built-in shell commands, in a script assigned to the rule.
- You can change the user’s logon shell to the rush client (no authorization request are sent when the user logs on). This provides an invisible method of integration to the user. To do this, use the tool provided in the UNIX or Linux environment to set it to:

```
/ usr/bin/rush
```

The rush client executes as a normal Korn Shell. Functions and aliases that replace normal system commands are read from /etc/profile.rush. When the user issues a command that needs privileges to run, it is authorized through the Framework.

If using the second option, you also need to perform the following steps:

1 To ensure configured commands are authorized at the Framework add the following line to either the user's .profile or to the central profile.rush in the /etc directory on the appropriate UNIX or Linux servers:

```
set - o remote
```

IMPORTANT: `set - o remote` forces all commands which are not built in to the user's shell to be authorized at the Framework. Commands for which a defined rule does not exist are not permitted to execute. To prevent all commands in the profile.rush or .profile from being passed to the Framework for authorization, add the `set - o remote` command at the end of the file.

---

2 You can set the following additional options:

To specify that all commands that are authorized are executed on the host defined if permitted, use:

```
set -o host <hostname>
```

To specify that all commands that are authorized are executed as the user defined if permitted, use:

```
set -o user <username>
```

NOTE: Rule definitions override these settings. If a successfully matched rule specifies a run user and/or a run host, this user and/or host is used to execute the command, and not those specified in the `set -o` commands.

You can use rule conditions to match the run user and/or run host to the username and/or hostname defined using these commands (see “Setting Conditions for a Rule” on page 67) but if a run user and/or run host is defined in the rule configuration, these are the ones that are used.

You can define a list of illegal commands, including built-in shell commands, in a script assigned to a rule. Users using the rush shell will not be able to run these commands, even if they are root.

4.3.4 Complete Session Capture Using crush

This method of integration provides the most auditing functionality. By changing the user's shell to the crush client, instead of the rush client, command control can be configured to capture the user's complete session, in addition to all other audit and control features.

When the user logs in to the server, the session is started using the crush client, which executes as a normal Korn Shell. A request is sent to the Command Control Manager for authorization, and you must define a "crush" rule which enables session capture, as described in the steps below. Functions

and aliases that can replace normal system commands are read from /etc/ profile.rush. When the user issues a command that needs privileges to run, it is executed through the command control system.

1. Use the tool provided in the UNIX or Linux environment to set the user logon shell to:

   ```
   / usr/bin/crush
   ```

2. Follow steps Step 1 on page 56 and Step 2 on page 57 above, or you can define the functions using a script assigned to a rule. For example, the following script will set up the set -o remote function:

   ```
   my $t=$meta->get_params('Ticket');
   if(! $t) {
   $t=$meta->add_param('Ticket');
   }
   $t->arg("remote","1");
   return 1;
   ```

3. Add a new rule called "crush" to the Framework (see "Adding a Rule" on page 65).
4. Set Session Capture to On for the rule (see "Modifying a Rule" on page 66).
5. Add the Run User and Run Host (see "Modifying a Rule" on page 66).
6. Add a new command called, for example, "Crush shell" (see "Adding a Command" on page 77).
7. Set the Command to "r;-crush" (see "Modifying a Command" on page 77).
8. Add the command to the rule conditions (see "Setting Conditions for a Rule" on page 67).
9. Add the submit user and host groups to the rule conditions (see "Setting Conditions for a Rule" on page 67).

4.3.5 Session Auditing

As well as providing command control functionality, the rush and crush clients can also audit every session command that the user types, whether run locally or authorized at the Framework. Commands which are built in to the user's shell are audited, and if you use the crush shell these commands are included in the session capture data.

The session auditing option is used in conjunction with the Command Risk list (see "Setting the Command Risk" on page 78).

1. To enable commands which have not been successfully authorized at the Framework to execute according to the local permissions in effect on the server where the command was issued, include the following line either in the user's `.profile` or in the central `profile.rush` in the `/etc` directory on the selected UNIX or Linux servers:

   ```
   set - o ignoreperm
   ```

2. To enable the audit function, include the following line either in the user's .profile or in the central profile.rush in the /etc directory on the selected UNIX or Linux servers:

   ```
   set - o audit 1
   ```

   The above command provides standard auditing. If you want to include built-in commands in the audit, use:

   ```
   set - o audit 2
   ```

You should note that using option 2 might affect login performance when using the command control shells.

NOTE: For security reasons, once enabled this function cannot be disabled within a user session.

If you are using the crush client, you can set up these functions using a script, as described in step Step 2 on page 57 in the previous section. For example:

```
my $t=$meta->get_params('Ticket');
if(! $t) {
$t=$meta->add_param('Ticket');
}
$t->arg("audit","1");
return 1;
```

You can also set the value to 0 so that the option is set to off.

4.3.6 Advanced Functions

The following advanced function is available:

- udsh command - invokes commands on a set of hosts concurrently.

4.4 Configuring Command Control

4.4.1 Configuration Overview

Command control uses rules to protect and control the use of user commands. When configuring a rule, you need to set rule conditions to determine which rule or rules are processed, depending, for example, on the command submitted or the user who submitted it. You also need to define what processing to do if the rule conditions are matched.

The components you can define and configure for a rule are as follows:

- The rule itself
- Account groups, which combine user and host groups
- User groups, containing submit users or run users
- Host groups, containing submit hosts or run hosts
- Commands
- Scripts, for additional functionality
- Access times, which define specific times during which access is denied or granted.

When you have defined and configured the rule components, you can then:

- Set the rule conditions for matching
- Assign scripts to the rule
- Configure script arguments and entities
- View the pseudocode for the rule, which is a simplified representation of the actual code that is processed when the rule is activated.

Refer to the appropriate topic for detailed descriptions.

NOTE: To enable access to the Command Control console for a Framework user and to control the level of access available, you must add the user to a group with the appropriate roles defined. See Section 3.4.3, "Configuring Roles," on page 48 for details.

Additional Options

The following additional options are provided to assist you with command control configuration:

- Sample commands and Perl scripts
- The ability to import and export Command Control configuration data
- Test suites for testing your rules
- Reporting facilities for extracting and emailing information about your database
- The ability to search for rules and rule components in your database
- The ability to disable components temporarily, rather than remove them altogether
- The ability to group rule components into categories for ease of use and maintenance.

Refer to the appropriate topic for detailed descriptions.

Audit Settings

There are various audit settings that can be configured for use with auditing and reporting facilities. You can:

- Use the *Audit Settings* option to enable encryption of sensitive password data in keystroke capture reports, and set various additional options which can be audited for each event

- Set a risk value for specific commands and rules that will be indicated in reports using different colors
- Define audit groups for rules for use by the Compliance Auditor.

Refer to the appropriate topic for detailed descriptions.

Command Control Transactions

Your Command Control database can optionally be protected through the use of the transactions feature. See "Command Control Transactions" on page 60 for further details.

4.4.2 Command Control Transactions

Your command control database can be protected through the use of the transactions feature, which automatically locks the database when you start making changes, preventing other Framework users from making any changes. You must then commit the transaction to save the changes and release the lock, and you are prompted by customized questions to provide information which can be viewed in the Compliance Auditor. You can cancel the transaction at any time.

To use this feature, you must first enable it and create a customized *Commit Transactions* screen as described in Transaction Settings.

To make command control configuration changes with transactions enabled:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Command Control* from the *Navigation Pane*.
3. Make the required configuration changes. A message will appear next to *Command Control* in the *Navigation Pane* to indicate that the Command Control database is locked, by whom, and when it was locked.
4. Select *Command Control* from the *Navigation Pane* and select *Commit Transaction*. Complete the fields as set up on the *Transaction Settings* screen and select *Finish*.

   Alternatively, if you do not want to keep the changes you have made, select *Cancel Transaction* from the *Task Pane* and select *Yes* to confirm. Any changes you have made since the database was locked are removed.

Transaction Settings

This option allows you to configure the Command Control Manager to require the transactions feature to be used when configuring command control rules. See "Command Control Transactions" on page 60 for details of how this feature is used.

You can configure your own Commit Transaction screen to be completed when a user commits a transaction, as described below. The data entered on the Commit Transaction screen can be viewed in the Compliance Auditor.

To configure this feature:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page
2. Select *Transaction Settings* from the *Task Pane*.
3. Check the *Enable transactions* checkbox to enable the use of command control transactions.
4. Select *Add*.

5 Enter a name for the field you want to display on the screen when a user commits a transaction. For example, you might want to request the user's name, so you could enter Name. A blank field called Name appears on the screen when a user commits a transaction.

6 Select *Text* if you want the user to enter one line of text, or TextArea if you want the user to be able to enter several lines of text.

7 Select *required* if you want to force the user to enter text in this field. The *Finish* button on the *Commit Transaction* screen does not become available until the user has entered text in this field.

8 Repeat Step 4 through Step 7 for any other fields you want to display when the user commits the transaction.

9 Select *Finish*.

Committing a Transaction

When you have completed the changes you require to your command control database, you must commit your transaction to save the changes and release the lock on the database. The *Commit Transaction* screen can be customized to request whatever information you require when a transaction is committed (see "Transaction Settings" on page 60 for details).

To commit a transaction:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page

2 Select *Commit Transaction* from the *Task Pane*.

3 Complete the customized fields according to company policies.

4 Select *Finish*.

4.4.3 Defining Audit Settings

The *Audit Settings* option allows you to configure the following:

- Enable encryption of sensitive password data in keystroke capture reports, and define a password that allows authorized Framework administrators to decrypt it
- Set various additional options which can be audited for each event.

To define audit settings:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Audit Settings* from the *Task Pane*.

4 Enable the *Password filter* field using the check box.

5 Enter the text that is used to prompt users for their passwords. For example, if your systems request a user's password using the word `Password`, enter `Password` in this field. If your systems use `password`, enter `password` in this field. If your systems use either, enter `assword` in this field. This ensures that the password the user enters in response to this prompt is encrypted in reports.

6 Enable the *Encryption password* field using the check box.

7 Enter the password to be used to decrypt the sensitive password data in the report. This password must be entered on the *Command Control Keystroke Report* screen to decrypt the password data.

8 Re-enter the password in the *Confirm password* field.

9 If required, set the additional audit options you require from the *MetaData audit settings* by checking the appropriate check boxes.

10 Select *Finish*.

4.4.4 Importing and Exporting Settings

- "Importing Command Control Settings" on page 62
- "Exporting Command Control Settings" on page 62
- "Importing Command Control Samples" on page 63

Importing Command Control Settings

Using the Import Settings option, you can restore a previously backed up version of your command control configuration settings, or import command control configuration settings from another Framework. Configuration settings are obtained using the Export Settings option, and you can then paste them into a text document for backup or for use on another Framework.

IMPORTANT: This process overwrites your existing configuration settings.

---

1 Access the Command Control configuration settings you require and copy the whole configuration.

2 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

3 Select *Command Control* from the *Navigation Pane*.

4 Select *Import Settings* from the *Task Pane*.

5 Click in the text area then paste in the copied settings using Ctrl+V, or right-click in the text area and select *Paste*.

6 Select *Finish*.

Exporting Command Control Settings

You can export your command control configuration settings to a text file for backup purposes, or for use in another Framework. To do this you must copy the configuration information from your Framework using the Export Settings option, and paste it into a text file. You can then use the Import Settings option to restore the backed up configuration settings, or to import the settings into another Framework.

NOTE: Novell recommends that you take frequent backups of your command control configuration settings in case of loss.

---

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Export Settings* from the *Task Pane*.

4 Select all your Command Control configuration settings using Ctrl+A, or right-click in the text window and select *Select All*.

5 Copy the settings using Ctrl+C, or right-click in the text window and select *Copy*.

6 Paste the text into a text document and save it.

7 Select *Finish*.

Importing Command Control Samples

Novell has provided a set of sample commands and Perl scripts to assist you with configuring your command control rules.

