LogMatrix

NerveCenter

Best Practices Guide for NerveCenter System and Security Administrators

Windows and UNIX

Version 5.x – Version 6.x

July 2011

NCBPSA5200-05

Copyright

Disclaimers

LogMatrix, Inc. ("LogMatrix") makes no representations or warranties, either expressed or implied, by or with respect to anything in this manual, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

These applications are available through separate, individual licenses. Not every feature or application described herein is licensed to every customer. Please contact LogMatrix if you have licensing questions.

No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, photocopying, recording or otherwise, without prior written consent of LogMatrix. While every precaution has been taken in the preparation of this book, LogMatrix assumes no responsibility for errors or omissions. This publication and the features described herein are subject to change without notice.

The program and information contained herein are licensed only pursuant to a license agreement that contains use, reverse engineering, disclosure and other restrictions.

Trademarks

LogMatrix is registered in the U.S. Patent and Trademark Office. NerveCenter and the LogMatrix Logo are trademarks of LogMatrix, Inc.

All other products or services mentioned in this manual may be covered by the trademarks, service marks, or product names as designated by the companies who market those products.

Contacting LogMatrix

LogMatrix, Inc.
4 Mount Royal Ave, Suite 250
Marlborough, MA 01752

Phone 508-597-5300
Fax 774-348-4953

info@logmatrix.com

Website: www.logmatrix.com
Forum: http://community.logmatrix.com/LogMatrix
Blog: www.logmatrix.com/blog

Best Practices for System Administrators and Security Administrators

This document discusses best practices and guidelines for Administrators who manage the system on which NerveCenter is installed and runs. The purpose of this document is to describe some of the processes and procedures that will help you get the most from your NerveCenter installation. This is by no means the only list of tasks you should consider but we hope this document helps you begin to manage your NerveCenter environment in an efficient, safe and prosperous manner.

This document is written assuming you are running NerveCenter release 5.1 or later. If you are running an earlier release of NerveCenter please contact Customer Support for more information on steps you can take to either upgrade or better manage your current environment.

What is NerveCenter?

NerveCenter is a Framework that can be used to build and customize the network management and application management that meets your business needs. NerveCenter is built using a finite state machine architecture. This architecture allows you to build simple yet elegant models to manage any type of device or application.

NerveCenter was designed to work either standalone as your Network Management platform or in conjunction with other Network management products and devices to provide a custom solution that can be individualized to your site’s needs.

At the heart of NerveCenter is its event correlation engine. For each device that it is monitoring, NerveCenter creates one or more finite state machines-or alarms-that define operational states it wants to detect. NerveCenter also defines rules that effect transitions between the operational states. These rules can be very simple; for example, a state transition can be caused by the receipt of a generic Simple Network Management Protocol (SNMP) trap. Or they can be quite complex and take advantage of NerveCenter's support for Perl expressions.

When deploying NerveCenter there are many areas to consider, just as with many large management products. We have tried to categorize the information into major activity groupings.

Recommended Platforms on Windows and UNIX for NerveCenter Server and Client applications

For a full description of Requirement for installing and running NerveCenter on UNIX and Windows platforms, please refer to the Release notes for your NerveCenter release. NerveCenter is supported on Linux, Solaris and Windows platforms.

NerveCenter as a Service

Before you even install NerveCenter at your site, you need to determine the specific goals of the installation. NerveCenter needs to be thought of as your Network or Application Management Service even if it is integrated into other Management components at your site. When deciding where and how to install NerveCenter, you need to think about the same questions as you would for a Database product, Domain Name Service (DNS), Inventory Control System or any other critical operation in your environment.

The health of your network depends on dedicating the appropriate resources to the task of managing your network. The same is true for managing critical applications.

Questions you should be asking are:

What types of devices or applications will I be monitoring?

For a large scale site with lots of Network devices and machines to monitor, a UNIX based system is a good choice. Smaller sites can choose either a Windows based server or Linux system running on a stand-alone host or Virtual machine.

Application monitoring can be done on either UNIX or Windows, but monitoring WMI applications needs a Windows based server. Currently there is no efficient way to monitor WMI applications remotely on a UNIX based platform.

How do I determine scale? How many devices can I manage on a NerveCenter Server?

