Technical White Paper
DATA CENTER

www.novell.com

Installing a Secure Server with SUSE® Linux Enterprise Server 9 and Novell® AppArmor

Novell®

Installing a Secure Server with SUSE Linux Enterprise Server 9 and Novell AppArmor

Table of Contents:

Introduction

The entire world is rapidly becoming IT-enabled. Wherever you look, computer technology has revolutionized the way things operate. But with all the opportunities the new information technologies open to society, there are also increasing risks. A lot of sensitive information passes through the Internet, such as credit-card data, mission-critical server passwords and important files. The havoc that computer failure, misuse or sabotage causes is greater than ever before. There is always a chance of someone viewing or modifying the data while it is in transmission. There are countless horror stories of what happens when outsiders get someone's credit card or financial information. They can use it in any way they like and could even destroy you and your business by taking or destroying all your assets.

But isn't there "the one and only" technique that is able to totally protect our IT environments? In a word—no. No machine connected to the Internet is 100-percent secure. This doesn't mean that you are helpless. You can take measures to avoid hacks, but you cannot avoid them completely. This is like the situation within your own home. If the doors and windows are open, the probability of a thief coming in his high. It the doors and windows are closed and locked, you are less likely to be robbed, even though this remains a possibility.

As we all know, "an ounce of prevention beats a pound of cure." To avoid critical situations, it is advisable to have a good security policy and security implementation.

Hardening SUSE Linux Enterprise Server 9

Linux* servers have proved themselves reliable and secure bastions in the Internet world of today. But while many administrators do know a lot about configuring their common systems, they rarely know and deploy all efficient mechanisms—especially new features—to strengthen security.

The goals of this white paper are to inform administrators about the tuning parameters of SUSE® Linux Enterprise Server 9 and to provide an easy step-by-step guide for improving system security. Any part of this guide can be skipped if the risk or threat scenario does not apply to your server. Conversely, depending on the installation and your specific needs, the security considerations of your system may not be covered completely in this document. System administrators should always closely evaluate the configuration changes they make on their systems.

Linux offers a wide range of security options for fine tuning system security. Because this guide cannot cover all mechanisms, it focuses on the most efficient ones.

If you want to install your SUSE Linux Enterprise Server 9 according to the Common Criteria-Controlled Access Protection Profile Evaluation Assurance Level 4+ (CCCAPP/EAL 4+) certification, disregard this guide. Instead, read the SUSE Linux Enterprise Server Security Guide and use the script ***/usr/lib/eal4/bin/sles-eal4*** for automatic reconfiguration.

Enhanced Security with Novell AppArmor

Many security vulnerabilities result from bugs in "trusted" programs. A trusted program runs with a privilege that some attacker would like to have, and the program fails to keep that trust if there is a bug in it that allows the attacker to acquire that privilege.

Novell® AppArmor is the most effective and easy-to-use security system for Linux applications. It is an intrusion-prevention system that protects both the Linux operating system and applications from external or internal attacks, viruses and malicious applications. As a result, businesses can protect key corporate data, reduce network administration costs and comply with government regulations.

Preparing the Infrastructure

A server should never be installed without a purpose. Usually, the purpose is to provide one or more network services to a group of users. The server and the services it provides must be placed in a proper environment. These are divided into different zones, each with its own security considerations. To guarantee the highest possible degree of protection, security must be consistently implemented in each zone.

The Infrastructure Zone

The infrastructure zone defines the position of the server within the network. This area must be protected from threats like data sniffing, network mapping and port scanning.

Furthermore, following a successful attack on an exposed server, it should not be possible to use this server to attack other important servers.

To this end, all servers offering Internet services must be protected by a central component and located in a separated network known as a demilitarized zone (DMZ). The protective component may be a complex firewall or a simple router for which restrictive filter rules are configured—a measure that should normally be adequate. This limits access only to certain server services. A very basic filter list might look like this if a Web server is the only service provided (anything else should be denied):

Filter Rules

Action	From	To	Services
ALLOW	Any location	Web server	HTTP, HTTPS, UDP highport, ICMP type 8
ALLOW	Administrative	Web server	SSH
ALLOW	Web server	Router	SSH (or telenet if not supported)
DENY	Any location	Router	Any service
ALLOW	Yes	Any location	DNS, SMTP, ICMP type 0
DENY	No	Any location	Any service

A switch with port security and flood protection for the DMZ provides an exceptionally high degree of security in this area. If you are concerned about physical security, make sure that the server is installed in a secure room or data processing center, and that all power, telephone and network lines are physically protected from access.

Limiting the possibility of attack is not enough. In case an attacker successfully penetrates the system, there should be mechanisms that detect the intrusion.

The Network Protocol Zone

Internet communication takes place almost exclusively by means of TCP/IP. The operating system kernel is responsible for communication and ensures a transparent communication flow. However, some protocol functions and vulnerable points can be misused for attacks or sabotage. The kernel must be configured to ward off such attacks. Although a firewall or router in front of the server may help to prevent many attacks, some Web server settings need to be adjusted.

The prevention of synchronize/start (SYN) flooding attacks is essential. Linux provides the most effective operating system solution: SYN cookies. Internet Control Message Protocol redirects and pings on broadcast addresses when they should not be accepted, and IP source routed packets should be denied. Use of additional kernel filter functions increases the security level.

The Service Zone

The service zone defines what services are required. Only services necessary for operation should be configured on servers; otherwise, attackers are provided with additional vulnerable spots.

