Interested in learning more about security?

SANS Institute InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Securing GroupWise 6.5 for SUSE Linux Enterprise Server 8 and Novell Nterprise Linux Services 1.0

The purpose of this document is to help readers properly secure a Novell GroupWise 6.5 (GroupWise) environment installed on SUSE Linux Enterprise Server 8 (SLES) using Novell Nterprise Linux Services 1.0 (NNLS). We begin by providing information pertaining to the installation environment that is used throughout the document.
We'll start working with OS security by properly securing our installation of SLES. Commonly required services that can cause extra vulnerabilities are also secured. Following the hardening of SLES...

Copyright SANS Institute

Author Retains Full Rights

AD

Securing GroupWise 6.5 for SUSE Linux Enterprise Server 8 and Novell Nterprise Linux Services 1.0

by Ben Austin
GIAC Security Essentials Certification (GSEC)
Practical Assignment, Version 1.4c. Option 1
Submitted: October 22, 2004.

© SANS Institute 2005, Author retains full rights.

© SANS Institute 2005 Author retains full rights.

Table of Contents

Abstract….....3
Installation Environment Details.....….3
SLES Environment.....…….3
NNLS Environment…….....4
eDirectory Environment……...5
GroupWise Environment…….6
Hardening SUSE Linux Enterprise Server 8.....…….….6
Disable Non-Essential Services.....6
Service Configurations…….....8
GRUB…….8
SSH……...10
Securing Novell Nterprise Linux Services……..11
NNLS Daemon Configurations……11
iManager……..11
Apache2……11
eDirectory……14
Filesystem Rights……14
Tripwire……14
Directory Security……15
LDAP……16
Securing GroupWise 6.5 for Linux……...18
Access Methods……..18
Password Settings……..19
Traffic Encryption……20
Spam……20
Viruses……..21
Conclusion……...21
References……..22
Appendix A - /etc/opt/novell/nterprise_linux_services_install.conf…….23

© SANS Institute 2005, Author retains full rights.

Abstract

The purpose of this document is to help readers properly secure a Novell GroupWise 6.5 (GroupWise) environment installed on SUSE Linux Enterprise Server 8 (SLES) using Novell Nterprise Linux Services 1.0 (NNLS). We begin by providing information pertaining to the installation environment that is used throughout the document. We'll start working with OS security by properly securing our installation of SLES. Commonly required services that can cause extra vulnerabilities are also secured. Following the hardening of SLES, we continue by addressing NNLS security, particularly the components required to support the subsequent installation of GroupWise. Finally, the initial installation of GroupWise will be hardened to provide a more secure environment. Once the above steps are complete, a functionally secured GroupWise environment will be ready for deployment. In conclusion, we discuss expanding this scenario to include such issues as growth and scalability.

Securing every aspect of each component is worthy of complete books (if not volumes) on each subject. For this reason, we will be applying best-practice security measures that secure the environment in general, as a whole. Resources for more detailed and complex configurations are noted throughout this document.

Installation Environment

The installation environment used as a base for all examples and scenarios throughout this paper is laid out here. SUSE Linux Enterprise Server 8 will be installed using a customized install that included several variations.

First, we will modify the recommended partition scheme. The root (/) Filesystem should be able to "boot, restore, recover, and/or repair the system"[1] in the event of any daemon related problems. For this reason, the partition scheme was changed to provide the following mount points: /, /var, /opt, /groupwise, and swap (approximately 2 times the amount of RAM up to 1GB). One form of DoS attacks is to fill up the root (/) partition, sometimes hanging the host OS or at least associated daemons. By creating independent /var and /opt partitions, it enables NNLS to primarily run and log to these areas leaving root (/) alone. This will help limit the effects of eDirectory attacks or log overflows. The /groupwise partition allows all mail to reside on a non root (/) mount point. If the server isn't enforcing quotas correctly or falls victim to a massive Spam attack, it can fill up the mount point completely. While these scenarios will definitely affect the services specifically attacked, administrators would most likely be able remotely clean up invalid content and restart the services involved to minimize downtime.

In the "Software Selection" section, we picked "Minimum graphical system (without KDE)". This gave us a basic setup leaving out most unneeded

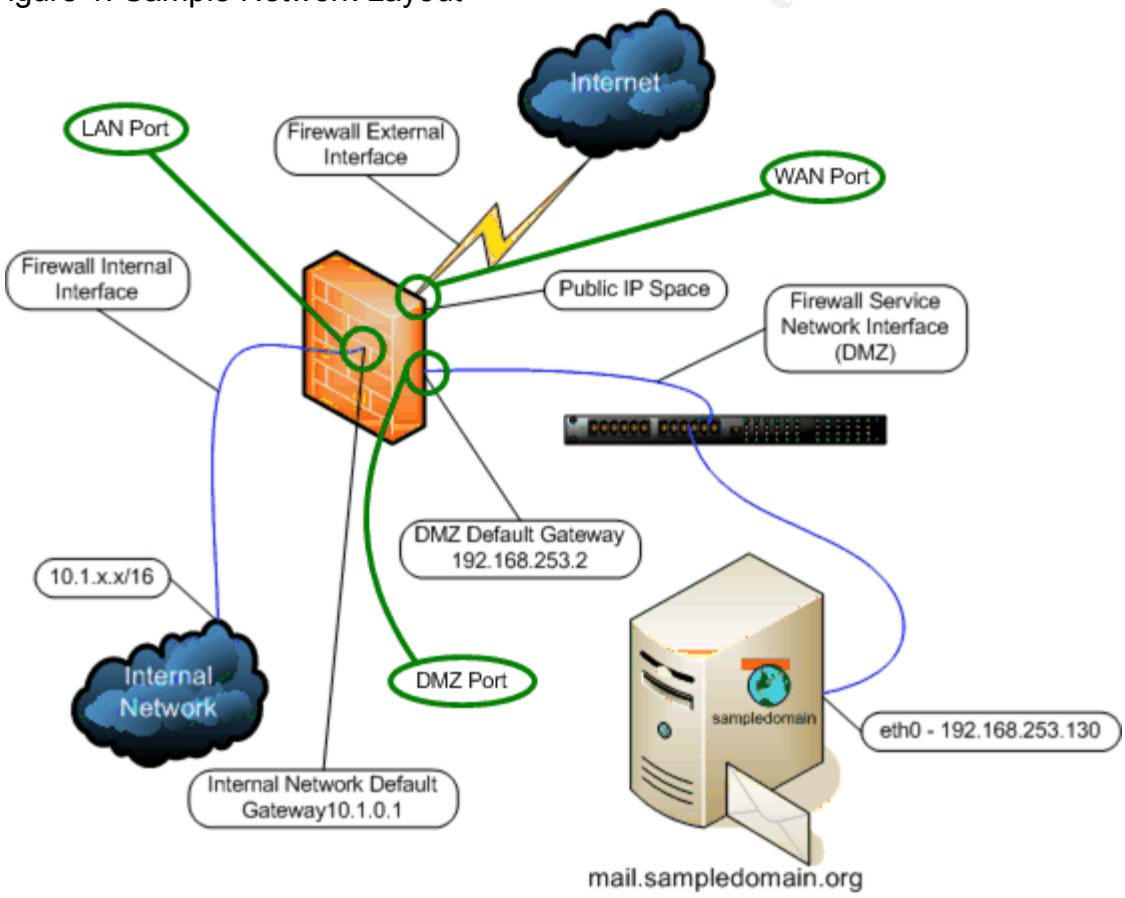
[1] Filesystem Hierarchy Standards – http://www.pathname.com/fhs/pub/fhs-2.3.html#THEROOTFILESYSTEM

© SANS Institute 2005 Author retains full rights.

packages. The only additional packages required are the “C/C++ Compiler and Tools” section and “compat” packages. I added these components as a part of the installation, but they can be added after the installation completes if necessary.