To add these samples to your configuration:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Import Samples* from the *Task Pane*.

4 Select the samples you require.

TIP: To select multiple samples in a folder, display the samples then hold down the Ctrl key and select the required samples one at a time, or hold down the Shift key to select a consecutive list of samples. You cannot import samples by selecting a folder.

---

5 Select *Finish*. The samples are added to the appropriate section of the configuration.

4.4.5 Categories

- "Adding a Category" on page 63
- "Deleting a Category" on page 63

Adding a Category

You can group your account groups, user groups, host groups, commands, scripts, and access times into categories for ease of use and maintenance, using the appropriate Add Category option.

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select the section to which you want to add a category. You can also add sub-categories to existing categories.

3 Select the *Add Category* option from the *Task Pane*.

4 Enter a name for the category.

5 Select *Finish*.

Deleting a Category

Before deleting a category you must delete or move the items and sub-categories which it contains.

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select the category you want to delete.

3 Select the *Delete Category* option from the *Task Pane*. The category is deleted.

4.4.6 Finding a Reference

The *Find References* option allows you to find where in the database a specific account group, user group, host group, command, script, or access time is referenced. For example, you could use this option to find out which account group or groups a specific user group belongs to.

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select the entity to which you want to find references.
3. Select *Find References* from the *Task Pane*. The groups and/or rules in which the entity is referenced are displayed.
4. To go to one of the listed groups or rules, double-click it. Alternatively select *Close*.

4.4.7 Defining Custom Attributes

Custom attributes can be defined for account groups, user groups, host groups, commands and access times to provide additional parameters for use in scripts. For example, you could set an expiration date as a custom attribute for a user group, check for this date in your script, and then expire the user group when the date is reached.

To define custom attributes:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select the entity you want to add custom attributes to.
3. Select *Custom Attributes* from the *Task Pane*.
4. Select *Add*.
5. In the *Name* field, type the name of the custom attribute, for example, Expiration date.
6. In the *Value* field, type the value for the attribute, for example, the date you want the entity to expire.
7. Repeat Step 4 through Step 6 for any other custom attributes you want to add.
8. Select *Finish*.

4.4.8 Rules

- “Viewing Pseudocode” on page 71
- “Deleting a Rule” on page 71

Adding a Rule

Rules provide the means by which you can control commands using command control. Commands can be authorized to run, or not authorized to run, by setting rule conditions based on:

- The command being submitted
- The user and host submitting the command
- The user and host assigned to run the command
- The time the command is being submitted
- The contents of Perl scripts you have defined.

See “Setting Conditions for a Rule” on page 67 for details.

If a rule's conditions are met, there are a number of options you can set to determine how the rule processes the command. You can configure a rule to:

- Display a message to the user submitting the command
- Capture the user session for reporting and auditing purposes
- Authorize or not authorize the command to be run
- Specify what further rule processing to do. For example, if the command is authorized, the rule could specify that processing of further rules can stop.

You can also:

- Specify the user and host to run the command
- Set a risk level for use with keystroke reports
- Assign an audit group to the rule for use with the Compliance Auditor.

See “Modifying a Rule” on page 66 for details.

You can also create and assign Perl scripts to the rule to provide additional functionality (see “Adding a Script” on page 80 and “Assigning a Script to a Rule” on page 68 for details).

NOTE: If you are using a different user (run user) to run an authorized command from the user who submitted the command (submit user), by default the submit user's environment variables are used for the run user. If you want to use the environment variables associated with the run user, you can add a script to your rule containing the following text:

```
$meta->get_params("Job")->arg("job_default_env",0);
return 1;
```

To add a new rule:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Rules* from the *Navigation Pane*.
3. To add a rule at the top level, select *Add Rule* from the *Task Pane*. To add a rule as a child of another rule, select the rule and select *Add Rule* from the *Task Pane.*

4 Enter a name for the rule.

5 Select *Finish*. The new rule is added to the bottom of the list.

6 Move the rule to the correct position according to the order in which you want to process your rules.

When a user issues a command under command control, the following rule processing takes place:

The conditions set for the first rule in the hierarchy are checked.

If there is a match The rule is processed. Depending on how the rule is configured, processing of further rules takes place or stops. If rule processing is not stopped, the next rule for which conditions are checked is the child of this rule. Rule checking and processing continues until stopped by a rule, or all appropriate rules have been processed.

If there is no match The conditions for the next rule at the same hierarchical level as the first rule are checked, and this continues until a match is found. Rule processing then takes place as described above.

You can change the default order of rule processing on the *Modify Rule* screen, or using scripts (see "Modifying a Script" on page 80).

7 Configure the rule using the *Modify Rule* option.

Modifying a Rule

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Select the rule you want to modify.

4 Select *Modify Rule* from the *Task Pane*.

5 You can:

- Change the *Name* of the rule.
- Disable the rule by checking the *Disabled* box. A disabled rule is shown grayed out.
- Enter a *Description* of the rule.
- Enter a *User Message* to be displayed to the user when this rule is processed, before any commands are run.
- Set *Session Capture* to *On* or *Off*. Setting *Session Capture* to *On* will allow the Audit Manager to perform keystroke logging for the rule. To view a captured session from a command control report, an Auditing Manager and the Reporting Console must be installed.
- Set *Authorize* to *Yes* or *No*, depending on whether you want the command protected by the rule to be authorized or not authorized if the rule conditions are met.

  You can define what happens next if the command is authorized, or not authorized, using the adjacent drop down list, as follows:

  - **Blank:** the next rule in the hierarchy is checked

    Stop: no more rules are checked for the command

    Return: the next rule to be checked is up one level in the hierarchy from the current rule

Stop if authorized: no more rules are checked for the command if Authorize is set to Yes

Stop if unauthorized: no more rules are checked for the command if Authorize is set to No

- Define a *Run User*: the name of the user you want to run this command (this overrides any username defined using a set command)
- Define a *Run Host*: the name of the host on which you want to run this command (this overrides any hostname defined using a set command)
- Set a *Risk Level* of 0 to 99. This option allows you to set a value representing the relative risk of a rule when using the rush or crush clients with the session auditing option (see Section 4.3, "Integrating Command Control," on page 55). When viewing a Command Control Keystroke Report, you see commands controlled by rules with different risk values represented in different colors.
- Define an *Audit Group*. This setting is for use in Compliance Auditor reports.

6 Select *Finish*. The settings you have defined for the rule are displayed in the console.

Setting Conditions for a Rule

You can set a number of conditions for a rule which determine whether the rule is processed or not. A simple example is that you can set a particular command as a condition, and only process the rule if any user enters that command.

There are two ways of setting conditions for a rule:

- Dragging an entity onto the rule
- Using the *Edit Condition* option, as described in the steps below.

NOTE: When you drag an entity onto a rule, you might need to edit the condition to ensure that the condition logic is as you require. If you want to use a script in rule conditions, you must set it to Conditional first (see "Modifying a Script" on page 80).

To set conditions using the *Edit Condition* option:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Display the rules in the left hand pane and select the rule for which you want to set conditions.

4 Select the currently defined condition in the right hand pane. If you have not yet defined a condition, this is *Match All*.

5 Select *Edit Condition* from the *Task Pane*.

6 Select the type of condition you require from the *Add Condition* drop-down list. The condition is displayed on the screen.

7 Set the condition to the value and logic you require. For example, if setting a condition to match a run user to a user group:

7a Change *user* (submit user) to *run user*.

7b Leave the logic setting as *IN*.

7c Select the user group you require from the user group drop-down list.

8 Repeat Step 6 and Step 7 for any other conditions you require to be set. Set the condition logic as required.

   You can use parentheses to group conditions according to the required logic. To do this:

   Select the parentheses ( ) entry from the *Add Condition* drop down list. The opening and closing parentheses are displayed.

   8a Select the opening parenthesis.

   8b Select the condition type you want to place inside the parentheses and set it as required.

   8c Select the opening parenthesis again.

   8d Select another condition type to place inside the parentheses and set it as required.

   8e If required, change OR to AND.

   8f Repeat Step 8d through Step 8f for any other conditions you require inside this set of parentheses. You can also place parentheses within parentheses.

9 Select *Finish*.

Assigning a Script to a Rule

You can use Perl scripts to provide additional, customized functionality to your rules (see "Adding a Script" on page 80). To assign a script to a rule, use drag and drop as described below.

NOTE: If you drag a script that has been set to Conditional, the script will be added to the rule conditions.

---

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Display the rules in the left hand pane and display the rule to which you want to assign a script.

4 Select *Scripts* from the *Navigation Pane*

5 Select the script you want to assign to the rule.

   TIP: To select multiple scripts in the same category, hold down the Ctrl key and select the required scripts one at a time, or hold down the Shift key to select a consecutive list of scripts.

   ---

6 Drag the selected scripts and drop onto the rule.

7 Configure script arguments and entities for the scripts if required.

Configuring Script Arguments and Entities for a Rule

You can configure script arguments and entities for the scripts assigned to a rule, before or after assigning the scripts. You can define only one set of arguments and entities which applies to all scripts assigned to a rule.

1 Select *Command Control* from the Navigation Pane on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Select the rule for which you want to add script arguments.

4 Select *Script Arguments* from the *Task Pane*.

To add script arguments:

5 Select *Add*.

6 In the *Name* field, enter a name for the argument.

7 In the *Value* field, enter a value for the argument.

8 To add more arguments, repeat Step 5 through Step 7.

9 Select *Finish*, or to add script entities go to Step 10.

To add script entities:

10 Click the arrow under *Add Script Entity* to display the list of available entities and select the one you require. A drop-down list will be displayed in the *Script Entities* table.

11 Select the entity you require from the drop-down list.

12 To add more entities, repeat Step 10 and Step 11.

13 Select *Finish*.

Removing Script Arguments and Entities

To remove a script argument, select the required argument and select *Remove*.

To remove a script entity, select the icon next to the name of the entity and select *Remove*.

Finding a Rule

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 To find a rule from the whole list of rules, select *Find Rule* from the *Task Pane*. Alternatively, to find a rule in a set of rules, select the parent rule and then select *Find Rule*.

4 In the *Rule Filter* field, type the name of the rule you are looking for and select *Find*. You can use wildcard characters * and ?, for example, rul* will find the first rule beginning with rul. This field is case sensitive.

5 If the rule name you are looking for is displayed, double-click it and you are returned to the *Navigation Pane* with the rule selected. Alternatively, select *Close*.

Moving a Rule

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Select the rule you want to move.

TIP: To select multiple rules in the same group, make sure the rules are displayed in the right hand pane of the Navigation Pane, then hold down the Ctrl key and select the required rules one at a time, or hold down the Shift key to select a consecutive list of rules.

---

4 Drag the selected rules to the required location.

Copying a Rule

You can create a copy of an existing rule in your rule hierarchy, either so you can use the same rule in more than one place in the hierarchy, or so you can create a new rule based on your existing rule.

NOTE: If you want to use the same rule in more than one place and you want any changes you make to the rule to be reflected in the other copy or copies, you should link the rule instead.

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Rules* from the *Navigation Pane*.
3. Select the rule you want to copy.

   TIP: To select multiple rules in the same group, make sure the rules are displayed in the right hand pane of the Navigation Pane, then hold down the Ctrl key and select the required rules one at a time, or hold down the Shift key to select a consecutive list of rules.
4. Hold down the Ctrl key and drag the selected rules to the required location to create a copy.
5. If required, rename and/or modify the copy using the *Modify Rule* option.
6. Move the rule to the correct position according to the order in which you want to process your rules (see "Adding a Rule" on page 65 for details).

Linking a Rule

If you require a specific rule to be used in different places in your rules hierarchy, you can create a linked rule. Any changes you make to the linked rule are reflected in all the instances of the rule in your hierarchy. If you simply copied the rule, any changes made to the original rule or to one of its copies would not be reflected in the other copies.