Only services guaranteeing a sufficient level of security should be used. Services with insufficient authentication—such as rexec or services transmitting sensitive data without encryption, including Telnet, File Transfer Protocol (FTP) and credit card details via the World Wide Web—should be replaced with secure services like Secure Shell (SSH), Secure Sockets Layer (SSL) FTP or Secure HTTP.

The Application Zone

Each service must be individually configured for security. An incorrectly configured mail service can be used for spamming, or a Web server for the execution of all kinds of commands. High privilege services (root) should not be established. Consult the manuals of the software used for any information on this subject.

The Operating System Zone

The final protective mechanism is the operating system itself. If the security measures for the application zone are applied consistently, the attacker does not have any administrative authorization even after penetrating the computer. Installed programs, especially privileged programs, should be limited to those absolutely necessary for the operation of the system. Many privileged programs can also be deprived of high-level authorizations because these are not needed by the standard user accounts in the system.

Limiting the possibility of attack is not enough. In case an attacker successfully penetrates the system, there should be mechanisms that detect the intrusion. This is called host-based intrusion detection. It should also be possible to monitor and record file manipulations in the system. Regular backups should not be neglected and old backups should be kept. Backups prevent data loss and also enable system manipulation to be tracked. If several administrators supervise the server, a mechanism recording who executed what action should be available for later reference.

Installing SUSE Linux Enterprise Server 9

After booting from the CD and selecting New Installation, there are three install options you can choose to enhance the security of the system: Partitioning, Software Packages and System Settings.

Partitioning

For each file system, use ext3 or reiserfs for enhanced reliability. Create the following six partitions:

/
/var
/tmp
/home
/svr
/usr/local

You can create more if necessary. If the log files are flooded, the service and the system should not be impaired, so ***/var*** should be a separate partition. If logging is mandatory for an application for security reasons, it should stop providing its service until it can log again. The ***/tmp*** directory can be flooded by accident or on purpose because it is writable by everyone, so it could impair system and service stability as well. ***/home*** and the service directories ***/svr*** should be separated for the same reason. Finally, keeping ***/usr/local*** separate makes updating the installation easy if you want to reinstall the system from scratch with a new version of SUSE Linux Enterprise Server.

The following table lists the special security flags you might want to set on each partition. A question mark indicates that some software might not work if this flag is set.

Mount Point	Mount Operations
/	
/var	nosuid
/tmp	nosuid
/home	nosuid, nodev, noexec?
/svr	nosuid?, nodev?, noexec?, ro? [after installation]
/usr/local	nosuid?, nodev?, ro? [after installation]

Proprietary software might fail its installation process with these limitations, for example, if files in ***/tmp*** cannot be ***suid*** or devices do not work in ***/usr/local***. In such cases, remount those partitions temporarily with security deactivated.

Software Packages

Select the minimal system and then manually add the RPM packages required for the services. You might also want to select the following packages to increase security:

Package	Description
acct	Process accounting (for auditing purposes)
arpwatch	Address Resolution Protocol (ARP) spoofing detection on the LAN
compartm	Tool to run services in a security compartment
laus pam-laus	Linux Audit-Subsystem (LAuS), enabling fine-grained auditing mechanisms on SUSE Linux Enterprise Server 9
checkpolicy policycoreutils	Security-Enhanced Linux (SELinux)
freeswan ipsec-tools	IPsec VPN software
logsurfer	Log checks, automatic actions and so on
scanlogd	Port scanning detection
seccheck	Daily, weekly and monthly security checks
snort	A powerful network intrusion detection tool

continued on next page

Package	Description
stunnel	An SSL wrapper for unencrypted services
sudo	Replacement for su that defines which administrator is allowed to do what
tripwire	A file integrity checker
xntp	Network time tools

Also, select a text editor that suits you. When you are finished selecting the software, start installing the system.

System Settings

After you have performed the basic installation, you are asked for a root password. Select a strong and hard-to-guess password and then press Expert Options. In the dialog box, choose MD5 hashing of the passwords. In a later screen, you can add a user account. Here, go into Password Settings and change the settings for the maximum number of days for a password to a value between 30 and 100, and the value for how many days a login with an expired password is usable to a number less than 30.

Customizing the Installation

To customize the security of your installation according to your requirements, follow the steps indicated below.

Software Packages

Removing Packages

After the installation process is finished, you might want to remove some unnecessary software packages. With ***rpm -e PACKAGE***, you can remove the following packages:

- ***cpp***
- ***rsh (insecure)***

Several more packages can be removed if not required. View the list of remaining packages with the command ***rpm –qa***.

Updating Packages

Before configuring anything, check whether updates are available for any of the installed packages and install the updates, if necessary. Normally, you would perform an online update (YaST2 -> Software -> Online Update), but it is recommended that you do not connect the server to the network until all packages are up to date. The most secure way to update is by downloading the relevant RPMs, writing them to media and installing the updates from there to the server. Update information can be found at the SUSE Linux Portal at: www.novell.com/linux/suse/migration.html

Standard Services

Disabling Services

If your server is not offering or accessing any Remote Procedure Call service, disable portmap with ***insserv -r portmap***. Because Service Locator Protocol is usually not used either, remove it as well with ***insserv -r slpd***.