The server’s hostname is ‘mail.sampledomain.org’. Its IP address is 192.168.253.130/24. The default gateway is set to 192.168.253.2. These parameters should be changed for other environments (when prompted during the installation). For this particular environment, we assume an existing network infrastructure and will be placing this server on a perimeter level DMZ as illustrated in Figure 1. The firewall used should have several basic features to support this scenario. It should be able to support service networks (DMZ). It should also support basic rule lists and NAT to entities on the DMZ.

Figure 1: Sample Network Layout



Once the operating system was installed, SP3 CDs were installed via **Y**aST **O**nline **U**pdate (YOU). Once patch CDs were completely installed, YOU was executed again to update the system from the SUSE update site. The initial update will require a reboot to refresh all updated services and boot from a new kernel.

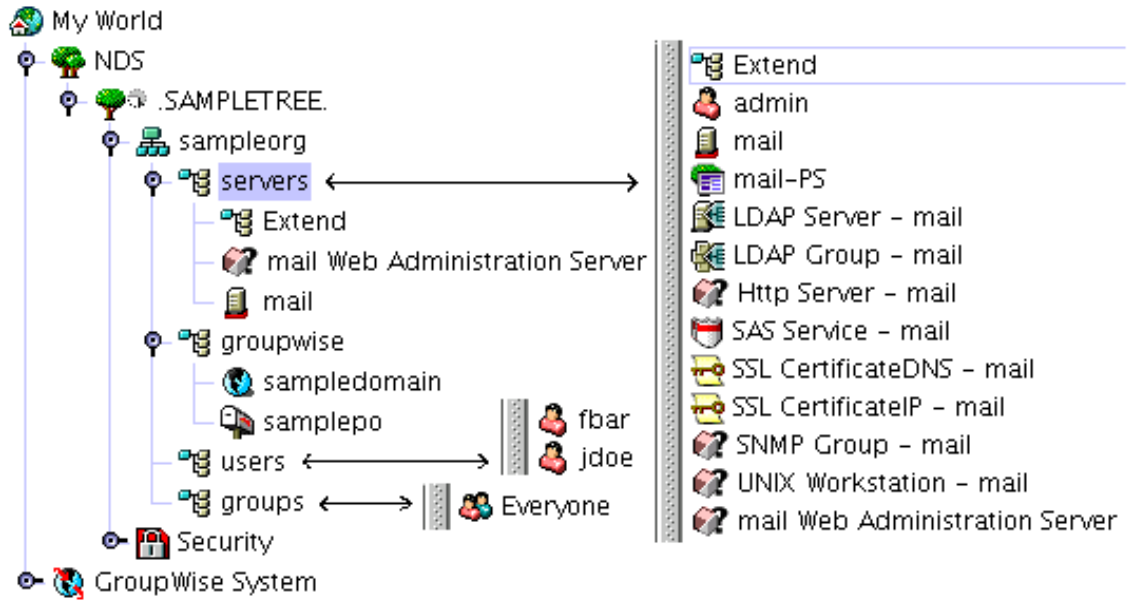
NNLS was installed using the 1.0 tarball distributed from Novell. Defaults were

© SANS Institute 2005 Author retains full rights.

used for all port assignments and daemon related questions. The complete installation configuration file can be found in Appendix A. This file can be used as a reference when answering questions through your own install. NNLS was installed with the bare minimum components required for a successful installation of GroupWise. I installed and configured Apache, eDirectory, iManager, JVM, LUM, and Red Carpet.

A new eDirectory tree named “SAMPLETREE” was created. The completed tree structure is listed in Figure 2.

Figure2: Sample eDirectory Tree



Both users ‘fbar’ and ‘jdoe’ are members of the ‘Everyone’ group. Neither user was created as a LUM user. Since the primary purpose for our server will be to provide GroupWise services, users will not need access other services. Not assigning users to a LUM groups protects the underlying Linux system because non-LUM users do not exist anywhere besides eDirectory. Within our eDirectory tree, all non-administrative users have been placed in their own OU (ou=users.o=sampleorg) and have no special permissions. They have all been assigned to the GroupWise domain ‘sampledomain’.

Post installation, NNLS was also updated using the Red Carpet update client ‘rug’. During NNLS setup, the server will automatically be subscribed to the channel ‘Novell Nterprise Linux Services 1.0’. Typing ‘rug lu’ from a console prompt fetches and displays a list of available updates. Read the list of available updates and apply only the ones applicable to your installation. For example, the SAMPLETREE NNLS installation did not include installing iFolder. So when we see the package ‘novell-ifolder-imanager-plugin’, it should be skipped as it does not apply to this server. If you’re unsure about whether or not a specific patch applies to your server, typing ‘rug info $PACKAGE_NAME’ at a console gives extended information about the package (replace $PACKAGE_NAME with

© SANS Institute 2005 Author retains full rights.

package name you wish to query about). Once all related daemons are restarted following the update(s), NNLS should be completely patched and ready for GroupWise components.

The GroupWise environment used throughout this document consists of a single Domain and Post Office. Other components installed and active include: Internet Agent, Webaccess, Webpublisher, GroupWise Monitor, and GroupWise Messenger. The Domain is named 'sampledomain', and was installed to '/groupwise/sampledomain'. The Post Office is named 'samplepo' and was installed to '/groupwise/samplepo'. The Post Office security level chosen was "High Security". eDirectory user .admin.servers.sampleorg was assigned postmaster. Once the initial installation was complete, I also installed support pack 2 (gwlinux652.tar.gz). This was the latest support pack available at time of writing. I recommend using the latest support packs available while following this document. To search for newer patches visit Novell's patch database at http://support.novell.com/filefinder/. All available updates can be searched for and may also be listed on the "Minimum Patch List" link under "GroupWise 6.5". Some downloads require a login depending on export laws and Novell licensing. GroupWise 6.5 for Linux Support Pack 2 requires one such login. The account signup is free and may not be required for other downloads.

Hardening SUSE Linux Enterprise Server 8

When securing GroupWise services, the operating system (OS) that it resides on must also be secured. Without applying security best practices to the underlying OS, some areas may be vulnerable to attack and have nothing to do with GroupWise. These unnecessary vulnerabilities can eventually take the system down by affecting integrity and/or availability. Since securing every available service and aspect of SLES8 is beyond the scope of this document, you can find much more information in SUSE's "SLES Security Guide" [2]. This guide identifies how each part of the SLES OS should be configured to ensure "Common Criteria" [3] compliancy. A system can be considered to be "CC compliant" if it matches an evaluated and certified configuration[4].