Changes to sub rules of a linked rule are not linked. For example if you add or modify a rule under a linked rule, the change is not reflected in other instances of the linked rule.

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Rules* from the *Navigation Pane*.
3. Select the rule you want to link.

   TIP: To select multiple rules in the same group, make sure the rules are displayed in the right hand pane of the Navigation Pane, then hold down the Ctrl key and select the required rules one at a time, or hold down the Shift key to select a consecutive list of rules.
4. Hold down the Ctrl key and the Shift key together, and drag the selected rules to the required location to create links. A linked rule is shown with an arrow  .

Removing Conditions for a Rule

You can remove all the conditions set for a rule, or you can remove individual conditions using the Edit Condition option.

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Rules* from the *Navigation Pane*.
3. Display the rules in the left hand pane and select the rule for which you want to set conditions.
4. Select the currently defined condition in the right hand pane.

To remove all conditions:

5. Select *Remove Conditions* from the *Task Pane*.

6 Select *Yes* to remove all the conditions. The rule condition is returned to *Match All*.

To remove individual conditions:

7 Select *Edit Condition* from the *Task Pane*.

8 Remove individual conditions by selecting the appropriate button on the left.

9 Select *Finish*.

Removing a Script from a Rule

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Display the rules in the left hand pane and select the rule from which you want to remove a script.

4 Select the script you want to remove in the right hand pane.

TIP: To select multiple scripts hold down the Ctrl key and select the required scripts one at a time, or hold down the Shift key to select a consecutive list of scripts.

---

5 Select Remove *Script* from the *Task Pane*.

6 Select *Yes* to confirm the removal. The scripts are removed from the rule.

Viewing Pseudocode

The pseudocode for a rule provides a simplified representation of the actual code that is processed when the rule is activated. For complex rules, this can assist you with understanding what happens in different situations.

To view the pseudocode for a rule:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Select the rule for which you want to view the pseudocode.

4 Select *Pseudocode* from the *Task Pane*.

TIP: If required, you can copy the pseudocode using Ctrl+A then Ctrl+C, and paste it into a document for printing.

---

5 Select *Close*.

Deleting a Rule

The Delete Rules option deletes the selected rule and all the rule children.

1 Select *Command Control* from the Navigation Pane on the Framework Console home page.

2 Select *Rules* from the *Navigation Pane*.

3 Select the rule you want to delete.

TIP: To select multiple rules in the same group, make sure the rules are displayed in the right hand pane of the Navigation Pane, then hold down the Ctrl key and select the required rules one at a time, or hold down the Shift key to select a consecutive list of rules.

4 Select *Delete Rules* from the *Task Pane*.

5 Select *Finish* to delete the rule and all rule children.

4.4.9 Account Groups

Adding an Account Group

Account groups can be used to combine host groups and user groups to be used together in setting rule conditions. Account groups can also contain other account groups. You can also use account groups as script entities.

To add a new account group:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Account Groups* from the *Navigation Pane*.

3 To add an account group at the top level, select Add Account Group from the *Task Pane*. To add an account group to a category, select the category and select *Add Account Group* from the *Task Pane.*

4 Enter a name for the account group.

5 Select *Finish*. Account groups are represented by this icon  . Configure the account group using the *Modify Account Group* option.

Modifying an Account Group

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Account Groups* from the *Navigation Pane*.

3 Select the account group you want to modify.

4 Select *Modify Account Group* from the *Task Pane*. You can:

- Change the *Name* of the group.
- Disable the account group by checking *Disabled*. A disabled account group is shown grayed out.
- Add or change the *Description*.

- Enter the name, telephone number and email address of the manager of the users in this account group, if required. The manager details can be used in the Compliance Auditor.

  TIP: If these details have been entered in the manager's Framework user account details (see "Modify User: Account Details" on page 43), they can be entered automatically by selecting the manager's user name from the drop-down list. This option is only available if you belong to a Framework user group with the *read* role defined for the auth module (see Section 3.4.3, "Configuring Roles," on page 48).

- Select or remove the *User Groups*, *Host Groups* and *Account Groups* to be included in this account group from the lists of groups you have already defined. You can also add groups to an account group by dragging the groups to the account group in the *Navigation Pane*.

5 Select *Finish*. You can now use this account group in rule conditions, and/or as a script entity.

Deleting an Account Group

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Account Groups* from the *Navigation Pane*.

3 Select the required account group.

TIP: To select multiple account groups, display the groups in the right hand pane, hold down the Ctrl key and select the required account groups one at a time, or hold down the Shift key to select a consecutive list of account groups.

4 Select *Delete Account Group* from the *Task Pane*. The selected account groups are listed.

5 Select *Finish*. The account groups will be deleted, and will also be removed from any other account group, rule conditions and script entities in which they have been defined.

User Groups

Adding a User Group

User groups contain users who are allowed, or not allowed, to submit or run commands controlled by your rules. You can add user groups to your rule conditions to control whether the rule is processed, depending on the user who is submitting a command or the user who is specified to run a command. You can also use user groups as script entities.

To add a new user group:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Account Groups* and then *User Groups* from the *Navigation Pane*.

3 To add a user group at the top level, select *Add User Group* from the *Task Pane*. To add a user group to a category, select the category and select *Add User Group* from the *Task Pane.*

4 Enter a name for the user group.

5 Select *Finish*. User groups are represented by this icon  . Configure the user group using the *Modify User Group* option.

Modifying a User Group

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Groups* and then *User Groups* from the *Navigation Pane*.
3. Select the user group you want to modify.
4. Select *Modify User Group* from the *Task Pane*. You can:
   - Change the *Name* of the group.
   - Disable the user group by checking *Disabled*. A disabled user group is shown grayed out.
   - Add or change the *Description*.
   - Enter the name, telephone number and email address of the manager of this user group, if required. The manager details can be used in the Compliance Auditor.

     TIP: If these details have been entered in the manager's Framework user account details (see "Modify User: Account Details" on page 43), they can be entered automatically by selecting the manager's user name from the drop-down list. This option is only available if you belong to a Framework user group with the 'read' role defined for the 'auth' module (see Section 3.4.3, "Configuring Roles," on page 48).

   - Add or change the *Users* you want to include in this group. You can type the user names, one on each line, or paste them from elsewhere. You can use the *Sort* button to sort the list of users into alphabetical order.
   - Select the *User Groups* you want to include as sub groups of this user group from the list of groups you have already defined. You can also add sub user groups to a user group by dragging the groups to the user group in the *Navigation Pane*.
5. Select *Finish*. You can now use this user group in rule conditions, and/or as a script entity.

Deleting a User Group

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Groups* and then *User Groups* from the *Navigation Pane*.
3. Select the required user group.

   TIP: To select multiple user groups, hold down the Ctrl key and select the required user groups one at a time, or hold down the Shift key to select a consecutive list of user groups.

   ---

4. Select *Delete User Group* from the *Task Pane*. The selected user groups are listed.
5. Select *Finish*. The user groups will be deleted, and will also be removed from any account group, rule conditions and script entities in which they have been defined.

Host Groups

Adding a Host Group

Host groups contain hosts that are allowed, or not allowed, to submit or run commands controlled by your rules. You can add host groups to your rule conditions to control whether the rule is processed, depending on the host that is submitting a command or the host specified to run a command. You can also use host groups as script entities.

To add a new host group:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Groups* and then *Host Groups* from the *Navigation Pane*.
3. To add a host group at the top level, select *Add Host Group* from the *Task Pane*. To add a host group to a category, select the category and select *Add Host Group* from the *Task Pane.*
4. Enter a name for the host group.
5. Select *Finish*. Host groups are represented by this icon  . Configure the host group using the *Modify Host Group* option.

Modifying a Host Group

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Groups* and then *Host Groups* from the *Navigation Pane*.
3. Select the host group you want to modify.
4. Select *Modify Host Group* from the *Task Pane*. You can:

   Change the *Name* of the group.

   Disable the host group by checking *Disabled*. A disabled host group is shown grayed out.

   Add or change the *Description*.

   Add or change the *Hosts* you want to include in this group. You can type the host names, one on each line, or paste them from elsewhere. You can use the *Sort* button to sort the list of hosts into alphabetical order.

   Select the Host Groups you want to include as sub groups of this host group from the list of groups you have already defined. You can also add sub host groups to a host group by dragging the groups to the host group in the Navigation Pane.
5. Select Finish. You can now use this host group in rule conditions, and/or as a script entity.

Deleting a Host Group

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Groups* and then *Host Groups* from the *Navigation Pane*.
3. Select the required host group.

   TIP: To select multiple host groups, hold down the Ctrl key and select the required host groups one at a time, or hold down the Shift key to select a consecutive list of host groups.

   ---
4. Select *Delete Host Group* from the *Task Pane*. The selected host groups are listed.
5. Select *Finish*. The host groups will be deleted, and will also be removed from any account group, rule conditions, and script entities in which they have been defined.

Copying a Group

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Groups* from the *Navigation Pane*. If you are copying a host group, select *Host Groups*. If you are copying a user group, select *User Groups*.
3. Select the group you want to copy.

   TIP: To select multiple groups in the same category or group, make sure the groups are displayed in the right hand pane of the Navigation Pane, then hold down the Ctrl key and select the required groups one at a time, or hold down the Shift key to select a consecutive list of groups.
4. Hold down the Ctrl key and drag the selected groups to the required location to create a copy.
5. If required, rename and/or modify the copy using the appropriate *Modify Group* option.

Moving a Group

You can move a group from one category to another by dragging.

You can also drag account groups, user groups and host groups into an account group (this does not delete the groups from their original place). You can also add groups to account groups using the Modify Account Group option.

To move a group:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Groups* from the *Navigation Pane*. If you are moving a host group, select *Host Groups*. If you are moving a user group, select *User Groups*.
3. Select the group you want to move.

   TIP: To select multiple groups in the same category or group, make sure the groups are displayed in the right hand pane of the Navigation Pane, then hold down the Ctrl key and select the required groups one at a time, or hold down the Shift key to select a consecutive list of groups.
4. Drag the selected groups to the required location.

4.4.10 Commands

Adding a Command

Command definitions contain the commands you want to control using the command control. A command definition can contain a single command, or several commands that you want to control in the same way. You can also specify a command that you want to run in place of a submitted command.

You can add command definitions to your rule conditions to control whether the rule is processed, depending on the command that has been submitted by the user. You can also use commands as script entities.

To add a new command:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Commands* from the *Navigation Pane*.
3. To add a command at the top level, select *Add Command* from the *Task Pane*. To add a command to a category, select the category and select *Add Command* from the *Task Pane.*
4. Enter a name for the command. This can be different from the name of the actual command you want to control.
5. Select *Finish*. Configure the command using the *Modify Command* option.

Modifying a Command

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Commands* from the *Navigation Pane*.
3. Select the command you want to modify.
4. Select *Modify Command* from the *Task Pane*.
5. You can:
   - Change the *Name* of the command.
   - Disable the command by checking *Disabled*. A disabled command is shown grayed out.
   - Enter a *Description* of the command.
   - Define a command in the *Rewrite* field which will be used in place of the commands listed in the *Command* field below. You can also enter command arguments. Positional parameters can be used, as described in "Command Rewrite Examples" on page 78.
   - Define one or more commands in the *Command* field that could be entered by users, one on each line. You can also enter command arguments. For example:

     ```
     vi *
     /usr/bin/vi *
     ```

     You can copy and paste a list of commands from elsewhere. You can use the *Sort* button to sort the commands into alphabetical order.
   - Select the *Sub Commands* you want to include in this command definition from the list of command definitions you have already defined. You can also add sub commands to a command definition by dragging them to the command definition in the *Navigation Pane*.
6. Select *Finish*.

Command Rewrite Examples

The following table provides examples showing how the command rewrite functionality, provided on the Modify Command screen, can be used with positional parameters to replace the submitted command and parameters. The examples use the echo command as the rewritten command, to display the selected parameters on the screen.