Enabling Services

If you want to enable the following services, use ***insserv SERVICE***:

Service	Description
acct	Process accounting
arpwatch	ARP spoof detector
scanlogd	Port scan detector
snort	Network intrusion detection
SuSEfirewall2_init SuSEfirewall2_setup SuSEfirewall2_final	Firewalling

Hardening Services

Along with the services you enabled in the previous section, there should be just one other service running now—SSH. Hardening OpenSSH service requires two steps:

1. **TCP Wrapper Setup.** First, deny any access that is not from the local host to any service with TCP wrapper support. ***echo "ALL : ALL" >> /etc/hosts.deny***

   Next, add the IP addresses from which logins are permitted, for example, from a single host and a subnet. ***echo "sshd : 10.0.0.10 10.10.10.0/24" >> /etc/hosts.allow***

   Finally, allow everything that comes from the local host. This can be required for portmap local mounts and other things. ***echo "ALL : 127.0.0.1" >> /etc/hosts.allow***

   Use only IP addresses, not DNS names. For more information, refer to ***man 5 hosts_access***.

2. **SSHD Configuration.** Edit ***/etc/ssh/sshd_config*** and add or change the following lines:

   – PermitRootLogin ***no***
   – X11Forwarding ***no***
   – UsePrivilegeSeparation ***yes***
   – Banner ***/etc/issue.net***

   Additionally follow the directions in the upcoming "Login" section.

System Hardening

Login

Now set login and password security parameters. Edit /etc/login.defs and change the following lines:

```
PASS_MAX_DAYS 60 # a value between
    30 and 100
PASS_MIN_LEN 7
UMASK 077
```

Then configure a warning message to display for authorized usage. Change this example message to your corporate standard and local law requirements.

```
# cd /etc
# cat > motd
Unauthorized logins and misuse of this
    system is prohibited!
^D
# rm issue issue.net
# ln -s motd issue
# ln -s motd issue.net
```

You can further fine tune user and login security by using the ***access.conf***, ***chroot.conf*** and ***limits.conf*** files in the /etc/security directory. Here, define what users are allowed to login from where and chroot to what directory and what system resource limits to apply to what users.

Permissions

In a default SUSE Linux Enterprise Server 9 installation, about 50 ***suid*** binaries are installed. These provide a security risk for local attackers. Therefore, you should harden the file system permissions. Fortunately, this is easy on SUSE Linux. Just change the ***PERMISSION_SECURITY*** parameter line in ***/etc/sysconfig/security*** to:

```
PERMISSION_SECURITY="secure local"
```

Then run ***SuSEconfig***.

The system can be stripped of further privileged suid and sgid programs. Simply do this by entering programs that should not have these privileges in ***/etc/permissions.local*** and subsequently starting ***SUSEconfig***.

PAM

All login mechanisms should use the Pluggable Authentication Mechanism (PAM). The default configuration is fine except that it allows logins with accounts that have no passwords set. This is not secure, so change it as follows:

```
perl -pi -e 's/nullok//' /etc/pam.d/*
perl -pi -e 's/nullok//'
    /etc/security/pam_pwcheck.conf
perl -pi -e 's/nullok//'
    /etc/security/pam_unix2.conf
```

sudo

Administrators should each have their own user accounts, because it is impossible to know who did what when working under the root identity. In addition, incorrectly entering a command as root can affect the entire system. Therefore, operations with high levels of authority should be performed only when really necessary. A direct root login over the network is impossible given the modifications to the SSH service, and administration itself can only be performed in an encrypted manner with SSH. The next step in this process is to configure sudo, a program that helps administrators do their jobs while at the same time keeping a record of the commands.

This program also enables a detailed authorization structure. For example, as "user oracle" user A is entitled to restart the database and view the system log files under root, but nothing else. Subsequently, the administrators' user accounts are admitted to sudo by means of the visudo program. The following line, which allows the administrator to do whatever he wishes, is added in the editor:

```
username ALL=(ALL) ALL
```

man 5 sudoers defines a range of settings with which the authorizations can be restricted.

It is important for the administrators to use sudo and not shift to the root identity with ***su root***. For this reason, the root password should be disclosed to as few people as possible.

Syslog

Logging data is very important. All important log messages from the Web server and the router should be sent to a central log host from which the status of the computers can be monitored. This makes it difficult for an attacker to hide his trail.

Enable remote reception at the central log host by adding the ***-r*** switch to ***syslogd***. You can do this in ***/etc/sysconfig/syslog***.

To send all log messages to a central log host, add the following line to ***/etc/syslog.conf***:

```
*.* @IP-ADDRESS-OF-LOGHOST
```

NTP

There is a saying that reads, "Logs are only as good as their time stamp." To correlate events from different machines or log sources, such as router, firewall and proxy, the timestamps need to match. Therefore, one server should provide the time for all other servers. The easiest way to connect to this time server is by running a cron job that synchronizes the time.

```
cat > /etc/cron.hourly/ntpdate
    #!/bin/bash
    MAILTO=""
    /usr/sbin/ntpdate <IP address of time
        server>
    ^D
    chmod 700 /etc/cron.hourly/ntpdate
```

Boot Security

If you also need to protect the server from local attacks, equip the GRUB boot loader with a password to protect it from selecting different root systems, init processes and so on. You can achieve this by starting the GRUB command line interface and then entering the ***md5crypt*** command as follows:

```
# grub
grub> md5crypt
Password: **********
Encrypted: $1$4f34f...some.hash...
grub> quit
# echo 'password --md5
   $1$4f34f...some.hash...'
      >> /boot/grub/menu.lst
```

Note that the echo command uses single quotes. If you start the general purpose mouse ***(gpm)*** service ***(rcgpm start)*** prior to entering the ***md5crypt*** command, copying and pasting with a mouse is easier. Read the GRUB information pages ***(info grub)*** for more security features, such as menu locking.