Disable Non-Essential Services

During the installation, we performed a "Minimum graphical system (without KDE)" with a few other required packages. Even though we picked a minimal installation, SUSE still installed and started some services that we really don't need or want. A place to start narrowing these services down is through the runlevel editor included in YaST (Yet another Setup Tool). We can accomplish this by logging into Xfree86 or 'X', opening a terminal, and typing 'yast2'. This will launch the GUI version of YaST. If you prefer to use it from the console (ncurses based), just type 'yast' at a terminal prompt. The choice layout between the

[2] SLES Security Guide, http://www.suse.com/de/security/eal3/SLES8_EAL3_SecurityGuide.pdf
[3] SLES Security Guide, http://www.suse.com/de/security/eal3/SLES8_EAL3_SecurityGuide.pdf
[4] SLES Security Guide, http://www.suse.com/de/security/eal3/SLES8_EAL3_SecurityGuide.pdf

© SANS Institute 2005 Author retains full rights.

graphical and console versions is identical. First, select ‘System’ in the left window pane. Once selected, you should select ‘Runlevel Editor’ from the middle pane. Once this launches, you are taken to a screen that allows selection of the default runlevel after boot. I recommend that production server environments start in runlevel 3 and only start X when required. If started every time the system boots, X and xdm/kdm/gdm can become a security issue. Although there are certain measures that can be taken to protect the windows environment, it’s still for the most part unused in a server environment, therefore taking up resources and causing an unnecessary vulnerability. The GroupWise MTA, POA, and GWIA daemons can all be started with ‘—show’ parameters that launch GUI front-ends as part of the daemon processes, but they aren’t necessary for everyday production use. When experiencing problems with your server, it’s easy enough to start X on demand by typing ‘startx’ from a console and restart the GroupWise daemons with the ‘—show’ parameter to view potential problems as they occur. After changing the runlevel to 3, click on “Runlevel Properties” to customize daemon actions for each runlevel. Now all services in the /etc/init.d directory and inetd/xinetd facility are listed with available runlevels displayed. This list of services should be thoroughly scrutinized to determine which services are valid for everyday operations. Any services that are configured to start on boot and are not essential to GroupWise services can be disabled here. For example, In this case Postfix is enabled on boot by default. Postfix listens on port 25 as does GroupWise. If Postfix is left enabled, the GroupWise Internet agent will be in contention for the same TCP port and problems will arise. To keep a particular service from starting on boot, clear all ‘x’ marks from boxes 0 through 6 that are located on the same line and to the right of each service. For a production GroupWise environment, there are specific services that should and should not be started. Figure 3 shows all services set to start on boot and the runlevels in which they are to be started.

© SANS Institute 2005 Author retains full rights.

Figure 3: Runlevel editor list of enabled services with associated runlevel(s).

```
atd              0:off  1:off  2:on   3:on   4:off  5:on   6:off
cron             0:off  1:off  2:on   3:on   4:off  5:on   6:off
fbset            0:off  1:on   2:on   3:on   4:off  5:on   6:off
grpwise          0:off  1:off  2:off  3:on   4:off  5:on   6:off
grpwise-ia       0:off  1:off  2:off  3:on   4:on   5:on   6:off
grpwise-ma       0:off  1:off  2:off  3:on   4:on   5:on   6:off
grpwise-wa       0:off  1:off  2:off  3:on   4:on   5:on   6:off
hotplug          0:off  1:on   2:on   3:on   4:off  5:on   6:off
hwscan           0:off  1:off  2:on   3:on   4:off  5:on   6:off
kbd              0:off  1:on   2:on   3:on   4:off  5:on   6:off
namcd            0:off  1:off  2:on   3:on   4:on   5:on   6:off
ndsd             0:off  1:off  2:on   3:on   4:on   5:on   6:off
network          0:off  1:off  2:on   3:on   4:off  5:on   6:off
novell-httpd     0:off  1:off  2:on   3:on   4:on   5:on   6:off
novell-rmrcdlink 0:off  1:off  2:on   3:on   4:on   5:on   6:off
novell-tomcat4   0:off  1:off  2:on   3:on   4:on   5:on   6:off
novell-webadmin  0:off  1:off  2:on   3:on   4:on   5:on   6:off
rcd              0:off  1:off  2:off  3:on   4:on   5:on   6:off
rpmconfigcheck   0:off  1:on   2:on   3:on   4:off  5:on   6:off
slpuasa          0:off  1:off  2:off  3:on   4:off  5:on   6:off
sshd             0:off  1:off  2:off  3:on   4:off  5:on   6:off
syslog           0:off  1:off  2:on   3:on   4:off  5:on   6:off
```

Notice that sshd is the only non-novell related service that actually listens on an IP port. Since this server is meant to facilitate a GroupWise environment, all non-essential services are disabled and/or subsequently uninstalled.

Service Configurations

Since physical security is site dependant, I have included very little information pertaining to the subject. For our purposes, I assume the server is mounted on a lockable rack cabinet in a proper server room that is clearly labeled and has appropriate room entry validation equipment installed (locks, biometrics, cameras, etc). After physical security risks have been properly mitigated, the next part of the boot process is initiating the BIOS. Most main stream server manufacturers provide the ability to password protect local access to the BIOS. If this is possible, it could keep someone from altering the hardware configuration, possibly affecting the overall availability of your server.

GRUB

Once the boot process runs through the BIOS, it utilizes a boot loader called GRUB (GRand Unified Bootloader) to access available operating systems. Grub is very powerful and should be protected against unauthorized access.

One powerful capability GRUB has is that it utilizes a built in command shell that can be used for extensive pre-boot operations. It can also access files on local

© SANS Institute 2005 Author retains full rights.

filesystems. For this reason, GRUB provides a *password* feature[5]. The password is requested anytime a user tries to edit the configuration, access the GRUB shell, or boot from a medium other than the production kernel and permitted media. To enable this feature, the line “password --md5 PASSWORD”, must be added to “/boot/grub/menu.lst”. The “PASSWORD” parameter should be replaced by your md5 hash version of a clear text password used to access these extra capabilities. Md5 password hashs can be generated from clear text on the GRUB command line. To access the grub command line, type “grub” at a console prompt. Once you’re in a grub shell, type “md5crypt” and press enter. It will ask for your clear text password. When you hit enter after entering your password, md5crypt dumps the md5 password on the next line (as shown in Figure 4).

Figure 4: Converting a clear text password to an md5 encrypted form.

```
mail:~ # grub

    GRUB  version 0.93  (640K lower / 3072K upper memory)

 [ Minimal BASH-like line editing is supported.  For the first word,
   TAB lists possible command completions.  Anywhere else TAB lists
   the possible completions of a device/filename. ]

grub> md5crypt

Password: ******
Encrypted: $1$n7p8b0$8Ol5iqtv4Odlg1hJW/OBu.

grub> quit

mail:~ #
```

The new md5 password created by this process should be copied to your clipboard and placed in “/boot/grub/menu.lst” as illustrated in Figure 5.

[5] GRUB Manual, http://www.gnu.org/software/grub/manual/html_node/Security.html#Security

© SANS Institute 2005 Author retains full rights.

Figure 5: Sample locked down /boot/grub/menu.lst.

```
color white/blue black/light-gray
default 0
timeout 8
password --md5 1$n7p8b0$8Ol5iqtv4Odlg1hJW/OBu.

title linux
   kernel (hd0,0)/vmlinuz root=/dev/sda5  pci=noacpi  vga=791
   initrd (hd0,0)/initrd
title floppy
   lock
   root (fd0)
   chainloader +1
title failsafe
   lock
   kernel (hd0,0)/vmlinuz.shipped root=/dev/sda5 ide=nodma apm=off acpi=off vga=normal
nosmp noapic maxcpus=0 3
   initrd (hd0,0)/initrd.shipped
```

To prevent someone from booting to a particular stanza, add the “lock” line underneath the title to force password protected booting. For example, in Figure 5 above, a password is required to edit any configuration or booting from any stanza other than ‘linux’.

By default, the “/boot/grub/menu.lst” menu file is world readable (644 octal). This file should only have read/write permissions for the owner. Group and world should have no permissions. Type “chmod 600 /boot/grub/menu.lst” to give it read/write to the owner (root) only.

SSH

The default installation of OpenSSH permits several features that make it more vulnerable[6]:
- Remote root login should be disabled.
- Protocol 2 should be the only version allowed.
- SSH clients should not be set up setuid root.
- Per-user passwords and public key authentication should be the only types of authentication available.
- PAMAuthenticationViaKbdInt MUST be disabled.

Figure 6 lists the resulting “/etc/ssh/sshd_config”. This is a combination of pre-existing default plus the above additions. The sshd daemon must be restarted for the changes to take affect. This can be accomplished by typing “/etc/init.d/sshd restart” at a console prompt. Once restarted, make sure to test connections to validate the changes.