Function	Rewrite	Submitted Command	Executed Command
Insert all arguments (note $0 is not displayed)	echo $*	ls passwd shadow fstab	echo passwd shadow fstab
Insert argument ’r;n’	echo $3	ls passwd shadow fstab	echo fstab
Insert all BUT argument 'n' (note $0 is not displayed)	echo ${^2}	ls passwd shadow fstab	echo passwd fstab
Insert arguments from 'n' to end	echo ${2-}	ls passwd shadow fstab	echo shadow fstab
Insert arguments from 0 to 'n'	echo ${-2}	ls passwd shadow fstab	echo ls passwd shadow
Insert arguments from 'm' to 'n'	echo ${1-2}	ls passwd shadow fstab	echo passwd shadow
Insert the total number of arguments	echo $#	ls passwd shadow fstab	echo 3
Insert contents of argument $#	echo ${$#}	ls passwd shadow fstab	echo fstab

Example using ufsdump

In this example, the administrator usually does a backup of the system using the following command:

```
ufsdump -0f /dev/rmt/0 /usr
```

A new tape drive is then installed on the host and this must be used instead. Also the administrator must make sure that it is working correctly by using the -v flag to verify the tape.

To avoid the administrator having to remember the changes, the following can be entered in the Rewrite field in a command definition for the above command:

```
$0 -v $1 /dev/rmt/1 ${$#}
```

When the administrator enters the usual command, the following command will be run instead:

```
ufsdump -v -0f /dev/rmt/1 /usr
```

Setting the Command Risk

This option allows you to set a value representing the relative risk of a command when using the rush or crush clients with the session auditing option (see Section 4.3, “Integrating Command Control,” on page 55). When you view a Command Control Keystroke Report, the commands with different risk values are represented in different colors.

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Commands* from the *Navigation Pane*.
3. Select *Command Risk* from the *Task Pane*.

4. Select *Add*.
5. Set a value for the command *Risk*.
6. To use a regular expression instead of a simple pattern match on the command, select the *Regex* checkbox.
7. Type the *Command* you want to set a risk value for, or the regular expression. You can use wildcard symbols.
8. If you want to base the risk level on the directory in which the command is running, define a *Working Directory*.
9. If you want to base the risk level on who is running the command, define a *User*.
10. If you want to base the risk level on the host where the command is running, define a *Host*.
11. Use the arrow buttons to change the order in which the commands are listed, if required.
12. Select *Finish*.

Removing a Command Risk

To remove a command risk setting, select the required entry and select *Remove*.

Copying a Command

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Commands* from the *Navigation Pane*.
3. Select the command you want to copy.

   TIP: To select multiple commands in the same category, hold down the Ctrl key and select the required commands one at a time, or hold down the Shift key to select a consecutive list of commands.
4. Hold down the Ctrl key and drag the selected commands to the required location to create a copy.
5. If required, rename and/or modify the copy using the *Modify Command* option.

Moving a Command

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Commands* from the *Navigation Pane*.
3. Select the command you want to move.

   TIP: To select multiple commands in the same category, hold down the Ctrl key and select the required commands one at a time, or hold down the Shift key to select a consecutive list of commands.
4. Drag the selected commands to the required location.

Deleting a Command

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Commands* from the *Navigation Pane*.
3. Select the required command.

TIP: To select multiple commands in the same category, hold down the Ctrl key and select the required commands one at a time, or hold down the Shift key to select a consecutive list of commands.

4 Select *Delete Command* from the *Task Pane*. The selected commands are listed.

5 Select *Finish*. The commands will be deleted, and will also be removed from any rule conditions and script entities in which they have been defined.

4.4.11 Scripts

Adding a Script

You can use Perl scripts to provide additional, customized functionality to your rules. For example, you could create a Perl script that displays on the screen the name of the run user who executes the command (see the sample provided called 'Display the Run User Name'). You can also use scripts in rule conditions.

You can add your own custom attributes for account groups, user groups, host groups, commands, and access times to provide additional parameters for use in your scripts. See "Defining Custom Attributes" on page 64 for details.

For further details about creating your own scripts for use with command control rules, contact PrivilegedUserSupport@novell.com.

To add a new script:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Scripts* from the *Navigation Pane*.

3 To add a script at the top level, select *Add Script* from the *Task Pane*. To add a script to a category, select the category and select *Add Script* from the *Task Pane.*

4 Enter a name for the script.

5 Select *Finish*. Configure the script using the *Modify Script* option.

Modifying a Script

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Scripts* from the *Navigation Pane*.

3 Select the script you want to modify.

4 Select *Modify Script* from the *Task Pane*.

5 You can:

- Change the *Name* of the script.

- Set the script to be conditional by checking *Conditional script*. Scripts defined as conditional can be used in rule conditions. The return codes are limited to 1 for true and 0 for false.
- Disable the script by checking *Disabled*. A disabled script is shown grayed out.
- Enter the text of your script in the Script field by typing, or pasting from elsewhere. The return codes you can return from your script for processing by the command control software are shown below this field. Contact PrivilegedUserSupport@novell.com for details about creating your own scripts.

6 Select *Finish*. You can now assign your script to a rule, and/or, if you have set this script as conditional, you can specify it in rule conditions.

Copying a Script

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Scripts* from the *Navigation Pane*.

3 Select the script you want to copy.

TIP: To select multiple scripts in the same category, hold down the Ctrl key and select the required scripts one at a time, or hold down the Shift key to select a consecutive list of scripts.

4 Hold down the Ctrl key and drag the selected scripts to the required location to create a copy.

5 If required, rename and/or modify the copy using the *Modify Script* option.

Moving a Script

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Scripts* from the *Navigation Pane*.

3 Select the script you want to move.

TIP: To select multiple scripts in the same category, hold down the Ctrl key and select the required scripts one at a time, or hold down the Shift key to select a consecutive list of scripts.

4 Drag the selected scripts to the required location.

Deleting a Script

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Scripts* from the *Navigation Pane*.

3 Select the script you want to delete.

TIP: To select multiple scripts in the same category, hold down the Ctrl key and select the required scripts one at a time, or hold down the Shift key to select a consecutive list of scripts.

4 Select *Delete Script* from the *Task Pane*. The selected scripts are listed.

5 Select *Finish*.

4.4.12 Access Times

- “Adding an Access Time” on page 82

Adding an Access Time

You can restrict the times during which a rule is valid by defining an access time and adding it to the rule conditions. You can also use access times as script entities.

To add a new access time:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Access Times* from the *Navigation Pane*.
3. To add an access time at the top level, select *Add Access Time* from the *Task Pane*. To add an access time to a category, select the category and select *Add Access Time* from the *Task Pane.*
4. Enter a name for the access time, for example, 'Office hours'.
5. Select *Finish*. Configure the access time using the *Modify Access Time* option.

Modifying an Access Time

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Access Times* from the *Navigation Pane*.
3. Select the access time you want to modify.
4. Select *Modify Access Time* from the *Task Pane*.
5. You can:
   - Change the *Name* of the access time.
   - Enter a *Description* of the access time.
   - Disable the access time by checking *Disabled*. A disabled access time is shown grayed out.
   - Set the access time as described in the following step.
6. You can set access times in multiples of half-hourly intervals. The access time is set to Deny Access for the whole week by default, shown in the calendar as blue.
   - To allow access at specific times, click and drag across the days and times you require, and repeat until the hours during which you want to grant access are shown in green,
   - To allow access for the majority of times and deny access for specific times, click the Grant Access box below the table to grant access for the whole week, then click and drag across the days and times you require, and repeat until the hours during which you want to deny access are shown in blue.

   For example, if you want to allow access only during the hours from 9:00 to 18:00 from Monday to Friday:

   6a Ensure the whole week is set to Deny Access (blue).

   6b Click in the calendar on 9 on Monday morning, and drag along to 18 and down to Friday and release. This creates a green block representing the times in which access is allowed.
7. Select *Finish*. You can now use this access time in rule conditions, and/or as a script entity.

Copying an Access Time

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Access Times* from the *Navigation Pane*.
3. Select the access time you want to copy.

   TIP: To select multiple access times in the same category, hold down the Ctrl key and select the required access times one at a time, or hold down the Shift key to select a consecutive list of access times.

4. Hold down the Ctrl key and drag the selected access times to the required location to create a copy.
5. If required, rename and/or modify the copy using the *Modify Access Time* option.

Moving an Access Time

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Access Times* from the *Navigation Pane*.
3. Select the access time you want to move.

   TIP: To move multiple access times in the same category, hold down the Ctrl key and select the required access times one at a time, or hold down the Shift key to select a consecutive list of access times.

4. Drag the selected access times to the required location.

Deleting an Access Time

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Access Times* from the *Navigation Pane*.
3. Select the access time you want to delete.

   TIP: To select multiple access times in the same category, hold down the Ctrl key and select the required access times one at a time, or hold down the Shift key to select a consecutive list of access times.

4. Select *Delete Access Time* from the *Task Pane*.
5. Select *Finish*. The access time will be deleted, and will also be removed from any rule conditions and script entities in which it has been defined.

4.4.13 Reports

Adding a Command Control Report

You can configure customized reports of the contents of the command control configuration database, which are dynamically created and emailed to the specified person at defined intervals. You can use Perl template scripting to extract the required information and format it into an email for the target person. An option is available for sending your reports to the Compliance Auditor for escalation management.

To use this feature, you must provide details of your email server to the Messaging Component (msgagnt) so that reports can be emailed. See "Configuring the Messaging Component" on page 84 for details.

To add a new report:

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Reports* from the *Navigation Pane*.
3. To add a report at the top level, select *Add Report* from the *Task Pane*. To add a report to a category, select the category and select *Add Report* from the *Task Pane.*
4. Enter a name for the report.
5. Select *Finish*. Configure the report using the *Modify Report* option.

Configuring the Messaging Component

To use the command control reporting feature, you must provide details of your email server to the Messaging Component (msgagnt) so that reports can be emailed, as follows:

1. Select *Hosts* from the *Navigation Pane* on the Framework Console home page.
2. Select the host where Command Control and the Messaging Component are installed.
3. Select *Packages* to view details of the packages installed on this host.
4. Select the *Messaging Component (msgagnt)*.
5. Select *SMTP Settings* from the *Task Pane*.
6. In the *SMTP Host* field, enter your email server IP address.
7. In the *SMTP Port* field, select or type the required port number.
8. If you are using a Lotus Notes server, in the *SMTP Domain* field enter the name of your SMTP domain.
9. Select *Finish*.

Modifying a Command Control Report

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Reports* from the *Navigation Pane*.
3. Select the required report.
4. Select *Modify Report* from the *Task Pane*.
5. If required, you can change the name of the report in the *Report Name* field.
6. If required, you can disable the report using the *Disabled* check box. A disabled report is shown grayed out.

7 Set the *Run Report* settings to determine the time of the first report and subsequent frequency of each report. You can set the initial date using the calendar and type in the time, then set the frequency as required.

8 In the *Email To* field, enter the email address of the person you want to send the report to.

9 In the *Email From* field, enter the email address of the person you want to send the report from.

10 In the *Email Subject* field, enter a subject for the email.

11 If you want the email to be displayed using HTML, check the *HTML* check box.

12 If you require a receipt, check the *Receipt* check box.

13 If you want the report to be available to be audited by the Compliance Auditor, check the *Audit* check box.

14 Enter a Perl script in the *Report Template* field to control how the email will be formatted and what it will contain.

For further information about report templates, contact PrivilegedUserSupport@novell.com.

15 If you want to send an email while testing this report, select the *Send email* check box.

16 Select *Test Report* to view the report that will be sent to the defined email address. Note that the report will not be shown here in HTML format. If there are errors in the Report Template, these will be shown.