Miscellaneous

This section discusses some less common security measures. Using these can further improve the security of your system. First, ***edit /etc/inittab***. You should disable the ***ctrlaltdel*** and keyboard request functions by commenting them:

```
#ca:.ctrlaltdel:.....
#kb::kbrequest:.......
```

If you want a "last resort" login via the serial port, add the following line:

```
S0:1235:respawn:/sbin/agetty -L 57600
   ttyS0 xterm
```

Also, put the serial port into root's allowed login list: for example, with ***echo ttyS0 >> /etc/security***.

If you are not using the recommended SuSEfirewall2, at least create this script to secure your TCP/IP settings:

```
# cat > /etc/init.d/interface_security
#!/bin/sh
#
# /etc/init.d/interface_security
#
### BEGIN INIT INFO
# Provides: interface_security
# Required-Start: $network $local_fs
# X-UnitedLinux-Should-Start: route
   dhclient dhcpcd named
# Required-Stop: $local_fs
# X-UnitedLinux-Should-Stop:
# Default-Start: 3 4 5
# Default-Stop: 0 1 2 6
# Short-Description: Secure TCP/IP
   settings
# Description: This script performs
   extensive TCP/IP security settings
### END INIT INFO

# basic security
echo 0 > /proc/sys/net/ipv4/ip_forward
echo 1 > /proc/sys/net/ipv4/icmp_
   echo_ignore_broadcasts
echo 1 > /proc/sys/net/ipv4/tcp_syncookies
echo 1 > tcp_secure_pmtu 2> /dev/null
echo 0 > /proc/sys/net/ipv4/tcp_ecn 2>
   /dev/null

for i in /proc/sys/net/ipv4/conf/*; do
echo 0 > $i/accept_redirects
echo 0 > $i/secure_redirects
echo 0 > $i/accept_source_route
# The following line should be commented
   out, if you use IPSEC !!!
echo 1 > $i/rp_filter
echo 0 > $i/mc_forwarding 2> /dev/null
done

# For more security
for i in /proc/sys/net/ipv4/conf/*; do
echo 1 > $i/log_martians
echo 0 > $i/bootp_relay
echo 0 > $i/proxy_arp
done

# Configure the following values to your
   needs!
echo 16384 > /proc/sys/net/ipv4/ip_
   conntrack_max
echo 5 > /proc/sys/net/ipv4/icmp_
   echoreply_rate
echo 5 > /proc/sys/net/ipv4/icmp_
   destunreach_rate
```

```
echo 5 > /proc/sys/net/ipv4/icmp_
    paramprob_rate
echo 6 > /proc/sys/net/ipv4/icmp_
    timeexceed_rate
echo 20 > /proc/sys/net/ipv4/ipfrag_time
echo 1 > /proc/sys/net/ipv4/igmp_max_
    memberships
echo "1024 29999" > /proc/sys/net/ipv4/ip_
    local_port_range
^D
# insserv interface_security
```

Service Hardening

Apache

The Web software and pages are the core to protect. You need to make sure that nobody gains unauthorized access to data or changes the pages. For this purpose, the pages are equipped with a special protection and Apache is furnished with a secure configuration.

The site administrator should supervise all pages, and the pages should be locally write protected for everybody else. It is important that the Web server runs under a different user than the one supervising the pages. In this way, an attacker who manages to sneak through the Web still cannot change the pages. Therefore, you should set up a user and generate a cron job that runs every day and makes sure that all pages belong to the site supervisor and have the correct authorizations.

```
# useradd -m wwwdocs
# cat > /etc/cron.daily/wwwdocs
#!/bin/sh
/bin/chown -R -h wwwdocs /srv/www/*
/bin/chmod -R go-w /srv/www/*
/bin/chmod -R a+r /srv/www/*
^D
# chmod 700 /etc/cron.daily/wwwdocs
```

Because Apache is preconfigured with reasonable defaults, few changes are necessary in the configuration to make it secure. If you use Apache 2.x, first edit **/etc/sysconfig/apache2** and change the following parameter to:

```
APACHE_MODULES="access actions
    aliases autoindex auth env expires
    include log_config mime negotiation
    setenvif ssl"
APACHE_SERVERSIGNATURE=off
APACHE_SERVERTOKENS=ProductOnly
```

After this, run **SuSEconfig** and **rcapache2 start**.

If you use Apache 1.x, first edit **/etc/sysconfig/apache** and change the following parameter to:

```
ENABLE_SUSECONFIG_APACHE=no
```

This way, you can tighten the configuration file yourself. Then, change the **/etc/httpd/httpd.conf** file. Most important are hiding the server signature, disabling all CGI executables and removing all unnecessary modules. You can save the following **diff -u0** output into a file and use it for patching **(cd /etc/httpd/httpd.conf; patch < PATCH)**. Some parts of this patch may fail, depending on the modules you have installed. The failed junks can safely be ignored.

```
---httpd.conf.orig 2005-01-23
    18:06:55.095112792 +0100

+++ httpd.conf 2005-01-23
    18:17:32.999136640 +0100

@@ -238,2 +238,2 @@

-LoadModule status_module/usr/lib/
    apache/mod_status.so

-LoadModule info_module/usr/lib/
    apache/mod_info.so

+#LoadModule status_module/usr/lib/
    apache/mod_status.so
```

