Global Information Assurance Certification Paper

Installing a Secure Network DHCP Registration System

GCUX Practical Assignment version 3.0

Option 1 - Securely Administering UNIX

Pamela Fournier

January 18, 2005

Table of contents

Abstract
One limitation of DHCP is that there is no accountability for IP address usage. NetReg is a Network DHCP Registration System which provides a means of linking user information to MAC and IP addresses on the network. Many educational institutions are using NetReg on their student networks. This paper focuses on the deployment of a NetReg server in an open network environment similar to what is found at many educational institutions. The following tasks will be covered in this paper:

- perform a risk analysis for a NetReg server deployed in an open network environment
- determine measures to be taken to secure the NetReg server
- provide step-by-step instructions to build a secure NetReg server
- describe procedures for maintaining the server
- describe procedures for auditing the server

Introduction
I am the Systems Administrator at an educational institution. We have been using a very customized version of NetReg for almost three years. NetReg is a software package developed by Peter Valian and documented by Peter Valian and Todd K. Watson both of Southwestern University in Georgetown, Texas. NetReg requires that computers be registered before being issued a routable IP address. This makes it possible to locate a problem computer on the network, and solve the problem or terminate the network connection. As part of the registration process, the person registering the machine must agree to abide by the displayed network policy, and must be authenticated by the chosen authentication mechanism.

NetReg requires that a DHCP, web and DNS server be installed on the NetReg server. The NetReg DHCP server is configured with two address pools per subnet – one for registered machines containing routable IP addresses and one for unregistered machines containing non-routable IP addresses. When a machine configured to use DHCP is connected to the network, it broadcasts a request for an IP address. The NetReg DHCP server receives the request. If the MAC address of the machine is listed in the DHCP server configuration file, the machine is considered registered and is assigned a routable IP address and is provided functional TCP/IP information. If the MAC address of the machine is not listed in the DHCP server configuration file, the machine is considered unregistered and is assigned a non-routable IP address and the NetReg nameserver (*Jaques, p. 4*).

The NetReg nameserver resolves all names to the NetReg server. URL requests from an unregistered machine, are redirected to the NetReg Network Registration page. The user completes the registration form, and the MAC address of the user's machine is added to the NetReg DHCP configuration file. Future DHCP requests from the now registered machine, will result in the

assignment of a routable IP address and functional TCP/IP information (*Jaques, p. 4*).

Because the NetReg server requires that a DNS, DHCP, and web server all reside on the same server box, the security risks for the NetReg server are increased. Each server application has vulnerabilities that must be addressed. This paper will lead you through the process of building a secure NetReg server. For security reasons, all IP addresses used in this paper are fictitious. The placement of spaces within file records is very important in many of the configuration files included in this paper. If you choose to use any of the sample files included in this paper, be very careful to enter spaces where spaces are indicated. Failure to do so may cause problems starting or running services.

Risk Analysis and Mitigation

Risk Analysis

The use of DHCP at educational institutions poses security risks intensified by the open network environment common at many such institutions. Students and faculty bring in their personal computers and plug them into the institution's network. Some of the computers have already been comprised, or are running services that can be easily exploited.

A security issue at many institutions is lack of security concern. Passwords are easily guessed, written down, or stored on computers in an unencrypted format. People fail to log out of programs. Doors left open and rooms left unoccupied, allow unauthorized people to have access to computers on the network.

Network users can pose a threat to servers, especially in an open network environment. Many users are computer savvy and curious. Some of the users like to poke at servers and explore the network. Free scanning software is widely available on the Internet and is often easy to use. Packet sniffers are also available for download on the Internet. The network users are located inside the enterprise firewall, so are not blocked by the firewall.

Wireless networks are being deployed at many institutions. The use of wireless networks can increase security risks. Unless the wireless devices are properly configured and secured, people may be able to access the network not only from within buildings, but from outside as well. This means that the network may be physically open to the world.

Rogue DHCP servers are becoming a common problem. Many wireless hubs have the capability of functioning as DHCP servers. Although the use of unauthorized wireless hubs is often banned, people still bring in their own wireless hubs. They either connect them to the wired network, or use them strictly in wireless mode. Some people don't realize that they have a DHCP server running on their wireless hub. Others simply don't care. Either way,

rogue DHCP servers are posing a real problem on networks.

DNS version queries and zone transfer attempts are threats to security which must be guarded against. A successful zone transfer can provide a hacker with the names and IP addresses of all equipment connected to the network. A successful version query can provide a hacker with the information needed to hack the nameserver. In addition, the potential risks of cache poisoning and IP spoofing need to be guarded against.

Web servers commonly experience probes and scans from curious people and hackers. Some hackers are interested primarily in defacing web pages while others are after whatever information they can find. Misconfigured web servers, improper permissions on web accessible files, and poorly written software all pose threats to server security.

Risk Mitigation

Because NetReg servers at educational institutions often reside in an open network environment, great lengths must be taken to secure the servers. The following steps will be taken to secure this NetReg server.

- The server will be physically located in a locked server room
- All unnecessary software will be removed and all unnecessary services will be turned off
- Only system administrators will be allowed to access the server via the console or ssh
- LogDog will be used to monitor system messages
- Snort will be used to detect rogue DHCP servers
- Administrative interfaces will be opened in a new window with an idle time window closing
- HTTPS will be used for administrative interfaces
- Iptables will be used to block pings and port scans
- Machines used to access administrative interfaces must have anti-virus software and all operating system patches installed
- Accounts will become locked after five failed login attempts
- IT staff logins will be disabled after hours
- Passwords will be required to be changed periodically and be hard to guess
- The DNS configuration file will be set to prevent zone transfers
- The DNS configuration file will be set to respond to version queries with false information
- To prevent poisoning of DNS cache, recursive lookups will not be allowed
- Access to the NetReg server will be blocked at the corporate firewall
- The DNS and DHCP servers will be chrooted

Server Requirements

Hardware Requirements

NetReg should be run on a dedicated server because of its use of DNS to resolve IP addresses to itself. The minimum hardware requirements for the NetReg server are a 200 MHz CPU, a 4 GB Hard Drive, 32 MB of RAM, and a 10/100 Ethernet Adapter (*Jaques, p. 5*). The NetReg server provides a critical network service, so it is desirable to install the server on a reliable, up-to-date machine. For our server, I selected a computer with a 2.4 GHz CPU, an 80 GB Hard Drive, 512 MB of RAM, and two 100mbps Ethernet Adapters.

Software

NetReg is compatible with RedHat Linux and other Linux platforms. When building a new server, I always use the latest version of the operating system available that the software will run on. Therefore, I have chosen Fedora Core 2 as the operating system for this server. I chose Apache 2 as our web server because of the ease of implementing SSL. NetReg uses ISC BIND and ISC DHCP. In addition to the software required for the installation of NetReg, a number of security related software packages will be installed.

Anti-virus

Although linux is not plagued by viruses as some operating systems are, I still choose to install anti-virus software on our linux servers. Clam AntiVirus provides anti-virus protection via a multi-threaded daemon, and command line scanner (*Clam AntiVirus*).

Syslog Monitoring

LogDog is a syslog monitoring tool that performs actions based on monitored words and phrases. The LogDog configuration file is used to specify which words and phrases to monitor and what actions to take (*Zehm "LogDog A daemon for monitoring syslogd messages and alerting administrators"*). System messages are read via a FIFO rather than reading logs which a hacker may have altered.

Email

SendEmail is an email program written in Perl. It can be used in bash scripts and Perl programs (*Zehm "SendEmail A tool for Sending SMTP Email from a Console"*). It is used in conjunction with LogDog to send LogDog alerts.

Intrusion Detection

Snort is a network intrusion detection system which can be used to detect attacks and probes against a server. Snort uses a rule set to determine which packets to collect or report. These rules can be adjusted to accommodate each server's particular configuration (*Caswell and Roesch*).