17 Select *Back* to return to the report configuration screen.

18 Select *Finish*.

Copying a Command Control Report

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Reports* from the *Navigation Pane*.

3 Select the report you want to copy.

TIP: To select multiple reports in the same category, hold down the Ctrl key and select the required reports one at a time, or hold down the Shift key to select a consecutive list of reports.

4 Hold down the Ctrl key and drag the selected reports to the required location to create a copy.

5 If required, rename and/or modify the copy using the *Modify Report* option.

Moving a Command Control Report

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Reports* from the *Navigation Pane*.

3 Select the report you want to move.

TIP: To select multiple reports in the same category, hold down the Ctrl key and select the required reports one at a time, or hold down the Shift key to select a consecutive list of reports.

4 Drag the selected reports to the required location.

Deleting a Command Control Report

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Reports* from the *Navigation Pane*.

3 Select the required report.

TIP: To select multiple reports in the same category, hold down the Ctrl key and select the required reports one at a time, or hold down the Shift key to select a consecutive list of reports.

4 Select *Delete Report* from the *Task Pane*. The selected reports are listed.

5 Select *Finish*. The reports are deleted.

4.4.14 Test Suites

Adding a Test Suite

Command control test suites allow you to test your rules by running specified commands, submit users and other input values through your rule configuration, and checking the result is as expected. Each test suite can contain a number of test cases in which you specify the expected outcome for one or more input values.

To add a new test suite:

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Test Suites* from the *Task Pane*.

4 Select *Add Test Suite* from the *Task Pane*.

5 Enter a name for the test suite.

6 Enter a description for the test suite.

7 Select *Finish*. You should now add test cases to your test suite.

Viewing a Test Suite

1 Select *Command Control* from the *Navigation Pane* on the *Framework Console* home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Test Suites* from the *Task Pane*.

4 Select the test suite you want to view. From here you can modify the test suite, add, modify and delete test cases, and run the test suite.

Modifying a Test Suite

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Command Control* from the *Navigation Pane*.
3. Select *Test Suites* from the *Task Pane*.
4. Select the test suite you want to modify.
5. Select *View Test Suite* from the *Task Pane*.
6. Select *Modify Test Suite* from the *Task Pane*.
7. You can:
   - Change the *Name* of the test suite.
   - Add or change the *Description*.
   - Change the order in which the test cases are run using the *Up* and *Down* buttons.
8. Select *Finish*.

Adding or Modifying a Test Case

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Command Control* from the *Navigation Pane*.
3. Select *Test Suites* from the *Task Pane*.
4. Select the test suite for which you want to add a test case, or modify an existing test case.
5. Select *View Test Suite* from the *Task Pane*.
6. Select one of the following:
   - Select *Add Test Case* from the *Task Pane* to add a new test case, or
   - Select the test case you want to modify and select *Modify Test Case*.
7. Enter, or modify, the values that you want to run through the rule configuration, and the expected results. For example, if you want to test whether a specific command would be authorized for a specific submit user:
   - **7a** Enter the command in the *Command* field.
   - **7b** Enter the username in the *Submit User* field.
   - **7c** Check the *Expected authorized* checkbox and set the value to *Yes*.

     The *User Input* field allows you to enter information that you might request from a user via a script.

     The *Custom Input* text box allows you to enter additional customized test data. Contact PrivilegedUserSupport@novell.com if you want to use this facility.
8. Select *Finish*. The input values are shown in the *Test Cases* table.
9. Repeat Step 6 through Step 8 for any further test cases you want to include or modify in this test suite.

You can now run the test suite.

Running a Test Suite

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Test Suites* from the *Task Pane*.

4 Select the test suite you want to run.

TIP: To select multiple test suites, hold down the Ctrl key and select the required test suites one at a time, or hold down the Shift key to select a consecutive list of test suites. Use Ctrl+A to select all test suites.

---

5 Select *Run Test Suites* from the *Task Pane*. The results are displayed for each test case as Success, or Failure and the reason for the failure.

6 Use the buttons on the left and right of the table to find previous successes and failures, and the next successes and failures from a long list.

7 To view further details on a specific entry, select the entry and select *Details*. The configuration for the test case is shown, and a list of rules that have been tested, with configuration settings for each rule. The *Matched* column shows true if the rule conditions were met, and false if the rule conditions were not met. Select *Back* to return to the main *Run Test Suite* screen.

8 Select *Cancel* to return to the list of test suites.

Deleting a Test Case

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Test Suites* from the *Task Pane*.

4 Select the test suite from which you want to delete a test case.

5 Select *View Test Suite* from the *Task Pane*.

6 Select the test case you want to delete.

7 Select *Delete Test Case* from the *Task Pane*.

8 Select *Yes* to confirm the deletion. The test case is deleted.

Deleting a Test Suite

1 Select *Command Control* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control* from the *Navigation Pane*.

3 Select *Test Suites* from the *Task Pane*.

4 Select the test suite you want to delete.

TIP: To select multiple test suites, hold down the Ctrl key and select the required test suites one at a time, or hold down the Shift key to select a consecutive list of test suites.

---

5 Select *Delete Test Suite* from the *Task Pane*.

6 Select *Yes* to confirm the deletion. The test suite is deleted.

Importing a Test Suite

Using the Import Test Suites option, you can import backed up test suites, or test suites from another framework. Test suite configuration details are obtained using the Export Test Suites option, and can then be pasted into a text document for backup or for use on another framework.

NOTE: When you import test suites, they are added to your existing configuration and do not overwrite your existing test suites. However, if you import a command control database using the Import Settings option, your existing test suites are overwritten.

1. Access the test suite data you require and copy it.
2. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
3. Select *Command Control* from the *Navigation Pane*.
4. Select *Test Suites* from the *Task Pane*.
5. Select *Import Test Suites* from the *Task Pane*.
6. Click in the text area then paste in the copied settings using Ctrl+V, or right-click in the text area and select *Paste*.
7. Select *Finish*.

Exporting a Test Suite

You can export your command control test suites to a text file for backup purposes, or for use in another framework. To do this you need to copy the configuration information from your framework using the Export Test Suites option, and paste it into a text file. You can use the Import Test Suites option to restore the backed up test suites, or to import the test suites into another framework.

1. Select *Command Control* from the *Navigation Pane* on the Framework Console home page.
2. Select *Command Control* from the *Navigation Pane*.
3. Select *Test Suites* from the *Task Pane*.
4. Select the test suite you want to export.

   TIP: To select multiple test suites, hold down the Ctrl key and select the required test suites one at a time, or hold down the Shift key to select a consecutive list of test suites. To select all test suites, use Ctrl+A.
5. Select *Export Test Suites* from the *Task Pane*.
6. Select the test suite data using Ctrl+A, or right-click in the text window and select Select All.
7. Copy the test suite data using Ctrl+C, or right-click in the text window and select *Copy*.
8. Paste the text into a text document.
9. Select *Finish*.

4.4.15 Functions

The udsh command invokes commands on a set of hosts concurrently. It issues a command control request concurrently for each host that is specified and returns the output from all the hosts, formatted so that command results from all hosts can be managed.

- “Synopsis” on page 90
- “Options” on page 90
- “Keywords” on page 90

Synopsis

```
udsh [-bcdqv] [-t timeout] [-l user] [-f num] [-w host,host,wildcard] [-g
hostgrp,hostgrp] [cmd ...]
```

Options

The following options can be specified only on the command line:

-b	Do not break lines to column width when displaying output
-c	Do not remove the host from the list if the command fails
-d	Add timestamp to displayed output
-f num	Specify the maximum number of concurrent processes to run
-g hostgrp,hostgrp,wildcard	Specify which Command Control host groups to retrieve the list of agents to run the command on. Wildcards must be properly escaped
-l user	Specify the user to run the command as
-q	Quiet - do not display output
-t timeout	Specify the timeout in seconds for the command to complete on each host
-v	Verbose output
-w host,host,wildcard	Specify which agents to run the command on. Wildcards must be properly escaped

If a command is not specified, the user is placed at a command prompt. Each entry run from this prompt is run separately on each host. If readline(3) is available, command line editing and history are provided. There are various macros that can be specified in the command to substitute keywords when the command is run on the remote host.

Keywords

${uid}$	Calling user's uid
${gid}$	Calling user's primary group id
${gecos}$	Calling user's gecos
${home}$	Calling user's home dir
${shell}$	Calling user's shell
${cwd}$	Calling user's current working dir
${lhost}$	Local host name
${rhost}$	Remote host name
${pid}$	Pid of the individual udsh call
${ppid}$	Pid of the udsh

Compliance Auditor

5

5.1 Compliance Auditor Overview

The Compliance Auditor collects, filters and generates reports of audit data for analysis and sign-off by authorized personnel. The Compliance Auditor can be used in conjunction with command control to enable auditors to view security transactions and play back recordings of user activity. Auditors can record notes against each record, creating permanent archives of activity.

Rules can be configured to pull any number of audit events matching a given filter into the Compliance Auditor at any specific frequency. Examples of filters include username, host and command for Command Control. Roles can be assigned to each rule to ensure auditors are only able to view extracted records with a matching role defined in their user account. In addition, Access Control Levels (ACLs) can be defined to restrict access to individual events, and to prevent users from auditing their own activity.

When an audit event has been viewed, auditors can authorize the event, or mark it as unauthorized, escalate it, and assign it to someone else. Each change is recorded in an indelible audit trail within each record, along with any notes made by the auditor. Automatic reports can be generated and emailed to the appropriate personnel, and can be used, for example, for daily reporting to managers on audit activity awaiting sign-off, or hourly reporting triggered by an escalation value to notify senior management of activity.

To use the Compliance Auditor:

- If required, define roles in user groups to control user access to the Compliance Auditor (see Section 5.3, “Controlling Access to the Compliance Auditor,” on page 92).
- Create one or more rules to pull the required events into the Compliance Auditor (see Section 5.5, “Audit Rules,” on page 96).
- If required, define ACLs for individual users (see Section 5.7.1, “Adding or Modifying a User ACL,” on page 103).
- When events are listed in the Compliance Auditor main screen, authorized personnel can view event records and authorize them, or mark them as unauthorized and define further action (see Section 5.4.1, “Compliance Auditor Event List,” on page 93).
- Configure auditing reports to be automatically emailed to the appropriate personnel (see Section 5.6.2, “Adding or Modifying an Audit Report,” on page 98).

5.2 Deploying the Compliance Auditor

The Compliance Auditor modules comprise the following packages:

- Compliance Auditor (secaudit): holds the compliance auditor rules and audit information.
- Compliance Auditor Console: installed into the Framework Console and required for configuring Compliance Auditor rules and for viewing audit information.

The Compliance Auditor is only shown as an available package on hosts with the Audit Manager (audit) deployed.

In addition, if you want to use the Compliance Auditor reporting facilities, you need to install the Access Manager (auth) on the host with the Compliance Auditor.

5.2.1 Steps Required

To deploy the Compliance Auditor:

1. Download the required packages to your local Package Manager. See Section 2.2, “Managing Module Distribution,” on page 28 for details.
2. Install the Audit Manager package on the host you want to be the Audit Manager, and then install the Compliance Auditor package on the same host. This can be on any operating system including Windows. See Section 2.2.2, “Deploying Packages to Hosts,” on page 30 for details. The auditing packages can be deployed to as many hosts as you wish to build an environment with load balancing and failover.
3. If you require reporting facilities, install the Access Manager package on the same host as the Compliance Auditor package.
4. Install the Compliance Auditor Console. See Section 2.3.2, “Adding Consoles to the Framework Console,” on page 34 for details.

The Compliance Auditor is now deployed and ready to use.

5.3 Controlling Access to the Compliance Auditor

Roles can be used to restrict the Compliance Auditor options available to Framework users. For example, you might want users to be able to audit events, but not administer rules, ACLs, or reports.