+#LoadModule info_module/usr/lib/apache/mod_info.so

@@ -243 +243 @@

-LoadModule cgi_module/usr/lib/apache/mod_cgi.so

+#LoadModule cgi_module/usr/lib/apache/mod_cgi.so

@@ -247 +247 @@

-LoadModule speling_module/usr/lib/apache/mod_speling.so

+#LoadModule speling_module/usr/lib/apache/mod_speling.so

@@ -250,1 +250,1 @@

-LoadModule rewrite_module/usr/lib/apache/mod_rewrite.so

+#LoadModule rewrite_module/usr/lib/apache/mod_rewrite.so

@@ -257,2 +257,2 @@

-LoadModule proxy_module/usr/lib/apache/libproxy.so

-LoadModule cern_meta_module/usr/lib/apache/mod_cern_meta.so

+#LoadModule proxy_module/usr/lib/apache/libproxy.so

+#LoadModule cern_meta_module/usr/lib/apache/mod_cern_meta.so

@@ -268 +268 @@

-Include /etc/httpd/suse_loadmodule.conf

+#Include /etc/httpd/suse_loadmodule.conf

@@ -285,2 +285,2 @@

-AddModule mod_status.c

-AddModule mod_info.c

+#AddModule mod_status.c

+#AddModule mod_info.c

@@ -288 +288 @@

-AddModule mod_autoindex.c

+#AddModule mod_autoindex.c

@@ -290 +290 @@

-AddModule mod_cgi.c

+#AddModule mod_cgi.c

@@ -292 +292 @@

-AddModule mod_imap.c

+#AddModule mod_imap.c

@@ -297,1 +297,1 @@

-AddModule mod_rewrite.c

+#AddModule mod_rewrite.c

@@ -304,2 +304,2 @@

-AddModule mod_proxy.c

-AddModule mod_cern_meta.c

+#AddModule mod_proxy.c

+#AddModule mod_cern_meta.c

@@ -321 +321 @@

-Include /etc/httpd/suse_addmodule.conf

+#Include /etc/httpd/suse_addmodule.conf

@@ -329 +329 @@

-ExtendedStatus On

+ExtendedStatus Off

@@ -467 +467 @@

- Options Indexes -FollowSymLinks

+Includes MultiViews

+ Options None

@@ -847,2 +847,2 @@

Options +ExecCGI -Includes

-SetHandler cgi-script

+Options None

+#SetHandler cgi-script

@@ -1346,2 +1346,2 @@

-SSLRandomSeed startup builtin

-SSLRandomSeed connect builtin

+#SSLRandomSeed startup builtin

+#SSLRandomSeed connect builtin

@@ -1349 +1349 @@

-#SSLRandomSeed startup file:/dev/urandom 512

+SSLRandomSeed startup file:/dev/urandom 512

@@ -1351 +1351 @@

-#SSLRandomSeed connect file:/dev/urandom 512

+SSLRandomSeed connect file:/dev/urandom 512

@@ -1387 +1387 @@

-SSLCipherSuite

ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:

+LOW:+SSLv2:+EXP:+eNULL

+SSLCipherSuite ALL:!ADH:!EXPORT56:RC4+RSA:

+HIGH:+MEDIUM

@@ -1555 +1555 @@

-Include /etc/httpd/suse_include.conf

+#Include /etc/httpd/suse_include.conf

You might need to disable ***mod_user*** if it is enabled. After you have hardened the configuration, enable the CGI and PHP modules and directories, if required. You should also change the MinSpareServers, MaxSpareServers, StartServers and MaxClients options to have a Web server that performs well.

The MaxClients option helps to ward off connect denial-of-service (DoS) attacks. However, if this option is set too low, regular visitors are denied access. If it is set too high, the administrator can have difficulties logging in and taking countermeasures in the event of an attack. Experimentation is the only way to find the correct value.

The activation of SSL and the generation of the certificate is described in ***/usr/share/doc/packages/apache/README.SUSE*** and ***/usr/share/doc/packages/apache/README.SSL***.

Note: *The SSL certificate should be protected with a password so that an attacker cannot copy and misuse it following a successful invasion. However, this requires the Web administrator to log in to start and restart Apache.*

As a general rule, make sure that no symlinks are used anywhere and disable the option ***FollowSymLinks***. CGI should only be found in the cgi-bin directory and should not be permitted or even executed anywhere else. Do not use the configuration option ***ExecCGI*** on any other directory.

In case certain document areas should be off limits, you can add the following lines in ***.htaccess*** files in the individual directories:

```
order deny,allow
deny from all
```

Postfix

If you want to run an e-mail server on your system, configure the domains accepted, relays and so on as necessary, and then make the following changes to ***/etc/sysconfig/postfix*** to chroot postfix and configure moderate spam protection:

```
POSTFIX_CHROOT="yes"
POSTFIX_UPDATE_CHROOT_JAIL="yes"
POSTFIX_RBL_HOSTS="blackholes.
    mail-abuse.org, relays.ordb.org,
    relays.osirusoft.com"
POSTFIX_BASIC_SPAM_PREVENTION=
    "medium"
```

After this, restart postfix with ***rcpostfix restart***.

Squid

For a secure Squid, which provides Web and FTP proxy service, edit ***/etc/squid/squid.conf***. By default, Squid only provides a proxy for local host clients. First, make the necessary configuration changes for your requirements.