[6] SLES Security Guide, http://www.suse.com/de/security/eal3/SLES8_EAL3_SecurityGuide.pdf

© SANS Institute 2005 Author retains full rights.

Figure 6: /etc/ssh/sshd_config

```
Protocol 2
Ciphers 3des-cbc
PermitRootLogin no
RSAAuthentication no
PubkeyAuthentication yes
IgnoreRhosts yes
RhostsRSAAuthentication no
HostbasedAuthentication no
PasswordAuthentication yes
PermitEmptyPasswords no
ChallengeResponseAuthentication no
KerberosAuthentication no
GSSAPIAuthentication no
PAMAuthenticationViaKbdInt no
X11Forwarding no
UsePrivilegeSeparation no
Banner /etc/ssh/banner
Subsystem       sftp       /usr/lib/ssh/sftp-server
```

Securing Novell Nterprise Linux Services

Since GroupWise requires a local eDirectory environment, NNLS installed iManager, LUM, Apache, Tomcat, eDirectory, and Red Carpet. Several components were left out in order to keep the installation as focused as possible.

LUM is installed, but no user objects are converted to “LUM Users” or added to a “LUM Group”. LUM adds the ability to allow eDirectory users access to local server services using eDirectory credentials. This comes in useful for administrators that want to allow specific eDirectory users access to other services on the machine, such as SSH.

iManager

iManager is Novell’s next-generation eDirectory administration utility. For client security, iManager utilizes SSL/TLS connectivity only. This ensures no clear text between client and server, regardless of location. Once logged in, iManager uses something called Role Based Services (RBS). RBS allow administrators to define scopes of tasks that can be assigned to individuals[7]. For example, if you have a local sysop that needs to have rights for clearing print jobs only, a role could be assigned to that particular user that contains just enough rights to clear print jobs. That user will only see tasks that are assigned to him/her.

Apache2

[7]Novell iManager 2.0.x, Http://www.novell.com/documentation/imanager20/imanager20/data/box3hrz.html

© SANS Institute 2005 Author retains full rights.

We purposefully left Apache out of the initial SLES installation so NNLS could install its own version for use with all subsequent Novell product installs. Apache2 is one of the server daemons responsible for products like iManager, Webaccess, and Webpublisher. Since we have a very specific use for our server, security should be tailored to support those applications. Novell's Apache2 installation places the configuration files in /etc/opt/novell/httpd/ and the document root in /var/opt/novell/httpd/htdocs. The primary configuration file where most changes are made is /etc/opt/novell/httpd/conf/httpd.conf.

Novell's Apache2 implementation is shipped with a feature called Dynamic Shared Objects (DSO). This feature allows directives to be defined in /etc/opt/novell/httpd/conf/httpd.conf by adding a LoadModule statement followed by the DSO name. By default, a fully patched server, with all SLES, NNLS, and GroupWise components installed, loads the following modules:

Figure 7: DSO modules loading in /etc/opt/novell/httpd/conf/httpd.conf

```
LoadModule access_module modules/mod_access.so
LoadModule include_module modules/mod_include.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule mime_magic_module modules/mod_mime_magic.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule mime_module modules/mod_mime.so
LoadModule autoindex_module modules/mod_autoindex.so
LoadModule vhost_alias_module modules/mod_vhost_alias.so
LoadModule negotiation_module modules/mod_negotiation.so
LoadModule dir_module modules/mod_dir.so
LoadModule actions_module modules/mod_actions.so
LoadModule userdir_module modules/mod_userdir.so
LoadModule alias_module modules/mod_alias.so
```

I've found multiple sources of information regarding which modules are safe to remove, but when looking at the specific functions of each module, there are really only three that can and should be disabled. Removing these may help avoid future security incidents as they are installed services that just aren't used or configured. Be aware that disabling these three are not supported by Novell and may cause issues if an incident is opened. That being said, I have listed the following modules along with their purpose and the reason we are removing/remarking them:

- **mime_magic_module** – This directive works much like the Unix "file" command. It looks at the first few bytes of a file to figure out the MIME type for Apache[8]. It requires another directive called MimeMagicFile that defines a "magic" file on the filesystem used to aid in these lookups. The MimeMagicFile directive however is not defined by default, thus making mime_magic_module loaded but unusable.

[8] Apache HTTP Server Version 2.0 Documentation, http://httpd.apache.org/docs-2.0/mod/mod_mime_magic.html

© SANS Institute 2005 Author retains full rights.

- **userdir_module** – This module provides the capability for users to reach their home directories from a web page[9]. This option is of no use for our installation since the only users that actually have Linux home directories are root and our initial user. In addition to the LoadModule line, one more line needs to be removed in order to restart Apache2 with no problems. Search for and remove the line that reads "UserDir public_html". The UserDir directive is only available when this DSO is loaded. If you remove one and not the other, Apache2 will not restart.
- **actions_module** – Used to assist in executing CGI scripts based on media type or method[10]. This is the only modules I have reservations about. I've had luck removing it and have experienced no adverse affects. But again, use with caution as this is not formally supported by Novell.

Besides the aforementioned modules, there are several directives that should be changed or removed completely. I've listed each below with an explanation:

- LogLevel – This directive affects the amount of information logged by Apache2. The Apache2 manual suggests nothing below the level "crit" be used. I've decided to go with the level "notice" to gain even more verbosity. Make sure to check your logs regularly to ensure that too much information isn't being recorded. For example, this setting may need to be lowered if you're not interested in seeing messages pertaining to core dumps.
- ServerTokens – Determines the amount of information that is returned as the HTTP response header. "Full" gives the maximum amount of information to clients. "Prod" gives the least. We want this set to "Prod". When scanned remotely with a tool such as nmap, these result sets are returned:

Figure 8: nmap –sV –P0 –p80 mail.sampledomain.org with "ServerToken Full"

```
80/tcp open  http    Apache httpd 2.0.48 ((Unix) mod_ssl/2.0.48
OpenSSL/0.9.6k mod_jk/1.2.5)
```

Figure9: nmap –sV –P0 –p80 mail.sampledomain.org with "ServerToken Prod"

```
80/tcp open  http    Apache httpd
```

- ServerSignature – Displays a footer on server generated documents. This is another form of information leakage and can be stopped by setting to "Off".

Apache2 file permissions are also vital to its security. By default, all files in the document root are owned by root. The aliased application directories are mostly owned and writable by the user novlwww. For enhanced security, all documents should be owned by a user other than the one used to run the server

[9] Apache HTTP Server Version 2.0 Documentation, http://httpd.apache.org/docs-2.0/mod/mod_userdir.html
[10] Apache HTTP Server Version 2.0 Documentation, http://httpd.apache.org/docs-2.0/mod/mod_actions.html

© SANS Institute 2005 Author retains full rights.

(novlwww)[11]. At the present time, this cannot be changed due to Novell web application dependencies. Changing them would most likely stop services like iManager and Webaccess from functioning. The same permission rules apply, if not more so, to the Tomcat Servlet Engine. Tomcat is really the driving force behind almost all aspects Novell web applications. It's the actual program that facilitates filesystem writes and interactive CGI operations. When Red-Carpet applies patches, it is very likely that any file permissions modified will be overwritten. Until the functionality of Novell web applications is written in such a way to take advantage of proper permissions, we have no alternative but to accept the permission defaults.

eDirectory

There are two perspectives on securing eDirectory that should be addressed. Filesystem permissions and eDirectory tree rights should be checked for excessive permissions. Excessive permissions in either of these areas could represent a potential vulnerability.