To define roles for a user group to control their use of the Compliance Auditor, select Manage Users from the Navigation Pane, then either add a new group or modify an existing group and select the Roles section. Configure the roles you require according to the table under Compliance Auditor in the Configuring Roles topic. Any user belonging to this group will have these roles.

You can use roles in audit rules to restrict the events that users can view in the Compliance Auditor. See Section 5.5, “Audit Rules,” on page 96 for details.

Example

If you define the following combination of roles for a group, users belonging to this group will be able to access the Compliance Auditor console only, and will be able to view and edit records and review keystroke logs, but not manage rules, reports or ACLs.

Module	Role
secaudit	console
secaudit	audit
audit	read

5.4 Records

5.4.1 Compliance Auditor Event List

The Compliance Auditor main screen lists events collected according to defined audit rules (for an overview of the Compliance Auditor, see Section 5.1, “Compliance Auditor Overview,” on page 91).

- By default, all new and pending events are displayed, indicated in the *Status* column: to view authorized and/or unauthorized events, check the appropriate checkboxes and select *Refresh*. Pending events are events that have been viewed and their records edited, but they have not been set to authorized or unauthorized. You can sort by any of the column headings by selecting them.
- To view events for a specific time period, check the From and To checkboxes, select the required dates, enter the required times and select *Refresh*.
- The first column in the table displays color-coded indicators for Command Control command risk level and rule risk level, ranging from green (low) to red (high) (see “Setting the Command Risk” on page 78).
- The table also shows:
  - In the Level field, the escalation level set by the auditor editing the event record.
  - The Time of the event.
  - A description of the event in the Event column.
  - In the Note column, any notes made by the auditor when editing the event record.
  - In the Assigned column, the user the event has been assigned to by the auditor editing the event record.
  - The audit Rule that has pulled in the event.
  - The Type of event.
  - The unique Event ID.

To view and/or edit an event record, select the event then select View Record from the Task Pane. For further details, see one of the following:

- Section 5.4.2, "Viewing a Command Control Audit Record," on page 94
- Section 5.4.3, "Viewing a Command Control Keystroke Report," on page 94
- Section 5.4.4, "Viewing a Change Management Audit Record," on page 95
- Section 5.4.5, "Viewing a Report Audit Record," on page 95

5.4.2 Viewing a Command Control Audit Record

1. Select *Compliance Auditor* from the Navigation Pane on the Framework Console home page.
2. Select the Command Control record you want to view. The record type is shown in the *Type* column (you may need to scroll to the right to see this column).
3. Select *View Record* from the *Task Pane*. Record data for this event is shown, including the submit user and host, the run user and host, the command, whether it was authorized by Command Control, and whether the session was captured.
4. From here you can view a Command Control keystroke report, if it exists, or edit the record. If a keystroke report exists, you must review it before you can edit the record.

5.4.3 Viewing a Command Control Keystroke Report

1. Select *Compliance Auditor* from the Navigation Pane on the Framework Console home page.
2. Select the record for which you want to view a keystroke report.
3. Select *View Record* from the *Task Pane*.
4. Select *View Keystroke Report* from the *Task Pane*, or select the *Keystroke* button. The text that the user entered during the session is shown on the Input screen. The first column displays color-coded indicators for command risk level and rule risk level, ranging from green (low) to red (high) (see "Setting the Command Risk" on page 78 and "Modifying a Rule" on page 66).
5. You can:
   - Change the *Terminal Type* to the one you require, if it is set incorrectly.
   - Search for text in the report: enter the text in the search field and select *Find*.
   - If an encryption password has been defined on the Command Control *Audit Settings* screen to encrypt the sensitive password data in the reports (see "Defining Audit Settings" on page 61), enter this password in the *Decryption key* field and select *Refresh* to display the passwords.
   - Show or hide control characters on the screen using the *Show control characters* checkbox.
   - Show or hide the full list of audited commands using the *Show audited commands* checkbox. If enabled, the screen will show the actual commands that are being run when a user types a command. They can also be viewed for each input command individually by hovering over the command with your mouse.
6. If you want to see the keystroke text being played back with the screen output, select *Output*.

TIP: You can start the playback from a specific line in the input by selecting that line before selecting *Output*.

You can:

- Select *Play* to play the keystroke entries and view the output.
- Select *Rewind* to go back to the beginning.
- Select *Pause* to pause the playback.
- Select *Forward* to skip any pauses in the playback where the user may have taken a break from typing.
- Set the *Playback Speed* to *Real Time*, *Double Speed* or *Full Speed*.
- Set the *Scrollback* field to the amount of text you want to be able to scroll back through, in kilobytes.
- Change the *Terminal Type* to the one you require, if it is set incorrectly.

7 Select *Cancel* to return to the record list.

5.4.4 Viewing a Change Management Audit Record

1 Select *Compliance Auditor* from the Navigation Pane on the Framework Console home page.

2 Select the Command Control Change Management record you want to view. The record type is shown in the *Type* column (you may need to scroll to the right to see this column).

3 Select *View Record* from the *Task Pane*. Information about the Change Management action is displayed, including the name of the user who made changes to the database, and any entries the user made when committing the command control transaction.

4 From here you can edit the record.

5.4.5 Viewing a Report Audit Record

1 Select *Compliance Auditor* from the Navigation Pane on the Framework Console home page.

2 Select the Report record you want to view. The record type is shown in the *Type* column (you may need to scroll to the right to see this column).

3 Select *View Record* from the *Task Pane*. Record data for this report is shown, including the contents of the report sent.

4 From here you can edit the record.

5.4.6 Editing an Audit Record

For each event listed in the Compliance Auditor, you can edit the audit record to authorize the event, or mark it as unauthorized, escalate it, and assign it to another user. You can also add notes for display in the event record, and comments which will be permanently recorded in the event history.

NOTE: For command control events for which a keystroke report exists, you must view the keystroke report before editing the audit record.

To edit an audit record:

1 Select *Compliance Auditor* from the Navigation Pane on the Framework Console home page.

2 Select the record you want to edit.

3 Select *View Record* from the *Task Pane*.

4 Select *Edit Record*.

If you want to authorize the event

1 Check the *Authorized* checkbox.

2 If required, enter a note in the *Note* field which will be displayed on the event list and event record.

3 If required, enter a comment in the *Comment* field which will be permanently displayed in the *History* on the *View Record* screen.

If you want to mark the event as unauthorized

1 Check the *Unauthorized* checkbox.

2 If required, set an *Escalation Level* which will be displayed on the event list and can be used as a *Report Filter* when setting up reports (see Section 5.6.2, "Adding or Modifying an Audit Report," on page 98).

3 If required, assign the record to a different user in the *Assigned to* field.

4 Enter a *Note* and/or a *Comment* to explain why the event is unauthorized.

5 Select *Finish*.

5.5 Audit Rules

Audit rules specify the events to be pulled in to the Compliance Auditor for viewing and authorization. You can specify:

- The type of event, using one or more filters
- The number of events
- The time and frequency at which the events will be pulled in

You can also restrict access to records of events pulled in by a specific rule, using audit roles.

To adding or modifying an audit rule:

1 Select *Compliance Auditor* from the *Navigation Pane* on the Framework Console home page.

2 Select *Audit Rules* from the *Task Pane*.

3 If you want to add a new rule, select *Add Rule* from the *Task Pane*.

If you want to modify an existing rule, select the required rule and select *Modify Rule* from the *Task Pane* (the remaining steps assume you are adding a new rule).

4 In the *Rule Name* field, enter a name for your rule.

5 If required, you can disable the rule using the *Disable* check box. Disabled rules are not shown, by default, on the rule list. Note that you cannot delete a rule.

6 Set the number of records to be collected on each audit run in the *Records* field, or set it to all records using the *All Records* checkbox (the default).

7 If required, enter an *Audit Role* for the rule. Only users belonging to a group for which this Audit Role is defined will be able to view the events pulled in by the rule.

TIP: To define an Audit Role for a user group, select *Manage Users* from the *Navigation Pane*, then either add a new group or modify an existing group and select the *Roles* section. Your role should specify secaudit in the *Module* column, and a name of your choice for the role in the *Role* column, which can then be used in your audit rules. Any user belonging to this group will have this role.

8 Set the *Run Filter* to determine the time and frequency of each audit run. You can set the initial date using the calendar and type in the time, then set the frequency as required.

9 Select the *Audit Category* as required.

10 If required, select one or more filters from the *Add Filter* drop down list for the type of event you want this rule to pull in, and configure as required. The filters available and configuration options depend on the *Audit Category* selected. For example, for Command Control events you could choose to pull in only those events which have been submitted by a particular user and include a session capture.

You can add more than one filter of the same type for filters such as the Command Control Submit User, and select the logic you require from and or or. You can also set these filters to be inclusive or exclusive, using *matches* or *does not match*.

You can remove a filter by selecting the button to the left of the filter.

11 Select *Finish*.

5.6 Reports

You can configure customized reports of events which require auditing, which will be dynamically created and emailed to selected users at defined intervals. You can use filtering and Perl template scripting to extract the appropriate event information and format it into an email for each target user.

Audit reporting uses a tokens object which contains all the user information and other information. You can use keyword anchors in your report configuration which will be replaced by the appropriate values from the tokens object. It is also possible for the Perl code in the report template to set values in the tokens object. Sample report templates are supplied to assist you with creating your own.

5.6.1 Configuring the Messaging Component

To use this feature, you must provide details of your email server to the Messaging Component (msgagnt) so that reports can be emailed. To do this:

1 Select *Hosts* from the *Navigation Pane* on the Framework Console home page.

2 Select the host where the Compliance Auditor and Messaging Component are installed.

3 Select *Packages* to view details of the packages installed on this host.

4 Select the *Messaging Component (msgagnt)*.

5 Select *SMTP Settings* from the *Task Pane*.

6 In the *SMTP Host* field, enter your email server IP address.

7 In the *SMTP Port* field, select or type the required port number.

8 If you are using a Lotus Notes server, enter the name of your SMTP domain in *SMTP Domain*.

9 Select *Finish*.

5.6.2 Adding or Modifying an Audit Report

To use this feature, you must provide details of your email server to the Messaging Component (msgagnt) so that reports can be emailed. See Section 5.6.1, "Configuring the Messaging Component," on page 97 for details.

To add or modify an audit report:

1 Select *Compliance Auditor* from the Navigation Pane on the Framework Console home page.

2 Select *Audit Reports* from the *Task Pane*.

3 If you want to add a new report, select *Add Report* from the *Task Pane*.

If you want to modify an existing report, select the required report and select *Modify Report* from the *Task Pane* (the remaining steps assume you are adding a new report).

4 In the *Report Name* field, enter a name for the report.

5 If required, you can disable the report using the *Disable* check box. Disabled reports are not shown, by default, on the report list.

6 Set the *Run Report* settings to determine the time of the first report and subsequent frequency of each report. You can set the initial date using the calendar and type in the time, then set the frequency as required.

7 Set the *Report Category* to the category you require, or set to *All*.

8 If you want the report to be sent to a user or group of users who are defined as Framework users, check user report in the *Report Target* section and select the required user or [group] from the drop down list.

TIP: You could select a group on the basis of the Audit Role defined for it. To define an Audit Role for a user group, select *Manage Users* from the *Navigation Pane*, then either add a new group or modify an existing group and select the *Roles* section. Your role should specify secaudit in the *Module* column, and a name of your choice for the role in the *Role* column. Any user belonging to this group will have this role.

Ensure that the users' email addresses are defined in the *Account Details* section in the Framework User Account definitions. You must define a keyword anchor in the *Email To* field, as described below.

9 Set the *Report Filter* to include the required event records:

- Select one or more from *New*, *Pending*, *Authorized* and *Unauthorized*.
- Select the *age* of events you want to include in the report. Events older than the number of days you specify will be included.
- Select the *escalation level* of events you want to include in the report. Events at this escalation level and above will be included.