Unfortunately there is not simple formula for calculating the number of managed nodes on a NerveCenter server. You must consider the type of network you will be polling:

- Is it a single LAN?
- A WAN spread of a large geographic area?
- What is the bandwidth of the network backbone?
- What Poll rate (polls per second) will be required?
- What is the reliability of the network, does it vary depending on location?

- How much failover can you support (what rate of retry of polls is acceptable to you)?
- What type of polling will you do?

After you have calculated all the values above that you can, the next step is to setup a mock up server to test your upper limits. You may obtain a Trial license to perform the trial runs. Customer Support can help you determine if the system you have selected as your server will meet your minimum requirements.

Remember to factor in for some increase in polling. Once you or the NerveCenter development engineers begin to write models and understand the subtleties of your network, the models will undoubtedly become more complex.

Should my NerveCenter server be on a dedicated system?

If you plan to run NerveCenter to monitor a small number of nodes or applications you may consider having the server run on a machine with other applications. These applications should not require dedicated network bandwidth or CPU usage.

For large sites, NerveCenter should be on a dedicated system. This does not mean you cannot run other Management software on the same system, but try not to have a service like the Master DNS, your company Database server, the Active Directory Server, a NFS server or anything else requiring a lot of network bandwidth and dedicated CPU usage. These services will compete with NerveCenter for network resources and slow your polling performance.

What other applications or services should I install on my NerveCenter server?

All NerveCenter servers should have a Time Service or appropriate policy for ensuring time is in sync between the server and the managed objects (nodes, interfaces and applications.) Without this it will be virtually impossible to analyze the results of your Models to determine the true cause of problem reports.

A backup service is ideal, but a backup policy with manual intervention is also acceptable for smaller sites.

For sites requiring high levels of security, consider deploying an access control and auditing package. This can be a 3rd party package, or something built in-house.

Installation

For a first installation of NerveCenter:

1. Obtain your license key for the machine where you wish to run NerveCenter. This is done by contacting Customer support.
2. Read the release notes for the specific release of NerveCenter. Also read through the Installation manual for the specific Operating System where you will install NerveCenter.
3. If you wish to run your NerveCenter server on UNIX, but your client applications on Windows, make sure you obtain both the appropriate UNIX kit (Linux or Solaris) and the Windows client application kit.
4. Obtain a list of all the nodes you wish to manage, if you are installing a production system.
5. Gather any information needed to configure your SNMP Agent if desired. Refer to the installation manual for information on these steps. You can install your server and then bring up the ncadmin application SNMP Agent tab to see the information required to setup and configure your SNMP Agent in NerveCenter. (Refer to the online documentation for more assistance on configuring your SNMP Agent.)
6. Setup User accounts and groups (on UNIX) necessary for the installation. See the User accounts section of the Best Practices Guide and the Installation Guide for your NerveCenter Platform. For Windows installations prior to 6.0 release the User accounts will be created automatically.
7. Obtain a privileged account for the installation. Only the installation process on UNIX requires a privileged account. You can setup the system to run as "non-root". For Windows, an account with Administrator privileges is required for installation. This is necessary on Windows to update information in the Windows Registry.
8. Determine the name of the directory under which you wish to install NerveCenter on UNIX. For UNIX platforms, NerveCenter installs the server under */opt/OSInc* or will symbolically link */opt/OSInc* to a location of your choice. We recommend the latter choice with a directory name that signifies which version of NerveCenter you are installing. For example: ***/opt/OSInc_NerveCenter5.2*** would indicate a NerveCenter 5.2 installation on your system. Using this method allows you to install multiple versions of NerveCenter Server on a UNIX platform. ***Note: Only one NerveCenter server can be run at a time.***
9. Follow the appropriate installation steps and complete your installation.
10. Turn on Automatic start on Reboot (unless you have a strong desire to not use this feature.)
11. For a new Production system, copy over any Models you wish to install on this system if you have previously designed any on a Development platform.