Filesystem Rights

Filesystem permissions for primary eDirectory daemon files located in /var/nds have ample default permissions. All files are owned by root and have octal 600 permissions assigned to them. It is still recommended to monitor these files for permission changes. Some of these files, specifically the ones kept in /var/nds/dib, should be regularly checked. These files contain all eDirectory information. If seized by an attacker, password hashes could possibly be retrieved and cracked. The only file permissions related to eDirectory that I recommend changing are the NDS command line administrative tools. By default, all of these tools are located in /usr/bin and have octal permissions of 755. These should be changed to octal permissions 700 to prevent non-root users from having any access (as root, "chmod 700 /usr/bin/nds*").

Tripwire

I recommend the use of Tripwire to implement file integrity checking. Tripwire is a great tool for providing a means of non-repudiation for essential files. Any filesystem changes that violate the Tripwire policy trigger administrator configurable alerts. The version of Tripwire shipped with SLES is 1.2. For this example, I used the latest version of the RPM available from their website, which is 2.3-47 (http://www.tripwire.org/downloads/index.php). The following steps illustrate what I did to get a scheduled file integrity check with reports that are emailed to me.

- Edit /etc/tripwire/twcfg.txt and change the HOSTNAME variable to the local machines actual hostname (typing hostname at a console prompt will show this

[11] AppNote: Securing a Novell Nterprise Linux Services Server: Step-by-Step (SUSE 8, NNLS 1.0), http://www.novell.com/coolsolutions/nnlsmag/features/a_securing_server_nls.html#8.1

© SANS Institute 2005 Author retains full rights.

information).
- Edit /etc/tripwire/twpol.txt and altar to fit your server. I have found this to be the most lengthy part of the process as there are large quantities of information to read through. It was designed for Red Hat 7 and had many inconsistencies with SLES. The only major directory that should be added is /var/nds. It should be added with a "SEC_INVARIANT" flag. This is to make sure all permission and ownership changes in that directory are reported. Since this is where the eDirectory files are stored, we are verifying the integrity of our underlying directory database files.
- Next run twinstall.sh by typing "/etc/tripwire/twinstall.sh". This creates the initial binaries and asks for a site and local passphrase. These passphrases should be strong and unique. Type "tripwire –init" to initialize the database. After entering the passphrase, you will most likely see "file not found" errors. These errors are okay for now. What we're really looking for is "Wrote database file: /var/lib/tripwire/localmachinename.twd".
- Now we're ready to check filesystem data against the Tripwire database. Type "tripwire --check" from the console to initiate this process. Once it compares the filesystem to it's database, you should see a report containing change statistics and any errors that occurred (such as file not found).
- If errors are found (as they were in my case), you can edit /etc/tripwire/twpol.txt to remove or correct the errors and subsequently update the database. Updating your changes can be accomplished by typing "tripwire --update-policy /etc/tripwire/twpol.txt". You will be prompted to enter your passphrase. Once entered, the database will be updated and ready for another check.
- If errors were reported due to normal user/daemon activity, run "tripwire --update --twrfile /var/lib/tripwire/report/most_recent_report.twr", to update the database. It should no longer report hose as violations.
- Schedule a daily cron job that runs "tripwire --check --email-report". You should now have a daily Tripwire report emailed to you for review.

The instructions work on multiple systems, however if further customization is needed, the Red Hat Linux 9 Reference Guide[12] has very detailed instructions and was in part used as a reference point for the instructions above. They also have an excellent flowchart to help the reader understand how it all works.

Directory Security

The use of eDirectory in a GroupWise environment primarily deals with authentication and authorization using messaging clients only. When user objects are created, they are added to the GroupWise domain to allow access to messaging services (if the ConsoleOne snap-in is installed and you choose too). Even though our system is designed to primarily only allow user interaction to occur with GroupWise client components, attackers may use other tools to discover things about your eDirectory environment that could facilitate an attack. Overall directory design can play a large part in keeping these attacks from being

[12]Red Hat Linux 9 Reference Guide, http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/ch-tripwire.html

© SANS Institute 2005 Author retains full rights.

successful.

We'll start with renaming the admin account. The default administrative username in Netware is "admin". The name "admin" is used during several parts of the installation process, but after the initial installation, it should be changed to something unique. This way an attacker will not immediately know what username to try, instead of knowing it's admin and immediately have the opportunity to try cracking the account.

User object restrictions can provide a significant amount of protection for eDirectory. There's a tab named "Restrictions" at the top of all user object properties boxes. That tab contains several sub-categories that can be used to tighten eDirectory security. Password restrictions are among the most important. There you can define minimum password requirements, require password uniqueness, and Limit grace logins. Under some of the other tabs, you can do things such as limiting concurrent connections, and set up time based login restrictions if you wish. If users are created with a template, these measures can be put into place at time of account creation. This helps ensure that all user objects have identical account restrictions initially.

When someone does compromise the server, a very popular backdoor for the attacker is to create a hidden object with administrator level rights. Once the attacker is on the server, all they have to do is create an organizational unit with one user inside, then make that user the only object with any rights to the new organizational unit. Once they've done that, all they have to do is make sure the new user has administrative level rights. At this point, they have a completely invisible user, that even admin can't find. With traditional Netware servers, there are several NLM based utilities available that can find these objects and allow administrators to find and change the hidden object passwords (or at least give them rights to it). With SLES and NNLS, this becomes more difficult. The only tool that I'm aware of right now that can detect this on a Linux platform is BindView's bv-control for eDirectory[13]. It uses admin level privileges to remotely scan for vulnerabilities, with a hidden object query being among them.

LDAP

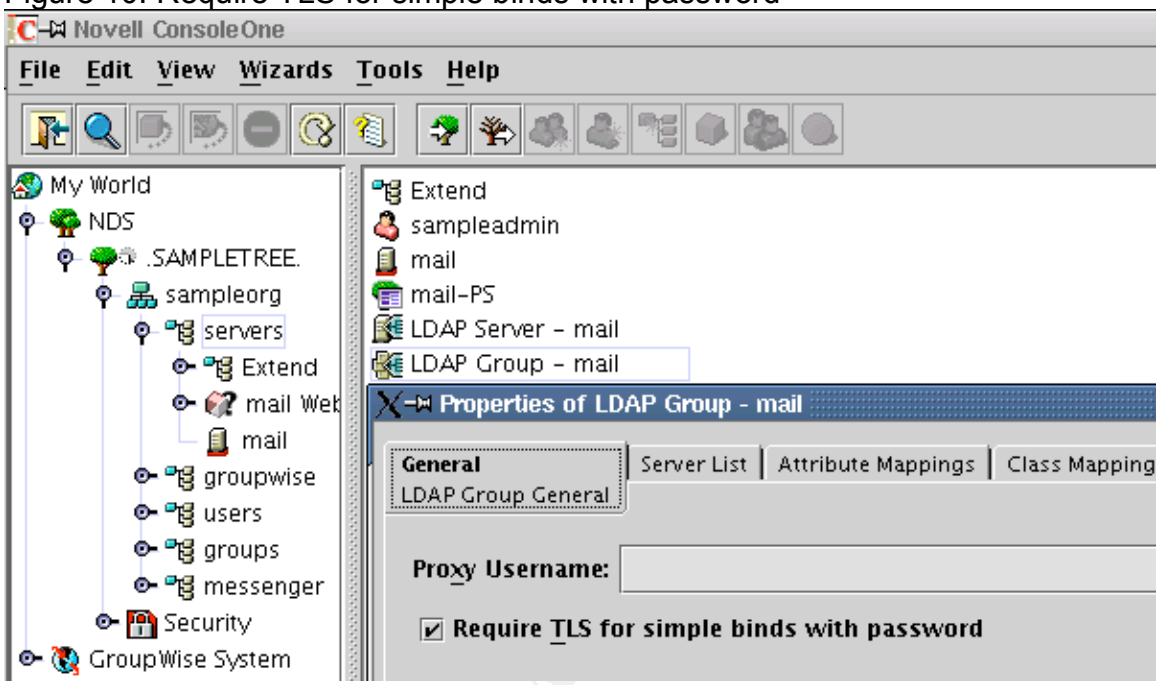
Securing LDAP for NNLS consists of disabling non-SSL authentication of any kind to the server. LDAP servers use a type of encryption called Transport Layer Security (TLS). TLS is Secure Socket Layer (SSL) encryption performed at the session layer[14]. To ensure no clear text authentication can take occur, several boxes must be checked in ConsoleOne. For the first, open ConsoleOne and navigate to the organizational unit where the server resides. Then right click on the "LDAP Group" object and select "Properties". Once the properties page is displayed, the "General – LDAP Group General" tab should be selected. Make