If this is a standalone proxy, meaning it does not need to communicate to other proxies, disable the Internet Control Protocol (ICP) feature by adding the following lines:

```
icp_port 0
icp_access deny all
```

DHCPD

Fortunately, the Dynamic Host Configuration Protocol (DHCP) service is already secure out of the box in SUSE Linux Enterprise Server. You do not need to configure anything other than your DHCP network setup. If you want to use DHCP in your network, beware of the following security hazards:

- *It is easy for an attacker in the LAN to deplete the IP resources with tools like thcrut. New valid clients requesting IP addresses cannot be served then and are unable to connect to any resource.*
- *It is even easier for an attacker in the LAN to set up a fake DHCP service that competes with your valid server. This, too, can disrupt your network.*

The advantage of using DHCP is that you can easily change Domain Name System (DNS) and default router configurations on DHCP client machines. Given that the mentioned denial-of-service attacks are rare, DHCP is recommended for workstations and mobile equipment.

Note that the tool arpwatch makes no sense when DHCP is used within a network.

There are no really interesting security settings for DHCP clients. However, the risks of receiving wrong IP addresses, default routers and time and name servers should be kept in mind.

Security Tools

Firewalling with SuSEfirewall2

To edit the firewall configuration, open **/etc/sysconfig/SuSEfirewall2** in your favorite editor. This example uses one interface and provides Web and Web via SSL services to the public and SSH access to an administrator subnet. For this, change the following values:

```
FW_DEV_EXT="eth0"
    # Config/Query number 1
FW_SERVICES_EXT_TCP="80 443"
    # Config/Query number 9
FW_TRUSTED_NETS="10.10.0.0/24, tcp,22"
    # Config/Query number 10
```

The command **/sbin/SuSEfirewall2** updates the rules. These are loaded with each boot.

Seccheck

If you install the seccheck package, your system is automatically checked for system changes. Most checks are performed daily; those that consume more time and resources, weekly. When seccheck runs for the first time, get an overview of the system. Future runs show only changes compared to the previous snapshot. A complete overview is generated monthly. All reports are sent via e-mail to root.

Compartment

Compartment is a tool for securely running untrusted services and programs. It supports chrooting, changing user and group IDs, Linux capabilities, init scripts and more. You can use this tool to securely run all your network services that are not running in a chroot environment already.

/usr/share/doc/packages/compartm/README also shows some examples for squid and bind.

Tripwire

Once you complete all work on the system, use the program tripwire to generate a database containing the checksums of all files. First, create the **/etc/tripwire** directory. The configuration is not easy. Refer to ***man twconfig*** and ***man twadmin*** for information about how to configure tripwire for your installation.

Before connecting to the Internet, the **/etc/tripwire/** directory and its contents along with ***tripwire.rpm*** should be saved to a secure medium, such as CD-ROM. If an attacker is suspected of having manipulated the system, tripwire can be used to track the manipulations. This should be done at regular intervals, because there is no other way to bust intelligent attackers. If you suspect that the server was compromised, you should boot from the SUSE Linux Enterprise Server CD1, mount the CD-ROM with the tripwire information, install the ***tripwire.rpm*** and copy the database and configuration file to their corresponding directories. Only then should you perform the integrity verification. Otherwise, an attacker can hide his modifications, for example by changing the tripwire database, the tripwire binary or libc or by installing a kernel module.

LAuS

You have the option of running LAuS, a process and permission auditing system. Because it is very powerful, an introduction is beyond the scope of this document. Refer to the Linux Audit Subsystem documentation with ***man 7 laus***.

Other Options

To secure your installation, you can also use the ext2 and ext3 file system flags ***append-only*** and ***immutable*** (by means of the ***chattr*** command) to define the kernel capabilities to protect log, boot and other files from changes.

Novell AppArmor Application Containment

Novell AppArmor is an application security solution designed specifically to provide the least privilege confinement to suspect programs. AppArmor allows the administrator to specify the domain of activities the program can perform by developing a security profile for that application: that is, a listing of files that the program may access and operations the program may perform. AppArmor provides sufficient security to prevent the exploitation of software vulnerabilities in Internet servers, while minimizing performance, implementation and administrative costs.

Choosing Applications to Profile

Effective hardening of a computer system requires minimizing the number of programs that mediate privilege and then securing those programs as much as possible. With Novell AppArmor, you need only profile the programs that are exposed to attack in your environment, which drastically reduces the amount of work required to harden your computer. AppArmor profiles enforce policies to make sure that programs do what they are supposed to do, but nothing else. To effectively secure your system, you should consider creating an AppArmor profile for each privilege-mediating program, such as:

- **Network agents**—*Programs (servers and clients) have open network ports, and network agents are server programs that respond to those ports. User clients (such as mail clients and Web browsers) also have open network ports and mediate privilege.*
- **Web applications**—*CGI PERL scripts, PHP pages and more complex Web applications can be invoked through a Web browser.*
- **Setuid programs**—*Setuid or setgid programs run as the user or group that owns the program file rather than as the user and group of the person invoking the program.*
- **Cron jbs**—*Programs that the cron daemon periodically runs read input from a variety of sources.*

Network Agents

These programs run with the privilege to write to the users' home directories and process input from potentially hostile remote sources, such as hostile Web sites and malicious code transmitted via e-mail. To find network server daemons that need profiling, inspect the open ports on your machine, consider the programs that are answering on those ports and provide profiles for as many of those programs as possible. If you provide profiles for all programs with open network ports, the attacker cannot get to the file system on your machine without passing through an AppArmor profile. You can scan for open network ports manually, from outside the machine, by using a scanner such as nmap or from inside the machine or by using netstat and then inspecting the machine to determine which programs are answering on the open ports.