10 Complete the *Email* fields as required:

- If you want the report to be sent to a user who is not defined as a Framework user, enter the user's email address in the *Email To* field, and complete the other fields as required.
- If you want the report to be sent to a user or group defined as the *Report Target* above, enter the following keyword anchor in the *Email To* field: $User.ACT_EMAIL.value$. Complete the other fields as required.

You can also use a keyword anchor in the *Email Subject* field and the *Email From* field. For example, if you wanted to display the target user's name in the email subject, you could enter

```
Report for $User.ACT_FULL_NAME.value$
```

in the *Email Subject* field.

You can view the format in XML of the full tokens object passed into the audit report by entering $<>$ in the *Report Template* field below and selecting *Test Report* (ensure you have defined a *Report Target*). To view just the user subtree, use `$<User>$`.

11 Enter a Perl script in the *Report Template* field to control how the email will be formatted and what it will contain. If you want the emails to be displayed using HTML, check the *HTML* checkbox.

For a sample report template, see Section 5.6.3, "Sample Command Control Report Template," on page 99.

If you are using the sample as a base for your own report templates, check *HTML* to display the emails correctly. The sample displays a message to the recipients of the emails requesting them to log on to the Compliance Auditor and review activity. It extracts selected events and list them in tables according to the age of the events, and provides information about the events.

TIP: As shown in the sample, you can use the user name keyword anchor $User.ACT_FULL_NAME.value$ to display a user's name in the email, if you are using the Report Target option above. You must ensure that a *Display name* is entered for the user in the *Account Details* section in the Framework User Account definitions.

For further information about report templates, contact PrivilegedUserSupport@novell.com.

12 Select *Test Report* to view the report that will be sent to each email target. Use the arrow keys to page through the reports if there is more than one. Note that the reports will not be shown here in HTML format. If there are errors in the *Report Template*, these will be shown.

13 Select *Back* to return to the report configuration screen.

14 Select *Finish*.

5.6.3 Sample Command Control Report Template

```
<%!
my @lvl0;
my @lvl1;
my @lvl2;
my @lvl3;
my @gt0;
my @gt5;
my @gt10;
my @gt20;
%>
<%
my @audit_records = @{$tokens->{'AuditRecords'}->{'AuditRecord'}} if
```

```
(defined($tokens->{'AuditRecords'}) && defined($tokens->{'AuditRecords'}-
>{'AuditRecord'}));
foreach my $ar (@audit_records) {
  my $age = $ar->{'age'};
  my $lvl = $ar->{'level'};

  if ($age > 5 && $age < 10) {
    push(@gt5,$ar);
  } elsif ($age >= 10 && $age < 20) {
    push(@gt10,$ar);
  } elsif ($age >= 20) {
    push(@gt20,$ar);
  } else {
    push(@gt0,$ar);
  }
  if ($lvl == 1) {
    push(@lvl1,$ar);
  } elsif ($lvl == 2) {
    push(@lvl2,$ar);
  } elsif ($lvl >= 3) {
    push(@lvl3,$ar);
  } else {
    push(@lvl0,$ar);
  }
}
%>
<%
my $total = @audit_records;
if ($total > 0) {
%>
<style type="text/css">
<!--
.style1 {
 color: #000000;
 font-family: Arial, Helvetica, sans-serif;
 font-size: 12px;
}
.style2 {
 color: #000000;
 font-family: Arial, Helvetica, sans-serif;
 font-size: 12px;
 font-weight:bold;
}
.style4 {
 color: #000000
}
-->
</style>
<p class="style1"> Hello $User.ACT_FULL_NAME.value$,<br/>
  <br/>
  This is an automated event notification email from the Compliance Auditor. <br/>
<br/>

  It is the responsibility of management to log into the Compliance Auditor each
day and review their team's keystroke logs.  <br/> <br/>

  Please log on to the Compliance Auditor at your earliest convenience using this
link: <a href="https://admin.company.com">https://admin.company.com</a></p>
```

```
<%
my $gt0 = @gt0;
%>
<span class="style2">Events &lt; 5 days old (<%= "$gt0" %>)</span>
<table border="1">
  <tr class="style1">
    <td>Time</td>
    <td>User</td>
    <td>Run As</td>
    <td>Host</td>
    <td>Command</td>
  </tr>
  <%
foreach my $ar (@gt0) {
  my $cmd = $ar->{'cmdctrl'}->{'cmd'};
  my $usr = $ar->{'cmdctrl'}->{'user'};
  my $ras = $ar->{'cmdctrl'}->{'runAs'};
  my $hst = $ar->{'cmdctrl'}->{'host'};
  my $tme = $ar->{'cmdctrl'}->{'time'};
 $tme = localtime($tme);
%>
  <tr class="style1">
    <td><%= "$tme" %></td>
    <td><%= "$usr" %></td>
    <td><%= "$ras" %></td>
    <td><%= "$hst" %></td>
    <td><%= "$cmd" %></td>
  </tr>
<%
}
%>
</table>
<br/>

<%
my $gt5 = @gt5;
%>
<span class="style2">Events &gt; 5 days old (<%= "$gt5" %>)</span>
<table border="1">
  <tr class="style1">
    <td>Time</td>
    <td>User</td>
    <td>Run As</td>
    <td>Host</td>
    <td>Command</td>
  </tr>
<%
foreach my $ar (@gt5) {
  my $cmd = $ar->{'cmdctrl'}->{'cmd'};
  my $usr = $ar->{'cmdctrl'}->{'user'};
  my $ras = $ar->{'cmdctrl'}->{'runAs'};
  my $hst = $ar->{'cmdctrl'}->{'host'};
  my $tme = $ar->{'cmdctrl'}->{'time'};
 $tme = localtime($tme);
%>
  <tr class="style1">
    <td><%= "$tme" %></td>
    <td><%= "$usr" %></td>
    <td><%= "$ras" %></td>
```

```
    <td><%= "$hst" %></td>
    <td><%= "$cmd" %></td>
  </tr>
<%
}
%>
</table>
<br/>

<%
my $gt10 = @gt10;
%>
<span class="style2">Events &gt; 10 days old (<%= "$gt10" %>)</span>
<table border="1">
  <tr class="style1">
    <td>Time</td>
    <td>User</td>
    <td>Run As</td>
    <td>Host</td>
    <td>Command</td>
  </tr>
<%
foreach my $ar (@gt10) {
  my $cmd = $ar->{'cmdctrl'}->{'cmd'};
  my $usr = $ar->{'cmdctrl'}->{'user'};
  my $ras = $ar->{'cmdctrl'}->{'runAs'};
  my $hst = $ar->{'cmdctrl'}->{'host'};
  my $tme = $ar->{'cmdctrl'}->{'time'};
 $tme = localtime($tme);
%>
  <tr class="style1">
    <td><%= "$tme" %></td>
    <td><%= "$usr" %></td>
    <td><%= "$ras" %></td>
    <td><%= "$hst" %></td>
    <td><%= "$cmd" %></td>
  </tr>
<%
}
%>
</table>
<br/>



<%
my $gt20 = @gt20;
%>
<span class="style2">Events &gt; 20 days old (<%= "$gt20" %>)</span>
<table border="1">
  <tr class="style1">
    <td>Time</td>
    <td>User</td>
    <td>Run As</td>
    <td>Host</td>
    <td>Command</td>
  </tr>
<%
foreach my $ar (@gt20) {
```

```
  my $cmd = $ar->{'cmdctrl'}->{'cmd'};
  my $usr = $ar->{'cmdctrl'}->{'user'};
  my $ras = $ar->{'cmdctrl'}->{'runAs'};
  my $hst = $ar->{'cmdctrl'}->{'host'};
  my $tme = $ar->{'cmdctrl'}->{'time'};
 $tme = localtime($tme);
%>
  <tr class="style1">
    <td><%= "$tme" %></td>
    <td><%= "$usr" %></td>
    <td><%= "$ras" %></td>
    <td><%= "$hst" %></td>
    <td><%= "$cmd" %></td>
  </tr>
<%
}
%>
</table>
<br/>

<p class="style2">Total Events = <%= $total %></p>

<%
}
%>
```

5.6.4 Deleting a Report

1. Select *Compliance Auditor* from the *Navigation Pane* on the Framework Console home page.
2. Select *Audit Reports* from the *Task Pane*.
3. Select the report you want to delete.
4. Select *Delete Report* from the *Task Pane*.
5. Select *Finish* to confirm the deletion.

5.7 Access Control Levels (ACLs)

You can define an Access Control Level (ACL) for your auditors which specifies which events they are allowed or not allowed to view, and also allows you to restrict the ability of auditors to authorize their own activity.

- Section 5.7.1, “Adding or Modifying a User ACL,” on page 103
- Section 5.7.2, “Deleting a User ACL,” on page 104

5.7.1 Adding or Modifying a User ACL

1. Select *Compliance Auditor* from the *Navigation Pane* on the Framework Console home page.
2. Select *Access Control* from the *Task Pane*.
3. If you want to add a new ACL, select *Add User ACL* from the *Task Pane*.

   If you want to modify an existing ACL, select the required *User* and select *Modify ACL* from the *Task Pane* (the remaining steps assume you are adding a new ACL).
4. Select the user from the *Username* drop down list for whom you want to define an ACL.

5. Select *Add*.
6. Select the attribute from the drop-down list which describes the entity to which you want to control access for the selected user. For example, if you do not want this user to be able to audit Command Control events involving a particular command, select *Command*.
7. In the *Matches* field, enter the value of the attribute you want to control access to. For example, if you do not want this user to be able to audit any Command Control events which involve the cat /etc/passwd command, enter this command in this field. You can use wildcard characters in this field.
8. Set the *Action* to allow or deny, as required.
9. Use the arrow buttons to move entries up and down the list. You might want to do this if, for example, you are allowing the user to access a restricted list of commands, and denying access to all other commands using the wildcard *. The 'allowed commands' entries must be above the 'deny all' entry. By default, all commands would be allowed.

   You can remove an attribute by selecting it and then selecting the *Remove* button.

   You can modify an entry by selecting it, then entering the required changes.
10. Select *Finish*.

5.7.2 Deleting a User ACL

1. Select *Compliance Auditor* from the *Navigation Pane* on the Framework Console home page.
2. Select *Access Control* from the *Task Pane*.
3. Select the *User* for whom you want to delete an ACL.
4. Select *Delete User ACL* from the *Task Pane*.
5. Select *Finish* to delete the ACL for the user.

Auditing Enterprise Users

6

The Framework enables auditing of events at several levels from keystroke logging through command authorization to logon success or failure. The reports are divided into three primary areas:

- Command Control Reports
- Account Logon Reports

This section describes the following features of auditing:

6.1 Audit Settings

It is possible to control the log files used to hold the audit data on the Framework.

1. Select *Reporting* from the *Navigation Pane* on the Framework Console home page.
2. Select *Audit Settings* from the *Task Pane*.
3. For each Audit Db file, you can set the rollover parameters in time or size at which the system clears the file and starts a new record.
4. Select the *Finish* button to complete the task.

IMPORTANT: This is an advanced user function. Do not amend the settings in this area prior to consulting PrivilegedUserSupport@novell.com.

6.2 Command Control Activity Reports

6.2.1 Add a Report

To add a report to the command control reports list, perform the following:

1. Select *Reporting* from the *Navigation Pane* on the Framework Console home page.
2. Select *Command Control Reports* from the *Navigation Pane*.
3. Select *Add Report* from the *Task Pane*.

4 Enter a name for the report in the provided field and enter a description.

5 Select the *Finish* button to complete the task.

6.2.2 Configuring a Report

- “Selecting Log files to be included in the report” on page 106
- “Selecting Hosts to be included in the report” on page 106

Selecting Log files to be included in the report

1 View the report data.

2 Select the Log Files tab from the *Navigation Pane*.

3 Select the log files which are required for the report.

TIP: To include all available log files select the box to insert a tick.

4 Select the *Apply* button to complete the task.

Selecting Hosts to be included in the report

1 View the report data.

2 Select the Filter tab from the *Navigation Pane*.

3 Enter the filter values required to generate the filtered report. You can use the following wildcard characters in your filter entries

Symbol	Effect
*	Matches any sequence of zero or more characters
?	Matches exactly one character
[...]	Matches one character from the enclosed list
[^...]	Matches one character not from the enclose list
[a-z]	Matches any single lower case character

4 Apply time and date filters if required.

5 Select the *Apply* button to complete the task.

6.2.3 Viewing Report Data

1 Select *Reporting* from the *Navigation Pane* on the Framework Console home page.

2 Select *Command Control Reports* from the *Navigation Pane*.

3 Select the required report from the *Navigation Pane*.

The *Navigation Pane* displays the Report Data for the selected report.

6.2.4 Keystroke Replay

Where a rule has been configured to capture session information, it is possible to review the entire session.

To review a captured session:

1. View the report data.
2. Select the session which you wish to review from the *Navigation Pane*.

   TIP: Commands for which session data has been captured are indicated by a "Yes" in the Capture column.

3. Select *Keystroke Replay* from the *Task Pane*.
4. Use the Play, Rewind and Pause buttons to review the session data.

6.2.5 Removing a Report

1. View the report data.
2. Select *Delete Report* from the *Task Pane*.
3. Select the *Finish* button to complete the task.

IMPORTANT: This action can not be undone.

6.2.6 Generating an Activity Report

1. View the report data.
2. Select *Activity Report* from the *Task Pane*.

   The *Navigation Pane* displays the selected Activity Report.

6.2.7 Printing Activity Report

1. Generate the Activity Report.
2. Select the *Print* button from the *Navigation Pane*.
3. Complete the Print dialogue on your host system to print the report.

6.3 Account Log on Reports

6.3.1 Adding a Report

To add a report to the command control reports list, perform the following:

1. Select *Reporting* from the *Navigation Pane* on the Framework Console home page.
2. Select *User Logon Reports* from the *Navigation Pane*.
3. Select *Add Report* from the *Task Pane*.
4. Enter a name for the report in the provided field and enter a description.
5. Select the *Finish* button to complete the task.

6.3.2 Configuring a Report

- "Selecting Log Files to Be Included in the Report" on page 108
- "Setting the Filters on the Report" on page 108

Selecting Log Files to Be Included in the Report

1. View the report data.
2. Select the *Log Files* tab from the *Navigation Pane*.
3. Select the log files which are required for the report.

   TIP: To include all available log files select the box to insert a tick.

4. Select the *Apply* button to complete the task.

Setting the Filters on the Report

1. View the report data
2. Select the *Filter* tab from the *Navigation Pane*.
3. Select the logon status and logon type values required.
4. Enter the filter values required to generate the filtered report. You can use the following wildcard characters in your filter entries

Symbol	Effect
*	Matches any sequence of zero or more characters
?	Matches exactly one character
[...]	Matches one character from the enclosed list
[^...]	Matches one character not from the enclose list
[a-z]	Matches any single lower case character

5. Apply time and date filters if required.
6. Select the *Apply* button to complete the task.

6.3.3 Viewing Report Data

1. Select *Reporting* from the *Navigation Pane* on the Framework Console home page.
2. Select *Account Logon Reports* from the *Navigation Pane*.
3. Select the required report from the *Navigation Pane*.

   The *Navigation Pane* displays the report data for the selected report.

6.3.4 Remove a Report

1. View the report data.
2. Select *Delete Report* from the *Task Pane*.
3. Select the *Finish* button to complete the task.

IMPORTANT: This action can not be undone.

6.3.5 Generating an Activity Report

1. View the report data.
2. Select *Activity Report* from the *Task Pane*.

   The *Navigation Pane* displays the selected activity report.

6.3.6 Printing an Activity Report

1. Generate the activity report.
2. Select the *Print* button from the *Navigation Pane*.
3. Complete the Print dialogue on your host system to print the report.

Load Balancing and Failover

7

The load balancing and failover features work by using an hierarchical view of the Hosts associated with the Framework.

The hierarchy of hosts is created using the Hosts console to group hosts into domains and sub domains, which are representative of your enterprise network structure. This effectively gives them a chain of command, where they always address requests to managers in their immediate sub domain before moving along a branch to another sub or parent domain.

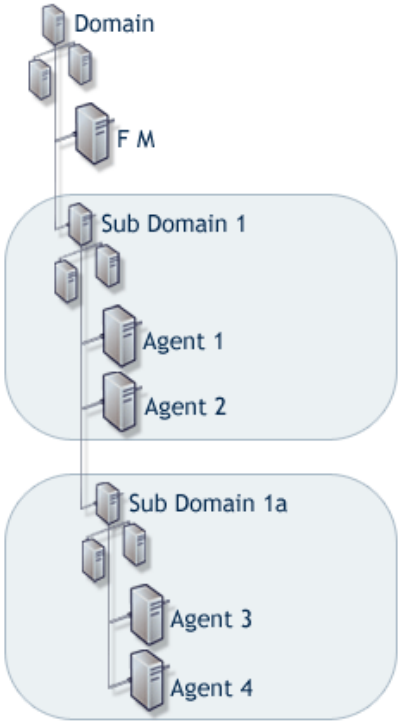
To achieve an effective load balancing and failover environment multiple Framework Manager packages must be deployed across the Framework. The licensing model is not based on how many managers or agents are deployed, but how many Hosts the Framework is deployed on. Therefore there are no restrictions on how many Framework Manager packages you can deploy.

The Registry Manager controls a database used to record the location and status of each package deployed on each of the hosts within the Framework. A copy of this information is held at each host by the registry agent package which is included as part of the agent installation. The distributed information is used to calculate the route to the appropriate manager for requests from any agent registered on the Framework. The structure of the registry data enables each host to determine which Framework Manager on the Framework should be the target of requests, and which Framework Manager to use in the event of failure or withdrawal of the initial selected Framework Manager.

7.1 Failover

The failover feature automatically and transparently redirects requests from a failed or withdrawn Framework Manager to the next available manager of the same type. The agent automatically connects to a manager that is next in line in accordance with your defined hierarchy.

Table 7-1 *Creating a Failover Environment*



This diagram is an example of a typical way to create a failover environment.

The example illustrates how to create an effective failover environment. The Framework Manager (FM) is a Windows host. All agents are UNIX hosts.

Deployment: You would deploy the Command Control Manager package on the following hosts. FM, Agent 1 & Agent 3

Who authenticates to whom: By Default each agent contacts the following host for command control authentication.

Agent 1 and 2 contact Agent 1.
Agent 3 and 4contact Agent 3.

IMPORTANT: Windows only supports the Command Control Manager package.

Examples

1. Agent 3 is downed for maintenance. Agent 4 would seek authentication from Agent 1.
2. Agent 1 is downed from a broken network card. Agents 2, 3 & 4 would seek authentication from FM.
3. The Command Control Manager package is removed from FM and Agent is still broken. Agents 2, 3 & 4 would seek authentication from Agent 3.

IMPORTANT: If a further Sub Domain was to be added, agents under Sub Domain 1 & 1a would then seek authentication from the New Sub Domain in the event no other Command Control Manager is available.

7.2 Load Balancing

Load balancing describes the ability to distribute processing and communications activity evenly across the Framework so that no single Framework Manager is overwhelmed by agent requests.

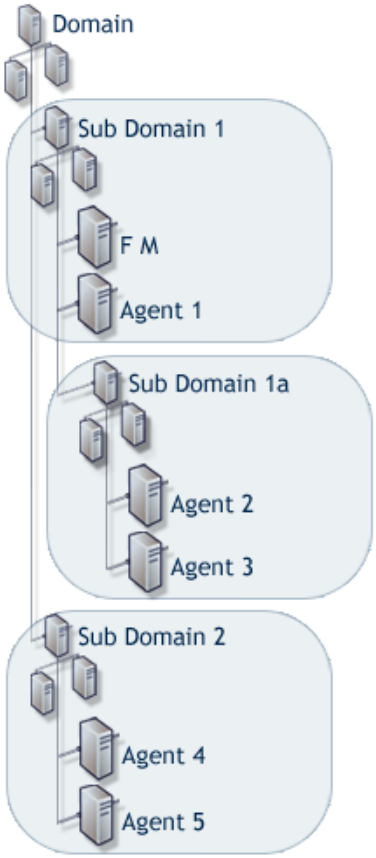
Load balancing is particularly important in situations where it is difficult to predict the number of requests that will be directed to a specific category of manager.

The Framework automatically replicates data from the defined primary manager to each additional manager that is deployed in the Framework. Replication takes place automatically when the manager is initially deployed and then again at any stage when the data on the primary manager is modified.

The following packages can be load balanced:

- Registry Manager - Maintains a database of all hosts and modules and provides certificate based registration features for the hosts.
- Package Manager - Manages a repository for packages.
- Administration Agent - Provides the functionality for the Web-based user interface. Consoles can be installed onto the Administration Agent and used to control product features.
- Access Manager - Maintains a list of Framework user accounts and provides authentication services for the Framework. Note: this package needs to be installed with a local Registry Manager in order to create a secure user authentication token.
- Command Control Manager – Maintains a database of all defined command control rules, commands, and scripts.

Table 7-2 *Creating a Load Balancing Environment*



This diagram is an example of a typical way to create a load balanced environment.

The example illustrates how to create an effective load balanced environment. The Framework Manager (FM) is a Windows host. All agents are UNIX hosts.

Deployment: You would deploy the Command Control Manager package on the following hosts. FM, Agent 2 and Agent 4

Who authenticates to whom: By default each agent contacts the following host for command control authentication.

Agent 1 contacts FM.
Agents 2 and 3 contacts Agent 2.
Agents 4 and 5 contacts Agent 4.

IMPORTANT: Windows only supports the Command Control Manager package.

Example of load balancing working with failover

1. Agent 2 is downed for maintenance. Agent 3 would seek authentication from FM.
2. Agent 4 is downed from a broken network card. Agents 5 would seek authentication from FM.

Command Control Components

8

This page explains the components of the Command Control module and how they interact with other Privileged User Manager modules.

Command Control Console	
Information	Provides the user interface with the command control database to allow creation and management of command control rules.
Interacts with	Framework Console - Provides the main user interface. Command Control Manager - Contains the command control information.
Platform support	All supported platforms.
Command Control Manager	
Information	Contains the database of command control rules.
Interacts with	Command Control Console - Provides a user interface to the Command Control module. Command Control Agents - Provides client functionality for the Command Control module.
Platform support	All supported platforms.
Command Control Agent	
Information	Provides the client end of command control including shells and remote execution binaries.
Interacts with	Command Control Manager - Obtains command authentication information. System Information Agent - Obtains remote host information from alternative hosts. Audit Manager - Sends collected audit information to the Audit Manager.
Platform support	All supported UNIX platforms.
System Information Agent	
Information	A background agent that gathers host and user information on the agents.
Interacts with	Command Control Agents - Supplies requested information regarding hosts and users.
Platform support	All supported UNIX platforms.
Audit Manager	
Information	Contains the database of all command control activity and user logged sessions.
Interacts with	Command Control Reports - Provides a user interface to command control audited events. Command Control Agents - Receives the auditing information.
Platform support	All supported platforms.
Reporting	
Information	Provides the primary user interface for auditing which you then add plug ins to view appropriate package information.

Interacts with	Framework Console - Provides the main user interface. Command Control Reports - Provides the command control auditing interface.
Platform support	All supported platforms.
Command Control Reports Console	
Information	Provides the command control auditing interface that reports on all successful/ unsuccessful command control events, as well as captured sessions along with replay.
Interacts with	Audit Manager - Contains the audited information. Reporting - Parent console of Command Control Reports Console.
Platform support	All supported platforms.