12. Populate your Node list and turn on alarms.

Upgrades

When upgrading NerveCenter, perform the following steps:

1. Read all the release notes prior to starting the upgrade
2. When upgrading to several releases higher than your current release, contact Customer Support if you have any questions about the procedure.
3. Backup all relevant data. This includes:
    a. MIB files and compilations
    b. NerveCenter node database – **nervecenter.node** file UNIX or SQL Server Database on Windows
    c. NerveCenter database – **nervecenter.ncdb** file on UNIX or SQL Server database on Windows
    d. NerveCenter configuration – nervecenter.xml file on UNIX.
    e. NerveCenter license - **<hostname>.dat** file on UNIX and Windows.
    f. Added Perl modules
    g. In-house scripts added to the current/prior NerveCenter installation.
    h. Models added to NerveCenter models library - */opt/OSInc/model* on UNIX; *C:\Program Files\LogMatrix\NerveCenter\Model* on Windows. For Windows 64-bit installations, look in *C:\Program Files(x86)* directory path.
4. If you are upgrading a system integrated with HP Openview NNM or IBM Tivoli Netcool make sure you understand all the requirements necessary or contact Customer Support for assistance.

When upgrading your site on UNIX, if possible, consider removing just the */opt/OSInc* symbolic link and installing into a new directory tree. You can then move over your database and configuration files while leaving your previous server is place as a backup during the process. You can switch back between the two (by changing the symbolic link) in case you need to verify an unexpected behavior.

If this is not allowed due to site policy, make a backup of the entire directory tree prior to the upgrade (with NerveCenter server not running.)

User Accounts for NerveCenter on UNIX

NerveCenter requires a user account called **nervectr** for NerveCenter to use for installation. Prior to installing NerveCenter you should create this account and two group entries in the */etc/group* file: **ncusers** and **ncadmins**

Once you have created the group entries on your UNIX platform, make sure to place the user account **nervectr** into the group file.

> Sample */etc/group* NerveCenter group entries:
> **ncadmins::78833;nervectr,root,my_admin_account**
> **ncusers::78834;nervectr, my_nervectr_user_account**

Once you have installed NerveCenter on your system, if you wish you can modify the **nervectr** account to be **nologin** and setup other accounts to use to access NerveCenter. The **nervectr** account will be used whenever you do an upgrade.

When you login to NerveCenter you are using a UNIX login account. In order to access NerveCenter administration functions (via **ncadmin** application) you must be in the **ncadmins** group. In order to access NerveCenter design and monitoring functions (via the **client** application) you must be in the **ncusers** group.

For security purposes, make sure you explicitly place the accounts needing NerveCenter group access in the */etc/group* file and not just the */etc/passwd* file.

User Accounts for NerveCenter on Windows

NerveCenter 5.1 plus releases of NerveCenter automatically create the user accounts necessary to run NerveCenter. For NerveCenter 6.0, a new installation process will be used. This document will be updated when NerveCenter 6.0 becomes available. Please refer to the release notes for the appropriate Windows release for additional information.

Database Files on NerveCenter Windows Application

For NerveCenter 5.1 Windows Server installations, there are two database choices, Microsoft Access (shipped with product) and Microsoft SQL server. The installation process will guide you through these choices.

For NerveCenter 6.0 Windows Server Installations, the database selection option will be changing. Please contact your sales representative or customer support for more information about enhancements to NerveCenter 6.0 on Windows.

This section of the document will be updated when NerveCenter 6.0 becomes available.

General Issues Regarding Login to the NerveCenter Server

NerveCenter user accounts and passwords should follow the guidelines for logins under your site security policy. Like any password protected account, passwords selected should be unique, alphanumeric, changed on an appropriate basis.

Some sites may have additional restrictions on user names and passwords. This requirement, as long as they follow the basic principles of logins for your Operating system platform, will work seamlessly with NerveCenter.

For UNIX platforms, NerveCenter uses the PAM architecture for login validation. See the later section on Securing your NerveCenter Installation for more details. For Windows platforms, NerveCenter logins can be integrated with Active Directory or other 3rd party products. Refer to the documentation for your 3rd party product for information about proper configuration with applications using standard Operating System logins.

The NerveCenter login process can be integrated with the following:

- Active Directory
- NIS+
- LDAP
- 3rd Party Products like Centrify
- Kerberos V5
- RSA

Backups

Backups on NerveCenter should be done on a consistent basis. The frequency will depend on your site policy. We recommend a full NerveCenter backup after every upgrade, new Model deployment or major configuration change.

For UNIX Production systems, backup the following on a more frequent schedule (at least bi-weekly):

- The **nervecenter.xml** file in the */conf* directory
- The **nervecenter.node** file in the */db* directory
- The **nervecenter.ncdb** file in the */db* directory
- The */model* directory
- Any scripts you have updated specific to NerveCenter
- Any Perl enhancements made to the Perl library
- The MIB files in the */mibs* directory

For Windows Production systems, backup the following on a more frequent schedule (at least bi-weekly):

- The SQL Server database
- The MIB files in the */mib* subdirectory
- The */model* directory
- Any Perl enhancements made to the Perl library

Also, it is extremely important to back up your **license files** and store them in a safe location.

Backups of all files can generally be done "in place", that is without shutting down the server. However, for full system backups, please schedule during a time when you can shut down the server and do a complete system backup.

System backups do require physical access to the NerveCenter server. You can do incremental backups of the Node database and Models through the **Export Objects and Nodes** feature in the **client** application. (Refer to the online documentation for more information on **Export Objects and Node Features**.)

File Layouts

The layout of files for Windows platforms and UNIX platforms is similar but not exactly the same. This is due to the specific file naming conventions on the various platforms.

Not all directories used within NerveCenter are described here. Those locations where you need to perform backups or where you find critical files are listed.

For UNIX systems, the follow table shows the important file locations used on a NerveCenter Server.

Directory Name	Description
/opt/OSInc	Directory on UNIX where NerveCenter server is

	installed, or symbolic link to directory where NerveCenter is installed.
/opt/OSInc/userfiles/logs	Default local for Log files created automatically as an Action in an Alarm
/opt/OSInc/userfiles	Default location of User environment scripts. These files can be modified by the user or copied and placed in any startup environment script.
/opt/OSInc/mibs **	MIB files and compiled MIB file are stored here
/opt/OSInc/db **	The NerveCenter Server database files are found here.
/opt/OSInc/man	NerveCenter Man pages
/opt/OSInc/conf **	The NerveCenter Server configuration files are found here. The license file is also located in this directory
/opt/OSInc/model **	Location of the Models stored on the Server (there is a similar directory on Client only systems.)
/opt/OSInc/lib/perl5 **	Location of the Perl libraries
/var/opt/NerveCenter	Maintains the logs, statistics and control files for the Poller processes, NerveCenter statistics and performance files
/var/opt/NerveCenter/ncsnmppoller.<#>	Directory holding control information and statistics on each NerveCenter Poller. This information is important if troubleshooting a poller process
/var/opt/NerveCenter/reports *	
/var/opt/NerveCenter/tables *	
/var/opt/NerveCenter /tables/polls *	
/var/opt/NerveCenter/tables/nodes *	
/var/opt/NerveCenter/log/	
/var/opt/NerveCenter/log/protocol *	Statistics on NerveCenter Running Protocols
/var/opt/NerveCenter/log/polling *	Statistics on NerveCenter Polls

*NerveCenter 5.2 release and later.

**Backups recommended.

For Windows platforms, the following table shows the directories of importance. For 64-bit architectures, replace \Program Files with \Program Files (x86).

Directory	Description

\Program Files\LogMatrix\NerveCenter\MIB**	MIB files storage Location
\Program Files\LogMatrix\NerveCenter\conf**	Configuration files including the license file
\Program Files\LogMatrix\NerveCenter\Model**	Model files stored on the system

**Backups recommended.

NerveCenter Startup and the rc.d system on UNIX platforms

Starting NerveCenter on UNIX platforms can be done automatically through the rc.d system. The recommendation is to stop NerveCenter at rc2 level and start NerveCenter at rc3 level. The defaults are:

```
/etc/rc3.d/S94ncservice
/etc/rc2.d/K94ncservice
```

The ncservice file can be found in **/etc/init.d/ncservice**. Do not edit this file without complete understanding of the consequences. If you wish to edit this file, make a backup of the file before starting. Do not run this file directly or put the **/etc/init.d** directory in your path.

When the UNIX platform boots, reboots or shuts down, the appropriate file will be executed by the Operating System.

Starting NerveCenter from the Command Line on UNIX

When starting or stopping NerveCenter from the command line on UNIX, use the **ncstart** and **ncstop** control scripts. These scripts are written to appropriately stop the NerveCenter server and all its dependencies.

Example **ncstart** output:

```
# /opt/OSInc/bin/ncstart
brassd stopped
```

The **brassd** stopped message will be seen if the **brassd** services had been left running for some reason. This is not an error.