A more automated method is to use the AppArmor tool unconfined. From a root prompt, type the command ***unconfined***. The tool inspects your open ports from inside your computer (using the command ***netstat -nlp***), detects the associated programs, inspects the set of AppArmor profiles you have loaded and reports these programs (along with their associated AppArmor profiles). Web Applications Confining each Web application with its own AppArmor profile minimizes each application's privileges and thus the attacker's opportunities to hijack the program.

However, you can also choose to spend less effort hardening your system (at the expense of security) by choosing instead to run a Web application within the Apache AppArmor profile. The selection of whether a Web application has its own profile or "inherits" Apache's profile happens in the profiling utilities described in the "Profile Building" section.

Novell AppArmor provides sufficient security to prevent the exploitation of software vulnerabilities in Internet servers, while minimizing performance, implementation and administrative costs.

Scripting Languages

Many Web applications are written in interpreted scripting languages such as PERL, PHP or Python. To enhance performance, many Web sites use mod_perl, mod_php and mod_python to place interpreters for these programming languages directly inside the Apache Web server. This improves performance because Apache no longer has to execute a large interpreter program to run a small script. It also compromises security, however, because these Web applications run inside the Apache process using Apache's privileges. AppArmor can confine individual Web applications even though they are executed inside Apache. AppArmor allows Apache to change to a subprofile corresponding to the name of the script about to be executed. If no specific profile for the script is found, a default profile associated with the interpreter can be used. This can increase security by confining all PHP pages, for example, to a similar profile permissive enough for all of the PHP pages to work but more restrictive than the Apache profile. Setuid Programs Programs that are setuid or setgid should be confined with AppArmor because they enable any user to assume the privileges of the setuid or setgid settings. The only line of defense is the correctness of the programs: if there is a bug that allows a non-privileged user to force the program to run arbitrary code by presenting "creative" input, that user can gain root permissions. AppArmor confinement ensures that the program can only perform the tasks it needs to, making attacks by non-privileged users futile.

Cron Jobs

A cron is normally used to schedule a job that is executed periodically: for example, to send out a notice every morning. It is also a daemon process, meaning that it runs continuously, waiting for specific events to occur. Cron jobs might run with special privilege, sometimes with as much as root privilege. To find programs that will be run by cron, you need to inspect your local cron configuration. Periodic cron jobs are run from these files:

/etc/crontab
/etc/cron.d/*
/etc/cron.daily/*
/etc/cron.hourly/*
/etc/cron.monthly/*
/etc/cron.weekly/*

For root's cron jobs, you can edit the tasks with "***crontab -e***" and list root's cron tasks with "***crontab -l***."

Profile Building

Once you have selected the programs to profile, the AppArmor utilities will automate most of the profile development process and ask you interactive questions about security decisions to complete the program profiles. AppArmor utilities can be run from the command line or through the YaST interface. To run the AppArmor tools from YaST, open the YaST Control Center and click on the Novell AppArmor icon. For more information on using the YaST interface or AppArmor tools, refer to the AppArmor User's Guide at: www.novell.com/documentation/apparmor

Enabling AppArmor

To enable AppArmor, click on the AppArmor Control Panel icon, click on Configure, click the Enable radio button, and click Done when you are finished. Once AppArmor is enabled, security profiles that are present in the ***directory /etc/subdomain.d*** will be enforced. **Note:** *Individual security profiles can be placed in "enforce" mode or "complain" mode. Complain mode will log violations to the profile but will not enforce the profile rules. To determine the state of profiles on your system, at a command prompt type* ***cat /subdomain/profiles | less*** *and the list of profiles will appear followed by the notation "(enforce)" or "(complain)." Individual profiles can easily be placed in complain or enforce mode by issuing the command enforce* ***profile_name*** *or* ***complain profile_name*** *at the command prompt.*

Profile Wizard

The AppArmor Profile Wizard is the place to start. Click on the Add AppArmor Profile Wizard icon, and a dialog box will appear asking you to choose the name of the program you want to profile. The Profile Wizard will scan your program, produce an initial estimate of the program's profile and then set the profile into "learning mode," where the profile rules are not enforced but violations of the rules are logged. The Profile Wizard will invite you to run your program in another window, and as you run the program through its operation, AppArmor builds up a log file of events that characterize the correct behavior of your program. Run your program through a thorough quality assurance cycle, exercising its major functionality and being careful not to run any attacks against the program. When you are done, return to the Profile Wizard window and click the Scan button. The Profile Wizard will then ask you a series of questions about how to respond to various file-access events. Typically, your program will have accessed some file and the Profile Wizard will ask you if you want to grant explicit access to precisely that literal file name or if you would like to grant access to some file pattern. The pattern might include wild cards or a set of rules (#include statement) that satisfy not just this event but others in the log file and future events. When you are finished answering questions, click Finish and answer Yes to exit the Profile Wizard. Your newly created profile will be loaded in enforce mode.

Updating Profiles

From time to time, AppArmor profiles may need to be supplemented. The Update Profiles Wizard utility works very much like the Add Profile Wizard utility except that it is designed for the ongoing improvement of AppArmor profiles rather than for initial generation. When you run the Update Profiles Wizard, it scans your current system log for AppArmor events and asks you what to do with each event, suggesting patterns as above.

Result Confirmation

The last step of all computer security hardening procedures is to verify the security of your configuration, a principle AppArmor follows. To verify the security of your AppArmor profiling efforts, run the unconfined program again and inspect the output to see that all programs exposed to attack have been profiled. If your computer system is a network server, your threats likely come from the network. Thus, the standard output of unconfined reporting by all network services listening to network ports exactly reflects the threats to which your computer is exposed. When all of the programs that produce unconfined reports are associated with AppArmor profiles, it is impossible for an attacker to directly access your file systems without going through the AppArmor policies you have set. For a worst-case analysis of the damage an attacker could inflict on your computer system, inspect each profile listed by unconfined. Viewing the profiles in vim is ideal, as that will show the profiles highlighted in color. The entire set of files an attacker can corrupt on your system is represented by the writable files (highlighted in yellow). This set is a great deal smaller than the set a network attacker could access without the AppArmor enforcement.

Summary

AppArmor proactively protects the operating system and applications from external or internal threats, even zero-day attacks, by enforcing good behavior and preventing even unknown application flaws from being exploited. AppArmor security profiles completely define what system resources individual applications can access and with what privileges. All threatened programs listed as unconfined should have an associated AppArmor profile. A number of default profiles are included with AppArmor, and using a combination of advanced static analysis and learning-based tools, AppArmor policies for even very complex applications can be deployed successfully in a matter of hours.

The Running System

Log Files

All log messages should be sent to a central log host, including application logs. These should be reviewed regularly to identify attacks and intrusions. Because the number of messages can be overwhelming, tools like swatch can help remove all the uninteresting stuff.

Updates

Update regularly. Subscribe to the important mailing lists (see the list in “Conclusion”) and install security patches as soon as possible. You can do this with YaST by running ***yast2 online_update***. Without timely updates, your system is certain to be compromised eventually. Almost all successful intrusions could have been prevented if security patches were installed as soon as they became available.

Conclusion

Using the suggestions outlined in this white paper, you can create a customized, secure environment with SUSE Linux Enterprise Server 9.

When you combine Novell AppArmor with SUSE Linux Enterprise Server 9, which has earned the industry's highest level of Linux security accreditation, your systems gain another layer of security. Novell AppArmor is included with SUSE Linux Enterprise Server 9 SP3 and is available as an add-on security solution for versions prior to SP3.

It is simple to install, and it automatically plugs into the Linux Security Module interface in the SUSE Linux Enterprise Server 9 kernel. A typical Novell AppArmor installation on SUSE Linux Enterprise Server 9 takes only a few minutes, and the result is far easier configuration and enforcement across Linux servers on the network. Visit www.novell.com/apparmor or contact your Novell sales representative for more information.

To keep your secure installation up to date and secure, you should at least subscribe to the most important security mailing lists:

- **suse-security**—*SUSE discussion list containing security-related subjects and security announcement; to subscribe, send a blank e-mail to: suse-security-subscribe@suse.com*
- **suse-security-announce**—*Security announcements only; to subscribe, send a blank e-mail to: suse-security-announce-subscribe@suse.com*
- **bugtraq**—*Discussion list addressing the latest security problems; to subscribe, send an e-mail with the following content to: listserv@securityfocus.com: "subscribe bugtraquser@domain"*

Finally, make sure that you have performed a system backup and a reboot of the system before the system is attached to the Internet. Always keep an eye on your system and logs.

Appendix

Tools

The following tools were developed by SUSE and can be downloaded free of charge:

SUSE Security Software

Name of Program (RPM)	Function	Included in the SUSE Distribution Since	Works on Other Linux Distributions	Download
SUSE Firewall2 (firewall)	A packet filter that also creates complex firewall systems and is very easy to configure	SUSE Linux 6.3	Yes (for the other distributions, init.d and start-up scripts must be adapted)	SUSE FTP Server
Security Checker (seccheck)	Checker that reviews the local security on a daily basis	SUSE Linux 5.2	Usually	SUSE FTP Server
Compartment (compartm)	Security wrapper for programs that supports chrooting, assignment of privileges and capabilities	SUSE Linux 7.0	Yes	SUSE FTP Server
Linux Audit Subsystem (laus)	Kernel audit module	SUSE Linux 8 SP3	Yes	SUSE FTP Server
Resource Manager Daemon (resmgrd)	Resource manager that allows applications to access and lock device files	SUSE Linux 8.2	Yes	SUSE FTP Server

Links

In addition to the security mailing lists, there is a comprehensive chapter about security in the SUSE Linux manual accompanying the distribution. Furthermore, have a look at the following links for security information:

- **SUSE security page:** *www.novell.com/linux/security/securitysupport.html*
- **SUSE Linux Enterprise Server updates and patches:** *www.novell.com/linux/suse/migration.html*
- **Security focus:** *www.securityfocus.com/*
- **Packetstorm:** *http://packetstormsecurity.org/*
- **Apache Group:** *www.apache.org/httpd.html*
- **Powerful SSL encryption for Apache:** *www.modssl.org/*
- **Novell AppArmor product information:** *www.novell.com/products/apparmor/*
- **Novell AppArmor technical whitepaper:** *www.novell.com/collateral/4821055/4821055.pdf*
- **Novell AppArmor documentation:** *www.novell.com/documentation/apparmor*
- **SUSE Linux Enterprise Server product page:** *www.novell.com/products/linuxenterpriseserver/*

www.novell.com

Contact your local Novell Solutions Provider, or call Novell at:

1 888 321 4272 U.S./Canada
1 801 861 4272 Worldwide
1 801 861 8473 Facsimile

Novell, Inc.
404 Wyman Street
Waltham, MA 02451 USA

Novell®