[13] Top Ten Threats to Novell NDS eDirectory, http://www.bindview.com/resources/WhitePapers/TopTenSecThreats_NDS_WP.pdf
[14] Novell eDirectory 8.7 Administration Guide, http://www.novell.com/documentation/edir87/edir87/data/agtxhz5.html#aimoz8c

© SANS Institute 2005 Author retains full rights.

sure the box labeled “Require TLS for simple binds with password” is checked as illustrated in Figure 10.

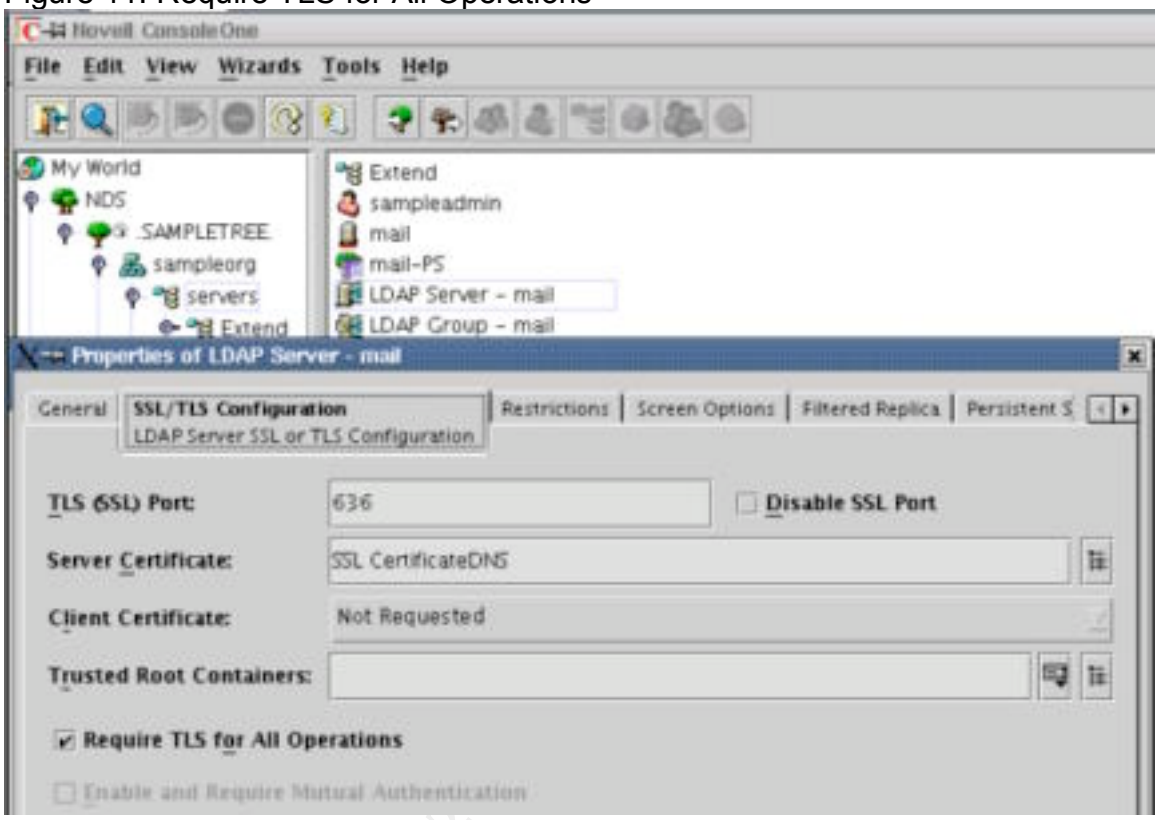
Figure 10: Require TLS for simple binds with password



The second change that needs to occur is in the object properties for the LDAP Server object. This object is located in the same container as the LDAP Group object above. Once located, right click on the “LDAP Server” object and select “Properties”. Next, click the tab labeled “SSL/TLS Configuration”. Ensure the box labeled “Require TLS for All Operations” is checked. Once complete, the object properties page should reflect the one displayed in Figure 11.

© SANS Institute 2005 Author retains full rights.

Figure 11: Require TLS for All Operations



Securing GroupWise 6.5 for Linux

Once installed, administration of GroupWise 6.5 for Linux is functionally very much like GroupWise 6.5 for Netware. For this reason, many possible security settings are the same between both environments. The filesystem represents the most apparent difference between the two environments. Since we have no native Linux or LUM users, we've already taken a good step by alienating the underlying OS from eDirectory users. Most files on SLES pertaining to GroupWise are owned by root and many have at least octal 755 permissions. Although it would be helpful to see these permissions reduced, they are tied to GroupWise functionality in such a way that changing permissions at this time could cause far more problems than would be avoided through securing it.

Access Methods

In the sample environment used throughout this paper, the GroupWise server is located on a corporate DMZ. For this reason, access will be possible via two specific means. The GroupWise X11/Windows client will be the primary means of access for clients located on the LAN port of the firewall (From Figure 1: Sample

© SANS Institute 2005 Author retains full rights.

Network). LAN clients also have the option of using Webaccess if a local install of the GroupWise client is unavailable. Remote users only have access to Webaccess.

Firewall rules should reflect client availability. Incoming connections from LAN to DMZ (specifically to the GroupWise server) should include the following TCP/UDP ports:

TCP
443 – HTTP protocol over TLS/SSL
524 – NCP Connectivity
1677 – GroupWise Client (POA)

UDP
524 – NCP Connectivity

Incoming connections from Internet to DMZ should only include one TCP port:

TCP
443 – HTTP protocol over TLS/SSL

Https connections (port 443) are used primarily for secure Webaccess connectivity. They are also used for iManager administration, which is protected by eDirectory logins and role based services. The NCP Connectivity port is only listed if administrators plan on using NCP authentication services to facilitate the use of ConsoleOne from the internal LAN. Even though it represents a certain level of convenience, I would recommend against this. Narrowing down services visible to other networks makes it inherently more secure. ConsoleOne can still be accessed from the server console to perform maintenance. Port 1677 is required for GroupWise client connectivity and should only be available to the internal LAN.