Example **ncstop** output:

```
# /opt/OSInc/bin/ncstop
```

```
Stopping NC Server … <pid>
Waiting 3 seconds for ncserver to terminate.
Waiting 3 seconds for ncserver to terminate.
brassd stopped
```

The <**pid**> is the process id of the NerveCenter server process. Do not worry if you see the line *"Waiting 3 seconds for ncserver to terminate."* The **ncstop** script is waiting for the Server to respond that it has cleanly shut down its sub-processes and threads.

NerveCenter Startup on Windows platforms

NerveCenter is installed on Windows using the Windows installer guidelines. For older versions of the NerveCenter installation, you will find the NerveCenter applications and Server under the Start Button in **OpenService NerveCenter**.

The newer installations of NerveCenter can be found under the Start Button in **LogMatrix NerveCenter**.

The NerveCenter Server on Windows platforms is always started as a service. This insures that the server will continue to run after the user has logged out of the Windows system.

If you are installing the NerveCenter Applications Windows installation kit, it will be installed under **LogMatrix NerveCenter.**

The NerveCenter Applications Windows kit provides only the NerveCenter **client** and **ncadmin** applications and the documentation set.

Using Virtual Machines

NerveCenter can be deployed on Virtual machines as easily as on a stand-alone system. The only caveat relates to the resources available to each Virtual machine. The actual Processing which runs all your Virtual Images is limited by its hardware configuration. Do not assume because you have virtual machines, you can suddenly run 10 NerveCenter servers and 5 database servers on a set of Virtual machines running on the same VM server.

We generally recommend **1** production NerveCenter server per VM server. Running the NerveCenter server on a system with other services that require a lot of resources requires careful analysis. There are no hard and fast rules for how much you will be able to run on your platform. Using the NerveCenter statistics gathering may help you calculate your upper bound limit.

Virtual Machines are very useful for Development platforms. Assuming you are not running an extremely heaving load, you could run a development system or two on the same Virtual Machine as a small production system or a Backup server. Having the ability to do snapshots of your development environment and install multiple versions of NerveCenter for comparative analysis is made much easier with Virtual machines.

Securing your NerveCenter Installation

There are a number of things to consider when securing your NerveCenter installation. There is the physical security of the system. This guide will not cover that topic as it should be covered by other corporate security policies. Nothing written in this section is intended to override any security policies written for your company.

There are several steps you can take when security NerveCenter to allow you to meet or exceed the security requirements for your site.

UNIX Systems Security

On UNIX based installations, you can configure NerveCenter to run under a non-root account. Refer to the document ***Securing a NerveCenter Installation to allow non-root Management of NerveCenter*** for more information on the steps needed to run NerveCenter from a non-root account. You may obtain a copy of this document by contacting Customer Support.

For user accounts on UNIX, NerveCenter implements logins with NerveCenter client and NerveCenter ncadmin through the use of PAM (Pluggable Authentication Modules.) Refer to Appendix A for a Description of PAM and its use in NerveCenter.

Additionally on UNIX, you can and should use good practices to manage password and group files. NerveCenter can be integrated with 3rd party user and password management products. As long as these products are integrated using PAM, their use is seamless to the user.

When running NerveCenter on all platforms, DNS configuration is critical. All systems should ensure that the NerveCenter server platform and managed Nodes can be uniquely identified by NerveCenter through the resolver. At a minimum, the host file must be configured correctly. The use of an appropriately managed Master DNS which provides name resolution is ideal. That ensure the IP addresses of systems are correct on discovery and you will know which host is being polled or 'pinged'.

A time service should be running in your environment. This service is necessary if you are running enhanced security packages using tickets or credentials. It is also necessary if to ensure the timestamps written to log files are understandable. Having a clock out of sync makes analyzing data very difficult. It will be impossible to trust which actions happened first on different platforms. Many security packages require accurate timestamps and hostname to IP Address resolution to guarantee secure transmissions.

Windows Security

On Windows platforms, the Windows Application kit has minimal impact on the security of your system. In order to update the Registry, you should install the NerveCenter Windows Application kit with administrator privileges, but it can be run by any users for whom you have allowed access.

The Windows Application kit installs the following:

- Ncadmin application
- Client application
- Nccmd application
- mibTool application
- NerveCenter on-line and PDF documentation

The NerveCenter Server kit for Windows (full kit) has a greater impact on the system and requires more careful planning. The Windows Firewall must allow access for all the Ports required to run NerveCenter. See the section on NerveCenter and Firewalls. The installation must be done with an account running with Administrator rights. This is necessary to update the Windows Registry. Currently, NerveCenter on Windows should be run under and account with Administrator rights. The Server must have access to the database selected (MS Server or Access) to be able to write updates to the system. The server must also be able to update entries in the registry.

NerveCenter and Firewalls

There are a few requirements for system access required to run NerveCenter. In 5.2 and future releases NerveCenter installation is attempting less restrictive on access and resources. NerveCenter server does require the ability to access some well know ports on the system, namely SNMP, SNMP trap and well defined NerveCenter ports. If you are running a firewall on your NerveCenter server, you must allow access to specific ports.

The Ports requiring access are:

Port Name	Port Number (default)

NerveCenter Administrator/Client	32504
NerveCenter Inform	32505
NerveCenter Command Line Application (nccmd)	32506
OV-Trapper (for communicating between NerveCenter and OpenView NNM, if needed)	32507
SNMP Poll	161
SNMP Trap	162

You may modify these ports number values through the NerveCenter **ncadmin** program if desired.

Appendix A: PAM and NerveCenter

PAM is the Pluggable Authentication Module provided on UNIX platforms for authenticating applications and services. PAM is a set of shared libraries which an application is linked against when it needed to authenticate a user. The design of PAM allows for extensions to the libraries to add support for additional security checking. In addition, the libraries are designed in a secure manner to prevent outside attack.

The following sections give a detailed description of the steps used when validating a user with PAM. At the end of the section is a sample PAM configuration file which is supplied by the System Administrator or Security Administrator and defines the login security protocol for the system.

The information in the following selections was gathered from several sources. These include the Red Hat man pages for PAM, the Solaris man pages for PAM and the Linux forums.

To learn more about PAM check out the following sites:

- http://linux.die.net/man/3/pam
- http://www.kernel.org/pub/linux/libs/pam/Linux-PAM-html/ - The Linux-PAM Guides
- http://www.kernel.org/pub/linux/libs/pam/Linux-PAM-html/Linux-PAM_SAG.html – Linux PAM System Administrators’ Guide
- http://www.kernel.org/pub/linux/libs/pam/modules.html - lists of PAM modules available for use. Please research a module very carefully before using it in our environment. All modules should be cleared by a Security Administrator or equivalent.

- http://content.hccfl.edu/pollock/AUnix2/PAM-Help.htm

Please refer to your Operating Systems latest documentation on PAM for complete details on implementation of the libraries and configuration files.

PAM Modules Description

Authentication Service module ← provides functionality to authenticate a user and set up user credentials.

Account Management module ← provides functionality to determine if the current user's account is valid. This includes checking for password and account expiration, or will as verifying access hour restrictions

Session Management module ← provides functionality to setup and terminate login sessions

Password Management module ← provides functionality to change a user's authentication token or password

Format of PAM definition in the PAM configuration file:

Service name module type control flag module path options

Options ← pass in module specific arguments to the module through the PAM Framework

Pam_dhkeys ← authentication Diffie-Hellman keys management module (pam_dhkeys.so.1) The service module provides functionality to two PAM Services: Secure RPC authentication and Secure RPC authentication token management.

Secure RPC authentication differs from regular UNIX authentication because NIS+ and other ONC RPCs user Secure RPC as the underlying security mechanisms.

Authentication Services

If the user has Diffie Helman keys, pam_sm_authenticate() establishes secret keys for the user specified by the PAM_USER (equivalent to running keylogin(1)), using the authentication token found in the PAM_AUTHTOK item.

pam_authtok_get.so.1 ← the service provides password prompting functionality to the PAM stack. It implements pam_sm_authenticate() and pam_sm_chauthtok() providing functionality to both the Authentication stack and the Password Management stack.

How Authentication Service Works

The implementation of the pam_sm_authenticate (3PAM) prompts the user name if not set and then tries to get the authentication token from the PAM handle. If the token is not set, it then prompts the user for a password and stores it in the PAM item PAM_AUTHTOK. This module is meant to be the first module on an authentication stack where users are to authenticate using a keyboard.

Password Management Service

Due to the nature of the PAM Password Management stack traversal mechanism, the pam_sm_chauthtok(3PAM) function is called twice. Once with the PAM_PRELIM_CHECK flag and once with the PAM_UPDATE_AUTHTOK flag.