AIDE is an intrusion detection program which can be used to detect changes in

a file. This includes changes in permissions, owner, group, and file size.

Firewall
Iptables can be used to build firewalls based on packet filtering. Because the NetReg server will reside in an open network environment, it is desirable to install a firewall on the server.

Communications
OpenSSH uses encryption to provide secure communications between networked systems. Even though communications with this server take place strictly behind the enterprise firewall, the dangers of packet sniffing still exist.

Installation

Physical Installation

The server will be located in a locked – password protected room. To prevent unauthorized people from booting the server, the bios password will be set.

Operating System Installation

Installation - Fedora Core 2

The checksum or signature of all software packages downloaded should be verified before installation.

- Begin by booting off the system cd

I do not install desktops on our linux servers and have no mouse connected to the machine, so this will be a text installation.

- At the boot prompt type: linux text

The next series of screens will allow you to test your media, select the language used during installation, and provide information pertaining to your keyboard and monitor. You will then be asked to select an installation type. To avoid installation of packages that are not needed, a custom installation will be performed.

- Installation Type: Custom

You are provided the option of having the installation process partition the hard drive. I prefer to partition the drive myself, therefore, I select the option to use Disk Druid to partition the drive. I build our standard linux servers with 4 partitions – a swap partition, system partition, log partition, and software partition. I always place our system logs in a separate partition to prevent growing logs from filling the system partition and causing server failure. I store our software and data in a partition separate from the system files. If for any reason I have to rebuild the system partition, the software and data remain intact.

- Disk Partitioning Setup: Disk Druid

The next series of prompts pertain to boot loader configuration. I use GRUB rather than LILO. If you use GRUB, you should always be able to boot your system. This is because GRUB allows you to fully specify boot properties from the boot prompt (*BG<willygilly@attbi.com>*). You can set a boot loader password to prevent people from passing options to the kernel at boot time.

You need a static IP address for your server. Hardware requirements for NetReg specify one 10/100 Ethernet Adapter (*Jaques, p. 5*). I have two ethernet adapters installed in our NetReg server. One adapter provides server access for unknown DHCP clients. The other adapter provides server access for known DHCP clients and administrative users. The use of two ethernet adapters facilitates the use of iptables to limit unregistered users' access to ports on the NetReg server.

I prefer to use IP addresses in the higher range of the subnet for servers. Many DHCP servers award leases in ascending order, usually starting with 1 or some number below 10. This means if a rogue DHCP server assigns IP addresses in our address space, it will probably be discovered before it assigns the IP addresses in use by our servers.

- Network Configuration for eth0: Enter information for the network adapter
- Network Configuration for eth1: Enter information for the network adapter
- Miscellaneous Network Settings:
  Gateway → Enter Gateway
  Primary DNS → Enter the IP address of this NetReg server
  Secondary DNS → leave blank

I recommend that you do not name a server based on its function. In other words, don't name your DHCP server DHCP1.yourdomain.com. If you name your servers based on their function, there is no need for a hacker to port scan them. The hacker will know by the name of the server which ports to expect to see open. Even if you block address sweeps, hackers will be able to find your servers simply by their name.

- Hostname Configuration: Select "Manually" and enter a hostname

You are given the option of enabling a firewall. I choose not to enable the firewall at this point. I find troubleshooting software installations to be easier without the complexities of blocked ports. I build our servers on an isolated network so lack of a firewall at this point does not pose a security risk.

The next prompts allow you to specify language support, time zone choices, and root password. The root password must be one that is not easily cracked. A password based on a phrase is easy to remember and can be hard to break.

- Root Password: Enter a root password.

Potentially, all software packages have vulnerabilities. In addition, a vulnerability in one package can make way for exploiting a vulnerability in another package.

Therefore, only those packages that are necessary, should be installed on the NetReg server. I begin my server installations by doing a "Custom Installation" and then unchecking all packages.

- Package Group Selection: uncheck all packages

The next screens are informational. The installation will begin. When the installation is complete, you will be instructed to reboot your server.

Remove Unnecessary Software Packages

Although, I performed a custom installation and selected not to install any packages, there is still software installed which I will need to uninstall. I uninstall editors except for vi, schedulers except for cron, compilers except for perl, and remote connection software except for ssh. I uninstall finger, telnet, ftp, sendmail, talk, rmt, rdist, rsh, rdate, and any other software which will not be used on this server.

Install Patches

Next, I install patches that pertain to the software packages I have installed on the server. The server is not yet fully configured and secured, and so it remains on an isolated network. Because it is on an isolated network, I manually check for patches and then move them to the isolated server via cdrom. Patches for Fedora 2 can be downloaded from http://download.fedora.redhat.com/pub/fedora/linux/core/updates/2/i386.

Disable Services

IPV6 causes a serious slow down in DNS queries. Because we do not need support for IPV6, I disable IPV6. To do this add the line “alias net-pf-10 off” to the end of the /etc/modprobe.conf file (*Miranda*).

Assign Non-functional Shells to System Accounts

As part of the installation, Fedora Core 2 assigns non-functional shells to system accounts and locks the accounts so that they cannot be used to log into the server. Verify that all system accounts have non-functional shells and are locked.

Anti-virus Software Installation

The first thing I do after installing the operating system is install the Clam AntiVirus software. This way, the remainder of the software being installed is scanned for viruses before installation. The Clam binary can be downloaded at **http://www.clamav.net/**. You must have curl installed prior to installing Clam.

- Install the binary
- Change the settings in the /etc/clamd.conf file to match your desired level of protection

It is recommended that clamd be run in local mode for security reasons (*Example config file for the Clam AV daemon*).

- In the /etc/clamd.conf file, uncomment the "LocalSocket /var/run/clamav/clamd.sock" line
- In the /etc/clamd.conf file, comment out the "TCPSocket 3310" line

Automatic updating of virus definitions can be done through an Internet connection. /usr/bin/freshclam is used to perform Clam updates. Setup a cron job to execute freshclam on at least a daily basis. We execute freshclam on an hourly basis.

- 47 * * * * /usr/bin/freshclam (*Kojm*)

If you receive an error stating that Clamd could not be notified, edit the /etc/freshclam.conf file and comment out the line "Notify Clamd" (*Snow*).

ISC BIND Installation

Installation

NetReg requires the installation of ISC BIND. It can be downloaded at http://www.isc.org/index.pl?/sw/bind/. Before installing BIND, you must have gcc-c_++, libstdc++-devel, openssl, and openssl-devel installed.

- Remove all existing BIND software from your NetReg server
- Change directory into the BIND source directory

SO_BSDCOMPAT is an obsolete define in the 2.6 kernel and causes an error to be generated by the BIND process. To eliminate the error messages, remove all references to it in the BIND socket.c file before compiling BIND.

- Edit the /source directory/lib/isc/unix/socket.c file
- Around line 1297 you should find the following line:
  #if defined(USE_CMSG) || defined(SO_BSDCOMPAT)
- Change the line to:
  #if defined(USE_CMSG)
- Around line 1384 you should find the SO_BSDCOMPAT subroutine.
- Remove the entire subroutine (*Warne*).

To protect the NetReg server against damage caused if the nameserver is compromised, the nameserver is placed in a chroot jail. A chroot jail is a means of limiting a process's access to resources outside of a specified area by changing the root directory of the process to the one specified (*Friedl*).

- Begin the BIND installation by running configure. Because I am going to chroot BIND, I use /opt/chroot as the installation target directory
  ```
  ./configure --prefix=/opt/chroot \
  --sysconfdir=/opt/chroot/etc \
  --disable-threads \
  --localstatedir=/opt/chroot/var/state \
  --with-libtool \
  --with-openssl=/usr/ssl
  ```
- Run "make" and then "make install" to install BIND

The BIND process should not be run as root. If you run the nameserver as root, and the server is comprised, the hacker will have root access to your server. Add a user called named and add it to a group called named. Lock the named account so that it cannot be logged into.