Password Settings

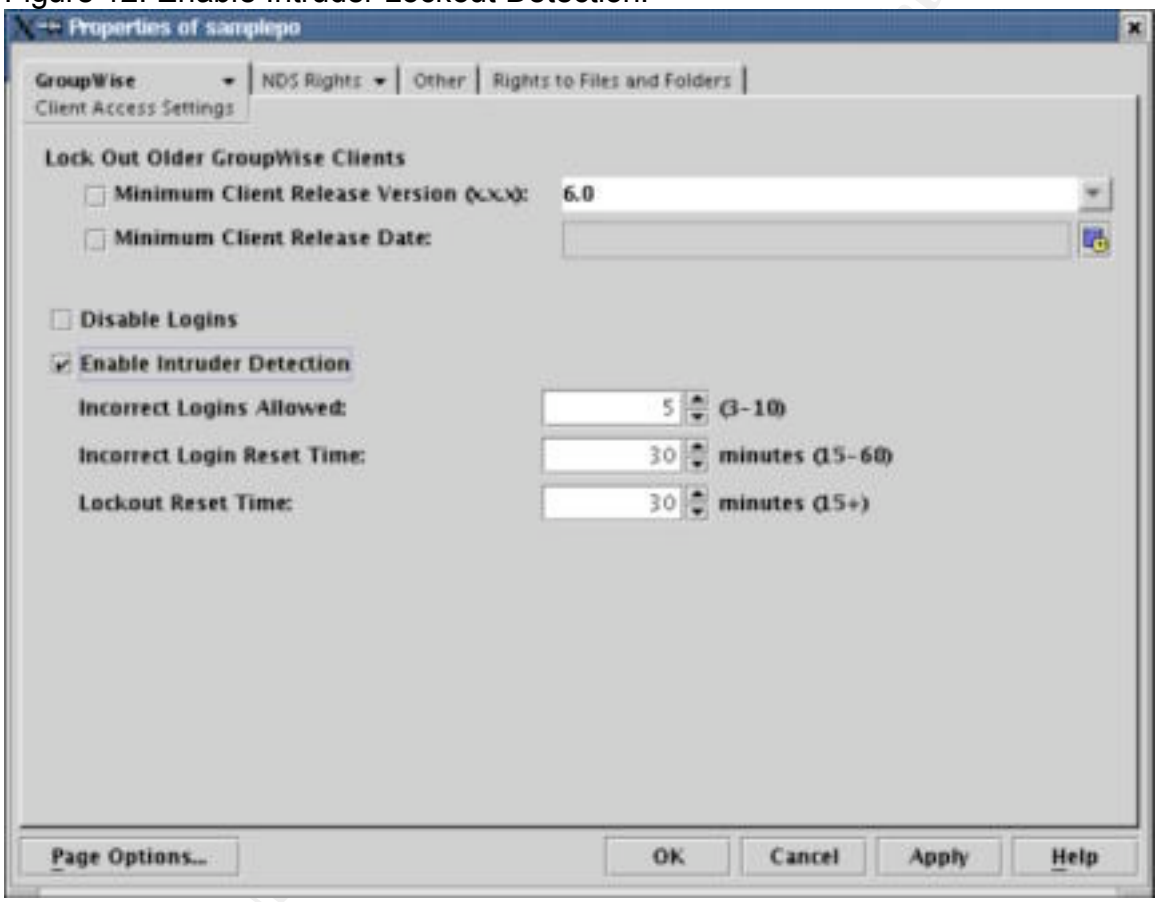
As stated in the installation environment section on page 6, “High Security” was selected as part of the initial installation process. This setting requires initial eDirectory authentication to access passwordless mailboxes. Authentication to most facets of GroupWise can be accomplished securely in multiple ways. Two of the most notable are eDirectory and LDAP. For this scenario, we are standardizing on eDirectory and GroupWise credentials for traditional clients and GroupWise credentials for Webaccess. To allow eDirectory authentication from traditional clients, a setting in ConsoleOne needs to be changed. In the left pane of ConsoleOne, select the Post Office object. With the Post Office object highlighted, select the “Tools” menu, then “GroupWise Utilities” and “Client Options”. That will display a window where you need to select “Password”. From there, make sure the box labeled “Allow eDirectory authentication instead of password” is checked, and then click “OK”. Once all users are created and added to the GroupWise system, they can login from eDirectory clients using their

© SANS Institute 2005 Author retains full rights.

eDirectory account.

By Default intruder lockout detection is enabled for Webaccess. It allows 5 unsuccessful logins before locking the account for 10 minutes. The waiting period is required and can only be bypassed by restarting the Webaccess agent. The GroupWise client does not have intruder lockout detection enabled by default. It has to be manually turned on in the Post Office properties > GroupWise > Client Access Settings section as shown in Figure 12.

Figure 12: Enable Intruder Lockout Detection.



Traffic Encryption

By default, the GroupWise client using a proprietary encryption for all traffic to and from the Novell client. Since the traditional client is only capable of connecting from the LAN, this should be ample. Webaccess traffic is also encrypted by default. It utilizes SSL/TLS to encrypt all traffic between client and server.

Spam

GroupWise uses several approaches to address spam. It can utilize real-time blacklists (RBLs), access control lists (ACLs), mailbomb protection, and

© SANS Institute 2005 Author retains full rights.

unidentified host rejection.

RBLs are configured in the properties of the GWIA eDirectory object in ConsoleOne. Once in properties, select the “Access Control” > “Blacklists” tab from the top. There are lots of free lists out there, but you should read the description of each carefully prior to putting it in place. Some RBLs may block too much while others not enough. This setting depends entirely on the environment in which it’s being placed. For this scenario, I used “blackholes.mail-abuse.org”, “relays.ordb.org”, and “bl.spamcop.net”.

ACLs can be used as a method of implementing a custom blacklist. This is configured in the “GWIA” > “Access Controls” > “Settings” object properties page. Select “Default Class of Service, then click “Edit”. Click “Create” in the “Prevent Messages From” section on the bottom. From here, add any host that this server should not be allowed to receive mail from.

Mailbomb protection and unidentified host rejection are configured in the same area. We’re still working within the “GWIA” object properties, but now we’re on the “SMTP/MIME” > “Security Settings” tab. When this tab comes up, you will notice two checkboxes. The box labeled “Enable mailbomb protection” should be checked to prevent any single external host from sending in excess of the defined threshold in the user set amount of time. This setting should have no negative impact on normal operations. The box labeled “Reject mail if sender’s identity cannot be verified” however can cause legitimate email to be discarded if the remote server inadvertently has reverse DNS problems. This setting causes GWIA to reject all incoming mail that cannot pass a reverse DNS lookup. If any valid record is returned, mail is allowed. Use this setting with caution.

Viruses

Virus protection, by Novell’s own admission, is only truly dealt with via third party products. These products integrate directly with the GroupWise system. One of the virus scan products recommended by Novell is Gwavix, by Beginfinate. There are other products out there, and each should be tested individually to form a preference.

Conclusion

In conclusion, we have just discussed how to secure GroupWise 6.5 on SUSE Linux Enterprise Server 8 and Novell Nterprise Linux Services 1.0. The above scenario works well for shops up to a certain size (small to medium businesses). When an organization looks at issues such as hardware fault tolerance and high availability, new security issues arise that must be addressed as the system grows. For instance, it’s very possible to encounter a situation where an internal eDirectory tree is either existing, or is implemented afterwards. This could enable a situation where it may be advantageous to split up agents and run only the required agents in the DMZ. One could also incorporate the use of SMTP relays to filter incoming messages for viruses and spam. There are many other ways to

© SANS Institute 2005 Author retains full rights.

implement these products as your environment grows, but hopefully guide will provide you a good starting point.