Return values on authentication service calls:
PAM_SUCCESS ← successfully obtains authentication token
PAM_SYSTEM_ERR ← fails to retrieve username. Username is NULL or empty

pam_unix_auth ← PAM authentication module for UNIX (pam_unix_auth.so.1) The pam_unix_auth module impliments pam_sm_authenticate() which provides functionality to the PAM authentication stack. It provides functions to verify that the password contained in the PAM item PAM_AUTHTOK is the correct password for the user specified in the item PAM_USER. If PAM_REPOSITORY is specified, the users' password is fetched from that repository. Otherwise the default nsswitch.conf repository is searched for that user. The following options can be passed to the module:

- Server-policy – if the account authority for the user, as specified by PAM_USER, is a server, do not apply the UNIX policy from the password entry in the name service switch

Returned Values:

- PAM_AUTH_ERR ← authentication failure
- PAM_BUF_ERR ← Memory buffer error
- PAM_IGNORE ← ignore module, not participating in result
- PAM_PERM_DENIED ← permission denied
- PAM_SUCCESS ← successfully obtains authentication token
- PAM_SYSTEM_ERR ← system error
- PAM_USER_UNKNOWN ← no account present for the user

pam_unix_account.so.1 ← PAM account management module for UNIX. Pam_unix_account provides functionality to the PAM account management stack. The function retries password aging information from the repositories specified in the nsswitch.conf (4) and verifies that the user's account and password have not expired. The options for use with this function are:

- nebug ← turn on debugging
- nowarn ← turn off warnings

- server_policy ← if the account authority for the user asspecified by PAM_USER, is a server, do not apply the UNIX policy from the password entry in the name service switch.

Errors:

- PAM_AUTHTOK_EXPIRED – password expired an no longer usable
- PAM_BUF_ERR – memory buffer error
- PAM_IGNORE – ignore module, not participating in result
- PAM_NEW_AUTHTOK_REQD – obtain new authentication token from the user
- PAM_SERVICE_ERR – error in underlying service module
- PAM_SUCCESS – successfully obtains authentication token

pam_ldap.so.1 ← authentication account and password management PAM module for LDAP

Role Account Management Module

The Role account management component provides a function to check for authorization to assume a role. It presents direct logins to a role, It uses the user_atr(4) database to specify which users can assume which roles. The only option used is *debug*. This module is generally stacked above pam_unix.so.1

pam_roles.so.1 ← The Role Account Management module for PAM, pam_roles.so.1, provides functionality for ONE PAM module: Account Management. The pam_roles.so.1 is a shared object that can be dynamically loaded to provide the necessary functionality upon demand. Its path is specified in the PAM configuration file.

Projects Account Management Module

The project account management component provides a function to perform account management, pam_sm_accnt_msnt(). This function uses the getdefaultproj() function (see getproject(3PROJECT)) to retrieve the user's default project entry from the project(4) database. It then sets the project ID attribute of the calling process, using the settaskid(2) system call.

If the user does not belong to any project defined in the project(4) database or if the settaskid() system call failed to set the project ID attribute of the calling process, the module will display an error message and will return error code PAM_PERM_DENIED.

pam_projects.so.1 ← account management PAM module for projects (pam_projects.so.1)
The projects service module for PAM, pam_projects.so.1, provides functionality for the account management PAM module. The pam_projects.so.1 module is a shared object that can be dynamically loaded to provide the necessary functionality upon demand. Its path is specified in the PAM configuration file.

pam_projects.so.1 is designed to be stacked on top of the pam_unix_account.so.1 module for all services. This module is normally configured as “required”, implying that any user lacking a default project will be denied login.

PAM Usage in NerveCenter – Sample file

```
#%PAM-1.0
# Sample NerveCenter/RHEL PAM configuration using /etc/pam.d/system-auth
auth       required     pam_stack.so service=system-auth
auth       required     pam_nologin.so
account    required     pam_stack.so service=system-auth
```

The previous example shows the basic recommendation for PAM usage with NerveCenter. You can modify this file to make your site more secure by incorporating 3rd party products or adding your own Modules. You are responsible for testing 3rd party integration or libraries not distributed with the standard UNIX distribution. The program **nctestlogin** provided in NerveCenter can help test and verify your PAM configuration.