- groupadd named
- mkdir -p /opt/chroot/var/named
- useradd -d /opt/chroot/var/named -g named -s /bin/false named
- passwd -l named (*Friedl*)

Subdirectories need to be created under the chroot jail to house the files necessary for the BIND process to run.

- cd /opt/chroot
- mkdir dev
- mkdir etc
- mkdir logs
- mkdir -p var/run (*Friedl*)

When a process is run in a chroot jail, all files that the process needs to run must reside in the chroot jail. This includes device files and the timezone file.

- mknod dev/null c 1 3
- mknod dev/zero c 1 5
- mknod dev/random c 1 8
- cp /etc/localtime etc (*Friedl*)

BIND Configuration File Creation

Next, the named configuration file needs to be created. This file should be created in the /opt/chroot/etc directory. When running a process in a chroot jail, all paths are relative to the jail. To prevent version information from being devulged, set version to "0.0.0". The NetReg nameserver will resolve all names to itself so "recursion" should be set to no.

- cd /opt/chroot/etc
- Create a named.conf file. Following is the named.conf file I used.

Sample named.conf File

```
options {
        directory "/var/named";
        dump-file "/var/tmp/named_dump.db";
        pid-file "/var/run/named.pid";
        statistics-file "/var/tmp/named.stats";
        memstatistics-file "/var/tmp/named.memstats";
       datasize 20M;
       allow-query {any;};
       listen-on {172.16.0.254;};
       recursion no;
       version "0.0.0";
};
logging {
   channel netreg_sec_chan {
     file "/logs/dns_logs" versions 5 size 10M;
     severity info;
     print-category yes;
     print-severity yes;
```

```
    print-time yes;
  };
  category default {
    netreg_sec_chan;
  };
};
# The root nameservers
zone "." in {
  type master;
  file "db.root";
  allow-transfer { none; };
};
```

Note: To test the DNS server locally, you must comment out the "listen-on" line in the named.conf file or set "listen-on" to include the 127.0.0.1 interface.

Zone File Creation

- In the /opt/chroot/var/named directory, create the db.root file. Following is a sample db.root file.

Sample db.root File

```
$TTL 3600
. IN SOA netreg.yourdomain.com root.netreg.yourdomain.com. (
            2           ;serial
        10800           ;refresh
         3600           ;retry
       604800           ;expire
        86400           ;default_ttl
)
                        IN  NS  netreg.yourdomain.com
netreg 86400    IN  A   172.16.0.254
*.     86400    IN  A   172.16.0.254          (Jaques, p. 23)
```

Remove Unnecessary Files

RNDC is a control utility for BIND. The NetReg nameserver is a very basic one. There are no zone updates or transfers. I do not find a need for a tool which provides all the functionality of RNDC on this nameserver. I, therefore, have chosen not to use RNDC. If you are not using RNDC, you will see two errors in the /var/log/messages file. One states that the rndc.key wasn't found, and the other states that the command channel 127.0.0.1 couldn't be added. This will not affect the workings of the nameserver.

Dynamic DNS can be used with BIND DNS, however, we have encountered a number of problems with using dynamic DNS in our environment.

a. Duplicate host names can cause problems on a network. From time to time, a machine will attempt to connect to our network with a

machine name that is already in use. If the machine is a member of a domain at another institution and the user uses a roaming profile to login to the machine, you cannot change the name of the machine. Changing the machine name results in lose of the roaming profile which means the user can no longer login to the machine.

b. The number of people who have a wireless card and an ethernet card in their machine is increasing. Because entries for known clients in the DHCP configuration file are based on MAC addresses, NetReg actually registers the network card not the machine. So, each network card has to have a separate registration. Once again we encounter the duplicate name problem. Two network cards require two registrations and two entries in DNS, but the machine only has one computer name.
c. BIND DNS does not allow underscores in host names. Unfortunately, people do use the underscore when naming their machines. These machines need to be renamed in order to have their machine name used for dynamic DNS. Again, we have the problem where if people change the name of their machine, and they are using a roaming profile to log into their machine, they lose their roaming profile and their ability to login.
d. We have a wide variety of machines and operating systems connecting to our network. Some of the machines do not pass their machine name in their lease request. Without a machine name, an entry cannot be made in DNS for the machine.

NetReg attempts to solve these problems by assigning names to computers which are based on the username used to register them. The problem with this is that people can tell simply by looking at the machine name, who owns the machine and can target that machine with hack attempts. We use static entries for DHCP addresses and avoid having to use dynamic DNS. For each IP address in the known pool, an entry is made in DNS. These entries are made regardless of whether or not the IP address is in use. The DNS names assigned are the last two ocets of the IP address preceded by the letters DHCP. For example, the DNS name for 123.123.161.10 is DHCP16110. With this naming scheme, people who are using Windows sharing, or other services on another machine, can easily determine the DNS name of the machine they are trying to access. They obtain the IP address of the machine from the owner of the machine they are trying to access, and then follow our naming scheme to determine the DNS name of the machine.

Files that are not necessary for the nameserver to function properly should be removed from the chroot jail. This includes utilities for nameserver control and zone file updating. Remember, this nameserver does not do zone transfers or zone file updates.

- Move nslookup, dig, and host from the /opt/chroot/bin directory to the /bin directory
- Remove nsupdate and isc-config from the /opt/chroot/bin directory

- Remove named-checkconf, named-checkzone, rndc, rndc-confgen, dnssec-keygen, dnssec-signzone, and lwresd from the /opt/chroot/sbin directory.

Permissions and Startup Files

Proper permissions need to be set on the directories and files contained in the chroot jail. You can use the following script provided by Steve Friedl on the unixwiz.net web site. Create this script in a directory other than the chroot jail directory because it is not used by the BIND process. A few minor modifications were made to the script to accommodate our chosen chroot jail directory.

named.perms

```
# named.perms
#
#   Set the ownership and permissions on the named directory
#
cd /opt/chroot
# By default, root owns everything and only root can
# write, but dirs have to be executable too. Note that
# some platforms use a dot instead of a colon between
# user/group in the chown parameters
chown -R root:named .
find . -type f -print | xargs chmod u=rw,og=r # regular files
find . -type d -print | xargs chmod u=rwx,og=rx   # directories
chmod o= etc/*.conf
find var/named/ -type f -print | xargs chown named:named
find var/named/ -type f -print | xargs chmod ug=r,o=
# the var/run business is for the PID file
chown root:named  var/run/
chmod ug=rwx,o=rx var/run/
# change the named file to executable
chmod a+x /opt/chroot/sbin/named
# named has to be able to create logfiles
chown root:named  logs/
chmod ug=rwx,o=rx logs/                                    (Friedl)
```

- chmod a+x named.perms
- Run the permissions script using the following command:
  sh named.perms

The nameserver must be started with the –t parameter which specifies the chroot jail, the –u parameter which specifies the user the process will run as, and the –c parameter which specifies the configuration file. These parameters can be specified in the startup script in the /etc/rc.d/init.d directory, or in a separate script which will be called by the startup script. The sample startup

script provided by Steve Friedl on the unixwiz.net web site follows:

named.start

```
# named.start
#
#       Note: the path given to the "-c" parameter is relative
#       to the jail's root, not the system root.
#       Add "-n2" if you have multiple CPUs
# usage: named [-c conffile] [-d debuglevel] [-f|-g] [-n
number_of_cpus]
#            [-p port] [-s] [-t chrootdir] [-u username]
cd /opt/chroot
# make sure the debugging-output file is writable by named
touch named.run
chown named:named named.run
chmod ug=rw,o=r   named.run
PATH=/opt/chroot/sbin:$PATH named  \
- t /opt/chroot \
- u named \
- c /etc/named.conf                                    (Friedl)
```

- chmod a+x named.start
- Start the nameserver

To start the nameserver automatically at boot time, requires the creation of an init file in the /etc/rc.d/init.d directory and symbolic links to the file in the rc.d directory.

- Create a startup file called named and place it in /etc/rc.d/init.d directory. I used the following startup file.

Sample Startup File

```
#!/bin/bash
# Init file for named server daemon
# chkconfig: 2345 11 25
# description: named server daemon
# source function library
. /etc/rc.d/init.d/functions
RETVAL=0
case "$1" in
    start)
          echo -n "Starting named: "
          /opt/named.start
          echo
          ;;
    stop)
         echo -n "Shutting down named: "
         killproc named
```

```
        echo
        [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/named
        ;;
  restart)
        $0 stop
        $0 start
        ;;
  *)
        echo "Usage: named {start|stop|restart}"
        exit 1
esac
exit $RETVAL
```

- chmod a+x named
- cd /etc/rc.d
- ln –s ../init.d/named rc2.d/S11named
- ln –s ../init.d/named rc3.d/S11named
- Reboot the NetReg server to verify that the nameserver starts properly at boot

To prevent other nameservers from talking to the NetReg nameserver, add a server statement in the other nameservers' configuration file making the NetReg nameserver a bogus server.

Web Server Installation

The NetReg server has a web based registration page and an administrative GUI. Authentication is required when using these interfaces. To provide secured transmission of usernames and passwords, SSL will be used with the Apache web server. The latest version of Apache 2 will be installed as the web server. The Apache packages do not include the SSL module, so you must install the mod_ssl rpm (*Apache 2 RPMs*).

- Use rpms to install the latest stable versions of Apache and mod_ssl

You can purchase a certificate from a certificate authority, or generate your own certificate. Even if you choose to purchase a certificate, a self-created security certificate is useful for configuring and testing the new web server. Generate a CA certificate using the CA script located in the /usr/share/ssl/misc directory.

- cd /usr/share/ssl/misc
- ./CA –newca
- Press enter to create
- Enter a phrase to use as a passphrase – remember this phrase
- Answer the prompts

The next step is to create a certificate signing request.

- ./CA –newreq
- Enter a phrase to use as a passphrase – remember this phrase
- Answer the prompts

- You will be prompted for a challenge password and an optional company name – if you do not need these you can just hit enter through them.

The certificate signing request needs to be signed.

- ./CA –sign
- When prompted, enter the passphrase you used in generating the certificate signing request
- Answer “y” to sign the certificate
- Answer “y” to commit the certification

The certificates need to be copied to the proper Apache directories.

- Copy newcert.pem to /etc/httpd/conf/ssl.crt/server.crt
- Copy newreq.pem to /etc/httpd/conf/ssl.key/server.key
- Enter the NetReg server name in the /etc/httpd/conf.d/ssl.conf file ("*Apache 2 RPMs*")

When the NetReg web server is started, you will be prompted for the certificate passphrase. The need to enter a passphrase prevents the web server from starting at boot. If the web server must start automatically at boot, you can disable the passphrase prompt. To disable the passphrase prompt, decrypt the server key using the following commands:

- cd /etc/httpd/conf/ssl.key
- cp server.key server.bak
- openssl rsa –in server.bak –out server.key ("*Apache 2 RPMs*")

Make changes in the /etc/httpd/conf/httpd.conf file to secure the web server.

- Set ServerSignature to Off
- Comment out the /var/www/manual directory section
- Comment out modules which are not neccessary to have loaded

Set the web server to start at boot

- In the /etc/rc.d/rc2.d and /etc/rc.d/rc3.d directories, create symbolic links to /etc/rc.d/init.d/httpd
- Reboot the NetReg server to verify that the web server starts properly at boot

ISC DHCP Installation

Chrooting DHCP

ISC DHCP is required for the installation of NetReg. It can be downloaded at http://www.isc.org/index.pl?/sw/dhcp/. The DHCP server will be placed in a chroot jail to protect the NetReg server against damage caused if the DHCP server is compromised. There is a chroot patch for ISC DHCP which can be downloaded from http://www.episec.com/people/edelkind/patches/ or http://www.linuxfromscratch.org/patches/downloads/dhcp/dhcp-3.0pl2-chroot-1.patch. You must apply the patch before installing the DHCP server.

- mkdir /opt/dhcpd

- cd /opt/dhcpd
- mkdir -p etc var/run var/state
- Download and extract the DHCP source code
- Copy the dhcp-3_0+paranoia_patch to the server directory in the DHCP server source directory
- Change directory to the server directory in the DHCP server source directory
- Apply the patch to dhcpd.c
  patch -Np0 < dhcp-3_0+paranoia_patch (*Gifford*)

Install DHCP

- Change directories into the source directory for the DHCP server
- ./configure --copts -DPARANOIA
- make
- make install

Create a lease file in the /opt/dhcpd/var/state directory (*Gifford*).

- touch /opt/dhcpd/var/state/dhcpd.leases

DHCP Configuration File Creation

The instructions for the NetReg DHCP server installation states that the DHCP server should be configured with two address pools per subnet (*Jaques, p. 3*). Our NetReg server is configured with only two address pools – one for unknown clients using one subnet and one for known clients using multiple subnets. Unknown clients are assigned IP addresses from a private address space, whereas known clients are assigned public IP addresses. There are two reasons for this. First of all, we do not use NAT on our network and have a limited number of public IP addresses. We do not want to use our public IP addresses for clients who have not been authorized to use our network. Secondly, by using different subnets for known and for unknown clients, we can tell very quickly if a machine is registered on our network by what IP address it has been assigned.

Many devices with built-in DHCP servers have a default IP address pool in the 192.168.0.0 to 192.168.255.255 or 10.0.0.0 to 10.255.255.255 address space. If you use a private address space other than either of those for your unknown address pool, detecting the presence of a rogue DHCP server will be much easier. When machines start popping up on your network with addresses from the 192.168.0.0 or 10.0.0.0 address space, it will be obvious that there is rogue DHCP server on your network.

- Create a dhcpd.conf file in the /opt/dhcpd/etc directory. The paths contained in the dhcpd.conf file are relative to the root of the chroot jail.

We want leases to renew in one minute for unknown clients because once a computer is registered on our network, we want it to obtain a routable IP address very quickly. We often have registered computers connect to the

network for short periods of time. Therefore, leases for known clients are set to 8 hours.

- Set max-lease-time and default-lease-time to 120 seconds for unknown clients.
- Set default-lease-time and max-lease-time for known clients to 28800 seconds.

The default is to allow bootp. We do not require bootp on our network, therefore, I set bootp to deny. This avoids possible security issues with bootp.

- Set bootp to deny

The ddns-update options need to be set to reflect the fact that we will not be using dynamic DNS on our network.

- Set ddns-update-style to none
- Set ddns-updates to off

Following is the dhcpd.conf file I used.

Sample dhcpd.conf File

```
## /etc/dhcpd/dhcpd.conf
## ISC DHCPD v3 Configuration file
##
deny bootp;
option domain-name “yourdomain.com”;
ddns-update-style none;
ddns-updates off;
subnet 172.16.0.0 netmask 255.255.255.0 {
   authoritative;
# Unknown clients get this pool.
pool {
   option domain-name-servers 172.16.0.254;
   max-lease-time 120;
   default-lease-time 120;
   range 172.16.0.5 172.16.0.250;
   allow unknown clients;
 }
}
shared-network yournetwork {
subnet 123.123.160.0 netmask 255.255.252.0{
   authoritative;
# Known clients get this pool.
pool {
   option domain-name-servers 123.123.161.251,
    123.123.161.252;
   option routers 123.123.160.1;
   option netbios-name-servers 123.123.161.253;
```

```
    max-lease-time 28800;
    default-lease-time 28800;
    range 123.123.161.10 123.123.161.250;
    deny unknown clients;
  }
}
subnet 123.123.160.0 netmask 255.255.252.0{
   authoritative;
# Known clients get this pool.
pool {
   option domain-name-servers 123.123.161.251, 123.123.161.252;
   option routers 123.123.160.1;
   option netbios-name-servers 123.123.161.253;
   max-lease-time 28800;
   default-lease-time 28800;
   range 123.123.162.1 123.123.162.254;
   deny unknown clients;
  }
 }
}
```

Permissions and Startup Files

The DHCP server needs to run as the same user that the web server runs as. The web server runs as user apache and group apache.

- chown -R apache:apache /opt/dhcpd/

To start the DHCP server automatically at boot time, requires an init file and symbolic links in the rc.d directory.

- Create a startup file called dhcpd and place it in /etc/rc.d/init.d directory. The startup file that I used follows.

Sample Startup File

```
#!/bin/bash
# Init file for dhcpd server daemon
# chkconfig: 2345 15 25
# description: dhcpd server daemon
# processname:  dhcpd
# config:  /etc/dhcpd/dhcpd.conf
#source function library
. /etc/rc.d/init.d/functions
RETVAL=0
case “$1” in
    start)
          echo -n “Starting DHCP Server: “
          daemon dhcpd -user apache -chroot /opt/dhcpd \
             -cf /opt/dhcpd/etc/dhcpd.conf  \
```

```
            -lf /var/state/dhcpd.leases -q
         echo
         ;;
    stop)
         echo -n "Shutting down dhcpd: "
         if [ -f /opt/dhcpd/var/run/dhcpd.pid ] ; then
              killproc dhcpd
         fi
         echo
         [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/dhcpd
         ;;
   restart)
         $0 stop
         $0 start
         ;;
   *)
        echo "Usage: dhcpd {start|stop|restart}"
        exit 1
esac
exit $RETVAL
```

- chmod a+x dhcpd
- cd /etc/rc.d
- ln –s ../init.d/dhcpd rc2.d/S15dhcpd
- ln –s ../init.d/dhcpd rc3.d/S15dhcpd
- Reboot the NetReg server to verify that the DHCP server starts properly at boot

NetReg Installation

Installation

Before installing NetReg, the following perl modules need to be installed:

Libnet → libnet-1.18.tar.gz
Mail::POP3Client → POP3Client-2.13.tar.gz
Net::IMAP → Net-IMAP-Simple-0.93.tar.gz
Authen::SASL → Authen-SASL-2.06.tar.gz
Convert::ASN1 → Convert-ASN1-0.18.tar.gz
IO::Socket::SSL → IO-Socket-SSL-0.95.tar.gz
Net::SSLeay → Net_SSLeay.pm-1.25.tar.gz
XML::NamespaceSupport → XML-NamespaceSupport-1.08.tar.gz
XML::SAX → XML-SAX-0.12.tar.gz
MIME::Base64 → MIME-Base64-3.00.tar.gz
Net::LDAP → perl-ldap-0.31.tar.gz (*Jaques, p. 5*)

The NetReg software can be downloaded from http://www.netreg.org. There are two packages that need to be downloaded – netreg-1.3rc2.tar.gz and netreg-cidr.tar.gz.

- Extract the netreg-1.3rc2.tar.gz file into /usr/local/src/netreg-1.3rc2 directory
- Place the netreg-cidr.tar.gz file in the /usr/local/src/netreg-1.3rc2 directory
- Extract the netreg-cidr-tar-gz file (*Jaques, p. 7*)

Installation of NetReg is a manual process. There is no installation script. The NetReg files need to be copied to various directories on the server.

- cd /usr/local/src/netreg-1.3rc2
- cp –r usr/local/apache/htdocs/* /var/www/html
- cp –r usr/local/apache/htsdocs/gfx /var/www/html
- cp usr/local/bin/* /usr/local/bin
- cp usr/sbin/* /usr/sbin
- cp –r usr/local/apache2/cgi-bin/ /var/www
- mkdir /etc/netreg
- cp etc/netreg/subnet.dat.example /etc/netreg/subnet.dat
- cp –r usr/lib/perl5/site_perl/NetReg /usr/lib/perl5/site_perl
- rename the /var/www/html/index.html to /var/www/html/register.html (*Jaques, p. 11*)
- cp usr/local/man/man3/* /usr/man/man3

Registration Web Page

When an unregistered machine makes a browser request, the NetReg nameserver resolves the name to the NetReg server. The default web page displayed by the NetReg server is a redirection to the registration web page.

- Create a web page, /var/www/html/index.html, to redirect requests to the registration web page. Replace 192.168.0.33 with the IP address of your NetReg server. Following is the sample index.html file provided by Patrick M. Jaques.

Sample index.html File

```
<html>
<head>
<title> Online Network Registration</title>
<meta http-equiv=”pragma” content=”no-cache”>
<meta http-equiv=”refresh” content=”0;
url=https://192.168.0.33/register.html”>
</head>
<body>
Redirecting.. please wait. If you are not redirected automatically,
    then click
<a href=”/register.html”> here</a>
</body>
</html>
```

(*Jaques, p. 12*)

NetReg Configuration Files

NetReg offers 5 different methods of authentication. The Variables.pm file in the /usr/lib/perl5/site_perl/NetReg directory contains variables for setting the authentication method.

- Edit the Variables.pm file and set the variables for your chosen method of authentication.
- Under "MISCELLANEOUS SETTINGS" adjust the settings to match your environment.

The /etc/netreg/subnet.dat file contains information pertaining to the subnets on your network that will use DHCP. You must edit this file and make changes to accommodate your subnets.

- Add a line for each subnet on your network that will use DHCP. The format for each entry is:
  - Network address and subnet mask for the subnet for registered clients
  - Location of the subnet
  - Number of leases available on the subnet
  - Network address and subnet mask for the subnet for unregistered clients (If the this is the same as for registered clients, leave this blank) (*Jaques, p. 13*)

Tailor the web pages to reflect your network settings and policies.

- Change the /var/www/html/register.html file to display your network policy
- Change the /usr/lib/perl5/site_perl/NetReg/Html.pm to reflect your helpdesk settings
- In the /usr/lib/perl5/site_perl/NetReg/Variables.pm file, change the myLOGO variable to point to your institution's logo file.

NetReg Control Files

The /usr/sbin/dhcpdctl file is used by NetReg to stop and start the DHCP server. Edit the dhcpdctl file changing the variables in the dhcpdctl file to point to the files in the DHCP chroot jail.

- Change the PIDFILE variable to /opt/dhcpd/var/run/dhcpd.pid
- Change the DHCPD variable to '/usr/sbin/dhcpd -user apache -chroot /opt/dhcpd -cf /opt/dhcpd/etc/dhcpd.conf -lf /var/state/dhcpd.leases -q'

When a machine is registered on the network, an entry for the machine's MAC address is made in the DHCP configuration file. The DHCP server must be restarted to activate changes in the configuration file. Occasionally, an entry to the configuration file is faulty resulting in the inability of the DHCP server to restart. To avoid problems caused by faulty entries, entries are not made directly to the DHCP configuration file. Instead, new entries are made to the dhcpd.conf.new file. Before using the new configuration file, it is tested to verify that it is not faulty. If the new configuration file is found to be faulty, it is copied to dhcpd.conf.bad and the current configuration file is copied over the new

configuration file. If the new configuration file is found not to be faulty, the current configuration file is copied to dhcpd.conf.bak. The dhcpd.conf.new file is copied to dhcpd.conf and the DHCP server is restarted.

- Copy the /opt/dhcpd/etc/dhcpd.conf file to /opt/dhcpd/etc/dhcpd.conf.new (*Jaques, p. 19*)
- Make sure the user which is running apache has read/write permissions to the /opt/dhcpd/etc/dhcpd.conf.new file

A refresh-dhcpdconf file comes with NetReg. It is located in the /usr/local/bin directory. Rather than use that file, I created my own refresh-dhcpdconf file which is based on the one provided by NetReg. The refresh-dhcpdconf file I used follows.

refresh-dhcpdconf

```
#!/bin/bash
if [ /opt/dhcpd/etc/dhcpd.conf.new -nt /opt/dhcpd/etc/dhcpd.conf ];
then
  echo "dhcpd.conf.new is newer than dhcpd.conf...testing
config..."
  /usr/sbin/dhcpd -t -cf /opt/dhcpd/etc/dhcpd.conf.new
  if [ "$?" -ne 0 ]; then
    echo "Error in dhcp configuration... leaving server running!"
    cp /opt/dhcpd/etc/dhcpd.conf.new
/opt/dhcpd/etc/dhcpd.conf.bad
    cp /opt/dhcpd/etc/dhcpd.conf /opt/dhcpd/etc/dhcpd.conf.new
    exit1
  else
    echo "Copying dhcpd.conf to dhcpd.conf.bak"
    cp /opt/dhcpd/etc/dhcpd.conf /opt/dhcpd/etc/dhcpd.conf.bak
    echo "Copying dhcpd.conf.new to dhcpd.conf"
    cp /opt/dhcpd/etc/dhcpd.conf.new /opt/dhcpd/etc/dhcpd.conf
    echo "Reloading the server..."
    /usr/sbin/dhcpdctl stop
    sleep 1
    /usr/sbin/dhcpdctl start
  fi
fi
```

- chmod a+x refresh-dhcpdconf
- Test refresh-dhcpdconf to verify that it works properly
- Add a cronjob to run refresh-dhcpdconf every minute.
  0-59/1 * * * * /usr/local/bin/refresh-dhcpdconf (*Jaques, p. 24*)

Administrative Interface

NetReg comes with an administrative interface which allows you to manage DHCP leases. It is important that this interface be kept as secure as possible. .htaccess is used to set password security on the administrative interface.

- Create a .htaccess file in the /var/www/cgi-bin/admin directory. Following is the .htaccess file I used.

Sample .htaccess File

```
AuthName "NetReg"
AuthType Basic
AuthUserFile /data/netreg
SSLRequireSSL  (Jaques, p. 15)
require valid-user
```

- Using htpasswd, create the password file you specified in the .htaccess file.

Most of our IT staff work only during business hours. For added security, access to the administrative interface for those individuals is not available after hours or on weekends. This is accomplished via a cronjob which renames the htaccess password file during off hours. If you have several shifts of staff members, you can have multiple htaccess password files which you manipulate in a similar fashion. Following are the cronjobs I used.

```
00 19 * * 1-5 mv /data/netreg /data/netreg.off
00 06 * * 1-5 mv /data/netreg.off /data/netreg
```

Change the /etc/httpd/conf/httpd.conf file to limit access to the administrative interface to the subnet used by the IT staff. Enter the following lines in httpd.conf. Change the settings to match your network.

Add to the httpd.conf File

```
<Directory "/var/www/cgi-bin/admin">
      AuthName "NetReg"
      AuthType Basic
      AuthUserFile /data/netreg
      AllowOverride authconfig
      Order deny,allow
      deny from all
      allow from 123.123.164.0
</Directory>
```

To prevent the NetReg administrative interface from being left open, the administrative interface is opened in a new window with an idle timeout set. This is done by creating a web page in the /var/www/html directory which calls the administrative interface. Following is the web page I used to do this.

Sample Web Page

```
<html>
<head>
<title>NetReg</title>
<meta http-equiv="Content-Type" content="text/html;charset=iso-8859-1">
<meta http-equiv=\"PRAGMA\" CONTENT="no-cache\">
<Script Language="JavaScript"><!--
      function winclose() {
```

```
            newwin.close()
      }
      function opendhcp() {
            newwin=window.open("https://123.123.164.254/cgi-
bin/admin/admin.cgi",
            "dhcp","height=600, width=800, scrollbars, resizable, toolbar=yes")
            self.setTimeout('winclose()',600000)
      }
--></script>
</head>
<body bgcolor="#FFFFFF" text="#660000" link="#0000FF" vlink=#"9933ff"
alink="#FF0000">
<center><img src="/gfx/newits1.gif" width="348" height="105"
border="0"></center>
<form method=post action="JavaScript: opendhcp()">
<p>
<center><font size="6" face="Arial,Helvetica,sans-serif"
color="#000000">NOTICE TO USERS<p></font>
<font size="4" color="#000000">
      Place a warning here telling unauthorized people to exit
</font>
<br><br><center><input type=submit name="submit" value="Continue">
</form></center></head></html>
```

LogDog Installation

You can download the source code for logdog from http://caspian.dotconf.net/menu/Software/LogDog/.

- Download and extract the source code for logdog.
- Change directory into the source directory for logdog.
- make install
- Edit the /etc/syslog.conf file and append the following line to the file:
  *.info |/var/log/fifo.logdog (*Zehm "Installing LogDog"*)
- Restart syslog
- Edit the /etc/logdog.conf file. Configure the alerts and monitors sections to meet your security needs. You can specify that LogDog stop processing alerts after an alert situation has occurred, by using the “stop” alert. I do not want monitoring to stop when an alert situation occurs. By stopping logdog monitoring, I lose some ability to assess the situation. For example, if a person attempts to break into your server using password guessing, you will receive an alert with each login attempt. If you use the “stop” alert you will only be notified of the first failed attempt and will not be notified of future failed attempts or successful logins.
- Start logdog

To start logdog automatically at boot time, requires the creation of an init file in the /etc/rc.d/init.d directory and symbolic links to the file in the rc.d

directory.

- Create a startup file called logdog. The startup file I used follows.

Sample Startup File

```
#!/bin/bash
# Init file for logdog daemon
# chkconfig: 2345 54 25
# description:  logdog
#source function library
. /etc/rc.d/init.d/functions
RETVAL=0
case “$1” in
    start)
          echo -n “Starting logdog: “
          /usr/local/sbin/logdog.pl -d
          echo
          ;;
    stop)
          echo -n “Shutting down logdog: “
          if [ -f /var/run/logdog.pid ] ; then
                killproc named
          fi
          echo
          [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/named
          ;;
   restart)
          $0 stop
          $0 start
          RETVAL=$?
          ;;
   *)
         echo “Usage: logdog {start|stop|restart}”
         exit 1
esac
exit $RETVAL
```

- chmod a+x logdog
- cd /etc/rc.d
- ln –s ../init.d/logdog rc2.d/S54logdog
- ln –s ../init.d/logdog rc3.d/S54logdog
- Reboot the NetReg server to verify that logdog starts properly at boot

SendEmail Installation

You can download sendEmail from http://caspian.dotconf.net/menu/Software/SendEmail/.

- Download and extract the source code for sendEmail.
- Change directory into the source directory for sendEmail

- Copy sendEmail to the /usr/local/bin directory
- chmod a+x /usr/local/bin/sendEmail (*Zehm sendEmail README file*)

Snort Installation

You can download the binary for snort and the snort rules from http://www.snort.org/dl/

- Install the snort rpm
- Extract the snort rules
- Copy the entire rules directory to /etc/snort overwriting the existing /etc/snort/rules directory
- Edit the /etc/snort/snort.conf file and change the variables to reflect your network environment
- In the snort.conf file, comment out the rule sets you choose not to use and uncomment the rule sets you choose to use
- Change the settings in the /etc/sysconfig/snort file to match your network settings

Create a snort rule which detects rogue DHCP servers.

- Add the following to /etc/snort/snort.conf. Change the IP addresses to the IP addresses of your NetReg server.
  # List of DHCP servers on your network
  var RES_NET [172.16.0.254/32,123.123.161.254/32]
- Add the following rule to /etc/snort/rules/dns.rules.
  alert idp !$RES_NET 67 → any 68 (msg:"Rogue DHCP server";) (*Currier*)

Create the log directory and edit the startup file.

- Create the /var/log/snort directory.
- Make the snort user the owner of the directory.
- Grant the snort user permissions to read and write to the /var/log/snort directory.
- Reboot the NetReg server to verify that snort starts properly at boot

Configure iptables

Even if you selected "No Firewall" during the installation of the operating system, iptables is installed as part of the base installation. This means you don't have to install iptables, you just have to configure it. Following are the set of rules I used on our NetReg server (IP addresses have been changed for security reasons).

Firewall Rules

```
-A INPUT -i eth0 -d 123.123.161.254 -j DROP
-A INPUT -i eth1 -d 172.16.0.254 -j DROP
-A INPUT  -i eth0 -p tcp -m tcp --sport 68 --dport 67 -j ACCEPT
-A INPUT -s 123.123.161.0/255.255.255.0 -i eth1 -p tcp -m tcp --sport 68 --dport
67 -j ACCEPT
-A INPUT -s 123.123.162.0/255.255.255.0 -i eth1 -p tcp -m tcp --sport 68 --dport
67 -j ACCEPT
-A INPUT -s 172.16.0.0/255.255.255.0 -i eth0 -p udp -m udp --dport 53 -j
```

```
ACCEPT
-A INPUT -s 123.123.164.0/255.255.255.0 -i eth1 -p tcp -m tcp --dport 443 -j
ACCEPT
-A INPUT -s 172.16.0.0/255.255.255.0 -i eth0 -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -s 127.0.0.1 -j ACCEPT
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -j DROP
```

Unregistered machines communicate with the NetReg DHCP server via eth0. Registered machines with 123.123.0.0 addresses communicate with the NetReg DHCP server via eth1. The 123.123.164.0 subnet is used for static IP addresses. Static IP address are used for administrative machines which require static IP addresses for access to such things as the NetReg Administration page. The NetReg Administration interface is available only via https. Because only the IT staff should have access to the Administration interface, only the 123.123.164.0 subnet is granted access to port 443. Once a machine is registered, there is no need for it to access the NetReg Registration page which is accessed via port 80. Therefore, only the 172.16.0.0 network has access to port 80. Only the 172.16.0.0 IP addresses needs access to the NetReg DNS server.

SSH Installation

Openssh is installed with the base installation of Fedora Core 2.

- Upgrade the openssh packages to the latest version.

There are known security issues with Protocol 1 so Protocol 1 should be disabled.

- Change #Protocol 2,1 to Protocol 2 (*rickn*)

Do not allow remote root login. If you must have root privileges, use su to become root.

- Change #PermitRootLogin yes to PermitRootLog no (*rickn*)

A challenge-response buffer overflow vulnerability existed in OpenSSH that could lead to root compromise. Privilege separation protects against this compromise.

- Change #UsePrivilegeSeparation no to UsePrivilegeSeparation yes (*"Installing OpenSSH on Solaris"*)

Account Policies

A good password policy is basic to server security. We require that people use passwords that are hard to crack. We force users to change their password every 90 days. Three days of inactivity are allowed after a password expires before the account is locked. This period of inactivity is allowed to account for non-use of an account on weekends.

- Use the chage command to set the maximum number of days a password is valid, and the number of days of inactivity after a password expires before the account is locked.

chage -M 90 -I 3 username

Use pam_tally to set a lockout policy for accounts. On this server, accounts will be locked out after three failed login attempts.

- Edit the /etc/pam.d/system-auth and add the following lines:

auth required /lib/security/pam_tally.so onerr=fail no_magic_root
account required /lib/security/pam_tally.so deny=5 no_magic_root reset (*Nguyen*)

- Create the /var/log/faillog file
  touch /var/log/faillog (*Nguyen*)

Accounts will be reset every night. This is done via a cron job. I used the following cronjobs to save the account status to a file and then reset accounts.

```
53 19 * * * date>>/var/log/pam_tally
55 19 * * * /sbin/pam_tally > >/var/log/pam_tally
00 20 * * * /sbin/pam_tally --reset
```

Aide Installation

You must have bison, m4, flex, and mhash-0.8.17 (newer versions generate an error when compiling aide) installed. You can download mhash from http://mhash.sourceforge.net.

- ./configure
- make
- make install (*Lehti*)
- copy the aide.conf from the doc directory located under the aide source directory to /usr/local/etc
- mkdir /usr/local/aide

Edit the configuration file

- Set the TOPDIR variable to "/usr/local/aide"
- Set the database variable to "file:@@{TOPDIR}/aide.db"
- Begin with the rule set provided on linsec.ca

Rule set provided on linsec.ca

```
GLOG=>
DEV=p+n+u+g
CONF=R+sha1
BIN=R+sha1
LOG=p+n+u+g
All=R+a+sha1+rmd160+tiger+crc3
2
# main configuration
/bin                                    BIN
/lib                                    BIN
/sbin                                   BIN
/usr/bin                                BIN
/usr/lib                                BIN
```

/usr/sbin	BIN
/usr/local/bin	BIN
/usr/local/lib	BIN
/usr/local/sbin	BIN
!/usr/src	
!/usr/local/src	
/boot	BIN
/bin/System.map$	BIN-m-c
/dev	DEV
/etc	CONF
/etc/mtab$	LOG
/var/log	GLOG
/usr/local/etc/aide.conf$	All
/usr/local/bin/aide$	All
/usr/local/aide/aide.db$	All *Danen*)

- Delete
  !@@{TOPDIR}/src/.*\.o
  !@@{TOPDIR}/src/(aide|core)$ L
- Add rules for the DNS files
  /opt/chroot/etc CONF
  /opt/chroot/logs LOG
  /opt/chroot/var/named CONF
- Add rules for the DHCP files
  /opt/dhcpd/etc LOG
  /opt/dhcpd/var LOG
- Add rules for Apache
  /var/www CONF

Aide should be run immediately at the completion of the server and then periodically after that.

- Run "aide --init" to initialize the database
- cp /usr/local/etc/aide.db.new /usr/local/aide
- cp /usr/local/etc/aide.db.new /usr/local/aide/aide.db
- Run "aide --check" to verify that aide is working properly

A copy of the database should be stored off the server to prevent attackers from altering the database. Create a cron job to periodically perform a check of the system with aide. I run a check once in the morning, once at noon, and once at night.

```
00 6 * * * /usr/local/bin/aide --check >>/var/log/morning
00 12 * * * /usr/local/bin/aide --check >> /var/log/noon
00 20 * * * /usr/local/bin/aide --check >> /var/log/nite
```

Ongoing Maintenance

Annual Upgrade

I completely rebuild each of our servers once each year. This allows me to upgrade the hardware, operating system, and software, and install new software. The upgrade is built on a computer other than the current server box. The upgrade is throughly tested before the server is brought on-line. As part of the upgrade process, I review user accounts and software packages to verify all are still needed. I check permissions on files and directories to verify that all are set properly and any permissions that can be changed to tighten security are changed.

Operating System Maintenance

The Fedora Project web page (http://fedora.redhat.com/download/mirrors.html) contains a list of mirrors from which you can download updates to the operating system. You can periodically download updates manually or use one of the Fedora packages which provide automated updating. Verification of accounts and permissions should be done periodically throughout the year.

Software Maintenance

Software upgrades are performed when they are available and as needed. I never do a software upgrade on a live server without first testing it on a test server. I review upgrades and only do those which are needed for security reasons or server performance. Some upgrades are strictly feature oriented and unless we need the enhanced features, I refrain from performing the upgrade until our annual server rebuild. I subscribe to security mailing lists and keep abreast of security warnings. I periodically search the vendor web pages for updates to the software.

Backups and Redundancy

The NetReg server is backed-up to tape each night. In addition, I have a redundant NetReg server which syncs every 15 minutes to the NetReg server. If the NetReg server fails, I can switch to the redundant server. I backup the /opt/dhcpd/etc/dhcpd.conf file every 15minutes to a different directory on the NetReg server. Backups of the dhcpd.conf file are kept on the server for one week. I have found that most problems which occur with the NetReg server are a result of a problem with the dhcpd.conf file. Backing up this file allows me to quickly recover from a problem by restoring the file from the backup on the server.

System Auditing

I review server logs on a regular basis. This includes system logs, snort logs, and output from aide. I check permissions on the /etc/passwd and group files. I verify that the system date is correct. I use the "rpm -qa --last" command to verify that no new packages have been installed. A record of the audit is maintained so discrepancies can be detected and tracked.

I use the "ps -ef" command to verify that the following processes are running: snort, logdog, sshd, clamd, httpd, dhcpd, and named. Make sure the dhcpd and named processes are chrooted.

Use "netstat -a" to verify that only the following ports are open: ports 80 and 53 are opened only to the subnet for unknown clients; port 443 is opened only to the subnet for administrative users; tcp ports 3310, 22, udp ports 67 and 32768, and a raw icmp port will also be listed.

Use "iptables -L" to verify that the firewall rules are correct and in use.

A good test of the NetReg server is to connect to the network with an unregistered machine, and go through the registration process. Try entering an incorrect username or password to verify that authentication is working properly. Verify that the unregistered machine has an IP address from the pool for unknown clients. After registering, verify the machine has an IP address from the pool for known clients. While the machine is unregistered, try accessing the Administration web page to verify it is blocked. Try accessing the Administration web page using incorrect authentication information.

Summary

NetReg has provided Network Administrators with a means of managing their DHCP networks. It can be used as a starting point to communicate with users who the IT staff would otherwise have little or no contact with. The NetReg server at our institution emails a welcome message to users when they register their machine. The message provides information pertaining to email and helpdesk support.

Several institutions are using Nessus in conjunction with NetReg to perform security scanning of machines before they are allowed to register. Nessus is a vulnerability scanner available for download from http://www.nessus.org. In an open network environment, people bring their own machines in and connect them to the network. The use of Nessus in conjuction with NetReg provides IT staff with an opportunity to scan machines and resolve security issues before they cause problems.

Brown University has a software package called "Reggie" which works with NetReg to perform continual scans of network machines. When a machine is found to pose a security risk, the owner is first notified of the situation and given a grace period to fix it. If the situation goes unresolved, the machine's lease is terminated (*Ballem*). In an open network environment, this can be a very valuable tool.

NetReg is now a SourceForge project. The project web page can be accessed at http://sourceforge.net/projects/netreg/.

List of references

"Apache 2 RPMs - Installation and Configuration on Fedora Core." 26 May 2004. 29 Oct. 2004 <http://www.linux-sxs.org/internet_serving/apache2.html>.

Ballem, John P. "Reggie." 9 Jun. 2004. 3 Jan. 2005 <http://www.brown.edu/Facilities/CIS/Projects/netreg/reggie/>.

BG<willygilly@attbi.com>. "RE: Grub vs Lilo." E-mail to <redhat-list@redhat.com> 28 Jun. 2004. 5 Nov. 2004 <http://www.redhat.com/archives/redhat-list/2002-June/msg02729.html>.

Caswell, Brian., and Roesch, Marty. "Snort 2.3.0 RC2." 11 Jan. 2005. 12 Jan. 2005 <http://www.snort.org/about.html>.

"Clam AntiVirus." 17 Oct. 2004. 14 Nov. 2004 <http://www.clamav.net>.

"Example config file for the Clam AV daemon." 2 Nov. 2004 Clam AntiVirus configuration file - clamd.conf.

Currier, Robert. "Address unknown, part2." 27 Nov. 2000. ITworld.com. 31 Dec. 2004 <http://security.itworld.com/4363/ITW3542/page_1.html>.

Danen, Vincent. "AIDE: Advanced Intrusion Detection Environment." 28 Apr. 2003. 22 Nov. 2004 <http://linsec.ca/bin/view/Main/AiDE>.

Friedl, Steve. "Steve Friedl's Unixwiz.net Tech Tips Building and configuring BIND 9 in a chroot jail." 20 Nov. 2004 <http://www.unixwiz.net/techtips/bind9-chroot.html>.

Gifford, Jim. "dhcp-3.0pl2-chroot-1.patch." 14 Sep. 2003. 11 Dec. 2004 <http://www.linuxfromscratch.org/patches/downloads/dhcp/dhcp-3.0pl2-chroot-1.patch>.

"Installing OpenSSH on Solaris." 12 Dec. 2004 <http://www.techgirl-net.com/ssh.html>.

Jaques, Patrick M. "Installing NetReg v1.3rc2 HOWTO." 17 Jun. 2004. 25 Oct. 2004 <http://www.netreg.org/contrib/NetReg-HowTo-A2.pdf>.

Kojm, Tomasz. "Clam AntiVirus 0.80rc4 User Manual." 11 Oct. 2004. 14 Nov. 2004 <http://www.clamav.net/doc/0.80rc4/html/node21.html>.

Lehti, Rami. "The Aide manual." 16 Nov. 2004 <http://www.cs.tut.fi/~rammer/aide/manual.html#compilation>.

Miranda, Mauriat. "Personal Fedora Core 2 Installation Guide." 12 Jul. 2004. 15 Nov. 2004 <http://www.mjmwired.net/resources/mjm-fedora-fc2.shtml#ipv6>.

Nguyen, Sonny. "Security with pam_tally." 7 May 2004. 31 Dec. 2004 <http://fedoranews.org/contributors/sonny_nguyen/pam/>.

rickn. "A Quick Word on Securing SSH." 2 May 2004. 15 Dec. 2004 <http://www.geekspeek.org/modules.php?name=News&file=article&sid=36>.

Snow, Sam <snowsam@laurel-point.net>. "Re: freshclam reports error (because I'm not using clamd?)." E-mail to "Adam Funk" <a24061@yahoo.com> 16 Sep. 2004. 14 Nov. 2004 <http://lists.debian.org/debian-user/2004/09/msg01948.html>.

Warne, Nick. "LinuxHints/RedHat 2.4 To 2.6 Kernel Upgrade." 3 Feb. 2004. 23 Nov. 2004 <http://www.hants.lug.org.uk/cgi-bin/wiki.pl?LinuxHints/RedHat_2.4_To_2.6_Kernel_Upgrade>.

Zehm, Brandon. "LogDog A daemon for monitoring syslogd messages and alerting administrators." 14 Nov. 2004 <http://caspian.dotconf.net/menu/Software/LogDog/>.

Zehm, Brandon. "Installing LogDog." 6 Jan. 2003. 14 Nov. 2004 INSTALL file included in the logdog-v.200-RC5 package.

Zehm, Brandon. "SendEmail A Tool for Sending SMTP Email from a Console." 14 Nov. 2004 <http://caspian.dotconf.net/menu/Software/SendEmail/>.

Zehm, Brandon. "sendEmail." 5 Oct. 2005. 14 Nov. 2004 README file included in the sendEmail-v1.51 package.