References

Filesystem Hierarchy Standard Group. "Filesystem Hierarchy Standards." 29 Jan. 2004. URL: http://www.pathname.com/fhs/pub/fhs-2.3.html#THEROOTFILESYSTEM (24 Oct. 2004)

Weidner, Klaus. "SLES Security Guide." v2.33 04 Dec. 2003. URL: http://www.suse.com/de/security/eal3/SLES8_EAL3_SecurityGuide.pdf (24 Oct. 2004)

Free Software Foundation. "GRUB Manual." 14 Jun. 2004. URL: http://www.gnu.org/software/grub/manual/html_node/Security.html#Security (24 Oct. 2004)

Novell. "Novell iManager 2.0.x". 23 Aug. 2004. URL: http://www.novell.com/documentation/imanager20/imanager20/data/box3hrz.html (24 Oct. 2004)

The Apache Software Foundation. "Apache HTTP Server Version 2.0 Documentation." URL: http://httpd.apache.org/docs-2.0/mod/mod_mime_magic.html (24 Oct. 2004)

The Apache Software Foundation. "Apache HTTP Server Version 2.0 Documentation." URL: http://httpd.apache.org/docs-2.0/mod/mod_userdir.html (24 Oct. 2004)

The Apache Software Foundation. "Apache HTTP Server Version 2.0 Documentation." URL: http://httpd.apache.org/docs-2.0/mod/mod_actions.html (24 Oct. 2004)

Al Maslowski-Yerges. "AppNote: Securing a Novell Nterprise Linux Services Server: Step-by-Step (SUSE 8, NNLS 1.0)". 01 Oct 2004. URL: http://www.novacoast.com/resources/NNLS-SLES8-NovacoastGCUX.pdf (24 Oct. 2004)

Fuller, Johnray. "Red Hat Linux 9 Reference Guide". URL: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/ch-tripwire.html (24 Oct. 2004)

Loveless, Mark. "Top Ten Threats to Novell NDS eDirectory". URL: http://www.bindview.com/resources/WhitePapers/TopTenSecThreats_NDS_WP.pdf (24 Oct. 2004)

© SANS Institute 2005 Author retains full rights.

Novell "Novell eDirectory 8.7 Administration Guide."
URL:
http://www.novell.com/documentation/edir87/edir87/data/agtxhz5.html#aimoz8c
(24 Oct. 2004)

Appendix A – /etc/opt/novell/nterprise_linux_services_install.conf

```
CONFIG_APACHE=yes
CONFIG_DIRXML=no
CONFIG_EDIRECTORY=yes
CONFIG_EDIR_ADMIN_CONTEXT=cn=admin.ou=servers.o=sampleorg
CONFIG_EDIR_ADMIN_PASSWORD=
CONFIG_EDIR_EXISTING_PORT=524
CONFIG_EDIR_HTTPS_PORT=8010
CONFIG_EDIR_HTTP_PORT=8008
CONFIG_EDIR_LDAP_PORT=389
CONFIG_EDIR_LDAP_SECURE_PORT=636
CONFIG_EDIR_NFK_FILE=/root/nw65license/XXXXXXXXX.nfk
CONFIG_EDIR_SERVER_CONTEXT=ou=servers.o=sampleorg
CONFIG_EDIR_TREE_NAME=SAMPLETREE
CONFIG_EDIR_TREE_TYPE=New Tree
CONFIG_EGUIDE=no
CONFIG_IFOLDER=no
CONFIG_IMANAGER=yes
CONFIG_IMANAGER_ADDRESS=mail.sampledomain.org
CONFIG_IMANAGER_ADMIN_CONTEXT=cn=admin.ou=servers.o=sampleorg
CONFIG_IMANAGER_ADMIN_PASSWORD=
CONFIG_IMANAGER_LDAP_ADDRESS=mail.sampledomain.org
CONFIG_IMANAGER_LDAP_PORT=636
CONFIG_IMANAGER_WEBADMIN_HTTPS_PORT=8020
CONFIG_IMANAGER_WEBADMIN_HTTP_PORT=8018
CONFIG_INSTALL_COMPONENT_SELECTION=finish
CONFIG_INSTALL_EXPRESS=no
CONFIG_INSTALL_LICENSE_AGREEMENT=yes
CONFIG_INSTALL_POSTGLOBAL_ROUTINES=yes
CONFIG_INSTALL_PREGLOBAL_ROUTINES=yes
CONFIG_INSTALL_SAVE_CONF=yes
CONFIG_INSTALL_VIEW_README=no
CONFIG_INSTALL_VIEW_SUMMARY=yes
CONFIG_IPRINT=no
CONFIG_JVM=yes
CONFIG_LUM=yes
CONFIG_LUM_ADMIN_FDN=cn=admin.ou=servers.o=sampleorg
CONFIG_LUM_ADMIN_PASSWORD=
```

© SANS Institute 2005 Author retains full rights.

```
CONFIG_LUM_EDIR_IP_ADDR=mail.sampledomain.org
CONFIG_LUM_EDIR_SELECT_AUTH=finish
CONFIG_LUM_LDAPS_PORT=636
CONFIG_LUM_LDAP_PORT=389
CONFIG_LUM_PARTITION_ROOT=o=sampleorg
CONFIG_LUM_SERVICE_FTP=yes
CONFIG_LUM_SERVICE_LOGIN=yes
CONFIG_LUM_SERVICE_PASSWD=yes
CONFIG_LUM_SERVICE_RLOGIN=yes
CONFIG_LUM_SERVICE_RSH=yes
CONFIG_LUM_SERVICE_SSHD=yes
CONFIG_LUM_SERVICE_SU=yes
CONFIG_LUM_WS_CONTEXT=ou=servers.o=sampleorg
CONFIG_NETMAIL=no
CONFIG_RCD_PROXY=no
CONFIG_REDCARPET=yes
CONFIG_REDCARPET_EMAIL=admin@sampledomain.org
CONFIG_SAMBA=no
CONFIG_TOMCAT=yes
CONFIG_VO=no
```

© SANS Institute 2005, Author retains full rights.

Last Updated: May 24th, 2013

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Malaysia @ MCMC 2013	Cyberjaya, MY	Jun 03, 2013 - Jun 08, 2013	Live Event
SANS Pen Test Berlin 2013	Berlin, DE	Jun 03, 2013 - Jun 08, 2013	Live Event
Industial Control Systems Security Training - Houston	Houston, TXUS	Jun 10, 2013 - Jun 15, 2013	Live Event
Security Impact of IPv6 Summit 2013	Washington, DCUS	Jun 14, 2013 - Jun 16, 2013	Live Event
SANSFIRE 2013	Washington, DCUS	Jun 14, 2013 - Jun 22, 2013	Live Event
SANS Canberra 2013	Canberra, AU	Jul 01, 2013 - Jul 13, 2013	Live Event
Digital Forensics & Incident Response Summit 2013	Austin, TXUS	Jul 09, 2013 - Jul 16, 2013	Live Event
SANS London Summer 2013	London, GB	Jul 09, 2013 - Jul 16, 2013	Live Event
SANS Rocky Mountain 2013	Denver, COUS	Jul 14, 2013 - Jul 20, 2013	Live Event
SANS Mumbai 2013	Mumbai, IN	Jul 22, 2013 - Jul 27, 2013	Live Event
SEC528 SANS Training Program for the CompTIA® New Advanced Security Practitioner (CASP) Certification	Washington, DCUS	Jul 22, 2013 - Jul 26, 2013	Live Event
SANS San Francisco 2013	San Francisco, CAUS	Jul 29, 2013 - Aug 03, 2013	Live Event
SANS SEC 560: Network Penetration Testing @ Bangalore 2013	Bangalore, IN	Aug 05, 2013 - Aug 10, 2013	Live Event
SANS Boston 2013	Boston, MAUS	Aug 05, 2013 - Aug 10, 2013	Live Event
Critical Security Controls Summit	Washington, DCUS	Aug 12, 2013 - Aug 18, 2013	Live Event
Industrial Control Systems Security Training - DC	Washington, DCUS	Aug 12, 2013 - Aug 16, 2013	Live Event
SANS Thailand 2013	Bangkok, TH	Aug 19, 2013 - Aug 31, 2013	Live Event
SANS Virginia Beach 2013	Virginia Beach, VAUS	Aug 19, 2013 - Aug 30, 2013	Live Event
Mobile Device Security Summit 2013	OnlineCAUS	May 30, 2013 - Jun 06, 2013	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced