Basic Linux Commands for Everyday Usage

### 1.  List the contents of the current directory:

*# ls  
Desktop Documents Downloads Music Pictures Public Templates  
Videos*

### 2. Show the current location:

*# pwd  
/home/user01*

Presently, you are in user01’s home directory, so the output of the ls command was all directories that belong to user01. Let’s move out of user01’s home directory into one of the subdirectories. Using the cd command, you can move between different directories.

### 3. Move down one level into the Documents directory:

*# cd Documents/*

Note: The trailing slash (/) is optional when you’re using the cd command. It indicates that the name being specified is a directory.

### 4. Now view the current location again:

*# pwd  
/home/user01/Documents*

What if you want to move up one level to the directory you just came from? If you use ls, you don’t see the user01 directory listed. You can, however, view two special directories.

### 5. View all hidden directories with the ls –a command:

*# ls -a  
. ..*

Notice what seems like just a bunch of dots? They actually stand for two special types of directories. The first—the single .—stands for the current directory. The second—double ..—is the directory above where you  
currently are located.

### 6.To get back to the previous user01 directory, use the following:

*# cd ..*

### 7. Verify with the pwd command:

*# pwd  
/home/user01*

### 8. Check the file type of test1:

*# file test1  
test1: empty*

### 9. Check the type of the password file on the system:

*# file /etc/passwd  
passwd: ASCII text*

**10. View the beginning of the messages log file:**

*# head /var/log/messages  
Dec 5 03:13:06 RHEL01 dhclient: DHCPREQUEST on eth0 to*

*172.27.100.163 port 67  
Dec 5 03:13:10 RHEL01 dhclient: DHCPREQUEST on eth0 to  255.255.255.255 port 67  
Dec 5 03:13:20 RHEL01 dhclient: DHCPREQUEST on eth0 to 172.27.100.163 port 67  
Dec 5 03:13:29 RHEL01 dhclient: DHCPREQUEST on eth0 to 255.255.255.255 port 67  
Dec 5 03:13:30 RHEL01 dhclient: DHCPREQUEST on eth0 to 172.27.100.163 port 67  
Dec 5 03:13:43 RHEL01 dhclient: DHCPREQUEST on eth0 to 255.255.255.255 port 67  
Dec 5 03:13:44 RHEL01 dhclient: DHCPREQUEST on eth0 to 172.27.100.163 port 67  
Dec 5 03:13:50 RHEL01 dhclient: DHCPREQUEST on eth0 to 255.255.255.255 port 67*

### 11. View the end of the messages log file:

*# tail /var/log/messages  
Dec 11 08:11:04 RHEL01 dhclient: DHCPDISCOVER on eth0 to*

*255.255.255.255 port 67 interval 13  
Dec 11 08:11:04 RHEL01 dhclient: DHCPOFFER from 172.27.100.163  
Dec 11 08:11:04 RHEL01 dhclient: DHCPREQUEST on eth0 to  
255.255.255.255 port 67  
Dec 11 08:11:04 RHEL01 dhclient: DHCPACK from 172.27.100.163  
Dec 11 08:11:04 RHEL01 NET[26281]: /sbin/dhclient-script : updated  
/etc/resolv.conf  
Dec 11 08:11:04 RHEL01 dhclient: bound to 172.27.100.226 — renewal  
in 40864 seconds.  
Dec 11 08:18:00 RHEL01 abrt[26389]: saved core dump of pid 26388  
(/usr/libexec/fprintd) to /var/spool/abrt/ccpp-129207348026388.  
new/coredump (757760 bytes)  
Dec 11 08:18:00 RHEL01 abrtd: Directory ‘ccpp-1292073480-26388’  
creation detected  
Dec 11 08:18:00 RHEL01 abrtd: Crash is in database already (dup of  
/var/spool/abrt/ccpp-1291114420-26066)*

### 12. Copy the log file into the user01 home directory:

*# cp /var/log/messages /home/user01*

### 13. You also could use the following:

*# cp /var/log/messages .*

Remember that the dot (.) represents the current location. After the messages log file is copied over, you should probably rename it for safekeeping.

### 14. Rename the file by specifying the filename and the new name of the file:

*# mv messages messages.bak*

### 15. With the file renamed, move it to the test directory for safekeeping:

*# mv messages.bak test/*

Because you specified a directory this time, the file was moved instead of renamed. You can also verify that the file was moved correctly.

### 16  Deleteing a file i.e. messages.bak

*# cd test  
# rm messages.bak  
rm: remove regular file `messages.bak’? y*

Notice that you are prompted to delete the file? By using the –f option, you can skip the confirmation. While you’re deleting things, also remove the test directory.

### 17. Delete a directory and it’s subcontinents recursively

*# cd ..  
# rm -Rf test/*

**18. View file permissions for user01’s home directory:**

*# ll /home/user01  
total 32  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Desktop  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Documents  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Downloads  
-rw-rw-r–. 1 user01 user01 0 Dec 11 07:44 file1  
-rw-rw-r–. 1 user01 user01 0 Dec 11 07:44 file2  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Music  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Pictures  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Public  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Templates  
drwxr-xr-x. 2 user01 user01 4096 Dec 11 07:43 Videos*

### 19.  Change the owner of file1 from user01 to user02:

*# chown user02 file1*

### 20. Change the group of file2 from user01 to user02:

*# chown :user02 file2*

### 21. Now check the permissions again:

*# ll  
-rw-rw-r–. 1 user02 user01 0 Dec 11 07:44 file1  
-rw-rw-r–. 1 user01 user02 0 Dec 11 07:44 file2*

### 22.  Change the permissions in the “other” section to allow write access to this file:

*# chmod 666 file1*

# I/O direction commands in Linux

There are also a few characters you can use to direct or redirect output of commands.   
These characters are

* **>** Directs output to a file or device (overwrites if the file exists)
* **<** Directs input from the file or device
* **>>** Appends output or text to a file (creates if the file doesn’t exist)
* **|** Pipes the output of one command to another
* **&&** Combines commands

### 

### 1.  Use the echo command to output some text to a file:

*# echo “This is some sample text” > file\_example*

Normally, the echo command just displays the text you have given it back to the screen, but because you are using the output direction character,  the output is pushed to the file specified instead.

### 2. Verify that the text was output correctly by viewing the contents of the file:

*# cat file\_example   
This is some sample text*

### 3. Display the third field of the text using the space as a delimiter:

*# cut -d “ “ -f3 file\_example*

### 4. Combine the two commands into a single line:

*# cat file\_example | cut -d “ “ -f3 file\_example*

### 5. Execute one command and then another:

*# echo “This is some more text” > file\_example && cut -d “ “ -f3 file\_example*

Check the current contents of the sample file:

*# cat file\_example   
This is some more text*

### 6. Output some more text to this file:

*# echo “Different text” > file\_example*

### 7. Verify the contents of the file again:

*# cat file\_example   
Different text*

What happened to the original text? When you use the >, the output is sent to a file or device. However, it always overwrites what is in the current file or device. If you want to append text, you can use the same   
character twice.

### 8. Append text to the file instead of overwriting it:

*# echo “My original text” >> file\_example*

### 9. Verify the contents of the file:

*# cat file\_example   
Different text   
My original text*

### 10. Run the same exact command again and view the contents of the file:

*# echo “My original text” >> file\_example && cat file\_example   
Different text   
My original text   
My original text*

Notice there are now two lines with the same text. What if this was a config file for a service with duplicate data? In that case, you can use the uniq command to pull only unique lines from a file, making sure that   
there are no duplicates.   
Step 7. View only unique lines in the sample file, create a new file based on the output, and view the contents of this new file:

*# uniq file\_example > uniq\_file && cat uniq\_file   
Different text   
My original text*

# Installing and configuring SSH Service in Linux

You know that at some point or another you will need to be able to administer servers remotely, and you wouldn’t want to do this without some sort of security in place, right? SSH can protect the traffic that passes from your computer to a remote computer, making the tunnel secure for administration.

The tunnel that SSH uses for communication is encrypted, unlike protocols such as Telnet that don’t employ any encryption.

SSH is a fairly easy service to set up and is useful in many different ways. Although it is usually installed by default, you also should verify.

Step 1. Verify that the SSH server package is installed:

*# rpm -qa | grep ssh  
libssh2-1.2.2-7.el6.x86\_64  
openssh-5.3p1-20.el6.x86\_64  
openssh-server-5.3p1-20.el6.x86\_64  
openssh-clients-5.3p1-20.el6.x86\_64*

Step 2. If the SSH server package isn’t installed, do the following:

*# yum install -y openssh-server*

Step 3. Make sure to verify again:

*# rpm -qa | grep ssh  
openssh-server-5.3p1-20.el6.x86\_64*

Step 4. Ensure that the service is currently running (or start it if it isn’t):

*# service sshd status  
sshd (pid 678) is running…*

Now that the service is running, you should also make sure that the service is set to start when the system boots.

Step 1. Enable the service Ïduring boot:

*# chkconfig sshd on*

Step 2. Verify that the service is enabled during boot:

*# chkconfig –list sshd  
sshd 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

Now let’s move on to configuring the SSH service for use.

## Configuring SSH

After the SSH package is installed, you can start the configuration by looking at the  main config file.

The config file, which is located at /etc/ssh/ssh\_config, comes with a set of “safe” default settings. Before making any changes, however, you should make a backup copy of the original file in case something happens that you need to revert to a clean config file later and you want to restore the original.

Make a backup of the main config file:

*# cp /etc/ssh/sshd\_config /etc/ssh/sshd\_config.orig*

By default, Red Hat has the SSH service installed, running, and a firewall rule allowing incoming connections for the service. Because the config file also has a decent set of default options, you could log in at this point and begin doing work on your server.

Let’s look through the config file at some options:

*# cat /etc/ssh/sshd\_config  
…  
Port 22  
Protocol 2  
ListenAddress 0.0.0.0  
SyslogFacility AUTHPRIV  
PermitRootLogin yes  
PasswordAuthentication yes*

*ChallengeResponseAuthentication no  
UsePAM yes  
X11Forwarding yes  
Subsystem sftp /usr/libexec/openssh/sftp-server*

Let’s discuss some of the options laid out here:

All these options should be self-explanatory, but there are two options that I recommend changing before you use the SSH service:

*PermitRootLogin = No  
X11Forwarding = No*

Changing these two options provides a little extra security on your system. As you learned in Chapter 7, “User Administration,” you should never use the root account locally, so allowing that account to log in remotely is just not a good idea. After you finish making any changes to the SSH service, you need to restart the service for the  hanges to take effect.

Restart the SSH service:

*# service sshd restart  
Stopping sshd: [ OK ]  
Starting sshd: [ OK ]*

## Firewall and SELinux Configuration

When it comes to remote management in Linux, SSH is the standard. Because SSH is installed by default in Red Hat and there is a firewall rule already in place for you to begin remote management, you do not need to create any additional rules.

As is good practice, though, you should verify that the rule is, in fact, in place. Because SSH uses TCP port 22 for remote access, you should query any rule from the/etc/sysconfig/iptables file:

*# cat /etc/sysconfig/iptables | grep 22  
-A INPUT -m state –state NEW -m tcp -p tcp –dport 22 -j ACCEPT*

With the firewall rule already in place, you also need to look at SELinux restrictions on the SSH service.

Step 1. Query the Boolean values associated with SSH:

*# getsebool -a | grep ssh  
allow\_ssh\_keysign –> off  
sftpd\_write\_ssh\_home –> off  
ssh\_sysadm\_login –> off*

You need to change only one of the options here.

Step 2. Enable the required Boolean value:

*# setsebool -P allow\_ssh\_keysign=1*

Step 3. Verify that the value has changed:

*# getsebool -a | grep ssh  
allow\_ssh\_keysign –> on  
sftpd\_write\_ssh\_home –> off  
ssh\_sysadm\_login –> off*

At this point, the firewall and SELinux requirements are taken care of for SSH.

SSH doesn’t require too many changes to work out of the box, but as you work your way through the book, you will encounter services that require multiple changes to firewall rules and SELinux.

## SSH Security

SSH has many different options when it comes to security. First, let’s look at some host security. The SSH service can make use of the TCP Wrappers service for additional protection when you are setting it up. Suppose you want to allow connections only from the 172.168.1.0 /24 network to the RHEL01 host.

Step 1. Use TCP Wrappers to limit the hosts that can connect to the server:

*# echo “sshd: 172.168.1.” >> /etc/hosts.allow  
# echo “ALL: ALL” >> /etc/hosts.deny*

This allows all clients within the 172.168.1.0 /24 subnet to connect into the SSH server (provided they have a valid user account), and it disallows any other host outside this subnet.

Although TCP Wrappers is a good starting point for host-based security, you should also change a few of the options in the config file to really improve the security of your SSH server.

You should take into account the default port that the SSH service will use, the IP address that the server listens on, and the protocol  version.

Step 2. Change the options just discussed to improve security:

*Protocol 2  
Port 2222  
ListenAddress 172.168.1.1*

When you change these options, the default port isn’t known to everyone, and only the internal network adapter listens for connections. Be careful not to lock yourself out from the external network, though.  
Let’s switch focus for a second and look at user-based security. Although this type of security is not in the config file by default, you can also limit the users or groups that you’d like to connect to your SSH server.

The SSH service processes these options in the following order:

*DenyUsers AllowUsers DenyGroups AllowGroups*

Step 3. Add the following to your config file to allow only specific users to connect:

*AllowUsers user01,user02*

Now only the two users who have been listed are allowed to connect to the SSH server. Although security for SSH is great, make sure that you have documented your security somewhere because hardened systems tend to lead to connection issues if not planned out properly. You need to be able to review what security mechanisms you have in place when troubleshooting.

## Troubleshooting SSH

When it comes to troubleshooting SSH, the service tends to either work or not work. A key part of working with SSH is also being able to troubleshoot it.

The first place to always look with SSH is in the log file located at /var/log/secure. This file should provide you with any information as to why you can’t connect to a remote machine.

If you run into errors about the remote host refusing the connection, the problem is most likely something blocking your connection.

This could be due to the service not running, firewall rules being incorrect, or extra security measures such as TCP Wrappers. Another common error that occurs often when testing is the remote key of the server changing.

If the key changes on the remote host, you get a big warning message:

*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  
@ WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED! @  
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!  
Someone could be eavesdropping on you right now (man-in-the-middle attack)!  
It is also possible that the RSA host key has just been changed.  
If this is the case, you can open the ~/.ssh/known\_hosts file to edit or delete the line containing the key that has changed.*

Important Note: The known\_hosts file contains ONE key PER line!

The reason I’m warning you here is that not all displays are long enough to display the full length of the key, so they wrap it, giving the illusion of a multiline key. It would not be a good thing to try to delete the four- or five-line key from this file in the nano text editor because you would actually just end up erasing everything below it.

## Important Questions about SSH Service

1. What is SSH used for?

Ans. SSH is used for secure remote management of Linux systems

2. Should you allow remote root access? Why or why not?

Ans. You should never allow remote root access. Should your root account become compromised and you use the same password, someone could gain access to all your systems. You also don’t want the most powerful user of your system (with no accountability) logging in and making changes.

3. What happens if a host changes its IP address and the keys don’t match?

Ans. A large warning message appears indicating that the key doesn’t match the host you are connecting to. You have to remove the key/host pair from the known\_hosts file to proceed.

4. Which version of SSH should you use?

Ans. Version 2 is the latest and most secure version of SSH.

5. SSH can run only on TCP port 22. True or False?

Ans. False. Through its main config file, SSH can be configured to run on any port you’d like (provided that port is available).

6. TCP Wrappers can be used with SSH. True or False?

Ans. True. SSH does support TCP Wrappers.

7. What is the benefit of using public/private key authentication?

Ans. Public/private key authentication provides an additional layer of security because you need the correct key instead of just knowing someone’s password. Passwords combined with public/private keys take the security one additional step.

# Configuring Linux Web Services using Apache httpd

The most commonly used web server in the world today is Apache—and with good reason. Built with security in mind, Apache is a solid and stable web server that has been around for years. The module design allows for scalability and ease of use.

Apache can also be used to host multiple websites at a single time through the use of its virtual hosts feature. There is also an option to use the SSL protocol, making websites safe and secure. This secure base provides a platform for developers to use when writing secure code for banks, retail sites, and so on.

## Task 1: Installing Apache Webserver Packages

When you’re working with Apache, you need two packages.

-> The first is httpd, which actually installs the Apache web server.

-> The second is the mod\_ssl package, which provides the ability to create secure websites

Step 1. Install the two required packages:

# yum install –y httpd mod\_ssl

Step 2. Verify that the packages were installed correctly:

# rpm -qa | grep http  
httpd-2.2.15-5.el6.x86\_64  
httpd-tools-2.2.15-5.el6.x86\_64  
# rpm -qa | grep ssl  
mod\_ssl-2.2.15-5.el6.x86\_64  
openssl-1.0.0-4.el6.x86\_64

Step 3. With the packages installed, make sure that the service is set to start when the system boots:

# chkconfig httpd on

Step 4. Verify your changes:

# chkconfig httpd –list  
httpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off

## Task 2: Configuring the  Web Server

Now that the web server is installed, we can shift our attention to the config files and directories. During the installation, a directory (/var/www) is created with a set of subdirectories. This directory tree is the place where you store your websites.

There are also a few config files to look at:

* /etc/httpd/conf/httpd.conf                 Main config file
* /var/log/httpd                                       Log file directory for the web server
* /usr/lib64/httpd/modules                  Modules for Apache

The main config file for Apache is completely usable right out of the box, which is great if you’d like to just get up and running. You should spend some time looking through the main config file because it provides many options and good documentation in the comments. The top of the config file is grouped into three sections.

Below are the comments sectionfrom the /etc/httpd/conf/httpd.conf file:

# The configuration directives are grouped into three basic sections:  
# 1. Directives that control the operation of the Apache server process as a  
# whole (the ‘global environment’).

# 2. Directives that define the parameters of the ‘main’ or ‘default’ server,  
# which responds to requests that aren’t handled by a virtual host.  
# These directives also provide default values for the settings  
# of all virtual hosts.

# 3. Settings for virtual hosts, which allow Web requests to be sent to  
# different IP addresses or hostnames and have them handled by the  
# same Apache server process.

Below are some important directives that we might be interested to configure to setup as basic webserver:

* ServerRoot    Defines where the config files are held
* Timeout              Specifies the time before a request times out (120 seconds is the default)
* Listen                   Indicates the port number to listen on (default is 80)
* User                     Identifies the user to run the web server as
* Group                  Identifies the group to run the web server as
* LoadModule      Defines a module to load when the web server starts
* DocumentRoot Defines where the website files are located
* ServerName       Defines a server name or IP address and port number

Let’s start by defining the location of a website on the file system. By default, it is located in the /var/www/html directory, although this can be changed if you’d like.

In the main config file, you see a section denoted by the Directory option.

Section of /etc/httpd/conf/httpd.conf:

<Directory “/var/www/html”>  
Options Indexes FollowSymLinks  
AllowOverride None  
Order allow,deny  
Allow from all  
</Directory>

The options defined within this section apply specifically to the website and directory structure, as defined by the Directory option. For the preceding example, this means the site located in the /var/www/html directory. If you want to change the directory where your site is located, you need to change the  DocumentRoot option as well as the Directory option. The default option is

*DocumentRoot “/var/www/html”*

There is also a way to have multiple sites using virtual hosts, as you see later. When your config file is completely set up the way you want it, you can use the httpd service options to test your config file. Using the configtest argument, the service parses the main config file for any errors and reports back if something is found. It is always a good idea to check your config file before trying to use it because it will prevent the server from starting if it contains any errors.

Test the config file:

# service httpd configtest  
Syntax OK

There is also another cool option that the httpd service has (many services have it, but it is more useful here). Normally, for a service to use a new config file, it requires that the service is restarted. This restart process actually stops the service from running before starting it over again.

Normally, this isn’t a big deal, but when hundreds of people are hitting your site, can you afford to have even a two-second outage? To circumvent this issue, you can use the reload argument, which allows the main config file to be reread without the actual service being brought down.

This is why it is important to test your config files first.

To restart the httpd service, use the following command:

# service httpd restart  
Stopping httpd: [ OK ]  
Starting httpd: [ OK ]

To only reload the service and reread the config file, use this command instead:  
# service httpd reload  
Reloading httpd: [ OK ]

One other option to keep in mind is the graceful parameter. It restarts the web server, allowing it to read the new config file changes without disconnecting any currently connected clients. The only downfall here is that the currently active connections use the old config file until they terminate their connection and  reconnect.

You can use it as follows:  
# service httpd graceful

## Task 3 : Firewall and SELinux Configuration for apache webserver

For your web server to become fully functional, you need to make some security changes. First and foremost, the firewall needs to be opened on port 80.

Step 1. Use iptables to create the additional firewall rules:

*# iptables -I INPUT 5 -p tcp -m tcp –dport 80 -j ACCEPT*

Step 2. Save the firewall rules you have just created:

*# service iptables save  
Saving firewall rules to /etc/sysconfig/iptables: [ OK ]*

Step 3. Then restart the iptables service:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

Additionally, you need to look at SELinux protection for the web server.

# Squid Web Proxy configuration in Linux

A proxy server is a device that usually sits between a client and the destination the user is trying to reach. It can provide security, anonymity, and even protection for the client behind the proxy. To help in this process is Squid, which is a web proxy server for Red Hat. It sits between the client and web server that the user is trying to connect to.

Many times these devices are used when you want to control access to the Internet (think web filtering). As a web proxy, it can also cache data that users request from the Web and make it locally available, reducing the load on your external devices such as gateways and firewalls.

Here, we look at how to set up a web proxy, define access control lists, and troubleshoot it.

## Task 1: Installing Squid

Much as you did with the web server, you need to start by installing the package(s) needed for Squid. There is only one package required to install the Squid proxy server.  
Step 1. Install the package with the following command:

*# yum install -y squid*

Step 2. After it’s installed, verify:

*# rpm -qa | grep squid  
squid-3.1.4-1.el6.x86\_64*

Next, you should turn on Squid at boot time. You use the chkconfig command to do this.

Step 3. Enable Squid to start at boot:

*# chkconfig squid on*

Step 4. Verify the service will start at boot:

*# chkconfig squid —list  
squid 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

Now that you know the package is installed and will start at boot, you can turn your attention to configuration.

## Task2: Configuring Web Proxy Server

When setting up your proxy server, you need to know the following items:

* /etc/sysconfig/squid           Startup options for the config file
* /etc/squid/squid.conf         Main config file for the service
* /var/spool/squid                 Cache location on the proxy server
* /var/log/squid                     Log files for the proxy server

As with most services you configure, the first item on the agenda is the main config file.

I want to warn you first that although this config file has huge amounts of documentation and numerous examples, it contains over 4,000+ lines, so make sure you put aside some time if you plan to take on reading and going through this whole config file! As with Apache, configuring a web proxy server can be a daunting and sometimes lengthy process until you have it set up correctly.

Let’s look at some of the main configuration options:

* http\_port Specifies the port to listen on
* visible\_hostname Identifies the name of the Squid server
* hierarchy\_stoplist Provides a list of words that tell the Squid server to handle the request
* access\_log Keeps track of the web pages that are downloaded
* acl Defines an access control list
* http\_access Defines which system or networks have access

You can use the default port to run the Squid proxy, which will make testing a little easier. However, you can set the visible\_hostname option to the name of your server:

*# nano /etc/squid/squid.conf  
visible\_hostname = RHEL01*

You should also define the URL syntax for which the Squid server should not handle.

An example would be form data that you want the server to submit directly and not cache your private data:

*hierarchy\_stoplist cgi-bin ?  
Acl QUERY urlpath\_regex cgi-bin \?  
cache deny QUERY*

## Task 3: Firewall and SElinux Configuration for Squid Server

The firewall and SELinux requirements for Squid are actually quite simple. Squid uses port 3128 by default for its communication, so you should open this port on the firewall. Both the TCP and UDP protocols are used.

Step 1. Use the iptables command to create your firewall rules:

*# iptables -I INPUT 5 -p tcp -m tcp —dport 3128 -j ACCEPT  
# iptables -I INPUT 5 -p udp -m udp —dport 3128 -j ACCEPT*

Step 2. Save the rules you just created:

*# service iptables save  
Saving firewall rules to /etc/sysconfig/iptables: [ OK ]*

Step 3. Restart the firewall service for the changes to take effect:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

By default, you do not have to change SELinux for your Squid setup. You should know what the available options are, though:

*squid\_use\_tproxy    Allows Squid to run as a transparent proxy (TPROXY)  
squid\_connect\_any   Allows Squid to connect to all ports, not just HTTP, FTP, and Gopher ports*

If you want to enable either of these features, just make sure to adjust the SELinux Boolean value appropriately.

As you can see, the firewall rules and SELinux requirements are really light for Squid. Before starting the service, though, we need to cover Squid security a little more in depth.

# NFS Configuration in Linux

The Network File Systems (NFS) protocol works great when it comes to Linux systems because it allows for client flexibility, centralized management of files, and some other great features.

To get NFS working properly, you need to set up the NFS server first and then set up the client to test access to the server. As with any other service, you need to install a few packages before doing anything else.

The actual NFS service is the same for both servers and clients, with the difference being found in the services running and the config files.

There are four different versions of NFS; version 4 is the most current. Although you can disable what versions the server listens for, the client actually determines which version it will use when connecting to the server (unless, of course, the server is offering the version the client is asking for).

## Task 1 : Installation of NFS Services

Step 1. To begin the NFS server setup, install the required packages:

*# yum install -y nfs-utils nfs4-acl-tools*

Step 2. Verify the package installation:

*# rpm -qa | grep nfs  
nfs4-acl-tools-0.3.3-5.el6.x86\_64  
nfs-utils-1.2.2-7.el6.x86\_64  
nfs-utils-lib-1.1.5-1.el6.x86\_64*

Step 3. The NFS server uses three different services to function properly. You need to enable them all at boot for the NFS server to function the way it should:

*# chkconfig nfs on  
# chkconfig nfslock on  
# chkconfig rpcbind on*

Step 4. Verify that all three services are set to start on system boot:

*# chkconfig –list nfs  
nfs 0:off 1:off 2:on 3:on 4:on 5:on 6:off  
# chkconfig –list nfslock  
nfslock 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

*# chkconfig –list rpcbind  
netfs 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

You should also verify that the NFS service is currently stopped because you need to make some configuration changes before you can start it.

Step 5. Verify that the service is off:

*# service nfs status  
rpc.mountd is stopped  
nfsd is stopped  
rpc.rquotad is stopped*

## Task 2:  Configuring NFS Services

The nfs and rpcbind services both control a number of daemons on the system when they are started. Let’s look at the different daemons these two services are composed of:

* rpcbind        Forwards incoming requests to the appropriate subservice
* rpc.idmapd Maps the UID and GID to users and groups
* rpc.lockd     Manages file locks and releases in case of client disconnect
* rpc.nfsd       Responds to client requests for file access
* rpc.rquotad Provides statistics on disk quotas to clients
* rpc.statd      Works with rpc.lockd to provide recovery services

Let’s also look at the config files that you will be dealing with:

* /etc/sysconfig/nfs Contains the main config files for the NFS service
* /etc/exports Contains a list

Here are some additional files that you will use when working with NFS:

* /var/lib/nfs/etab Contains a list of currently exported resources
* /var/lib/nfs/rmtab Contains a list of remotely mounted resources

For the first configuration step, you need to make a few changes to the main config file.

Step 1. Open the main config file for editing:

*# nano /etc/sysconfig/nfs  
Step 2. Uncomment the following lines:  
MOUNTD\_NFS\_V1=”no”  
MOUNTD\_NFS\_V2=”no”  
MOUNTD\_NFS\_V3=”no”  
RPCNFSDARGS=”-N 2 -N 3”*

The first three lines disable the mountd daemon from accepting anything below version 4. The last line disables the NFS service from even advertising anything but version 4 as well.

Step 3. Save the file and exit.

Next, let’s work with the /etc/exports file because this defines what resources will be available to your clients. If the file doesn’t exist already, you can create it. The syntax of the /etc/exports file is

<mountpoint> <host><permissions/options>

Mount Options:

* rw Sets read/write permissions
* ro Sets read-only permissions
* insecure Allows the use of ports over 1024
* sync Specifies that all changes must be written to disk before a command completes
* no\_wdelay Forces the writing of changes immediately (useful for logs  if something crashes)
* root\_squash Prevents root users

As an example, you can use the following two locations to export to the clients:  
Step 4. Set up your exports in the /etc/exports files to be available to any client on the network:

*# nano /etc/exports  
/home \*(ro,sync)  
/opt/company\_data \*(rw,sync)*

Here, you define two resources that you will make available. The first line defines the /home directory to be exported and allows read-only access to all clients. The second line provides the /opt/company\_data directory to all of your clients with read and write permissions. After you finish defining all the resources you want to export, save and close the file.

Step 5. Start the two NFS services (rpcbind should be running already by default):

*# service nfslock start  
Starting NFS statd: [ OK ]*

*# service nfs start  
Starting NFS services: [ OK ]  
Starting NFS quotas: [ OK ]  
Starting NFS daemon: [ OK ]  
Starting NFS mountd: [ OK ]*

Step 6. Verify that the services have started successfully:

*# service rpcbind status  
rpcbind (pid 25068) is running…  
# service nfslock status  
rpc.statd (pid 17726) is running…  
# service nfs status  
rpc.svcgssd is stopped  
rpc.mountd (pid 17780) is running…  
nfsd (pid 17777 17776 17775 17774 17773 17772 17771 17770) is  
running…  
rpc.rquotad (pid 17764) is running…*

## Task 3: Manually export of Shares

If you already started the services before creating an /etc/exports file, you can also use the exportfs command to manually export any new resources added to the /etc/exports file.  
Syntax: exportfs [options]  
Options:

*-a Exports or unexports all directories  
-r Reexports all directories  
-u Unexports one or more directories  
-v Provides verbose output*

Step 1. Here is what a manual export of resources would look like:

*# exportfs -avr  
exporting \*:/opt/company\_data  
exporting \*:/home*

Step 2. Alternatively, you can also get the same effect by restarting only the NFS service, which in turn restarts all daemons:

*# service nfs restart  
Shutting down NFS mountd: [ OK ]  
Shutting down NFS daemon: [ OK ]  
Shutting down NFS quotas: [ OK ]  
Shutting down NFS services: [ OK ]  
Starting NFS services: [ OK ]  
Starting NFS quotas: [ OK ]  
Starting NFS daemon: [ OK ]  
Starting NFS mountd: [ OK ]*

Now that all the resources have been exported properly and the NFS service has been started, you can use the rpcinfo command to verify that all the parts of the NFS service are running properly.

Syntax: rpcinfo -p [host]

You can view both local and remote connection information with the rpcinfo command. Because you are looking for information about the local server, you don’t have to specify a host when calling the command. View the current running nfs daemons:

*# rpcinfo -p  
program vers proto port service  
100000 4 tcp 111 portmapper  
100000 3 tcp 111 portmapper  
100000 2 tcp 111 portmapper  
100000 4 udp 111 portmapper  
100000 3 udp 111 portmapper  
100000 2 udp 111 portmapper  
100024 1 udp 41853 status  
100024 1 tcp 40535 status  
100011 1 udp 875 rquotad  
100011 2 udp 875 rquotad  
100011 1 tcp 875 rquotad  
100011 2 tcp 875 rquotad  
100003 4 tcp 2049 nfs  
100003 4 udp 2049 nfs  
100021 1 udp 32769 nlockmgr  
100021 3 udp 32769 nlockmgr  
100021 4 udp 32769 nlockmgr  
100021 1 tcp 32803 nlockmgr  
100021 3 tcp 32803 nlockmgr  
100021 4 tcp 32803 nlockmgr*

Where you see the nfs daemon running, notice that only version 4 is listed. The reason is that you disabled all other versions in the config file. Although some of the other daemons listed use other versions as well, you should verify that at least the nfs daemon shows version 4.

## Task 4: Firewall  Configuration for NFS server

NFS is one of the many Red Hat services that can take advantage of TCP Wrappers as well as firewall rules for security. We don’t use TCP Wrappers here, but should you run into trouble on the exam with the NFS service not working, don’t forget to check to see whether anything is being filtered by TCP Wrappers. Because you are using NFS version 4 here, you need to create only a single firewall rule.  
Step 1. Use iptables to create the additional firewall rules:

*# iptables -I INPUT 5 -p tcp -m tcp –dport 2049 -j ACCEPT*

Step 2. Save the firewall rules you just created:

*# service iptables save  
Saving firewall rules to /etc/sysconfig/iptables: [ OK ]*

Step 3. Restart the iptables service:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

If you are using NFS4 , you can configure the firewall rules as per below steps:

you need to statically define ports in the /etc/sysconfig/nfs file for each of the four daemons required for the NFS service to run. You also need to add these ports to the /etc/services file and create a firewall rule for each one.

Step 1. Define static ports in /etc/sysconfig/nfs for each of the four required daemons NFS uses.  
Step 2. Create a firewall rule for the rpcbind server (TCP and UDP port 111).  
Step 3. Create a firewall rule for the MOUNTD\_PORT you specified (TCP and UDP).  
Step 4. Create a firewall rule for the STATD\_PORT you specified (TCP and UDP).  
Step 5. Create a firewall rule for the LOCKD\_TCPPORT you specified (TCP).  
Step 6. Create a firewall rule for the LOCKD\_UDPPORT you specified (UDP).

## Task 5: SELinux Configuration for NFS Server:

Step 1. Query for the Boolean value you need to change:

*# getsebool -a | grep nfs  
allow\_ftpd\_use\_nfs –> off  
allow\_nfsd\_anon\_write –> off  
git\_system\_use\_nfs –> off  
httpd\_use\_nfs –> off  
nfs\_export\_all\_ro –> off  
nfs\_export\_all\_rw –> off  
qemu\_use\_nfs –> on  
samba\_share\_nfs –> off  
use\_nfs\_home\_dirs –> off  
virt\_use\_nfs –> off  
xen\_use\_nfs –> off*

Step 2. Disable SELinux protection for only the options that you need:

*# setsebool -P nfs\_export\_all\_ro=1 nfs\_export\_all\_rw=1*

Step 3. Verify that the Boolean has changed:

*# getsebool -a | grep nfs  
allow\_ftpd\_use\_nfs –> off  
allow\_nfsd\_anon\_write –> off  
git\_system\_use\_nfs –> off  
httpd\_use\_nfs –> off  
nfs\_export\_all\_ro –> on  
nfs\_export\_all\_rw –> on  
qemu\_use\_nfs –> on  
samba\_share\_nfs –> off  
use\_nfs\_home\_dirs –> off  
virt\_use\_nfs –> off  
xen\_use\_nfs –> off*

# Troubleshooting NFS Configuration in Linux

There are three management commands that help with troubleshooting NFS from both the server and client sides:

* mountstats          Shows information about mounted NFS shares
* nfsstat                  Shows statistics of exported resources
* nfsiostat              Shows statistics of NFS mounted shares

First, let’s look at which resources are exported. For this, you can check the /var/lib/nfs/etab file.

### Step 1. View exported resources (whether or not they are mounted):

*# cat /var/lib/nfs/etab*

### Step 2. From the client side, you can use the nfsstat command for similar results.

*Syntax: nfsstat [options]  
Options:*

* *-m   Shows statistics on mounted NFS file systems*
* *-n    Shows NFS statistics*
* *-v    Provides verbose output*

### Step 3. To display the exported resources from the client (mounted only), use the following:

*# nfsstat -m*

After clients are connected to the server, you can again use the nfsstat command to get some statistical information.

### Step 4. View NFS version 4 stats from connected clients:

*# nfsstat*

# Samba Server configuration in Linux

## Task 1: Installing Samba Services

Samba, which uses the CIFS/SMB protocol, is commonly brought up when you want Linux and Windows machines to be able to share files together. Aside from the file sharing uses, Samba also has some built-in functionality to run as a member server on a Windows domain, print server, or file server.

Let’s get started with the setup.  
Step 1. Install the required packages for Samba:

*# yum install –y samba samba-common samba-client*

Step 2. Verify the package installation:

*# rpm -qa | grep samba  
samba-client-3.5.4-68.el6.x86\_64  
samba-3.5.4-68.el6.x86\_64  
samba-winbind-clients-3.5.4-68.el6.x86\_64  
samba-common-3.5.4-68.el6.x86\_64*

Step 3. Enable the service to start during boot:

*# chkconfig smb on*

Step 4. Verify that the service is set to start on boot:

*# chkconfig smb –list  
smb 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

## Task 2 : Configuring Samba Services

If you have never worked with Samba before, the number of options can seem overwhelming.  
First, let’s look at the two services responsible for running Samba:

* smbd Samba server daemon
* nmbd NetBIOS service daemon

There are also a handful of config files:

* /etc/samba/smb.conf Contains the main config file
* /etc/samba/smbusers Maps Samba and Red Hat users
* /etc/samba/smbpasswd Contains Samba user passwords

I’m sure you could have guessed by now that no service this complex comes without a group of management commands as well:

* mount.cifs Mounts a Samba resource without root privileges
* smbclient Connects to a Samba resource
* smbpasswd Configures Samba users and passwords
* smbstatus Displays the status of Samba connections
* testparm Tests the syntax of the main config file for issues
* umount.cifs Unmounts a Samba resource without root privileges

Now that you are completely overwhelmed with config files and commands, let’s take a step back and see what these things are actually used for. You need to edit the main config file to set up the Samba server and directories that you’d like to make into Samba shares.

Here is a sample /etc/samba/smb.conf config file you can use (just read through it for now):

***# cat /etc/samba/smb.conf***

*### Global Data Section ###  
[global]  
### Define our workgroup and hostname information ###  
workgroup = INET  
server string = My Samba Server  
netbios name = RHEL01*

*### Define the log file and its size ###  
log file = /var/log/samba/%m.log  
max log size = 50  
### Use a local password file (/etc/samba/smbpasswd) ###  
security = user  
passdb backend = tdbsam  
### Define printer settings ###  
load printers = yes  
printcap name = /etc/printcap  
cups options = raw  
### Samba Share for Company Data ###  
[company\_data]  
### Define a comment for the share ###  
comment = Directory for all employees within the company  
### Allow users to access the share and define its location ###  
browseable = yes  
path = /opt/company\_data  
### Make the share writable and define access for valid users ###  
valid users = user01  
writable = yes  
### Share for Samba printers ###  
[printers]  
### Define a comment for the share ###  
comment = All Printers*

*Allow users to access the share and define its location ###*

*browseable = no  
path = /var/spool/samba  
### Set permissions and user access ###  
guest ok = no  
writable = no  
printable = yes*

To set up the file, do the following:

Step 1. Make a backup of the main config file so you can review the comments in it later:

*# cp /etc/samba/smb.conf /etc/samba/smb.bk*

Step 2. Copy the sample file provided here into a new main config file:

*# nano /etc/samba/smb.conf*

Step 3. Save the file and exit. Now you need to check that the config file has no syntax errors by using the testparm command:

*Syntax: testparm [options] <config file> [hostname] [host IP]  
Options:*

*-s Suppresses the prompt  
-v Provides verbose output (shows the default options)*

Check the syntax of the config file:

*# testparm  
Load smb config files from /etc/samba/smb.conf  
Processing section “[company\_data]”  
Processing section “[printers]”  
Loaded services file OK.  
Server role: ROLE\_STANDALONE  
Press enter to see a dump of your service definitions  
[global]  
workgroup = INET  
netbios name = RHEL01  
server string = My Samba Server  
log file = /var/log/samba/%m.log  
max log size = 50  
printcap name = /etc/printcap  
cups options = raw  
[company\_data]  
comment = Directory for all employees within the company  
path = /opt/company\_data  
valid users = user01  
read only = No*

*[printers]  
comment = All Printers  
path = /var/spool/samba  
printable = Yes  
browseable = No*

There are no errors in the output shown here, but you can see the global options displayed, including the different shares that are accessible to users. Before you can start connecting clients, however, you also need to create Samba users because they are separate from system users.

You can use the smbpasswd command to create a new Samba user.

*Syntax: smbpasswd [options] [user]  
Options:*

*-a Adds a user*

*-d Disables a user*

*-e Enables a user*

*-x Deletes a user*

Note: Because you have specified to use the tdbsm back end, any user that you want to create for Samba must have an account locally on the Samba server.

Step 1. Create your first Samba user:

*# smbpasswd -a user01  
New SMB password:  
Retype new SMB password:  
Added user user01.*

Step 2. Verify that the user was created successfully by using the pdbedit command:

*# pdbedit -w -L  
user01:501:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:17601CAE62CBC5D649CF7D1  
951C42806:[  
U ]:LCT-4D498DE8:*

At this point, everything should be in place for your Samba server. You just need to make sure that the directories you specified to be a Samba share exists.

Step 1. If you haven’t done so already, start the Samba service:

*# service smb start  
Starting SMB services: [ OK ]*

Step 2. Verify that the service is running:

# service smb status  
smbd (pid 3145) is running…  
SELinux and Firewall Configuration

## Task 3: Configuring Firewall ports to allow smb traffic in Linux

If you are accustomed to Windows systems, you should already know what ports you need to open on the firewall.

Step 1. Use the iptables command to create your firewall rules:

*# iptables -I INPUT 5 -p tcp -m tcp –dport 137 -j ACCEPT  
# iptables -I INPUT 5 -p udp -m udp –dport 138 -j ACCEPT  
# iptables -I INPUT 5 -p udp -m udp –dport 139 -j ACCEPT  
# iptables -I INPUT 5 -p tcp -m tcp –dport 445 -j ACCEPT*

Step 2. Save the rules you just created:

*# service iptables save  
Saving firewall rules to /etc/sysconfig/iptables: [ OK ]*

Step 3. Restart the firewall service for the changes to take effect:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

These four ports are very common to Windows administrators, as they are heavily used in Windows environments.

Samba Client Configuration in Linux

### Step 1. Install the client packages:

*# yum install -y samba-client samba-common*

### \

### Step 2. Verify that the install was successful:

*# rpm -qa | grep samba  
samba-client-3.5.4-68.el6.x86\_64  
samba-winbind-clients-3.5.4-68.el6.x86\_64  
samba-common-3.5.4-68.el6.x86\_64*

### Step 3. Create a local directory where you will mount your Samba share.

For this example, make it the same as the Samba share directory to keep things simple:

*# mkdir /opt/company\_data*

Using the smbclient command, you can now mount the Samba share.

*Syntax: smbclient [options]  
Options:  
-L Lists Samba shares  
-U Defines the user to connect with  
-P Defines the password to connect with  
-A Gets credentials from a file*

#### Step 4. List the Samba shares on the RHEL01 Samba server:

*# smbclient -L 172.168.1.1 -U user01%<you password here>  
Domain=[RHEL01] OS=[Unix] Server=[Samba 3.0.33-3.29.el5\_5.1]  
Sharename Type Comment  
——— —- ——-  
company\_data Disk Directory for all employees  
within the  
company  
IPC$ IPC IPC Service (My Samba Server)  
Domain=[INET] OS=[Unix] Server=[Samba 3.5.4-68.el6]  
Server Comment  
——— ——-  
Workgroup Master  
——— ——-  
INET RHEL01*

You can see here that the share named company\_data is available for access. Recall from Chapter 4 that the mount command allows you to connect additional resources to the file system hierarchy. You can use the  
mount command here with the cifs option to mount the Samba share.

### Step 5. Mount the remote Samba share:

*# mount.cifs //172.168.1.1/company\_data /opt/test -o  
username=user01,password=<password>*

You can verify that the mount worked successfully by using the smbstatus command.

*Syntax: smbstatus [options]  
Options:*

*-p Shows processes only  
-v Provides verbose output  
-S Shows shares only  
-L Shows locks only*

### Step 6. Verify the mount was successful:

*# smbstatus  
Samba version 3.5.4-68.el6  
PID Username Group Machine  
——————————————————————-  
3378 user01 user01 172.168.1.20 (172.168.1.20)  
Service pid machine Connected at  
——————————————————-  
company\_data 3378 172.168.1.20 Tue Oct 12 16:06:47 2010*

*No locked files*

So far, you are doing well with the client being able to connect to the Samba server. You need to know that any share that is mounted without being added to the /etc/fstab file will be unmounted after the system is rebooted.

To remedy this situation, you can create an entry in the /etc/fstab file.

### Automatically mounting Samba shares during Linux Boot

Step 1. Create an entry in the /etc/fstab file:

*//172.168.1.1/company\_data /opt/test cifs user=user01,pass=<password> 0 0*

Can anyone find a problem with this entry? The problem here is that the username and password are exposed in clear-text for anyone to see.

Another way that you can create an entry in the /etc/fstab file without exposing credentials is to put the credentials inside a file to be referenced from the /etc/fstab file.

Step 2. Add the credentials that you’d like to use to a file:

*# echo “username=user01” > /etc/samba/smbcred  
# echo “password=password” >> /etc/samba/smbcred*

Step 3. Update the entry in the /etc/fstab file to reflect the changes to how the credentials are read:

*//172.168.1.1/company\_data /opt/test cifs  
credentials=/etc/samba/smbcred 0 0*

You can use the umount.cifs command to unmount the Samba share and reboot the system to make sure that the share mounts correctly when the system reboots

\

# SELinux Configuration for Samba

Due to the complexities of Samba and its integration with Windows, there are quite a few different Boolean values that you need to change.

### Step 1. Query for available Boolean options:

*# getsebool -a | egrep ‘(samba)|(smb)|(nmb)|(win)’  
allow\_httpd\_mod\_auth\_ntlm\_winbind –> off  
allow\_smbd\_anon\_write –> off  
samba\_create\_home\_dirs –> off  
samba\_domain\_controller –> off  
samba\_enable\_home\_dirs –> off  
samba\_export\_all\_ro –> off  
samba\_export\_all\_rw –> off  
samba\_run\_unconfined –> off  
samba\_share\_fusefs –> off  
samba\_share\_nfs –> off  
use\_samba\_home\_dirs –> off  
virt\_use\_samba –> off  
wine\_mmap\_zero\_ignore –> off*

### Step 2. You need to change only a few settings for the shares to work properly:

*# setsebool -P samba\_export\_all\_ro=1 samba\_export\_all\_rw=1*

### Step 3. Verify that the changes have been made:

*# getsebool -a | egrep ‘(samba)|(smb)|(nmb)|(win)’  
allow\_httpd\_mod\_auth\_ntlm\_winbind –> off*

*allow\_smbd\_anon\_write –> off  
samba\_create\_home\_dirs –> off  
samba\_domain\_controller –> off  
samba\_enable\_home\_dirs –> off  
samba\_export\_all\_ro –> on  
samba\_export\_all\_rw –> on  
samba\_run\_unconfined –> off  
samba\_share\_fusefs –> off  
samba\_share\_nfs –> off  
use\_samba\_home\_dirs –> off  
virt\_use\_samba –> off  
wine\_mmap\_zero\_ignore –> off*

Don’t forget that you can always look up the available Boolean options in the /selinux/booleans directory if you forget which options you need.

Another huge benefit with Samba is that if you read the comments in the main config file, it tells you which Boolean values need to be enabled for the different services that Samba can provide.

When creating directories that you’d like to make into a Samba share, you can mark them as a Samba share with the correct SELinux context:

*# chcon -Rt samba\_share\_t /opt/company\_data*

Now the directory is accessible to the Samba service.

# FTP configuration in Linux

There are many different ways you can share files with users on your network. This capability is important because you don’t always want your users storing things locally on their desktop or laptop.

Should something happen to your end users’ systems, they would lose all their work…not to mention that it would cause a backup strategy nightmare. An easier solution for management and security would be to store all your files in a centralized location.

To transfer a file using the FTP protocol, a user must log in to an FTP server, which can be done with credentials or anonymously. When the user is connected, she can traverse the directory structure for any directory or file for which she has permissions.

If the protocol is not configured properly, this can leave your entire system open to attack and make it hard to track if the attack is done through an anonymous connection!

The second big issue with the FTP protocol is that when the user logs in with a username and password, they are passed over the network in clear-text, meaning that anyone listening can see them.

So, why use the FTP protocol at all? It’s easy to set up, and when used correctly, it’s highly effective for delivering files to end users.

Almost all major computer makers (HP, Dell, Apple) offer drivers for their systems over FTP, which allows for simple download by end users and organized structure on the back end for the drivers themselves.

### Task 1  : Installing FTP Services

In RHEL6 and later versions,  we use the vsftpd package, which stands for Very Secure FTP Daemon.

This particular FTP server offers additional features that make it a more secure choice if you have to use FTP. As with all services that you’d like to offer to your network users, you need to make sure that the appropriate packages are installed. Here’s how.  
Step 1. Grab the required package:

*# yum install -y vsftpd*

Step 2. When the installation is complete, verify it was installed successfully:

*# rpm -qa | grep vsftpd  
vsftpd-2.2.2-6.el6.x86\_64*

Step 3. Ensure that the service will start on system boot:

*# chkconfig vsftpd on*

Step 4. Verify the service starts on boot:

*# chkconfig vsftpd –list  
vsftpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

### Task 2: Configuring VSFTP in Linux

Step 1. Look at which options are available in the config file:

*# grep -v ^# vsftpd.conf  
anonymous\_enable=YES  
local\_enable=YES  
write\_enable=YES  
local\_umask=022  
dirmessage\_enable=YES  
xferlog\_enable=YES  
connect\_from\_port\_20=YES  
xferlog\_std\_format=YES  
listen=YES  
pam\_service\_name=vsftpd  
userlist\_enable=YES  
tcp\_wrappers=YES*

Step 2. Now let’s go over what each of these options can be used for:

*anonymous\_enable=YES*

The default; sets security, although it should be changed to NO for better host-based security

*local\_enable=YES*

Allows local users to log in

*write\_enable=YES*

Enables users to write to directories

*local\_umask=022*

Sets the umask for all uploaded files

*dirmessage\_enable=YES*

Displays directory messages

*xferlog\_enable=YES*

Logs all transfer activity to /var/log/xferlog

*connect\_from\_port\_20=YES*

Forces port transfers to originate from port 20

*xferlog\_std\_format=YES*

Logs everything in standard transfer format

*listen=YES*

Allows the server to listen for connections

*pam\_service\_name=vsftpd*

Specifies the name used for the PAM service

*userlist\_enable=YES*

Enables the service to consult user\_list

*tcp\_wrappers=YES*

Allows incoming requests based on the TCP Wrappers configuration

userlist\_deny=YES

Enables users listed in user\_list to log in via FTP.  
These default settings for the vsftpd service allow you to get off the ground running with the FTP service.

At this point, any one of your system’s users is able to log in to the vsftpd service, but because the firewall is enabled by default, the connection will be denied.

Before you open the connection to your users, take some time to become familiar with the different options you can configure on your FTP server. The config file is heavily documented as to what each option does.

# Firewall and SELinux Configuration for FTP Server

Before you can begin using the FTP service, you need to make some firewall adjustments and SELinux changes. Let’s start with the firewall rules. FTP uses both TCP ports 20 and 21, which you can open on the firewall.

### Step 1. Use the iptables command to create your firewall rules:

*# iptables -I INPUT 5 -p tcp -m tcp —dport 20 -j ACCEPT  
# iptables -I INPUT 5 -p tcp -m tcp —dport 21 -j ACCEPT*

### Step 2. Save the rules you just created:

*# service iptables save*

Saving firewall rules to /etc/sysconfig/iptables: [ OK ]

### Step 3. Restart the firewall service for the changes to take effect:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

Now that the firewall rules are taken care of, let’s move on to SELinux.

Real World Information :   It is possible to get your FTP server to interact over a single firewall port. This is possible because of the different ways that FTP functions (active versus passive), but because this isn’t a requirement for the exams, we don’t discuss it here. Depending on what features you are trying to configure, you need to adjust SELinux accordingly.

For now, let’s enable the system users to have read/write access to the system.

## SELinux Configuration

### Step 1. Query for the Boolean value you need to change:

*# getsebool -a | grep ftpd\_full  
allow\_ftpd\_full\_access –> off*

### Step 2. Disable the SELinux protection:

*# setsebool -P allow\_ftpd\_full\_access=1*

### Step 3. Verify that the Boolean has changed:

*# getsebool -a | grep ftpd\_full  
allow\_ftpd\_full\_access –> off*

If you need to enable additional features for your FTP server, make sure to disable SELinux protection for that feature.

One of the features you need to be able to set up is anonymous access to your FTP server. For this feature to function properly, you need to make sure that you adjust the allow\_ftpd\_anon\_write Boolean. This allows you to have anonymous users upload files. This capability can be dangerous because there is no way to track which user is uploading or writing files. It is recommended to leave this option disabled unless you know what you’re doing.

## Configuring FTP Security using Host-based Access lists

When dealing with security for FTP, you can run into a little trouble if you don’t plan things out ahead of time.

### The FTP protocol supports two different types of file transfers.

* The first is known as active mode, which uses port 20 to connect back to the client.
* The second is known as passive mode, which uses a custom-defined range of ports above 1024.

Because there are two different modes for FTP, you need to decide which mode you want to use so that you can configure the correct security settings and open the correct ports on the firewall. Back in the  configuration section, the option connect\_from\_port\_20 is set to YES by default. This means that, by default, active mode is used for the vsftpd service.

Let’s look at some other options that can be used for basic security. You can disable the anonymous\_enable option to prevent nonauthorized users from accessing the FTP server. The local\_enable option, which is enabled by default, allows local system users to log in to the FTP server. Keeping this option is usually safe option so that you don’t need to maintain a second list of users that you want to be able to log in to the FTP server.

There is one other security step you should take for the FTP server.

*The userlist\_enable option, which is set to YES by default, allows the vsftpd service to consult the /etc/vsftpd/user\_list file.*

When this option is used in conjunction with the userlist\_deny option, all users in this file are denied access to the server and not even prompted for a password. This prevents them from submitting clear-text passwords over the network.

If you want to change this setting, however, you could set the userlist\_deny option to NO. Then all users except for those listed in /etc/vsftpd/user\_list are denied access.

This setting is useful if you want only select individuals to be able to log in and not all your system users.  
Here is what the file contains by default:

### user\_list file

***# cat user\_list*** *# vsftpd userlist  
# If userlist\_deny=NO, only allow users in this file  
# If userlist\_deny=YES (default), never allow users in this file, and  
# do not even prompt for a password.  
# Note that the default vsftpd pam config also checks /etc/vsftpd/ftpusers  
# for users that are denied.*

*root  
bin  
daemon  
adm  
lp  
sync  
shutdown  
halt  
mail  
news  
uucp  
operator  
games  
nobody*

As you can see, this file can get very confusing with options quickly, which is why you should plan out ahead of time what you want your FTP server policy to be.

Real World Information :  The file called /etc/vsftpd/ftpusers denies access to log in to the FTP server no matter what. A few system users are populated in this file when the vsftpd service is installed.

If you want to ban a user, you can use this file. The difference is that users in this file are not allowed to log in but still receive a login and password prompt allowing them to submit their credentials over the network in clear-text.

How to Setup Chroot SFTP in Linux (Allow Only SFTP, not SSH)

### . Create a New Group

Create a group called sftpusers. Only users who belong to this group will be automatically restricted to the SFTP chroot environment on this system.

# groupadd sftpusers

### 2. Create Users (or Modify Existing User)

Let us say you want to create an user guestuser who should be allowed only to perform SFTP in a chroot environment, and should not be allowed to perform SSH.

The following command creates guestuser, assigns this user to sftpusers group, make /incoming as the home directory, set /sbin/nologin as shell (which will not allow the user to ssh and get shell access).

# useradd -g sftpusers -d /incoming -s /sbin/nologin guestuser

# passwd guestuser

Verify that the user got created properly.

# grep guestuser /etc/passwd

guestuser:x:500:500::/incoming:/sbin/nologin

If you want to modify an existing user and make him an sftp user only and put him in the chroot sftp jail, do the following:

# usermod -g sftpusers -d /incoming -s /sbin/nologin john

On a related note, if you have to transfer files from windows to Linux, use any one of the sftp client mentioned in this [top 7 sftp client](http://www.thegeekstuff.com/2011/06/windows-sftp-scp-clients/) list.

### 3. Setup sftp-server Subsystem in sshd\_config

You should instruct sshd to use the internal-sftp for sftp (instead of the default sftp-server).

Modify the the /etc/ssh/sshd\_config file and comment out the following line:

#Subsystem sftp /usr/libexec/openssh/sftp-server

Next, add the following line to the /etc/ssh/sshd\_config file

Subsystem sftp internal-sftp

# grep sftp /etc/ssh/sshd\_config

#Subsystem sftp /usr/libexec/openssh/sftp-server

Subsystem sftp internal-sftp

### 4. Specify Chroot Directory for a Group

You want to put only certain users (i.e users who belongs to sftpusers group) in the chroot jail environment. Add the following lines at the end of /etc/ssh/sshd\_config

# tail /etc/ssh/sshd\_config

Match Group sftpusers

ChrootDirectory /sftp/%u

ForceCommand internal-sftp

In the above:

* Match Group sftpusers – This indicates that the following lines will be matched only for users who belong to group sftpusers
* ChrootDirectory /sftp/%u – This is the path that will be used for chroot after the user is authenticated. %u indicates the user. So, for john, this will be /sftp/john.
* ForceCommand internal-sftp – This forces the execution of the internal-sftp and ignores any command that are mentioned in the ~/.ssh/rc file.

### 5. Create sftp Home Directory

Since we’ve specified /sftp as ChrootDirectory above, create this directory (which iw equivalent of your typical /home directory).

# mkdir /sftp

Now, under /sftp, create the individual directories for the users who are part of the sftpusers group. i.e the users who will be allowed only to perform sftp and will be in chroot environment.

# mkdir /sftp/guestuser

So, /sftp/guestuser is equivalent to / for the guestuser. When guestuser sftp to the system, and performs “cd /”, they’ll be seeing only the content of the directories under “/sftp/guestuser” (and not the real / of the system). This is the power of the chroot.

So, under this directory /sftp/guestuser, create any subdirectory that you like user to see. For example, create a incoming directory where users can sftp their files.

# mkdir /sftp/guestuser/incoming

### 6. Setup Appropriate Permission

For chroot to work properly, you need to make sure appropriate permissions are setup properly on the directory you just created above.

Set the owenership to the user, and group to the sftpusers group as shown below.

# chown guestuser:sftpusers /sftp/guestuser/incoming

The permission will look like the following for the incoming directory.

# ls -ld /sftp/guestuser/incoming

drwxr-xr-x 2 guestuser sftpusers 4096 Dec 28 23:49 /sftp/guestuser/incoming

The permission will look like the following for the /sftp/guestuser directory

# ls -ld /sftp/guestuser

drwxr-xr-x 3 root root 4096 Dec 28 23:49 /sftp/guestuser

# ls -ld /sftp

drwxr-xr-x 3 root root 4096 Dec 28 23:49 /sftp

### 7. Restart sshd and Test Chroot SFTP

Restart sshd:

# service sshd restart

Test chroot sftp environment. As you see below, when gusetuser does sftp, and does “cd /”, they’ll only see incoming directory.

# sftp guestuser@thegeekstuff.com

guestuser@thegeekstuff's password:

sftp> pwd

Remote working directory: /incoming

sftp> cd /

sftp> ls

incoming

When guestuser transfers any files to the /incoming directory from the sftp, they’ll be really located under /sftp/guestuser/incoming directory on the system.

DHCP Server Configuration

***# cat /etc/sysconfig/network-scripts/ifcfg-eth1*** *DEVICE=”eth1”  
NM\_CONTROLLED=”yes”  
ONBOOT=yes  
HWADDR=00:0C:29:8E:F1:FD  
TYPE=Ethernet  
BOOTPROTO=static  
DEFROUTE=yes  
PEERDNS=yes  
PEERROUTES=yes  
IPV4\_FAILURE\_FATAL=yes  
IPV6INIT=no  
NAME=”System eth1”  
UUID=5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03*

Before you configure your Server as DHCP server, please make sure that you server already had configurted with a static IP address, however, if you have been following along in all the labs. With your static IP in hand, you can begin the installation of the DHCP server.

## Task 1: Installing a DHCP Server

Step 1. Start by installing the dhcp package:

*# yum install -y dhcp*

Step 2. Verify that the package is installed correctly:

*# rpm -qa | grep dhcp  
dhcp-4.1.1-12.P1.el6.x86\_64*

Step 3. With the package installed, make sure that the dhcpd service starts when the system boots as well:

*# chkconfig dhcpd on*

Step 4. Verify that the DHCP service starts on boot:

*# chkconfig dhcpd –list  
dhcpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

## Task 2: Configuring the DHCP Server

To start the configuration, let’s look at the important files that handle the options for the DHCP service:

* /etc/dhcp/dhcpd.conf Main config file for the DHCP service using IPv4 addresses
* /etc/dhcp/dhcpd6.conf Main config file for the DHCP service using IPv6 addresses
* /var/lib/dhcpd/dhcpd.leases IPv4 client lease file
* /var/lib/dhcpd/dhcpd6.leases IPv6 client lease file

MIGRATION TIP : In RHEL5, the location of the main config file was /etc/dhcpd.conf. Now in RHEL6, the config file has been moved into a directory (/etc/dhcp) because there are additional config files for IPv6 addresses.

The main config file is usually empty aside from a comment or two. The good news is that the package does provide a sample config file for you to use. This sample file provides examples and comments on how you can configure options for your DHCP server.

To copy the sample file, use the following command:

# cp /usr/share/doc/dhcp-4.1.1/dhcpd.conf.sample /etc/dhc/dhcpd.conf

If you are new to working with DHCP servers, this sample file will prove helpful in that the comments included in the file explain the different options. The following sample config file gives you the basics to get started. It is less cluttered than the sample, so it is easier to explain.

Here is the sample DHCP server config file:

*# Global Options  
ddns-update-style none;  
authoritative;  
# Subnet definition  
subnet 172.168.1.0 netmask 255.255.255.0 {  
# Parameters for the local subnet  
option routers 172.168.1.1;  
option subnet-mask 255.255.255.0;  
option domain-name “example.com”;  
option domain-name-servers 172.168.1.1;  
default-lease-time 21600;  
max-lease-time 43200;  
# Client IP range  
range dynamic-bootp 172.168.1.100 172.168.1.200;  
}*

Let’s break down this file into sections. The first section contains two options for the DHCP server itself, also called global options:

REAL-WORLD TIP : If the server doesn’t contain an authoritative option and the client switches subnets,  
it is not able to obtain a new IP address until its old lease has fully expired. There are also security benefits of sending a DHCPNAK to incorrectly configured clients.

The next section defines a subnet. Any options that you list in a subnet section are specific to the subnet for which you define them. If you want to set global options (such as authoritative), you need to define them outside the subnet section. In this section, the following options are used:

* ***option routers****Defines the default gateway to the subnet*
* ***option subnet-mask****Defines the subnet mask for the subnet*
* ***option domain-name****Defines the name of the domain*
* ***option domainname-servers****Defines the DNS server for the subnet*
* ***ddns-update-style:none****This means that the DHCP server won’t update client DNS records.*
* ***authoritative****This informs the client that the DHCP server contains legitimate information.*
* ***default-lease-time S****pecifies how long each client keeps its lease until a renewal is requested (in seconds)*
* ***max-lease-time****Specifies the maximum amount of time a client can keep a lease (in seconds)*
* ***range dynamic-bootp****Specifies the range of IP addresses that can be given out to clients*

With a config file in place, you have everything you need for the DHCP server to function properly.

Although this section does not describe anywhere near all the options available for the DHCP service, this is a good start to getting a DHCP server up and running. For the exams, you can always reference the documentation if you forget the name of an option.

Although I hate pointing you to man pages, it is impossible to remember everything for every service. Knowing where to find information when you need it is critical on the exams and in the real world. Here are the three man pages you should know:

*# man dhcpd.conf  
# man dhcpd.leases  
# man dhcp-options*

If you want to have multiple subnets, you can just define a new subnet section with its own set of options. There is no limit to the number of sections you can have.

REAL-WORLD TIP : If you want to set up multiple subnets, you usually have multiple interfaces on the  
server. If this is the case and you would like to service multiple subnets of clients, you need to adjust the /etc/sysconfig/dhcpd file. In this case, you need to edit the following daemon option:

*DHCPDARGS=”eth0”*

Change this option to include all the interfaces for which you want to offer clients leases in different subnets. If you have two interfaces in two different subnets, for example, your option might look like this:

*DHCPDARGS=”eth0 eth1”*

The options we’ve covered so far are really all you need to set up your DHCP server to work. Even though the DHCP server is set to run, there is another example to consider. It is common in the real world to have to reserve an IP address for a particular client. If you don’t want to make the IP address of the client static, you can reserve it instead on the DHCP server. Reservations are common when dealing with printers on networks, but they can be used for clients, too. Here is how you define a reservation for a client.

In your /etc/dhcpd.conf file, do the following:

*host client01 {  
option host-name “client01.example.com”;  
hardware ethernet 02:B4:7C:43:DD:FF;  
fixed-address 172.168.1.50;  
}*

Again, let’s look at each of these options and what they do. These three options are defined in a host subsection, just like you defined a subnet section earlier. With the config files in place, you are now ready to start the service.

The DHCP server offers a particularly nice feature built into the startup scripts; it can check the syntax of your config file for errors. This capability is nice because prior to actually starting the service, you can check to make sure nothing is set incorrectly.

Check the config file for any errors:

*# service dhcpd configtest  
Syntax: OK*

If the DHCP service does find errors, it attempts to tell you where in the config file the error exists. Open your config file and remove the brace (}) that ends the subnet section. Save your config file and run the syntax check on your config file again:

*# service dhcpd configtest  
Internet Systems Consortium DHCP Server V3.0.5-RedHat  
Copyright 2004-2006 Internet Systems Consortium.  
All rights reserved.*

*For info, please visit http://www.isc.org/sw/dhcp//etc/dhcpd.conf line 18: unexpected end of file  
^  
Configuration file errors encountered — exiting*

Here, the service points out that the } is missing from the config file by giving you the line where the issue occurs as well as a general description of what the problem is. Now replace the } again and restart the DHCP service:

*# service dhcpd start  
Starting dhcpd: [ OK ]  
Verify that the service is running:*

*# service dhcpd status  
dhcpd (pid 3366) is running…*

Securing DHCP server with Firewall

Step 1. Use iptables to create the required firewall rule:

*# iptables -I INPUT 5 -p udp -m udp –dport 67 -j ACCEPT*

Step 2. Save the firewall rule you just created:

*# service iptables save  
Saving firewall rules to /etc/sysconfig/iptables: [ OK ]*

Step 3. Then restart the iptables service:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

With the firewall rules in place, you can shift your focus to SELinux. Well, wouldn’t you know…there are no SELinux requirements for a DHCP server.

You should now have a fully functional DHCP server! You can boot up your clients and see if they pick up an IP address, or if the client is already powered on, you can use the dhclient command to request an IP  
address from the server.  
Step 4. Request a dynamic IP from the server:

*# dhclient*

Troubleshooting DHCP related in issues in Linux

When you have a client that has a DHCP address, you can view the leases file by using the following

*# cat /var/lib/dhcpd/dhcpd.leases  
lease 172.168.1.200 {  
starts 2 2010/05/11 12:24:10;  
ends 2 2010/05/11 18:24:10;  
tstp 2 2010/05/11 18:24:10;  
binding state free;  
hardware ethernet 08:00:27:74:5b:11;  
}  
lease 172.168.1.199 {  
starts 4 2010/05/20 11:24:03;  
ends 4 2010/05/20 17:24:03;  
tstp 4 2010/05/20 17:24:03;  
binding state free;  
hardware ethernet 08:00:27:2f:80:8c;  
}*

As you can see from the file output, two clients have active leases. When viewing the leases file, you should be careful that all times in the dhcpd.leases file are in UTC (GMT). The reason for this is that there is no daylight savings in that time zone, making it internationally usable.

Make sure you pay attention to this issue if you’re trying to troubleshoot with this file. As a final tip, make sure you use the configtest  
option to test the syntax of your config files before starting or restarting the service.

This prevents any errors from occurring when you try to start the service.

REAL-WORLD  configuration information :  You may want to consider setting up a secondary DHCP server as a failover. It provides you with a backup should your primary server fail (don’t just rely on this secondary server, though; make sure you test it). If you create a failover server, you should follow the 80/20 rule: 80% of your IP addresses should be used on your primary DHCP server, and 20% should be used on your secondary server

DNS CONFIGURATION

In the background, though, Domain Name Service (DNS) is what translates that website name into an IP address so that the site may be accessed. This translation also occurs when you are connecting to other systems on your network through their hostnames instead of their IP addresses.

DNS plays a critical role not only in your networks, but also on the Internet as a whole. Knowing how to set up, maintain, and troubleshoot such a server is vital to any network.

The root DNS servers for the world run BIND as their DNS software choice, so naturally Red Hat also includes it on the exams. This chapter covers BIND in great detail because it is such a critical network component.

## Task1 : Setting Up BIND

Just as with any other service, you need to install the packages for BIND first.

Step 1. Install the required packages:

# yum install -y bind bind-utils bind-libs

Step 2. Verify that the packages have been installed:

# rpm -qa | grep ^bind  
bind-utils-9.7.0-5.P2.el6.x86\_64  
bind-9.7.0-5.P2.el6.x86\_64  
bind-libs-9.7.0-5.P2.el6.x86\_64

Step 3. Ensure that the service is set to start on system boot:

# chkconfig named on

Step 4. Verify that the service is set to start on boot:

# chkconfig –list named  
named 0:off 1:off 2:on 3:on 4:on 5:on 6:off

At this point, you need to decide what type of DNS server you would like to set up. According to Red Hat, there are two types of nameservers:

■ Authoritative—These nameservers answer to resource records that are part of their zones only. This includes both primary (master) and secondary (slave) nameservers.

■ Recursive—These nameservers offer resolution services but are not authoritative for any zone. All query answers are cached in memory for a fixed period of time.

There is another way you can classify nameservers:

■ Master—This nameserver stores original and authoritative zone records for a particular namespace. It also answers queries about the namespace from other nameservers. Each domain must have at least one master.

■ Slave—Although this nameserver receives its namespace information from a master nameserver, this nameserver type can answer queries for which it has authority.  Used for load balancing and redundancy.

■ Caching—This nameserver has no authority and is primarily used for name-to-IP resolution. All resolutions are cached for a fixed period of time. Heavily used by Internet service providers (ISPs).

■ Forwarding—This nameserver has no authority and is used only to forward requests  
to specific nameservers for resolution.

Each domain for which your server has authority is called a zone and the information for that zone in kept in….wait for it…zone files.

The caching DNS server requires an additional package to be installed, so let’s start by setting up a master DNS server and its zone first.

Each zone file also requires that you use the fully qualified domain name (FQDN) when defining hostnames. Although we usually address the configuration first, let’s look at the SELinux and firewall requirements instead because the configuration of BIND can become complex.

## [Task2 : SELinux and Firewall Rules for DNS Server](http://unixadminschool.com/member/selinux-and-firewall-rules-for-dns-server/)

## Task3 : DNS Server Configuration

To begin configuring the DNS server, check out these key config files for a BIND server:

* /etc/named.conf Main config file
* /etc/rndc.key Key file
* /etc/rndc.conf Key config file
* /usr/share/doc/bind-9\*/sample Directory that holds sample files

Before you do anything, you should make a backup of the /etc/named.conf file:

*# cp /etc/named.conf /etc/named.conf.orig*

You should also remove this file for now because you will be making a new one in the coming sections:

*# rm /etc/named.conf*

The first step for configuration is to make sure that your system has a static IP address and that the /etc/resolv.conf file is pointing to localhost as the nameserver:

Step 1. Verify that the localhost is used for DNS queries on RHEL01:

*# cat /etc/resolv.conf  
search example.com  
nameserver 127.0.0.1  
nameserver 192.168.1.1*

The secondary server listed here is actually my personal router, which can provide DNS as well. Before going any further, you should also understand the different types of resource records used with DNS and why each one is important.

* /etc/named.conf Main config file
* /etc/rndc.key Key file
* /etc/rndc.conf Key config file
* /usr/share/doc/bind-9\*/sample Directory that holds sample files
* *A                 – Maps the hostname to an IP address*
* *NS             – Contains the IP address or CNAME of the nameserver*
* *MX           – Defines where mail for a particular domain goes*
* *PTR          – Maps the IP address to a hostname*
* *SOA         – Contains general administrative control for the domain*
* *CNAME  – Used as an alias*

There are four different DNS server types, so let’s start with the configuration of each one.

* [Master Server Configuration](http://unixadminschool.com/member/master-dns-server-configuration-in-linux/)
* Slave Server Configuration
* [Caching-only Server configuration](http://unixadminschool.com/member/caching-only-dns-server/)
* [Forwarding-Only configuration](http://unixadminschool.com/member/forwarding-only-dns-server-configuration-in-linux-rhel6/)

## Troubleshooting DNS Server Configurations.

For the server and client, there are a handful of utilities you can use to verify the functionality of DNS. These utilities include

* dig                – DNS lookup utility
* host             –  DNS lookup utility
* ping             –  Network or hostname verification utility
* nslookup    –   Utility to lookup a hostname from an IP addresses
* hostname   –  Utility to sets or show the system hostname (FQDN)

Please refer the below links for more information on using the above tools for troubleshooting.

* [Troubleshooting DNS server Configuration issues  using ping, host and nslookup commands](http://unixadminschool.com/member/troubleshooting-dns-using-nslookup-and-host-commands-in-linux/)
* [Troubleshooting DNS server Configuration issues using dig command](http://unixadminschool.com/member/?p=4560&preview=true)

## Setup a FQDN ( fully Qualified domain name) in Linux

It is important to have proper FQDN set for each host that is configured with DNS. If you still haven’t set up the FQDN for your system, you can use the hostname command to set it up.

*Syntax: hostname [FQDN]*

Step 1. Query the current FQDN of your system:

*# hostname  
rhel01*

Step 2. If you want to change it, you can again use the hostname command, but specify the FQDN after the command:

*# hostname rhel01.example.com*

This changes your system’s hostname to include the domain in the hostname. If you make any changes to the hostname, you should reboot your system before continuing. As a final troubleshooting step, make  
sure that your clients have the correct information located in the /etc/resolv.conf file.  
Step 3. Change these DNS settings for Client01:

*# cat /etc/resolv.conf  
search example.com  
nameserver 172.168.1.1*

Master DNS Server Configuration in Linux

A master server is the basis for the DNS infrastructure. It provides a place to define all DNS records and extend them to secondary servers that can help with load balancing and redundancy. For this master server configuration, you need to create a new /etc/named.conf file. You can use the sample /etc/named.conf file to get started. Here is what the named.conf looks like:

### *Sample /etc/named.conf file*

*/\* Global options for the BIND Server \*/  
options  
{  
directory “/var/named”; // the default  
dump-file “data/cache\_dump.db”;  
statistics-file “data/named\_stats.txt”;  
memstatistics-file “data/named\_mem\_stats.txt”;  
};  
/\* Logging options so you know where your logs are going \*/  
logging  
{  
channel default\_debug {  
file “data/named.run”;  
severity dynamic;  
};  
};  
/\* Our sample domain is example.com defined here \*/  
zone “example.com” {  
type master;  
file “example.com.zone”;  
allow-update { none; };  
};  
/\* This is a reverse lookup for our subnet 172.168.1.0/24 \*/  
zone “1.168.172.in-addr.arpa” {  
type master;  
file “example.com.revzone”;  
allow-update { none; };  
};  
/\* File containing root hints (points directly to root DNS servers) \*/*

*zone “.” IN {  
type hint;  
file “named.ca”;  
};  
/\* The zone file for our localhost (good for troubleshooting) \*/  
zone “localhost.” IN {  
type master;  
file “named.localhost”;  
allow-update { none; };  
};  
/\* The reverse lookup zone for our localhost (good for troubleshooting) \*/  
zone “0.0.127.in-addr.arpa.” IN {  
type master;  
file “named.loopback”;  
allow-update { none; };  
};*

In the sample template here, the zone files are kept in /var/named, but in the real world, these files should actually exist in /var/named/dynamic. If you run into write errors on RHEL6, you should move your zone and reverse zone files into the /var/named/dynamic directory and update your /etc/named.conf file to reflect the changes.

The comments included here should make this config file self-explanatory. Take particular notice of the logging section.

The locations defined here specify where your log files go for BIND (however, some information is also logged to /var/log/messages).

This information is extremely helpful for troubleshooting should the DNS server not work properly. You can change the logging options for more or less information, but it is set up here to initially include enough information for someone to be able to know exactly what is going on.

Now that you have an /etc/named.conf file, you need to create the zone and reverse zone files.

### Sample forward lookup zone file

In the /var/named directory, you can set up the following example.com.zone file:

***# nano example.com.zone*** *;  
; Zone file for example.com*

*;  
$TTL 86400  
@ IN SOA rhel01.example.com. root.example.com. (  
2010120710 ; Serial  
1d ; refresh  
2h ; retry  
4w ; expire  
1h ) ; min cache  
@ IN NS rhel01.example.com.  
@ IN A 172.168.1.1  
;  
; Network Hosts  
;  
rhel01 IN A 172.168.1.1  
rhel02 IN A 172.168.1.2  
client01 IN A 172.168.1.10  
client02 IN A 172.168.1.20*

There are a few issues to make note of here. First, notice that when defining the domain, you put a dot (.) at the end of each FQDN. The root.example.com is the email address of the administrator for the DNS server, which is defined here without an at sign (@).

In the middle of the file, you create two entries for this server, making it the primary nameserver for this domain. Now if a client were to ping the domain name or the server’s hostname (RHEL01), it would respond because it is the primary nameserver. The last few lines define the clients and their IP addresses (notice that these hostnames are not FQDNs).

Let’s now look at the reverse zone file that will allow the DNS server to map IP addresses to hostnames. Again, you can work with the sample provided here:

### Sample Reverse Lookup Zone File

*# nano /var/named/example.com.revzone  
;  
; Reverse Zone file for example.com  
;  
$TTL 86400  
@ IN SOA rhel01.example.com. root.example.com. (  
2010120710 ; Serial  
1d ; refresh  
2h ; retry  
4w ; expire  
1h ) ; min cache*

*@ IN NS rhel01.example.com.  
;  
; Network Hosts  
;  
1 IN PTR rhel01.example.com.  
2 IN PTR rhel02.example.com.  
10 IN PTR client01.example.com.  
20 IN PTR client02.example.com.*

This file looks almost the same as the forward zone except that instead of A records there are PTR records. The number you see for each client is actually the number in the fourth octet of the IP address (2 correlates to 172.168.1.2). After each PTR definition, you specify the FQDN.

At this point, you still need three files. Two are for the localhost zone and one is the root hints file. On RHEL6, these files are all provided for you in the /var/named directory by default, making the  configuration process much easier.

Everything is now in place for you to begin using your DNS server. Before starting the service, however, make sure that the config files don’t have any syntax errors.

### Starting DNS Service

Step 1. You can use the configtest option of the named command to accomplish this:

*# service named configtest*

*zone localhost/IN: loaded serial 42  
zone 0.0.127.in-addr.arpa/IN: loaded serial 2010120710  
zone example.com/IN: loaded serial 2010120711  
zone 1.168.172.in-addr.arpa/IN: loaded serial 2010120710*

Step 2. Because no errors are displayed, you can start the service:

*# service named start  
Starting named: [ OK ]*

Real World Information : Don’t forget that to allow zone transfers or zone files to be written, you need to  
change the value of the named\_write\_master\_zones Boolean. At this point, you should have a fully functional master DNS server.

Although you aren’t required to know how to set up a master DNS server for the Red Hat exams, this task is commonly performed in the real world. The master DNS server is only the first of four, so let’s move on to the second type: a slave DNS server.

SLAVE DNS server Configuration in Linux

A slave DNS server is similar to a master DNS server. It can help with load balancing and provide redundancy should the master DNS server fail. Because it serves as a “secondary” DNS server, it actually pulls the necessary files from its master counterpart, making configuration of a slave DNS server quite easy. Because the slave server pulls all the DNS records from the master, you need to set up the slave DNS server on RHEL02.

NOTE: On RHEL02, you need to install the BIND packages, make a backup of the /etc/named.conf file, and copy the following template.

When you have RHEL02 set up, you can use the following template for your /etc/named.conf file:

### Sample named.conf file

*/\* Global options for the BIND Server \*/  
options  
{  
directory “/var/named”; // the default  
dump-file “data/cache\_dump.db”;  
statistics-file “data/named\_stats.txt”;  
memstatistics-file “data/named\_mem\_stats.txt”;  
};*

*/\* Logging options so you know where your logs are going \*/  
logging  
{  
channel default\_debug {  
file “data/named.run”;  
severity dynamic;  
};  
};*

*/\* Our sample domain is example.com defined here \*/  
zone “example.com” {  
type slave;  
file “slaves/example.com.zone”;  
masters { 172.168.1.1; };  
};  
/\* This is a reverse lookup for our subnet 172.168.1.0/24 \*/  
zone “1.168.172.in-addr.arpa” {  
type slave;  
file “slaves/example.com.revzone”;  
masters { 172.168.1.1; };  
};  
/\* File containing root hints (points directly to root DNS servers) \*/  
zone “.” IN {  
type hint;  
file “named.root”;  
};  
/\* The zone file for our localhost (good for troubleshooting) \*/  
zone “localhost.” IN {  
type master;  
file “localhost.zone”;  
allow-update { none; };  
};  
/\* The reverse lookup zone for our localhost (again good for troubleshooting) \*/  
zone “0.0.127.in-addr.arpa.” IN {  
type master;  
file “named.local”;  
allow-update { none; };  
};*

Because the slave server can provide redundancy or load balancing, its /etc/named.conf is similar to that of the master DNS server. The difference here, though, is that the slave server doesn’t actually need the zone files to exist.

You can see here that you define a masters option, which is actually the master DNS server. The slave DNS server will periodically check with the master DNS server, pull down the data for a zone, and create the zone file if it doesn’t exist. The slave zone files exist in the /var/named/slaves directory.

NOTE: Make sure that your named.ca, named.localhost, and named.loopback files are in place on RHEL02.

### Starting DNS Service

Step 1. With the files in place, you can check for syntax errors:

*# service named configtest  
zone localhost/IN: loaded serial 42  
zone 0.0.127.in-addr.arpa/IN: loaded serial 1997022700*

Step 2. Start the named service:

*# service named start  
Starting named: [ OK ]*

Step 3. Check the /var/named/slaves directory to see if the zone files copied over from the master DNS server correctly:

*# ls /var/named/slaves  
example.com.revzone example.com.zone*

You can manually pull the zone files from the master DNS server by using the dig command to perform a zone transfer. We look at the full syntax of the dig command later, but here you can see how to manually transfer a zone file:

*# dig -t axfr example.com @rhel01  
; <<>> DiG 9.3.6-P1-RedHat-9.3.6-4.P1.el5 <<>> @rhel01 example.com axfr  
; (1 server found)  
;; global options: printcmd  
example.com. 86400 IN SOA rhel01.example.com.  
root.example.com. 2010120711 86400 7200 2419200 3600  
example.com. 86400 IN NS rhel01.example.com.  
example.com. 86400 IN A 172.168.1.1  
client02.example.com. 86400 IN A 172.168.1.20  
client01.example.com. 86400 IN A 172.168.1.10  
rhel02.example.com. 86400 IN A 172.168.1.2  
rhel01.example.com. 86400 IN A 172.168.1.1  
example.com. 86400 IN SOA rhel01.example.com.  
root.example.com. 2010120711 86400 7200 2419200 3600  
;; Query time: 50 msec  
;; SERVER: 172.168.1.1#53(172.168.1.1)*

*;; WHEN: Tue Feb 1 10:21:25 2011  
;; XFR size: 8 records (messages 1)*

If you get any errors, the slave DNS server is not able to pull the zone files from the master DNS server until the errors are resolved. In the “DNS Utilities and Troubleshooting” section later, you see how to resolve any errors that are thrown here.

Caching-Only DNS server

The setup of a name caching-only server is a little different from what you have done already. The first thing we mentioned before starting any configuration is that you back up the original /etc/named.conf file (RHEL6 only).

This original /etc/named.conf file is actually a sample named.conf file for name caching-only DNS servers! On RHEL01 again, make a backup of any current named.conf file:

*# cp /etc/named.conf /etc/named.conf.bk*

Restore the original named.conf file:

*# mv /etc/named.conf.orig /etc/named.conf*

Use the following as your caching-only nameserver /etc/named.conf config file:

### Sample named.conf file

*/\* General options for our caching-only name server \*/  
options {  
listen-on port 53 { 127.0.0.1; };  
listen-on-v6 port 53 { ::1; };  
directory “/var/named”;  
dump-file “/var/named/data/cache\_dump.db”;  
statistics-file “/var/named/data/named\_stats.txt”;  
memstatistics-file “/var/named/data/named\_mem\_stats.txt”;  
allow-query { localhost; };  
recursion yes;  
};*

*/\* Logging options so you know where your logs are going \*/  
logging  
{  
channel default\_debug {  
file “data/named.run”;  
severity dynamic;  
};  
};  
/\* This view statement forces cached lookups only \*/  
zone “.” IN {  
type hint;  
file “named.ca”;  
};  
include “/etc/named.rfc1912.zones”;*

Make sure to save this file as named.conf and ensure it is in the /etc/directory. For a caching-only nameserver, there is only one other file you need:

/etc/named.rfc.1912.zones (as defined in the sample file). This file should already exist by default in the /etc directory.

The named.rfc.1912.zones file looks for the named.localhost and named.loopback files created earlier. It also looks for a named.empty file, which should already exist in your /var/named directory.

### Starting DNS Service

Step 1. When you have everything set up, make sure that you test the config file for any errors:

*# service named configtest  
zone localhost/IN: loaded serial 42  
zone 0.0.127.in-addr.arpa/IN: loaded serial 1997022700*

Step 2. Now you can start the service:

*# service named start  
Starting named: [ OK ]*

Important Note: If you set up a caching-only nameserver, you should now switch your /etc/named.conf file back to the master DNS server configuration. The rest of this chapter deals with DNS security and troubleshooting assuming that you still have the master DNS server in place.

Forwarding-Only DNS server Configuration in Linux

### Sample named.conf file

*options {  
directory “/var/named”;  
fowarders { 172.168.1.0/24; forward first; };  
};*

This section sends all queries to the first DNS server in the match-list and to the root DNS servers if no match is made. You also can specify forward only to check only the match-list servers and no root DNS server.

If you don’t include the directive, it defaults to forward first. If you are going to allow the query of root DNS servers, you must have the /var/named/named.ca file, which contains a list of root DNS servers.

This is the entire setup required for a forwarding-only DNS server.

# SELinux and Firewall Rules for DNS Server

BY [RAMDEV](http://unixadminschool.com/blog/author/ramdevram/) · PUBLISHED DECEMBER 27, 2015 · UPDATED DECEMBER 29, 2015

For your clients to be able to query the DNS server, you need to open a single port on the firewall, but for both protocols. The DNS clients can use both TCP and UDP port 53.

### Step 1. Use the iptables command to create your firewall rules:

*# iptables -I INPUT 5 -p udp -m udp –dport 53 -j ACCEPT  
# iptables -I INPUT 5 -p tcp -m tcp –dport 53 -j ACCEPT*

### Step 2. Save the rules you just created:

*# service iptables save  
Saving firewall rules to /etc/sysconfig/iptables: [ OK ]*

### Step 3. Restart the firewall service for the changes to take effect:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

For the DNS server, there is only a single SELinux Boolean value that you can change.

*named\_write\_master\_zones Allows master zone files to be written*

If you would like to have a dynamic DNS server or allow zone file transfers, you  
need to disable this value; otherwise, you can leave the default protection as is.

For additional security, you can set SELinux to allow only the named system user to be able to read the /etc/named.conf file. This ensures additional security should you need it. To allow only the named user to be able to read the /etc/named.conf file, use the following command:

*# chcon -t named\_conf\_t /etc/named.conf*

Verify with this command:

*# ls -Z /etc | grep named.conf*

* [artion](http://unixadminschool.com/blog/category/quiz-center/)

* [Training](http://unixadminschool.com/member/training)

* [Who we are](http://wp.me/P48rad-y)

* [How to Use this Site ?](http://unixadminschool.com/blog/category/site-reference/)
* **[LINUX ADMIN](http://unixadminschool.com/blog/category/unix-administration/linux/) / [LINUX TUTORIALS](http://unixadminschool.com/blog/category/linux-tutorials/)**
* **[0](http://unixadminschool.com/blog/2015/12/troubleshooting-dns-using-nslookup-and-host-commands-in-linux/" \l "respond)**

# Troubleshooting DNS using nslookup and host commands in Linux

The most basic test after the DNS server has been set up properly is to ping the hostname of the nameserver and the domain itself. If both return a reply, your nameserver is querying properly.

Step 1. Ping the hostname of the nameserver for your network—in this case, RHEL01:

*# ping rhel01  
PING rhel01.example.com (172.168.1.1) 56(84) bytes of data.  
64 bytes from rhel01.example.com (172.168.1.1): icmp\_seq=1 ttl=64 time=0.036 ms  
64 bytes from rhel01.example.com (172.168.1.1): icmp\_seq=2 ttl=64 time=0.020 ms  
— rhel01.example.com ping statistics —  
2 packets transmitted, 2 received, 0% packet loss, time 1001ms  
rtt min/avg/max/mdev = 0.020/0.028/0.036/0.008 ms*

Step 2. Next, you can ping the domain name to ensure that the primary nameserver is again functioning properly:

*# ping example.com  
PING example.com (172.168.1.1) 56(84) bytes of data.  
64 bytes from rhel01.example.com (172.168.1.1): icmp\_seq=1 ttl=64 time=0.036 ms  
64 bytes from rhel01.example.com (172.168.1.1): icmp\_seq=2 ttl=64time=0.020 ms  
— example.com ping statistics —  
2 packets transmitted, 2 received, 0% packet loss, time 1001ms   
rtt min/avg/max/mdev = 0.020/0.028/0.036/0.008 ms*

Another useful tool that can help test whether your DNS server is functioning properly is the host  command.

*Syntax: host [option] HOSTNAME  
Options:  
-l Allows you to perform zone transfers  
-r Disables recursive processing  
-t Specifies the query type  
-v Provides verbose output*

You can use this command for two simple purposes to ensure functionality.

Step 3. First, perform a forward lookup to test the main zone file:

*# host rhel01  
rhel01.example.com has address 172.168.1.1*

Step 4. Second, perform a reverse lookup to test the reverse lookup zone file:

*# host 172.168.1.1  
1.1.168.172.in-addr.arpa domain name pointer rhel01.example.com.*

If the results that you are expecting don’t match what is displayed, or if the hostname or IP address can’t be found, it is a good indication there is a problem with that particular zone file.

This is a good way to test a few of the entries in each zone file to verify functionality of the DNS server. Aside from the host command, you can also use the nslookup command that offers slightly different information about lookups.

Step 5. Query the domain name again:

*# nslookup example.com  
Server: 172.168.1.1  
Address: 172.168.1.1#53  
Name: example.com  
Address: 172.168.1.1*

This time you see the nameserver for the example.com domain respond. This is a good way to find out which DNS server is responsible for a particular domain.

Step 6. You can also use nslookup like the host command to perform forward lookups:

*# nslookup rhel01  
Server: 172.168.1.1  
Address: 172.168.1.1#53  
Name: rhel01.example.com  
Address: 172.168.1.1*

Step 7. Plus, you can use it to perform reverse lookups:

*# nslookup 172.168.1.1  
Server: 172.168.1.1  
Address: 172.168.1.1#53  
1.1.168.172.in-addr.arpa name = rhel01.example.com.*

### Spread a word

Troubleshooting DNS server Configuration issues using dig command

Step 1. Do a forward lookup of your DNS server directly:

*# dig @RHEL01*

Step 2. Do a forward lookup of your domain name:

*# dig @RHEL01 example.com*

Step 3. Also check the reverse lookup of your domain name:

*# dig -x 1.168.172.in-addr.arp*

Step 4. Test for zone transfer functionality:

*# dig @RHEL01 example.com axfr  
; <<>> DiG 9.3.6-P1-RedHat-9.3.6-4.P1.el5 <<>> @rhel01*

## ****ISCSI Server Configuration on RHEL 6****

Overall Steps :

1. Install the RPM required for ISCSI Target Administration
2. Enable the ISCSI Target Services
3. Configure the ISCSI target to share the storage
4. Attach Luns to the ISCSI Target
5. Share the ISCSI Target to specific Client Machines
6. Add CHAP user authentication ISCSI target
7. Save the configuration to /etc/tgt/targets.conf , so that configuration won’t last after reboot
8. Make rules for IPTABLES to accept iscsi client connection on port 3260

### 

### ****Step 1: Install the RPM required for ISCSI Target Administration****

*[root@gurkulsan ~]# yum list scsi-target-utils  
Loaded plugins: product-id, refresh-packagekit, security, subscription-manager  
Updating certificate-based repositories.  
Unable to read consumer identity  
Installed Packages  
scsi-target-utils.x86\_64 1.0.24-2.el6 @anaconda-RedHatEnterpriseLinux-201206132210.x86\_64/6.3  
[root@gurkulsan ~]#*

**Check the files which are installed by scsi-target-utils rpm**

*[root@gurkulsan ~]# rpm -ql scsi-target-utils  
/etc/rc.d/init.d/tgtd  
/etc/sysconfig/tgtd  
/etc/tgt/targets.conf  
/usr/sbin/tgt-admin  
/usr/sbin/tgt-setup-lun  
/usr/sbin/tgtadm  
/usr/sbin/tgtd  
/usr/sbin/tgtimg  
/usr/share/doc/scsi-target-utils-1.0.24  
/usr/share/doc/scsi-target-utils-1.0.24/README  
/usr/share/doc/scsi-target-utils-1.0.24/README.iscsi  
/usr/share/doc/scsi-target-utils-1.0.24/README.iser  
/usr/share/doc/scsi-target-utils-1.0.24/README.lu\_configuration  
/usr/share/doc/scsi-target-utils-1.0.24/README.mmc  
/usr/share/man/man5/targets.conf.5.gz  
/usr/share/man/man8/tgt-admin.8.gz  
/usr/share/man/man8/tgt-setup-lun.8.gz  
/usr/share/man/man8/tgtadm.8.gz*

### ****Step 2  : Enable the ISCSI Target Services to auto start during boot. And also start the service****

*[root@gurkulsan ~]# /etc/init.d/tgtd start  
Starting SCSI target daemon: [ OK ]  
[root@gurkulsan ~]# chkconfig tgtd on*

### ****Step 3: Configure the ISCSI target as a base of shared storage****

We can configure ISCSI target using the target administration utility tgtadm. The tgtadm will have following syntax

[root@gurkulsan ~]# tgtadm –help  
Usage: tgtadm [OPTION]  
Linux SCSI Target Framework Administration Utility, version 1.0.24

–lld –mode target –op new –tid –targetname   
add a new target with and . must not be zero.  
–lld –mode target –op delete [–force] –tid   
delete the specific target with .  
With force option, the specific target is deleted  
even if there is an activity.  
–lld –mode target –op show  
show all the targets.  
–lld –mode target –op show –tid   
show the specific target’s parameters.  
–lld –mode target –op update –tid –name

–value  
change the target parameters of the specific target with .  
–lld –mode target –op bind –tid –initiator-address

–lld –mode target –op bind –tid –initiator-name   
enable the target to accept the specific initiators.  
–lld –mode target –op unbind –tid –initiator-address

–lld –mode target –op unbind –tid –initiator-name   
disable the specific permitted initiators.  
–lld –mode logicalunit –op new –tid –lun   
–backing-store –bstype –bsoflags   
add a new logical unit with to the specific target with . The logical unit is offered to the initiators. must be block device files (including LVM and RAID devices) or regular files.  
bstype option is optional.  
bsoflags supported options are sync and direct (sync:direct for both).  
–lld –mode logicalunit –op delete –tid –lun   
delete the specific logical unit with that the target with has.  
–lld –mode account –op new –user –password   
add a new account with and .  
–lld –mode account –op delete –user   
delete the specific account having .  
–lld –mode account –op bind –tid –user [–outgoing]  
add the specific account having to  the specific target with .  
could be or .  
If you use –outgoing option, the account will be added as an outgoing account.  
–lld –mode account –op unbind –tid –user   
delete the specific account having from specific target.  
–control-port use control port   
–help display this help and exit

**Check if there are any targets exists already**

*# [root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op show*

Nothing returns, means no shared storage configured so far.

**Configure the first Target with target id – 101**

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op new –tid 101 –targetname iqn.2013-04.com.unixadminshool:target1*

*Check the new Target Created*

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op show  
Target 101: iqn.2013-04.com.unixadminschool:target1  
System information:  
Driver: iscsi  
State: ready  
I\_T nexus information:  
LUN information:  
LUN: 0****<=== Lun 0 always refers to the target*** *Type: controller  
SCSI ID: IET 00650000  
SCSI SN: beaf1010  
Size: 0 MB, Block size: 1  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: null  
Backing store path: None  
Backing store flags:  
Account information:  
ACL information:*

### ****Step 4: Attach Luns to the ISCSI Target****

**Decide the local disks that you want to use for the shared luns creation.**

[root@gurkulsan ~]# fdisk -l|grep -i Disk  
Disk /dev/sda: 17.2 GB, 17179869184 bytes  
Disk identifier: 0x000ae2a8  
Disk**/dev/sdc**: 2147 MB, 2147483648 bytes   <==  I want to use this disk for lun-1  
Disk identifier: 0x00000000  
Disk **/dev/sdb**: 2147 MB, 2147483648 bytes  <==  I want to use this disk for lun-2  
Disk identifier: 0xaed24ed6  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_root: 12.4 GB, 12423528448 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_swap: 4227 MB, 4227858432 bytes  
Disk identifier: 0x00000000  
[root@gurkulsan ~]#

**Make sure no partitions and data exists on /dev/sdb and /dev/sdc**

*[root@gurkulsan ~]# fdisk -l /dev/sdb*

*Disk /dev/sdb: 2147 MB, 2147483648 bytes  
255 heads, 63 sectors/track, 261 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000*

*[root@gurkulsan ~]# fdisk -l /dev/sdc*

*Disk /dev/sdc: 2147 MB, 2147483648 bytes  
255 heads, 63 sectors/track, 261 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000*

**Create LUNS using the disks /dev/sdb ( i.e. lun-1 ) and /dev/sdc ( i.e  lun-2 )as backing-stores**

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode logicalunit –op new –tid 101 –lun 1 –backing-store /dev/sdb*

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode logicalunit –op new –tid 101 –lun 2 –backing-store /dev/sdc*

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op show  
Target 101: iqn.2013-04.com.unixadminschool:target1  
System information:  
Driver: iscsi  
State: ready  
I\_T nexus information:  
LUN information:  
LUN: 0  
Type: controller  
SCSI ID: IET 00650000  
SCSI SN: beaf1010  
Size: 0 MB, Block size: 1  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: null  
Backing store path: None  
Backing store flags:  
LUN: 1         <== Lun1 created with /dev/sdb  
Type: disk  
SCSI ID: IET 00650001  
SCSI SN: beaf1011  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path:****/dev/sdb*** *Backing store flags:  
LUN: 2     <== Lun2 created with /dev/sdc  
Type: disk  
SCSI ID: IET 00650002  
SCSI SN: beaf1012  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path:****/dev/sdc*** *Backing store flags:  
Account information:  
ACL information:*

### ****Step 5: Share the ISCSI Target to specific Client Machines****

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op bind –tid 101 –initiator-address 192.168.1.34*

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op show  
Target 101: iqn.2013-04.com.unixadminschool:target1  
System information:  
Driver: iscsi  
State: ready  
I\_T nexus information:  
LUN information:  
LUN: 0  
Type: controller  
SCSI ID: IET 00650000  
SCSI SN: beaf1010  
Size: 0 MB, Block size: 1  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: null  
Backing store path: None  
Backing store flags:  
LUN: 1  
Type: disk  
SCSI ID: IET 00650001  
SCSI SN: beaf1011  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path: /dev/sdb  
Backing store flags:  
LUN: 2  
Type: disk  
SCSI ID: IET 00650002  
SCSI SN: beaf1012  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path: /dev/sdc  
Backing store flags:  
Account information:  
ACL information:****192.168.1.34****<==  Client IP Address added here*

### ****Step 6: Enable CHAP Authentication to the target****

**Create a User named  sanclient with the password mypassword**

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode account –op new –user****sanclient****–password****mypassword***

**Attach the user sanclient to the target 101  ( i.e. target1)**

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode account –op bind –tid 101 –user****sanclient***

**Check the User was attached to target-1 (  i.e. tid – 101 )**

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode account –op show  
Account list:  
sanclient  
[root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op show  
Target 101: iqn.2013-04.com.unixadminschool:target1  
System information:  
Driver: iscsi  
State: ready  
I\_T nexus information:  
LUN information:  
LUN: 0  
Type: controller  
SCSI ID: IET 00650000  
SCSI SN: beaf1010  
Size: 0 MB, Block size: 1  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: null  
Backing store path: None  
Backing store flags:  
LUN: 1  
Type: disk  
SCSI ID: IET 00650001  
SCSI SN: beaf1011  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path: /dev/sdb  
Backing store flags:  
LUN: 2  
Type: disk  
SCSI ID: IET 00650002  
SCSI SN: beaf1012  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path: /dev/sdc  
Backing store flags:  
Account information:****sanclient****<== Authentication user name appears here  
ACL information:  
192.168.1.34*

### ****Step 7. Save the configuration to /etc/tgt/targets.conf , so that configuration won’t last after reboot****

**Take the backup of original file**

*[root@gurkulsan ~]# cp /etc/tgt/targets.conf /etc/tgt/targets.conf.orig*

**Get the current configuration dump using the command**

*[root@gurkulsan ~]# tgt-admin –dump  
default-driver iscsi*

*backing-store /dev/sdb  
backing-store /dev/sdc  
incominguser sanclient PLEASE\_CORRECT\_THE\_PASSWORD  
initiator-address 192.168.1.34  
initiator-address 192.168.1.35*

Copy  and paste the output of  “tgtadmin –dump” to the end of the file /etc/tgt/targets.conf, so that the configuration exists after reboot. Modified file will have following differences after modification

*[root@gurkulsan ~]# sdiff -s /etc/tgt/targets.conf /etc/tgt/targets.conf.orig  
                                                              <  
             <  
backing-store /dev/sdb                                        <  
backing-store /dev/sdc                                        <  
incominguser sanclient mypassword                             <  
initiator-address 192.168.1.34                                <  
initiator-address 192.168.1.35                                <  
                                                    <  
                                                              <*

**Check configuration existing after service restart**

*[root@gurkulsan ~]# /etc/init.d/tgtd restart  
Stopping SCSI target daemon: [ OK ]  
Starting SCSI target daemon: [ OK ]*

*[root@gurkulsan ~]# tgtadm –lld iscsi –mode target –op show  
Target 101: iqn.2013-04.com.unixadminschool:target1  
System information:  
Driver: iscsi  
State: ready  
I\_T nexus information:  
LUN information:  
LUN: 0  
Type: controller  
SCSI ID: IET 00650000  
SCSI SN: beaf1010  
Size: 0 MB, Block size: 1  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: null  
Backing store path: None  
Backing store flags:  
LUN: 1  
Type: disk  
SCSI ID: IET 00650001  
SCSI SN: beaf1011  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path: /dev/sdb  
Backing store flags:  
LUN: 2  
Type: disk  
SCSI ID: IET 00650002  
SCSI SN: beaf1012  
Size: 2147 MB, Block size: 512  
Online: Yes  
Removable media: No  
Prevent removal: No  
Readonly: No  
Backing store type: rdwr  
Backing store path: /dev/sdc  
Backing store flags:  
Account information:  
sanclient  
ACL information:  
192.168.1.34*

*Check the current ports used by tgtd daemon*

*[root@gurkulsan ~]# netstat -pantl|grep tgt  
tcp 0 0 0.0.0.0:3260 0.0.0.0:\* LISTEN 4516/tgtd  
tcp 0 0 :::3260 :::\* LISTEN 4516/tgtd*

*Note: netstat -pantl — used to check all the ports which are open and listening by process wise*

### ****Step 8: Add rules for IPTABLES to accept connections from iscsi clients****

**Check the current rules, and there is no rule exists for port 3260**

*[root@gurkulsan ~]# iptables -L -n  
Chain INPUT (policy ACCEPT)  
target prot opt source destination  
ACCEPT udp — 0.0.0.0/0 0.0.0.0/0 udp dpt:53  
ACCEPT tcp — 0.0.0.0/0 0.0.0.0/0 tcp dpt:53  
ACCEPT udp — 0.0.0.0/0 0.0.0.0/0 udp dpt:67  
ACCEPT tcp — 0.0.0.0/0 0.0.0.0/0 tcp dpt:67*

*Chain FORWARD (policy ACCEPT)  
target prot opt source destination  
ACCEPT all — 0.0.0.0/0 192.168.122.0/24 state RELATED,ESTABLISHED  
ACCEPT all — 192.168.122.0/24 0.0.0.0/0  
ACCEPT all — 0.0.0.0/0 0.0.0.0/0  
REJECT all — 0.0.0.0/0 0.0.0.0/0 reject-with icmp-port-unreachable  
REJECT all — 0.0.0.0/0 0.0.0.0/0 reject-with icmp-port-unreachable*

*Chain OUTPUT (policy ACCEPT)  
target prot opt source destination*

*[root@gurkulsan ~]# iptables -I INPUT -p tcp –dport 3260 -j ACCEPT  
[root@gurkulsan ~]# service iptables save  
iptables: Saving firewall rules to /etc/sysconfig/iptables:[ OK ]*

[root@gurkulsan ~]# iptables -L -n  
Chain INPUT (policy ACCEPT)  
target prot opt source destination  
**ACCEPT tcp — 0.0.0.0/0 0.0.0.0/0 tcp dpt:3260  <==  New rule added**  
ACCEPT udp — 0.0.0.0/0 0.0.0.0/0 udp dpt:53  
ACCEPT tcp — 0.0.0.0/0 0.0.0.0/0 tcp dpt:53  
ACCEPT udp — 0.0.0.0/0 0.0.0.0/0 udp dpt:67  
ACCEPT tcp — 0.0.0.0/0 0.0.0.0/0 tcp dpt:67

Chain FORWARD (policy ACCEPT)  
target prot opt source destination  
ACCEPT all — 0.0.0.0/0 192.168.122.0/24 state RELATED,ESTABLISHED  
ACCEPT all — 192.168.122.0/24 0.0.0.0/0  
ACCEPT all — 0.0.0.0/0 0.0.0.0/0  
REJECT all — 0.0.0.0/0 0.0.0.0/0 reject-with icmp-port-unreachable  
REJECT all — 0.0.0.0/0 0.0.0.0/0 reject-with icmp-port-unreachable

Chain OUTPUT (policy ACCEPT)  
target prot opt source destination

## ISCSI Client Configuration for RHEL6.3

Overall Steps :

Step 1: Install the ISCSI  Client Utilities rpm i.e. iscsi-initiator-utils

Step 2 : Enables the services for Aut0-start, and start the services

Step 3: Discovery the targets available from the ISCSI target server and bind to the target

Step 4: Configure the iscsi client configuration file, in case if the CHAP authentication enabled to access the Targets

Step 5:  Check the new iscsi storage visible in the server, partition and use them

### ****Step 1: Install the ISCSI  Client Utilities rpm i.e. iscsi-initiator-utils****

**Install the RPM**

*[root@gurkulrhel4 ~]# yum install iscsi-initiator-utils  
Loaded plugins: product-id, refresh-packagekit, security, subscription-manager  
Updating certificate-based repositories.  
Unable to read consumer identity  
ol6\_UEK\_latest | 1.2 kB 00:00  
ol6\_latest | 1.4 kB 00:00  
rhel-debuginfo | 1.2 kB 00:00 …  
rpmforge | 1.9 kB 00:00  
Setting up Install Process  
Resolving Dependencies  
–> Running transaction check  
—> Package iscsi-initiator-utils.x86\_64 0:6.2.0.872-41.el6 will be updated  
—> Package iscsi-initiator-utils.x86\_64 0:6.2.0.873-2.0.1.el6 will be an update  
–> Finished Dependency Resolution*

*Dependencies Resolved*

*==========================================================================================  
Package Arch Version Repository Size  
==========================================================================================  
Updating:  
iscsi-initiator-utils x86\_64 6.2.0.873-2.0.1.el6 ol6\_latest 655 k*

*Transaction Summary  
==========================================================================================  
Upgrade 1 Package(s)*

*Total download size: 655 k  
Is this ok [y/N]: y  
Downloading Packages:  
iscsi-initiator-utils-6.2.0.873-2.0.1.el6.x86\_64.rpm | 655 kB 00:03  
Running rpm\_check\_debug  
Running Transaction Test  
Transaction Test Succeeded  
Running Transaction  
Updating : iscsi-initiator-utils-6.2.0.873-2.0.1.el6.x86\_64 1/2  
Cleanup : iscsi-initiator-utils-6.2.0.872-41.el6.x86\_64 2/2  
Installed products updated.  
Verifying : iscsi-initiator-utils-6.2.0.873-2.0.1.el6.x86\_64 1/2  
Verifying : iscsi-initiator-utils-6.2.0.872-41.el6.x86\_64 2/2*

*Updated:  
iscsi-initiator-utils.x86\_64 0:6.2.0.873-2.0.1.el6*

*Complete!*

**Check the files installed alond with the RPM**

*[root@gurkulrhel4 ~]# rpm -ql iscsi-initiator-utils  
/etc/NetworkManager  
/etc/NetworkManager/dispatcher.d  
/etc/NetworkManager/dispatcher.d/04-iscsi  
/etc/iscsi  
/etc/iscsi/iscsid.conf  
/etc/logrotate.d/iscsiuiolog  
/etc/rc.d/init.d/iscsi  
/etc/rc.d/init.d/iscsid  
/sbin/brcm\_iscsiuio  
/sbin/iscsi-iname  
/sbin/iscsiadm  
/sbin/iscsid  
/sbin/iscsistart  
/sbin/iscsiuio  
/usr/lib64/libiscsi.so.0  
/usr/lib64/python2.6/site-packages/libiscsimodule.so  
/usr/share/doc/iscsi-initiator-utils-6.2.0.873  
/usr/share/doc/iscsi-initiator-utils-6.2.0.873/README  
/usr/share/man/man8/iscsi-iname.8.gz  
/usr/share/man/man8/iscsiadm.8.gz  
/usr/share/man/man8/iscsid.8.gz  
/usr/share/man/man8/iscsistart.8.gz  
/usr/share/man/man8/iscsiuio.8.gz  
/var/lib/iscsi  
/var/lib/iscsi/ifaces  
/var/lib/iscsi/isns  
/var/lib/iscsi/nodes  
/var/lib/iscsi/send\_targets  
/var/lib/iscsi/slp  
/var/lib/iscsi/static  
/var/lock/iscsi*

**Step 2 : Enables the services for Aut0-start, and start the services**

**Start the iscsid Service**

*[root@gurkulrhel4 ~]# service iscsid restart  
Stopping iscsid:  
[root@gurkulrhel4 ~]# service iscsid force-start  
Starting iscsid: [ OK ]*

**Enable the Service for Auto-startup during boot**

*[root@gurkulrhel4 ~]# chkconfig iscsid on  
[root@gurkulrhel4 ~]# chkconfig –list iscsid  
iscsid 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

### ****Step 3: Discovery the targets available from the ISCSI target server and bind to the target****

**Run the iscsiadm command in discoverydb mode to get the information of available targets from the iscsi server 192.168.1.30 ( i.e. gurkusan)**

*[root@gurkulrhel4 ~]# iscsiadm –mode discoverydb –type sendtargets –portal 192.168.1.30 –discover  
192.168.1.30:3260,1 iqn.2013-04.com.unixadminschool:target1*

Once the target discovered, bind to the target

*[root@gurkulrhel4 ~]# iscsiadm –mode node –targetname iqn.2013-04.com.unixadminschool:target1 –portal 192.168.1.30 –login  
Logging in to [iface: default, target: iqn.2013-04.com.unixadminschool:target1, portal: 192.168.1.30,3260] (multiple)  
iscsiadm: Could not login to [iface: default, target: iqn.2013-04.com.unixadminschool:target1, portal: 192.168.1.30,3260].  
iscsiadm: initiator reported error (24 – iSCSI login failed due to authorization failure)  
iscsiadm: Could not log into all portals*

above faults appears because the targets were secured by CHAP authentication. We need to provide the authenitcated username and password to bind to the target

**Step 4: Configure the iscsi client configuration file, in case if the CHAP authentication enabled to access the Targets**

Add following details to the file /etc/iscsi/iscsid.conf

*node.session.auth.authmethod = CHAP  
node.session.auth.username = sanclient  
node.session.auth.password = mypassword*

**And then rediscover the ISCSI targets**

*[root@gurkulrhel4 ~]# iscsiadm –mode discoverydb –type sendtargets –portal 192.168.1.30 –discover  
192.168.1.30:3260,1 iqn.2013-04.com.unixadminschool:target1*

**Whenever iscsid discovers new target, it will add corresponding information in the following directory**

*[root@gurkulrhel4 ~]# ls -lR /var/lib/iscsi/nodes/  
/var/lib/iscsi/nodes/:  
total 4  
drw——- 3 root root 4096 Apr 14 13:01 iqn.2013-04.com.unixadminschool:target1*

*/var/lib/iscsi/nodes/iqn.2013-04.com.unixadminschool:target1:  
total 4  
drw——- 2 root root 4096 Apr 14 13:01 192.168.1.30,3260,1*

*/var/lib/iscsi/nodes/iqn.2013-04.com.unixadminschool:target1/192.168.1.30,3260,1:  
total 4  
-rw——- 1 root root 1902 Apr 14 13:01 default*

**And the following file will have all the related information about the discovered target**

Note :  you can find our CHAP user and password information in highlights

*[root@gurkulrhel4 ~]# cat /var/lib/iscsi/nodes/iqn.2013-04.com.unixadminschool:target1/192.168.1.30,3260,1/default  
# BEGIN RECORD 6.2.0-873.2.el6  
node.name = iqn.2013-04.com.unixadminschool:target1  
node.tpgt = 1  
node.startup = automatic  
node.leading\_login = No  
iface.iscsi\_ifacename = default  
iface.transport\_name = tcp  
iface.vlan\_id = 0  
iface.vlan\_priority = 0  
iface.iface\_num = 0  
iface.mtu = 0  
iface.port = 0  
node.discovery\_address = 192.168.1.30  
node.discovery\_port = 3260  
node.discovery\_type = send\_targets  
node.session.initial\_cmdsn = 0  
node.session.initial\_login\_retry\_max = 8  
node.session.xmit\_thread\_priority = -20  
node.session.cmds\_max = 128  
node.session.queue\_depth = 32  
node.session.nr\_sessions = 1****node.session.auth.authmethod = CHAP******node.session.auth.username = sanclient******node.session.auth.password = mypassword*** *node.session.timeo.replacement\_timeout = 120  
node.session.err\_timeo.abort\_timeout = 15  
node.session.err\_timeo.lu\_reset\_timeout = 30  
node.session.err\_timeo.tgt\_reset\_timeout = 30  
node.session.err\_timeo.host\_reset\_timeout = 60  
node.session.iscsi.FastAbort = Yes  
node.session.iscsi.InitialR2T = No  
node.session.iscsi.ImmediateData = Yes  
node.session.iscsi.FirstBurstLength = 262144  
node.session.iscsi.MaxBurstLength = 16776192  
node.session.iscsi.DefaultTime2Retain = 0  
node.session.iscsi.DefaultTime2Wait = 2  
node.session.iscsi.MaxConnections = 1  
node.session.iscsi.MaxOutstandingR2T = 1  
node.session.iscsi.ERL = 0  
node.conn[0].address = 192.168.1.30  
node.conn[0].port = 3260  
node.conn[0].startup = manual  
node.conn[0].tcp.window\_size = 524288  
node.conn[0].tcp.type\_of\_service = 0  
node.conn[0].timeo.logout\_timeout = 15  
node.conn[0].timeo.login\_timeout = 15  
node.conn[0].timeo.auth\_timeout = 45  
node.conn[0].timeo.noop\_out\_interval = 5  
node.conn[0].timeo.noop\_out\_timeout = 5  
node.conn[0].iscsi.MaxXmitDataSegmentLength = 0  
node.conn[0].iscsi.MaxRecvDataSegmentLength = 262144  
node.conn[0].iscsi.HeaderDigest = None  
node.conn[0].iscsi.IFMarker = No  
node.conn[0].iscsi.OFMarker = No  
# END RECORD  
[root@gurkulrhel4 ~]#*

### ****Step 5:  Check the new iscsi storage visible in the server, partition and use them****

*[root@gurkulrhel4 ~]# fdisk -l|grep Disk  
Disk /dev/sda: 17.2 GB, 17179869184 bytes  
Disk identifier: 0x000ae2a8  
Disk /dev/sdb: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/sdc: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_root: 12.4 GB, 12423528448 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_swap: 4227 MB, 4227858432 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data1: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data2: 1073 MB, 1073741824 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data3: 524 MB, 524288000 bytes  
Disk identifier: 0x00000000  
Disk /dev/sdd: 2147 MB, 2147483648 bytes****<==  New disks added Here*** *Disk identifier: 0x00000000  
Disk /dev/sde: 2147 MB, 2147483648 bytes****<==  New disks added Here*** *Disk identifier: 0x00000000*

**if you want to remove the discovered luns, simply logout from the target**

*[root@gurkulrhel4 ~]# iscsiadm –mode node –targetname iqn.2013-04.com.unixadminschool:target1 –portal 192.168.1.30 –logout  
Logging out of session [sid: 2, target: iqn.2013-04.com.unixadminschool:target1, portal: 192.168.1.30,3260]  
Logout of [sid: 2, target: iqn.2013-04.com.unixadminschool:target1, portal: 192.168.1.30,3260] successful.*

**Check the available disks, and you will notice new disks disappeared**

*[root@gurkulrhel4 ~]# fdisk -l|grep Disk  
Disk /dev/sda: 17.2 GB, 17179869184 bytes  
Disk identifier: 0x000ae2a8  
Disk /dev/sdb: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/sdc: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_root: 12.4 GB, 12423528448 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_swap: 4227 MB, 4227858432 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data1: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data2: 1073 MB, 1073741824 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data3: 524 MB, 524288000 bytes  
Disk identifier: 0x00000000*

**Login ( bind) back to the target to make iscsi disks available**

*[root@gurkulrhel4 ~]# iscsiadm –mode node –targetname iqn.2013-04.com.unixadminschool:target1 –portal 192.168.1.30 –login  
Logging in to [iface: default, target: iqn.2013-04.com.unixadminschool:target1, portal: 192.168.1.30,3260] (multiple)  
Login to [iface: default, target: iqn.2013-04.com.unixadminschool:target1, portal: 192.168.1.30,3260] successful.  
[root@gurkulrhel4 ~]# fdisk -l|grep Disk  
Disk /dev/sda: 17.2 GB, 17179869184 bytes  
Disk identifier: 0x000ae2a8  
Disk /dev/sdb: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/sdc: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_root: 12.4 GB, 12423528448 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-lv\_swap: 4227 MB, 4227858432 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data1: 2147 MB, 2147483648 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data2: 1073 MB, 1073741824 bytes  
Disk identifier: 0x00000000  
Disk /dev/mapper/vg\_gurkulrhel2-data3: 524 MB, 524288000 bytes  
Disk identifier: 0x00000000****Disk /dev/sdd: 2147 MB, 2147483648 bytes******Disk identifier: 0x00000000******Disk /dev/sde: 2147 MB, 2147483648 bytes******Disk identifier: 0x00000000***

**Create partition in the New disks**

*[root@gurkulrhel4 ~]# fdisk /dev/sdd  
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel  
Building a new DOS disklabel with disk identifier 0xaed24ed6.  
Changes will remain in memory only, until you decide to write them.  
After that, of course, the previous content won’t be recoverable.*

*Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)*

*WARNING: DOS-compatible mode is deprecated. It’s strongly recommended to  
switch off the mode (command ‘c’) and change display units to  
sectors (command ‘u’).*

*Command (m for help): p*

*Disk /dev/sdd: 2147 MB, 2147483648 bytes  
67 heads, 62 sectors/track, 1009 cylinders  
Units = cylinders of 4154 \* 512 = 2126848 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0xaed24ed6*

*Device Boot Start End Blocks Id System*

*Command (m for help): n  
Command action  
e extended  
p primary partition (1-4)  
p  
Partition number (1-4): 1  
First cylinder (1-1009, default 1):  
Using default value 1  
Last cylinder, +cylinders or +size{K,M,G} (1-1009, default 1009):  
Using default value 1009*

*Command (m for help): w  
The partition table has been altered!*

*Calling ioctl() to re-read partition table.  
Syncing disks.*

*[root@gurkulrhel4 ~]#  
[root@gurkulrhel4 ~]# fdisk -l /dev/sdd*

*Disk /dev/sdd: 2147 MB, 2147483648 bytes  
67 heads, 62 sectors/track, 1009 cylinders  
Units = cylinders of 4154 \* 512 = 2126848 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0xaed24ed6*

*Device Boot Start End Blocks Id System  
/dev/sdd1 1 1009 2095662 83 Linux*

*[root@gurkulrhel4 ~]# mkfs.ext4 /dev/sdd1*

*mke2fs 1.41.12 (17-May-2010)  
Filesystem label=  
OS type: Linux  
Block size=4096 (log=2)  
Fragment size=4096 (log=2)  
Stride=0 blocks, Stripe width=0 blocks  
131072 inodes, 523915 blocks  
26195 blocks (5.00%) reserved for the super user  
First data block=0  
Maximum filesystem blocks=536870912  
16 block groups  
32768 blocks per group, 32768 fragments per group  
8192 inodes per group  
Superblock backups stored on blocks:  
32768, 98304, 163840, 229376, 294912*

*Writing inode tables: done  
Creating journal (8192 blocks): done  
Writing superblocks and filesystem accounting information: done*

*This filesystem will be automatically checked every 29 mounts or  
180 days, whichever comes first. Use tune2fs -c or -i to override.*

*[root@gurkulrhel4 ~]# mkdir /shared-data*

**Add entry to /etc/fstab to mount new filesystem during the boot**

*[root@gurkulrhel4 ~]# cat /etc/fstab  
#  
# /etc/fstab  
# Created by anaconda on Fri Sep 28 07:39:26 2012  
#  
# Accessible filesystems, by reference, are maintained under ‘/dev/disk’  
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info  
#  
/dev/mapper/vg\_gurkulrhel2-lv\_root / ext4 defaults 1 1  
UUID=5e925634-5366-455d-b337-eed668f7b2e6 /boot ext4 defaults 1 2  
/dev/mapper/vg\_gurkulrhel2-lv\_swap swap swap defaults 0 0  
tmpfs /dev/shm tmpfs defaults 0 0  
devpts /dev/pts devpts gid=5,mode=620 0 0  
sysfs /sys sysfs defaults 0 0  
proc /proc proc defaults 0 0  
### below entry will mount /dev/sd1 on /shared/data  
### the option \_netdev option says to mount /shared-data only after the network available  
/dev/sdd1 /shared-data ext4 defaults,\_netdev 0 0*

*[root@gurkulrhel4 ~]#*

**Reboot the Host , and  check the filesystem getting mounted automatically after the reboot**

*[root@gurkulrhel4 ~]# reboot*

*Broadcast message from root@gurkulrhel4  
(/dev/pts/0) at 13:10 …*

*The system is going down for reboot NOW!*

*After the reboot*

*[root@gurkulrhel4 ~]# df -h  
Filesystem Size Used Avail Use% Mounted on  
/dev/mapper/vg\_gurkulrhel2-lv\_root  
12G 7.1G 3.8G 66% /  
tmpfs 1003M 88K 1003M 1% /dev/shm  
/dev/sda1 485M 37M 423M 9% /boot****/dev/sdd1 2.0G 35M 1.9G 2% /shared-data***

Device Mapper Multipath

## Installation of DM-Multipath Components:

Step 1: If not already installed, install the device-mapper-multipath package:

*# yum install device-mapper-multipath*

Step 2: Enable device mapper multipath

Generate a default /etc/multipath.conf file using the mpathconf command. To determine the default settings for the installed version use the –help option:

*# mpathconf –help  
usage: /usr/sbin/mpathconf <command>*

*Commands:  
Enable: –enable   
Disable: –disable  
Set user\_friendly\_names (Default n): –user\_friendly\_names <y|n>  
Set find\_multipaths (Default n): –find\_multipaths <y|n>  
Load the dm-multipath modules on enable (Default y): –with\_module <y|n>  
start/stop/reload multipathd (Default n): –with\_multipathd <y|n>  
chkconfig on/off multipathd (Default y): –with\_chkconfig <y|n>*

Options that default to the required value do not need to be specified on the command line.

For example, to enable multipath with the user friendly names feature leaving all other options set to the default values, issue the following command:

*# mpathconf –enable –user\_friendly\_names y*

## How to BlackList Certain Storage from Multipath

Device mapper multipath supports a number of mechanisms to allow blacklisting of devices that match certain criteria. A blacklisted device will be skipped when detecting and creating multipath devices.

One use for this mechanism is to exclude local devices (built-in disks, or removable media, for example) from the multipath configuration. This allows those devices to continue to be used as normal single-path devices without interference from the multipath subsystem.

To Blacklist some Storage Devices, Edit the blacklist section in /etc/multipath.conf and add the device section with the vendor and model indicated in /var/log/messages :

*blacklist {*

*……*

*device {  
vendor Generic  
model External  
}  
}*

then, reload multipath configuration

*# service multipathd reload*

Connect the external hard drive and check if it can be mounted.

Device mapper multipath also includes a built-in blacklist to exclude devices that do not support multiple paths. This list is preconfigured and does not need to be included in multipath.conf to be effective. The current set of built-in blacklists is as follows:

*blacklist {   
devnode “^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]\*”   
}*

Versions of device mapper multipath in Red Hat Enterprise Linux 6 and later include the **‘find\_multipaths’** feature. This excludes any device from multipath configuration until at least two paths to the device have been detected simultaneously. This is an effective alternative to manual blacklisting that requires less configuration and maintenance for common scenarios. The feature is enabled by adding the following line to the global ‘defaults’ section of multipath.conf:

*defaults {  
…  
find\_multipaths yes  
…  
}*

## Customisation to the Storage Multipath Configuration

Device mapper multipath will apply different settings based on the make (vendor) and model (product) of storage in use. The default values for any given release may be found in the file /usr/share/doc/device-mapper-multipath-\*/multipath.conf.defaults. You can use this file as template file to configure appropriate settings to /etc/multipath.conf and modifying parameters as desired.

Note: We normally don’t need any changes to this defaults file, except for those storage arrays that may be configured to operate in multiple failover modes (for instance standards-based ALUA mode or a proprietary failover mode specific to the array vendor). In this case array-specific configuration steps may be required.

* To determine the make and model of any storage device attached to the system

*check  the information provided in the /proc and /sys file systems.*

* To view all attached SCSI devices:

*# cat /proc/scsi/scsi*

 For example, a system attached to an EMC CLARiiON storage array may include entries similar to the following:

*Attached devices:  
Host: scsi11 Channel: 00 Id: 00 Lun: 00  
Vendor: DGC Model: RAID 0 Rev: 0851  
Type: Direct-Access ANSI SCSI revision: 04  
…*

* To determine vendor and model (product) strings for a specific SCSI disk device (replace sda with the device in question):

*# cat /sys/block/sda/device/vendor  
DGC*

*# cat /sys/block/sda/device/model  
RAID 5*

* Finally, Review the configuration and amend as necessary based on device configuration and vendor recommendations.

## How do we Detect and configure Multipath devices in your RHEL6

In order to detail and configure multipath devices , we need to have a multipath.conf file. Once we have the basic file we can customize the configuration with appropriate settings.

Just Run the multipath command to detect and configure multipath devices:

*# multipath  
mpatha (3600601f0d057000018fc7845f46fe011) dm-1 DGC,RAID 0  
size=98G features=’0′ hwhandler=’1 emc’ wp=rw  
|-+- policy=’round-robin 0′ prio=1 status=active  
| – 11:0:0:0 sdd 8:128 active ready running  
| – 10:0:0:0 sde 8:64 active ready running  
-+- policy=’round-robin 0′ prio=0 status=enabled  
|- 8:0:0:0 sdc 8:32 active ready running  
– 9:0:0:0 sdf 8:80 active ready running*

*The command can print additional information which may be of use in diagnosing problems or creating custom configurations. This includes information on blacklist settings and the matching of storage devices to built-in configurations. Specifying a verbose option with the -v<number> option enables this additional output:*

*May 05 16:48:56 | Found matching alias [mpatha] in bindings file.  
Setting wwid to 3600601608e661a00e4710bf88370e211  
May 05 16:48:56 | ram0: device node name blacklisted  
May 05 16:48:56 | ram1: device node name blacklisted  
May 05 16:48:56 | ram2: device node name blacklisted  
<snip>  
May 05 16:48:56 | loop5: device node name blacklisted  
May 05 16:48:56 | loop6: device node name blacklisted  
May 05 16:48:56 | loop7: device node name blacklisted  
May 05 16:48:56 | sr0: device node name blacklisted  
May 05 16:48:56 | sda: device node name blacklisted  
May 05 16:48:56 | sdc: not found in pathvec  
May 05 16:48:56 | sdc: mask = 0x3f  
May 05 16:48:56 | sdc: dev\_t = 8:32  
May 05 16:48:56 | sdc: size = 10485760  
May 05 16:48:56 | sdc: subsystem = scsi  
May 05 16:48:56 | sdc: vendor = DGC  
May 05 16:48:56 | sdc: product = RAID 5  
May 05 16:48:56 | sdc: rev = 0220  
May 05 16:48:56 | sdc: h:b:t:l = 3:0:0:1  
May 05 16:48:56 | sdc: tgt\_node\_name = 0x50060160ba601693  
May 05 16:48:56 | sdc: serial = FCNPR063600652  
May 05 16:48:56 | sdc: get\_state  
May 05 16:48:56 | loading /lib64/multipath/libcheckemc\_clariion.so checker  
…*

**Display and review the configuration with:**

*#multipath-ll  
mpatha (3600601f0d057000018fc7845f46fe011) dm-1 DGC,RAID 0  
size=98G features=’0′ hwhandler=’1 emc’ wp=rw  
|-+- policy=’round-robin 0′ prio=1 status=active  
| – 11:0:0:0 sdd 8:128 active ready running  
| – 10:0:0:0 sde 8:64 active ready running  
-+- policy=’round-robin 0′ prio=0 status=enabled  
|- 8:0:0:0 sdc 8:32 active ready running  
– 9:0:0:0 sdf 8:80 active ready running*

Redhat Enterprise Linux Networking Troubleshooting – Quick Reference

## ****1. Network does not start due to “Device not managed by Network-Manager” error.****

*Generally, this error occurs when the ethernet cable does not connect to the interface well and the interface is not configured properly.*

*The solution for this issue is to ensure the link connectivity and configure network interface.*

*Also, restart ‘network’ and ‘NetworkManager’ services.  
Diagnostic Steps*

*Ensure the status of ‘network’ and ‘NetworkManager’ services and restart them.*

*# service network status  
# service NetworkManager status*

*To restart:*

*# service network restart  
# service NetworkManager restart*

*Check if there any error messages while restarting these services.*

*If the issue does not resolve by the solutions mentioned above, please follow the instructions below.*

*Go to System->Preferences->Network Connections->Wired and ‘edit’ eth0 to make sure that ‘Connect automatically’ is enabled. If not, please enable it and restart ‘NetworkManager’ service.*

## ****2. Frame errors in network Interface – What does it mean?****

*ifconfig output shows errors related to frame field of network interface and we are noticing connectivity and performance problems.*

*Frame Errors: Shows the number of packets received incorrectly having a CRC error and a noninteger number of octets. On a LAN, this is usually the result of collisions or a malfunctioning Ethernet device.*

*Check the “frame” and “errors” field in ifconfig command output:*

*# ifconfig  
According to output below, it’s possible to see that some interfaces had errors in the transfer of packets, as can be seen in the fields: “errors” and “frame”.*

*RX packets:93849080 errors:276295 dropped:0 overruns:0 frame:276295*

*For each interface belonging to bond, enter the command below to get NIC statistics:*

*# ethtool -S ethX*

*Look at “rx\_frame\_errors:”, this line will show frame errors too and you can collect more info about a lot of other data.*

*We need to check below things :*

* *Physical devices like network cables, NICs and switch ports and configuration.*
* *Switch configuration about speed and auto-negotiation are the same as in the server.*

*For example:  
Switch was working with HALF-DUPLEX without “Autonegotiation”  
In the above cases, switch configuration need to be changed to work in FULL-DUPLEX mode to get the connection established with expected performance and eliminate frame errors.*

* *Server was connected to a switch which had an incompatible configuration to establish communication or you have a damaged NIC, cable or port.*

## ****3. Network restart throws the error “some other host already uses address”****

*Scenario :  host1 ( with 199.9.200.101 IP ) reporting the error “ifup:Error, some other host already uses address x.x.x.x”  
And then the Servers won’t rejoin network after reboot. There are no other host in the network with the same IP.*

*Network will not start after reboot, says IP address in use*

*The above error message returned by the Following section from in script /etc/sysconfig/network-scripts/ifup, which is actuallu responsible to check duplicate IPS during network start*

*if ! arping -q -c 2 -w 3 -D -I ${REALDEVICE} ${IPADDR} ; then  
echo $”Error, some other host already uses address ${IPADDR}.”  
exit 1  
fi*

*To resolve the problem, we should run a packet sniffer such as tcpdump on the system would help to know about the source MAC address of such system/device. Then we should be able to match this ‘MAC’ address with the switch MAC table and find out which system/device is connected to that port.*

*# tcpdump -i any -w /tmp/tcpdump.pcap*

*::::::  output truncated ::::::*

*Packet 25#  26.820528 xx:xx:xx:xx:xx:xx ARP Who has 199.9.200.101 ? Tell 0.0.0.0*

*Packer 26 # 26.820533 xx:xx:xx:xx:xx:xx ARP Who has 199.9.200.101 ? Tell 0.0.0.0*

*Above packets are arp requests sent from the system (with MAC xx:xx:xx:xx:xx:xx ) asking who is having the ip address 192.168.0.1.*

*Packet 27#  26.820633 Intel\_yy:yy:yy ARP 199.9.200.101 is at yy:yy:yy:yy:yy:yy*

*Packet27 is a ARP reply from a Cisco device ( having MAC address of yy:yy:yy:yy:yy:yy ) saying 199.9.200.101 ip is at yy:yy:yy:yy:yy:yy , which is its own MAC.*

## ****4. How to map the physical network interface corresponds to which ethX device?****

*There are several ways of identifying network interfaces.*

*A Media Access Control (MAC) address, or hardware address, is unique to each interface. This can be used to identify an interface.*

*# ifconfig -a | grep HWaddr  
eth0 Link encap:Ethernet HWaddr 00:16:3E:7B:8B:49  
eth1 Link encap:Ethernet HWaddr 00:16:3E:7C:9B:49*

*The command ethtool’s -p option initiates adapter-specific action intended to enable an operator to easily identify the adapter by sight. Typically this involves blinking one or more LEDs on the specific Ethernet port. Note that not all network interfaces support this.*

*# ethtool -p eth0*

*The command ethtool can be used to determine if an interface is connected to an active device (it detects a link/carrier). If a link is not detected then the interface is not connected to the network.*

*# ethtool eth0 | grep Link  
Link detected: yes*

## ****5. How do I add/delete additional Logical IP addresses to a network interface in RHEL6?****

*Since NetworkManager is usually disabled in server environments, either we can configure the temporary IP’s using the IP command or create them persistently by modifying the configuration files.*

### *Adding or removing IP addresses manually:*

*Use the ip command provided by the iproute package.*

*To dislplay the existing IP addresses in a network interface:  
# ip addr show eth0*

*2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000  
link/ether 52:54:00:71:98:9d brd ff:ff:ff:ff:ff:ff  
inet 192.168.122.101/24 brd 192.168.122.255 scope global eth0  
inet 192.168.122.12/24 scope global secondary eth0  
inet 192.168.122.11/24 scope global secondary eth0  
inet 192.168.122.13/24 scope global secondary eth0  
inet6 fe80::5054:ff:fe71:989d/64 scope link  
valid\_lft forever preferred\_lft forever*

### *To delete an existing IP address:*

*# ip addr del 192.168.122.13/24 dev eth0  
# ip addr show eth0  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000  
link/ether 52:54:00:71:98:9d brd ff:ff:ff:ff:ff:ff  
inet 192.168.122.101/24 brd 192.168.122.255 scope global eth0  
inet 192.168.122.12/24 scope global secondary eth0  
inet 192.168.122.11/24 scope global secondary eth0  
inet6 fe80::5054:ff:fe71:989d/64 scope link  
valid\_lft forever preferred\_lft forever*

### *To add an IP address:*

*# ip addr add 192.168.122.13/24 dev eth0  
# ip addr show eth0  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000  
link/ether 52:54:00:71:98:9d brd ff:ff:ff:ff:ff:ff  
inet 192.168.122.101/24 brd 192.168.122.255 scope global eth0  
inet 192.168.122.12/24 scope global secondary eth0  
inet 192.168.122.11/24 scope global secondary eth0  
inet 192.168.122.13/24 scope global secondary eth0  
inet6 fe80::5054:ff:fe71:989d/64 scope link  
valid\_lft forever preferred\_lft forever*

### *Configuring additional IP addresses persistently:*

*Edit the corresponding /etc/sysconfig/network-scripts/ifcfg-eth<x> configuration file by using the following additional entries:*

*IPADDR<n>: the additional IP address.  
PREFIX<n>: the length in bits of the netmask for the additional IP address.   
NETMASK<n>: the explicit netmask value for the additional IP address.  
BROADCAST<n>: the broadcast address for the additional IP address.*

*and add as many additional IPADDR<n> and PREFIX<n> entries as additional IP addresses are required.*

*For example the following configuration file:*

*# cat /etc/sysconfig/network-scripts/ifcfg-eth1  
DEVICE=eth1  
BOOTPROTO=none  
NETMASK=255.255.255.0  
TYPE=Ethernet  
HWADDR=52:54:00:cc:de:0b  
IPADDR=192.168.100.101  
PREFIX=24  
IPADDR2=192.168.128.101  
PREFIX2=24  
IPADDR3=192.168.130.101  
PREFIX3=28*

*would give the following result:*

*# ip addr show eth1  
3: eth1: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000  
link/ether 52:54:00:cc:de:0b brd ff:ff:ff:ff:ff:ff  
inet 192.168.100.101/24 brd 192.168.100.255 scope global eth1  
inet 192.168.128.101/24 brd 192.168.128.255 scope global eth1  
inet 192.168.130.101/28 brd 192.168.130.111 scope global eth1  
inet6 fe80::5054:ff:fecc:de0b/64 scope link  
valid\_lft forever preferred\_lft forever*

## ****6. Network interface does not come up after reboot; must be manually brought up with ifup****

*Ensure the interface config file (for example, /etc/sysconfig/network-scripts/ifcfg-ethX) contains:*

*:::*

*ONBOOT=yes*

*::::*

*Furthermore, ensure that either the network service or the NetworkManager service are enabled.*

*# chkconfig network on*

## ****7. How do I run a script or program immediately after my network interface goes up?****

*The network control scripts in /etc/sysconfig/network-scripts allows for adding scripts that will run after the device is up.  
Notice the last few lines search for a file called /sbin/ifup-local. If that file is found, it is executed.*

*For example, looking at the /etc/sysconfig/network-scripts/ifup file, the last few lines are:*

*:::::::::::: snip  :::::::::::::*

*if [ “${IPX}” = yes ]; then  
/etc/sysconfig/network-scripts/ifup-ipx ${DEVICE}  
fi*

*exec /etc/sysconfig/network-scripts/ifup-post ${CONFIG} ${2}*

*A script /etc/sysconfig/network-scripts/ifup-post is the last item to be executed. Looking at that script:*

*if [ -x /sbin/ifup-local ]; then  
/sbin/ifup-local ${DEVICE}  
fi*

*exit 0*

*From the above code we can identify the file /sbin/ifup-local is the one that is available to run programs immediately after the interface is up.*

*For Example, create file /sbin/ifup-local and add the commands to run after the network is up*

*————————————-  
#!/bin/bash  
if [ “$1” == “eth0” ]; then  
/sbin/ethtool -G $1 rx 4096 tx 4096  
fi  
————————————-*

*Make /sbin/ifup-local executable*

*# chmod +x /sbin/ifup-local*

## ****8. How to Configure more than one default gateway for a multi-home system ( i.e. had two or more separate Network Interface Cards)?****

*Scenario : Each network card goes to a separate network (ie. eth0 goes to external network, eth1 goes to internal).Each interface needs to access the network gateway to gain access to other networks.*

*It is not logically possible to have more than one default gateway. A default gateway is more accurately described as a “catch-all” gateway*

*A single desired default GATEWAY= address should be configured in the /etc/sysconfig/network file. All other references to the GATEWAY= directive should be removed from ifcfg-eth\* files. Additionally, If you wish to specify which NIC device is used to carry gateway traffic, the GATEWAYDEV=ethX directive should be used.*

*If more than one default gateway is configured, network packets may be routed incorrectly, leading to unpredictable behaviour. Due to the way in which initscripts functions, when the GATEWAY= directive is defined in multiple ifcfg-eth\* network scripts, whichever device is brought up last is the one which will define the gateway used by the system.*

*Alternatively, if we can create a static route from the each interface to pass the specific network related traffic.*

*For Example: If we have a system with 3 ethernet interfaces(eth0, eth1 and eth2), and want to configure a single default gateway as 10.0.0.1 and gatewaydev as eth1.*

*# cat /etc/sysconfig/network  
NETWORKING=yes  
NETWORKING\_IPV6=yes  
HOSTNAME=localhost  
GATEWAY=10.0.0.1  
GATEWAYDEV=eth1  
# cat /etc/sysconfig/network-scripts/ifcfg-eth0  
DEVICE=eth0  
BOOTPROTO=static  
HWADDR=A1:B2:C3:D4:E5:F6  
IPADDR=192.168.0.55  
NETMASK=255.255.255.0  
NETWORK=192.168.0.0  
BROADCAST=192.168.0.255  
ONBOOT=yes  
# cat /etc/sysconfig/network-scripts/ifcfg-eth1  
DEVICE=eth1  
BOOTPROTO=static  
HWADDR=A1:B2:C3:D4:E5:E5  
IPADDR=10.0.0.66  
NETMASK=255.255.255.0  
NETWORK=10.0.0.0  
BROADCAST=10.0.0.255  
ONBOOT=yes  
# cat /etc/sysconfig/network-scripts/ifcfg-eth2  
DEVICE=eth2  
BOOTPROTO=static  
HWADDR=A1:B2:C3:D4:E5:E4  
IPADDR=10.0.0.66  
NETMASK=255.255.255.0  
NETWORK=10.0.0.0  
BROADCAST=10.0.0.255  
ONBOOT=yes*

*Note that none of the above ifcfg-eth\* files contain the GATEWAY= directive.*

*On all other interfaces which must use a separate gateway, create static routes to all networks requiring them.*

*Below shows a system, that when configured correctly as shown above, will show the gateway being used as 10.0.0.1 on device eth1.*

*$ route -n  
Kernel IP routing table  
Destination Gateway Genmask Flags Metric Ref Use Iface  
192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0  
10.0.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1  
10.0.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth2  
0.0.0.0 10.0.0.1 0.0.0.0 UG 0 0 0 eth1*

## ****9. What is the “ethtool” command and how can I use it to obtain information about my network devices and interfaces?****

*The ethtool command is a low-level interface manipulation tool with the power to directly effect firmware and driver behavior and functionality on the active interface.*

*Before performing any tasks it is recommend to review the network topology involved to ensure all hardware directly supports any task you are enabling, such as is the case with auto-negotiation or jumbo frames. Please note that each driver will behave differently and have different supported operations while in use, however the ethtool generic commands should be fully compatible across different driver varieties.*

*Most functions with ethtool can be performed live. Data collection has little to no impact on the running system, and direct manipulation of functions such as the ring buffer or the enabling or disabling of checksumming can typically be done live without any interrupt in service.*

*The ethtool facility, with no commands run, prints basic information about the Ethernet card for the interface in which it is run:*

*# ethtool eth0  
Settings for eth0:  
Supported ports: [ TP ]  
Supported link modes: 10baseT/Half 10baseT/Full  
100baseT/Half 100baseT/Full  
1000baseT/Full  
Supports auto-negotiation: Yes  
Advertised link modes: 10baseT/Half 10baseT/Full  
100baseT/Half 100baseT/Full  
1000baseT/Full  
[…]*

*If you suspect packet loss or if the output of a utility such as # ifconfig shows drops, overruns, or errors, you have the ability to directly poll the card for packet statistics at boot. Please keep in mind that this data is only current since the last functioning operational state of the Ethernet controller, typically cleared at the reboot of a system.*

*# ethtool -S eth0  
NIC statistics:  
rx\_packets: 2251518  
tx\_packets: 1410087  
rx\_bytes: 1295872987  
tx\_bytes: 288388610  
rx\_broadcast: 409192  
tx\_broadcast: 11  
rx\_multicast: 34163  
tx\_multicast: 123  
rx\_errors: 0  
tx\_errors: 0  
tx\_dropped: 0  
multicast: 34163  
[…]*

*Note: keep in mind these parameters are defined by the Ethernet controller, firmware, and driver that is specific to your Ethernet card, and the values contained in such may be calculated radically differently on alternative hardware. For assistance with this, please contact Red Hat Support or the driver vendor for an analysis.*

*Some other various data collection functions of ethtool consist of:*

*# ethtool -a eth\*  
=> shows pause parameters for interface listed\**

*# ethtool -g eth\*  
=> shows ring buffer parameters for interface listed\**

*# ethtool -i eth\*  
=> shows driver and firmware versions, as well as physical bus address for interface listed\**

*# ethtool -k eth\*  
=> shows offload information for functions such as tcp checksumming for interface listed\**

*The ethtool utility can also be used to make non-persistant operational changes to the working interface, typically replacing the data-collection flag with the capital letter flag of the same type. For example, to change the ring buffer on an active interface without the need to bring the interface down, we first view the available options for the buffer:*

*# ethtool -g eth0  
Ring parameters for eth0:  
Pre-set maximums:  
RX: 4096 <–Maximum size for the RX(receive) ring buffer  
RX Mini: 0  
RX Jumbo: 0  
TX: 4096  
Current hardware settings:  
RX: 256 <–Current value, which can be set to the maximum.  
RX Mini: 0  
RX Jumbo: 0  
TX: 256*

*And then we can use the ethtool command to make the change interactively, without the need to take down the interface:*

*# ethtool -G eth0 rx 4096*

*Our interface should then be set to maximum value on the ring buffer. Please note this is a low level change on the interface, and as such, will not persist over reboot. To ensure any changes remain persistent, either specify them in the /etc/rc.local script which is run every time the system reboots, or in the /etc/sysconfig/network-scripts/ifcfg-eth\* file specify the function you desire performed with the “ETHTOOL\_OPTS” tag:*

*# cat /etc/sysconfig/network-scripts/ifcfg-eth0  
DEVICE=”eth0″  
HWADDR=”00:00:00:00:00″  
NM\_CONTROLLED=”no”  
ONBOOT=”yes”  
ETHTOOL\_OPTS=”speed 100 duplex full autoneg off”*

## ****10.  How do I set up VLAN (802.1q) tagging on a network interface ?.****

*1. Ensure that the module is loaded:*

*# lsmod | grep 8021q*

*2. If the module is not loaded, load it with the following command:*

*# modprobe 8021q*

*3. Configure your physical interface in /etc/sysconfig/network-scripts/ifcfg-ethX, where X = the interface number:*

*DEVICE=ethX  
TYPE=Ethernet  
BOOTPROTO=none  
ONBOOT=yes*

*4. Configure the VLAN interface configuration in /etc/sysconfig/network-scripts.*

*The configuration filename should be the physical interface plus a ‘.’ character plus the VLAN ID number.  
For example, if the VLAN ID is 192, and the physical interface is eth0, then the configuration filename should be ifcfg-eth0.192:*

*DEVICE=ethX.192  
BOOTPROTO=static  
ONBOOT=yes  
IPADDR=192.168.1.1  
NETMASK=255.255.255.0  
USERCTL=no  
NETWORK=192.168.1.0  
VLAN=yes*

*If there is a need to configure second vlan (ID 193) on the same interface (eth0), add a new file with the name eth0.193 with the vlan configuration details.*

*5. Restart networking:*

*#service network restart*

## ****11. How can I route network traffic such that the packets go out via the same interface they came in?****

*Scenario: Example : server has 2 interfaces with IP address.*

* *eth0 – > inet addr:10.66.1.51 Bcast:10.66.255.255*
* *eth1 – > inet addr:10.67.1.51 Bcast:10.67.255.255*

*We want to forward network packets to the same network interface from where it came from. We can implement this feature by using policy-based routing (source routing). The iproute2 package provides the tools (/bin/ip) to configure this.*

*Setup 2 more route table with different table ids; 10.66.255.254 is the gateway for eth0 and 10.67.255.254 is the gateway for eth1.*

*# ip route add default via 10.66.255.254 dev eth0 table 1  
# ip route add default via 10.67.255.254 dev eth1 table 2*

*Above command creates 2 routing table. Create rules to forward all the packets entering via a particular nic to go out the appropriate routing table.*

*# ip rule add iif eth0 table 1  
# ip rule add iif eth1 table 2*

*Create rules to forward packets to go out the specific routing table.*

*# ip rule add from 10.66.1.51 table 1  
# ip rule add from 10.67.1.51 table 2*

*If you want to add more routing records, following is the command format*

*# ip route add to 192.168.100.0 via 10.66.0.203 dev eth0 table 1*

*Add the above commands into scripts file and make it run on next boot.*

## ****12. Setting persistent network interfaces using udev in Red Hat Enterprise Linux 6****

*Scenario : When the system boots network interfaces are sometimes reordered causing the interfaces to be named differently. During a system boot, the system orders network interfaces as they come up.Once the system is booted, other network dependent tasks can fail because interfaces are not named properly.*

*The DEVICE= option in the ifcfg file does not determine the mapping of subchannels to network device names.*

*Instead, the udev rules file /etc/udev/rules.d/70-persistent-net.rules determines which network device channel gets which network device name.If network dependent services such as bonding begin to act differently, ensure interface details are matching their names by using ifconfig*

*To resolve the isssue, we can create Persistent network interfaces which are  bound through the creation of udev rules.*

*First, configure the interfaces correctly, ensuring they are in the correct order.*

*Then the rules can be created by running below command:*

*Check the current Udev rules:*

*# cat /etc/udev/rules.d/70-persistent-net.rules*

*# echo add > /sys/class/net/ethX/uevent*

*run this command with all the interface in use, and replace the X with the interface number.*

*Verify the udev rules have been added by checking the rules file:*

*# cat /etc/udev/rules.d/70-persistent-net.rules*

*Which should output something similar to:*

*# PCI device 0x1af4:0x1000 (virtio-pci) (custom name provided by external tool) <br />SUBSYSTEM==”net”, ACTION==”add”, DRIVERS==”?*

*The restart the network:*

*# service network restart*

## ****13. Network interface failes due to the delay in bringing up the link from switch side.****

*Scenario : After a server boots up, a DHCP based network interface is not working and has no IP address The following error may be found in /var/log/dmesg:*

ADDRCONF(NETDEV\_UP): eth0: link is not ready

*Some switches do not bring up link quickly, so the network link layer is not available at the moment that the DHCP client attempts to retrieve an IP address. This can also occur in a peer-to-peer network, ie, when connecting two systems together directly via network interface. It has also been observed that some network interface cards are themselves slow to finish link negotiation during initialization post boot, leading to link not being available at the right moment.*

*To Workaround the problem, from the server side, we can increase the LINKDELAY value ( default is 10 seconds configured in /etc/sysconfig/network-scripts/network-functions)*

*Add below directive to /etc/sysconfig/network-scripts/ifcfg-ethX and tune for the most effective value:*

*LINKDELAY=60*

*The restart the interface manually*

*# ifdown ethX   
# ifup ethX*

## ****14. How to configure network interfaces, after fresh installation.They are not available after reboot?****

*On the first boot the interfaces are configured to not be enabled on boot. To allow the interfaces up on boot, you need to edit the configuration bellow:*

*1.Identify if your interface was named with ’em’ or ‘eth’ nomenclature:*

*# ls /etc/sysconfig/network-scripts/*

*2.Edit the file:*

*# vim ifcfg-emX  
or  
# vim ifcfg-ethX*

*3.Change the parameter ‘ONBOOT’ from ‘no’ to ‘yes’:*

*…  
ONBOOT=yes  
…*

*4.Restart the network service:*

*# service network restart*

*5.Verify if the interfaces were recognized:*

*#ifconfig  
eth0 Link encap:Ethernet HWaddr 02:00:00:00:00:01  
inet addr:192.168.70.87 Bcast:192.168.70.255 Mask:255.255.255.0  
inet6 addr: fe80::ff:fe00:1/64 Scope:Link  
UP BROADCAST RUNNING NOARP MULTICAST MTU:1492 Metric:1  
RX packets:23 errors:0 dropped:0 overruns:0 frame:0  
TX packets:3 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:644 (644.0 b) TX bytes:264 (264.0 b)*

*lo Link encap:Local Loopback   
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:16436 Metric:1  
RX packets:16204 errors:0 dropped:0 overruns:0 frame:0  
TX packets:16204 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:91855053 (87.5 MiB) TX bytes:91855053 (87.5 MiB)*

## ****15. Why does network interface show “rx\_errors” in ifconfig output or what is rx\_crc\_errors ?****

*Scenario : Ifconfig output of network interface ethX , shows following errors :*

*#ifconfig  
eth0 Link encap:Ethernet HWaddr 00:26:55:7B:FB:40  
inet addr:172.21.141.28 Bcast:172.21.141.255 Mask:255.255.255.0  
inet6 addr: fe80::226:55ff:fe7b:fb40/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:161417251 errors:5068645 dropped:0 overruns:0 frame:5068645*

*From “ethtool -S eth0” output, these rx\_errors are basically “rx\_crc\_errors: 5068645”.*

*$ ethtool -S eth0  
….  
rx\_crc\_errors: 5068645  
…*

*Normally CRC errors indicate a problem at the layer 1, To Troubleshoot the problem :*

* *Connect interface to a different port.*
* *Change the cable.*
* *Use a different NIC if available.*
* *Set DUPLEX=FULL SPEED=1000 AUTONEG=ON as per the capability of NIC.*

## ****16. How to calculate right Prefix for the Netmask to configure in the ifcfg-eth\* files, usin ipcalc?****

*Find the subnetmask from the prefix value:*

*[root@rhel5 networking]# $ ipcalc 192.168.10.0/24  
Address: 192.168.10.0 11000000.10101000.00001010. 00000000  
Netmask: 255.255.255.0 = 24 11111111.11111111.11111111. 00000000  
Wildcard: 0.0.0.255 00000000.00000000.00000000. 11111111  
Network: 192.168.10.0/24 11000000.10101000.00001010. 00000000  
HostMin: 192.168.10.1 11000000.10101000.00001010. 00000001   
HostMax: 192.168.10.254 11000000.10101000.00001010. 11111110  
Broadcast: 192.168.10.255 11000000.10101000.00001010. 11111111  
Hosts/Net: 254 Class C, Private Internet*

*Find the Prefix from the subnetmask and broadcast address value:*

*[root@rhel5 networking]# ipcalc -m -b 10.18.1.255 255.255.254.0 -n -p  
NETMASK=255.255.254.0  
PREFIX=23  
BROADCAST=10.18.1.255  
NETWORK=10.18.0.0*

## ****17. Connecting 2 network interfaces on the same subnet.****

*Scenario : When there are 2 interfaces on the same subnet there is no assurance as to which interface will be used to transmit traffic and the machine will accept traffic for either IP on either interface. This is because in Linux the IP address belongs to the host and is not associted with the interface.*

*Also, if you ping with “-I<dev>”, attempting to use a given interface, there is no guarantee the reply packet (if there even is one) will come back to the same interface.*

*/—- NIC 1 –> 10.64.208.180 –>*

*Linux — |                                                     – —-> Destination Hosts*

*—- NIC 2 –> 10.64.208.208 –> /*

*To Configure this setup, we have to add tables binding source IP address for each route and add those as default gateway for each network interface.   
In this setting, we would need application having capability to handle each network devices with balanced loads by itself.*

*Assuming these networking enviroment :*

*+——————————————+  
|                             Linux                              |  
| eth0                                                   eth1 |  
| 10.64.208.180       10.64.208.208 |  
+——————————————+*

*# ip addr show*

*1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 16436 qdisc noqueue state UNKNOWN  
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
inet 127.0.0.1/8 scope host lo  
inet6 ::1/128 scope host  
valid\_lft forever preferred\_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000  
link/ether 00:16:3e:74:8d:aa brd ff:ff:ff:ff:ff:ff  
inet 10.64.208.180/24 brd 10.65.211.255 scope global eth0  
inet6 fe80::216:3eff:fe74:8daa/64 scope link  
valid\_lft forever preferred\_lft forever  
3: eth1: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000  
link/ether 00:16:3e:74:8d:b2 brd ff:ff:ff:ff:ff:ff  
inet 10.64.208.208/24 brd 10.65.211.255 scope global eth1  
inet6 fe80::216:3eff:fe74:8db2/64 scope link  
valid\_lft forever preferred\_lft forever*

### *Configuring Policy based IP routing:*

*Step 1. Add tables in /etc/iproute2/rt\_tables*

*# vim /etc/iproute2/rt\_tables  
100 t1  
101 t2*

*Step 2. Add routing table to t1, t2 :*

*# ip route add 10.64.208.0/24 dev eth0 src 10.64.208.180 table t1  
# ip route add table t1 default via 10.64.208.254 dev eth0  
# ip route show table t1  
10.64.208.0 dev eth0 scope link src 10.64.208.180  
default via 10.64.208.254 dev eth0*

*# ip route add 10.64.208.0/24 dev eth1 src 10.64.208.208 table t2  
# ip route add table t2 default via 10.64.208.254 dev eth1  
# ip route show table t2  
10.64.208.0 dev eth1 scope link src 10.64.208.208  
default via 10.64.208.254 dev eth0*

*To make the Changes done in Step 1) and 2) set these in the file :*

*Below entries in File : ifcfg-eth0*

*DEVICE=eth0  
BOOTPROTO=none  
ONBOOT=yes  
NETMASK=255.0.0.0  
IPADDR=10.64.208.180  
GATEWAY=10.64.208.254  
TYPE=Ethernet*

*Below entries in File : ifcfg-eth1*

*DEVICE=eth1  
BOOTPROTO=none  
ONBOOT=yes  
NETMASK=255.0.0.0  
IPADDR=10.64.208.208  
GATEWAY=10.64.208.254  
TYPE=Ethernet*

*Below entries in File : /etc/sysconfig/network-scripts/route-eth0*

*10.0.0.0/8 dev eth0 src 10.64.208.180 table t1  
default via 10.64.208.254 dev eth0 table t1*

*Below entries in File : /etc/sysconfig/network-scripts/route-eth1*

*10.0.0.0/8 dev eth1 src 10.64.208.208 table t2  
default via 10.64.208.254 dev eth1 table t2*

*Step 3. Setting table*

*# ip rule add table t1 from 10.64.208.180  
# ip rule add table t2 from 10.64.208.208  
# ip route show  
10.64.208.0/24 dev eth0 proto kernel scope link src 10.64.208.180  
10.64.208.0/24 dev eth1 proto kernel scope link src 10.64.208.208  
169.254.0.0/16 dev eth1 scope link  
default via 10.64.208.254 dev eth0*

*To make the Changes done in Step 1 and 2 , persistent, set these in the below files, as shown :*

*Below entry in File : /etc/sysconfig/network-scripts/rule-eth0*

*table t1 from 10.64.208.180*

*Below entry in File : /etc/sysconfig/network-scripts/rule-eth1*

*table t2 from 10.64.208.208*

*4. Set both of interfaces ready for receiving reply.*

*# sysctl net.ipv4.conf.default.arp\_filter=1*

*To make thise Step 4 changes , persistent, add entries to the configuration files as shown below*

*Add below entry in the file : /etc/sysctl.conf*

*net.ipv4.conf.all.arp\_filter = 1*

*5. Checking ping with ‘-I <IPADDR>’*

*# ping -I 10.64.208.180 <DSTADDR>*

## ****18. How to enable jumbo frames for network interfaces in Red Hat Enterprise Linux?****

*To enable jumbo frames, first ensure the device driver supports custom MTU sizes. Edit the network configuration script of the relevant interface .*

*For example, /etc/sysconfig/network-scripts/ifcfg-eth0. Add a line similar to the following which specifies the size of the frame in bytes:*

*MTU=9000*

*If the change is made while the interface is active, it will need to be restarted (brought down then up) for the new maximum frame size to take effect.*

*The networking stack in Red Hat Enterprise Linux 4 and above supports custom maximum frame sizes (Maximum Transmission Units, or MTUs) for Ethernet network interfaces.*

*It is important to realize that for large frame sizes to be effective, every intermediate network device between the sender and receiver must support large frames. If one or more devices does not support large frames then network performance can be severely adversely affected.*

*Another factor to consider is that the Linux kernel is significantly faster at making allocations of two pages of memory (8192 bytes) than three pages. Since three pages would be necessary to contain a frame of size 9000 bytes, better performance might be achieved in some scenarios by setting an MTU of around 8000 bytes.*

## ****19. Loopback interface lo deleted with system-config-network****

*The loopback interface was deleted using the system-config-network GUI.  After a reboot The loopback interface (lo) is renamed by the system with dev386 (or any other random number).*

*Below steps will recover the the loopback interface (lo) renamed with the correct entry on your system:*

*Rename dev386 in lo:*

*# ip link set dev386 name lo*

*Add in the file /etc/modprobe.conf the following line for persistents at reboot :*

*alias lo tg3  
check if the file /etc/sysconfig/network-scripts/ifcfg-lo exist, If not please create it under this location with this content:  
DEVICE=lo  
IPADDR=127.0.0.1  
NETMASK=255.0.0.0  
BROADCAST=127.255.255.255  
ONBOOT=yes  
NAME=loopback*

*Enable the lo interface:*

*# ifup lo*

Linux Admin Reference : Yum Quick Reference

1. YUM BASIC usage ( as root)   
2. Configuring Yum / up2date repository to use locally mounted DVD  
3. Configure Customized repository, consists only the packages tested for your environment  
4. Configuring YUM to install the packages from the RHN subscribed Channels  
5. Configuring a third-party repository ( Non-Redhat)  
6. Configure Yum to list only the packages available in my local repository  
7. How to install all related packages together using YUM?  
8. How to Exclude Kernel related packages to automatically update, while updating my linux server using yum?  
9. How do fix yum errors Like ” is duplicate of ?  
10. Understanding YUM plugins

## ****1.  YUM BASIC usage ( as root) :****

*1) Install a package                            :        yum install package*

*Example:     # yum install httpd*

*2) Remove a package                      :        yum remove package*

*Example:     # yum remove httpd*

*3) Update a package                        :         yum update package*

*Example:     # yum update httpd*

*4) Search for a package                 :         yum search package*

*Example:     # yum search httpd*

*5) Find information about a package:            yum info package*

*Example:     # yum info httpd*

*6) List packages containing a certain term:    yum list term*

*Example:    # yum list httpd*

*7) Find what package provides a particular file:  yum whatprovides ‘path/filename’*

*Example:     # yum whatprovides ‘etc/httpd.conf’*

*8) Update all installed packages with kernel package :     yum -y update*

*Example:        # yum -y update*

*9) To update a specific package:      yum -y update*

*Example:     #  yum -y update openssh-server*

## ****2. Configuring  Yum / up2date repository to use locally mounted DVD****

### Procedure for RHEL4

*In /etc/sysconfig/rhn/sources, comment out the following line:*

*# up2date default*

*This line needs to be commented out otherwise up2date will keep on looking for information using the Red Hat Network first. Next, in the same file, create a line like the following:*

*dir mydvdrepo /media/cdrom/RedHat/RPMS*

Note:*The CD/DVD must be mounted and available in the directory /media/cdrom/ and the path must point to the directory containing the RPMS.*

### Procedure for RHEL 5

( Note: Please make sure YUM package version is yum-3.2.8-9 or higher. Earlier Versions have bugs )

*Create a dvd.repo text file in /etc/yum.repos.d/with the following content :*

*[dvd]  
mediaid=1170972069.396645\*  
name=DVD for RHEL5  
baseurl=file:///media/RHEL\_5%20i386%20DVD/Server  
enabled=1  
gpgcheck=0*

*[dvd-cluster]  
mediaid=1170972069.396645  
name=DVD for RHEL5 – Cluster  
baseurl=file:///media/RHEL\_5%20i386%20DVD/Cluster  
enabled=1  
gpgcheck=0*

*[dvd-cluster-storage]  
mediaid=1170972069.396645  
name=DVD for RHEL5 – ClusterStorage  
baseurl=file:///media/RHEL\_5%20i386%20DVD/ClusterStorage  
enabled=1  
gpgcheck=0*

*[dvd-vt]  
mediaid=1170972069.396645  
name=DVD for RHEL5 – VT  
baseurl=file:///media/RHEL\_5%20i386%20DVD/VT  
enabled=1  
gpgcheck=0*

Note 1 : DVD repo only supports only base rpm packages under the /server directory, and the remaining repo will support the additional packages from the DVD.

Note 2:  For example for RHEL 5.5 version baseurl line should look like baseurl=file:///media/RHEL\_5.5%20i386%20DVD/Server’

Note 3: The **mediaid=** value comes from the **.discinfo** file located in the root of the DVD.

### Procedure for RHEL 6

*Mount RHEL 6 ISO to /media/rhel6*

*# mount -o loop RHEL6.0-20101111.0-Server-x86\_64-DVD.iso /media/rhel6*

*Copy media.repo file from the RHEL 6 ISO to /etc/yum.repos.d/*

*# cp /media/rhel6/media.repo /etc/yum.repos.d/rhel6.repo*

*Configure the repo file to point to the /media/rhel6*

*# vi /etc/yum.repos.d/rhel6.repo*

*Add following line*

*baseurl=file:///media/rhel6/Server*

*Ifyou want to configure repository for HighAvailability , LoadBalancer , ResilientStorage and ScalableFileSystem then you can add following lines in rhel6.repo file*

*[dvd-HighAvailability]  
mediaid=1289489697.332694  
name=DVD for RHEL6 – HighAvailability  
baseurl=file:///media/rhel6/HighAvailability  
enabled=1  
gpgcheck=0*

*[dvd-LoadBalancer]  
mediaid=1289489697.332694  
name=DVD for RHEL6 – LoadBalancer  
baseurl=file:///media/rhel6/LoadBalancer  
enabled=1  
gpgcheck=0*

*[dvd-ResilientStorage]  
mediaid=1289489697.332694  
name=DVD for RHEL6 – ResilientStorage  
baseurl=file:///media/rhel6/ResilientStorage  
enabled=1  
gpgcheck=0*

*[dvd-ScalableFileSystem]  
mediaid=1289489697.332694  
name=DVD for RHEL6 – ScalableFileSystem  
baseurl=file:///media/rhel6/ScalableFileSystem  
enabled=1  
gpgcheck=0*

## ****3. Configure Customized repository, consists only the packages tested for your environment****

There are two methods for doing this:

Method 1 :

*1. Create the folder that will hold the repository:*

*# mkdir -p /usr/share/repository*

*2. Create the RPM repository, and copy all the RPMs you want to use into that directory:*

*# cd /usr/share/repository  
# createrepo .  
# yum clean all  
Note: The createrepo package needs to be installed on the system in order to run the following commands.*

*3. Create the file /etc/yum.repos.d/file.repo as follows:*

*# cat /etc/yum.repos.d/file.repo  
[RHEL\_Repository]  
baseurl=file:///usr/share/repository  
enabled=1*

*4. Show the package list:*

*# yum list*

*5. Now you can install any packages you want, for example:*

*# yum install httpd*

Method 2  : There is another way to build the repository without copying all the RPMs to disk. In the ISO file there are repo data directories, and you can use these directly.

*1. Mount the ISO file:*

*# mount -o loop,ro rhel-X-server-i386-dvd.iso /mnt/iso*

*2. Create the file /etc/yum.repos.d/file.repo:*

*# cat /etc/yum.repos.d/file.repo  
[RHEL\_Server\_Repository]  
baseurl=file:///mnt/iso/Server  
enabled=1*

*[RHEL\_VT\_Repository]  
baseurl=file:///mnt/iso/VT  
enabled=1*

## ****4. Configuring YUM to install the packages from the RHN subscribed Channels****

 To download software, updates, and security errata, your system should be registered to RHN Classic.

### ****Step1 : Register System to RHN****

 We have two commands  to register the system in RHN network so the system can get updates from RHN.

#### Method 1:

Using  an Interactive command ,  rhn\_register, where the registration program will ask the details one by one .

*We should have  /etc/sysconfig/rhn/up2date file configured, before we use rhn\_register command*

*# vi /etc/sysconfig/rhn/up2date   
serverURL=https://xmlrpc.rhn.redhat.com/XMLRPC   
sslCACert=/usr/share/rhn/RHNS-CA-CERT*

*Note: we can get /usr/share/rhn/RHNS-CA-CERT file by installing “****rhn\_client\_tools****” package*

*Then use the below command to go through an interactive process , to register the system into RHN*

*#rhn\_register*

#### Method 2.

Using a non-interactive command, rhnreg\_ks , which allow us to key-in all the related information with single command. And mostly, we use this command to register the system to RHN using an activation key that we have already created as given below

1. *Log in to your RHN account or Satellite server as a user with administrative privileges*
2. *Go to Subscriptions -> Classic Management -> Registered Systems -> Activation Keys -> Create New Key*
3. *Fill in the Description field with something meaningful (like “marketing,” or “developer”)*
4. *Set the Usage Limit if you wish to enforce a maximum number of systems that can use this key*
5. *Set Add-On Entitlements to “Provisioning” or “Management” depending on the needs of this system*
6. *Set Universal Default. Universal Default means that this key will be used for any system registered to this account that does not specify a key. Set this to “yes” if you want this key to be the default key; set this to “no” if you want this key to be used only when specified*

*Example :*

*# rhnreg\_ks –profilename=gurkulindia\_rhel\_system –activationkey=1-dfasdf345345sdfasdfdas275c93*

*or*

*# rhnreg\_ks –profilename=gurkulindia\_rhel\_system –username=gurkuluser –password=abc1234*

*Note: if you rhnreg\_ks with user and password information, that information might be stored in the shell history.So don’t forget clear the history using history -c ( it will clear entire system history)*

### ****STEP 2: Installing/Updating Packages using RHN****

*For RHEL5/RHEL6 :  Use the following command to update the operating system:*

*# yum update*

*To install a specific package, such as vsftpd, use the following command:*

*# yum install vsftpd*

*To update a specific package, such as bind, use the following command:*

*# yum update bind*

*For RHEL 4:  Launch the interactive Red Hat Update Agent with the following command:*

*# up2date*

*For a command line interface, use up2date-nox. To update non-interactively, use the following command:*

*# up2date-nox –update*

*To install a specific package, such as vnc, use the following command:*

*# up2date vnc*

*To update a specific package, such as ntp, use the following command:*

*# up2date -u ntp*

## ****5. Configuring a third-party repository ( Non-Redhat)****

Method 1 : Configuring third part repository by Installing  the  third party repository release package to enable yum to download the software.

For example, Extra Packages for Enterprise Linux (or EPEL) has an epel-release package that includes gpg keys for package signing and repository information.

*# wget -c http://download.fedoraproject.org/pub/epel/6/i386/epel-release-6-5.noarch.rpm  
# rpm -ivh epel-release-6-5.noarch.rpm  
warning: epel-release-6-5.noarch.rpm: Header V3 RSA/SHA256 Signature, key ID 0608b895: NOKEY  
Preparing… ########################################### [100%]  
1:epel-release ########################################### [100%]*

*# yum repolist  
Loaded plugins: rhnplugin  
repo id repo name status  
epel Extra Packages for Enterprise Linux 6 – x86\_64 5,979  
rhel-src Red Hat Enterprise Linux 6Server – x86\_64 – Source 2,640  
rhel-x86\_64-server-6 Red Hat Enterprise Linux Server (v. 6 for 64-bit x86\_64) 5,043  
repolist: 13,662*

Method 2: Manually Crating Customized Non-Redhat Package Repository

Create a file named /etc/yum.repos.d/epel.repo as following:

*# vi /etc/yum.repos.d/epel.repo  
[epel]  
name=Extra Packages for Enterprise Linux 5 – $basearch  
baseurl=http://download.fedoraproject.org/pub/epel/5/$basearch  
enabled=1  
gpgcheck=1*

*#gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL  
gpgkey=http://download.fedora.devel.redhat.com/pub/epel/RPM-GPG-KEY-EPEL*

All of this configuration information should be available on the site for which the yum repo is being created. Then import the gpg key for the repository as follows:

*# rpm –import http://download.fedora.devel.redhat.com/pub/epel/RPM-GPG-KEY-EPEL*

## ****6. Configure Yum to list only the packages available in my local repository****

By Default “yum list” will display all the packages both from local repository and the base channel enabled in /etc/yum/pluginconf.d/rhnplugin.conf.

If we want to display only packages available locally and ignore RHN, then we have to disable the base channel as mentioned below

 Edit***/etc/yum/pluginconf.d/rhnplugin.conf***  and disable base channel:

*[main]  
enabled = 0  
gpgcheck = 1*

*# You can specify options per channel, e.g.:  
#  
#[rhel-i386-server-5]  
#enabled = 1  
#  
#[some-unsigned-custom-channel]  
#gpgcheck = 0*

## ****7. How to install all related packages together using YUM?****

YUM supports grouping all the related rpm packages together for ease of installation. The group data is stored in a file called “comps.xml”  under the repodata folder.

### Process to Define a custom package group:

*1. Create the RPM repository, and copy all the RPMs you want to use into that directory:*

*# mkdir -p /usr/share/repository  
# cp \*.rpm /usr/share/repository*

*2. Run createrepo command to create the repodata folder :*

*# cd /usr/share/repository  
# createrepo .*

*Note: The createrepo package needs to be installed on the system*

*3.  Now to create the Groups file, by Creating the comps.xml file under repodata folder :*

*<comps>  
<!– <meta> –>  
<!– Meta information will go here eventually –>  
<!– </meta> –>  
<group>  
<id>gurkulgroup</id>  
<name>gurkulgroup</name>  
<default>true</default>  
<description>Description of group goes here</description>  
<uservisible>true</uservisible>  
<packagelist>  
<packagereq type=”mandatory”>package1  
<packagereq type=”default”>package2  
<packagereq type=”optional”>pacakge3  
</packagelist>  
</group>  
</comps>*

*Each group has an id, user visibility value, name, description, and package list. In the package list, there are three types package:*

* *mandatory   :     the packages marked as mandatory are always installed if the group is selected.*
* *default           :     the packages marked default are selected by default if the group is selected.*
* *optional        :     the packages marked optional must be specifically selected even if the group is selected.*

*4. Rerun createrepo so that group assignments are taken into account :*

*# createrepo -g repodata/comps.xml .*

*5. Create the file /etc/yum.repos.d/file.repo as follows:*

*# cat /etc/yum.repos.d/file.repo  
[myrepo]  
name=My Repo  
baseurl=file:///usr/share/repository  
enabled=1*

*6. To test :*

*#yum clean all  
# yum –noplugins groupinfo gurkulgroup*

### Install   Package Groups:

*To list all  available package groups:*

*# yum grouplist*

*Sometime, grouplist command can’t list all the groups that are provided when installing RHEL installation CD. And to  list all  groups, including hidden groups, run the following command:*

*# yum grouplist hidden*

*To install the packages under a specific group you can execute the command :*

*# yum groupinstall “”*

*Example:*

*# yum groupinstall “Chinese Support”*

*# yum groupinstall “KDE (K Desktop Environment)”*

## ****8.  How to Exclude Kernel related packages to automatically update, while updating my linux server using yum?****

The up2date command in RHEL 4 excludes kernel updates by default. The yum in RHEL 5 includes kernel updates by default.

To skip installing or updating kernel or other packages while using the yum update utility in RHEL 5 and 6, use following options

*Via command line (temporary):*

*# yum update –exclude=PACKAGENAME*

*For example, to exclude all kernel packages:*

*# yum update –exclude=kernel\**

*To make permanent changes, edit the /etc/yum.conf file. Follow these steps:*

*# vi /etc/yum.conf*

*Add a new line in the [main] section like this: exclude=package1,package2*

*For example:*

*…  
gpgcheck=1  
plugins=1  
installonly\_limit=3  
exclude=kernel\*  
Save file.*

## ****9. How do fix yum errors  Like   “package\_version-1\_1  is duplicate of  package-version-1” ?****

Use package-cleanup to remove duplicates.

*# package-cleanup –dupes  
# package-cleanup –cleandupes*

Update the system.

*# yum update*

## ****10.  Understanding YUM plugins****

 Yum provided plug-ins that extend and enhance yum functionality. Some plugins installed by default, and yum always gives the plug-ins used whenever we run the yum command as shown below

*# yum info yum Loaded plugins: product-id, refresh-packagekit, subscription-manager*

*:::::::::[output truncated]::::::::::::*

### Enable/Disable YUM plugins

*Open ‘/etc/yum.conf’.*

*To disable yum plugins, change the value for plugins to 0.*

*plugins = 0*

*After changing that,  clear  the yum cache by executing the following command.*

*# yum clean all*

*To enable yum plugins, change the value for enabled to 1*

### Protecting YUM Source Repository using ****yum**** plugins

Red Hat Enterprise Linux (5 and later version) comes with a tool to manage packages using repositories, yum. However, if you use two different repositories, each holding a different version of the same package, only the latest of the two will be installed if you use yum to update that package. This can lead to problems if the third party repository contains a newer version of a core system package than the Red Hat repository.

By using below yum plug-in , we can protect yum source packages from overriding by other available repositories in the server.

* Yum plugin, for RHEL5 :                yum-protectbase
* Yum plugin, for RHEL 6:                yum-plugin-protectbase

### Using the plugin to protect a repository:

Go to the repository configuration file located in ” /etc/yum.repos.d/repository.repo ” and add the line “protect=1” as shown in the example:

*[repository]  
::: output truncated:::::  
protect=1  
:::: output truncated:::::*

### Using the plugin to protect particular RHN channel:

Edit the file  “/etc/yum/pluginconf.d/rhnplugin.conf” and add “protect=yes” line, under the channel we want to protect, as shown in below example

*[rhel-x86\_64-server-5]  
enabled = 1  
protect = yes*

 Network Bonding

Bonding is a feature that can be enabled bu the Linux kernel, to allow system administrators to combine two or more network interface to form a single, logical “bonded” interface, for redundancy or increased throughput purpose.

We can configure Linux bonding in 7 different modes. The behavior of the bonded interfaces depends upon the mode.

Below are the available bonding modes

* Balance-rr (mode 0)
* Active-backup (mode 1)
* Balance-xor (mode 2)
* Broadcast (mode 3)
* 802.3ad (mode 4)
* Balance-tlb (mode 5)
* Balance-alb (mode 6)

## Configuring a bonding Channel between eth0 and eth1 using the default mode i.e. Mode-0 (balance-rr)

1. Network manager should not be running on the system, as NIM doesn’t support bonding. So please stop and disable the networkmanager service.

*# service NetworkManager stop*

*# chkconfig NetworkManager off*

2. Bonding module should be configured and loaded. To enable bonding module, create a bonding.conf file as below.

Note : you can add multiple entries to create multiple bonding channels like bond0, bond1, bond2…etc

*# cat > /etc/modprobe.d/bonding.conf*

*alias bond0 bonding*

*options bond0 mode=1 miimon=100*

3. Create a configuration file for the bonding channel i.e. bond0, as below

*# cat /etc/sysconfig/network-scripts/ifcfg-bond0*

*DEVICE=bond0  
IPADDR=192.168.50.111  
NETMASK=255.255.255.0  
USERCTL=no  
BOOTPROTO=none  
ONBOOT=yes  
BONDING\_OPTS=”mode=0 miimon=100″*

4. Create configuration files for all the interfaces  i.e. eth0 and eth1, which are participating in the bonding channel

*# cat > /etc/sysconfig/network-scripts/ifcfg-eth0*

*DEVICE=eth0  
BOOTPROTO=none  
HWADDR=aa:bb:cc:dd:ee:ff  
ONBOOT=yes  
MASTER=bond0  
SLAVE=yes  
USERCTL=no*

*# cat > /etc/sysconfig/network-scripts/ifcfg-eth1*

*DEVICE=eth1  
BOOTPROTO=none  
HWADDR=yy:xx:zz:aa:bb:cc  
ONBOOT=yes  
MASTER=bond0  
SLAVE=yes  
USERCTL=no*

5. Activating Bonding Channel

*a. With out rebooting*

*# modprobe bonding*

*# service network restart*

*b. with reboot*

*# reboot*

6. Verify that Bonding channel is active, by using the commands

*# cat /proc/net/bonding/bond0   
Ethernet Channel Bonding Driver: v3.0.2 (Nov 25, 2012)*

*Bonding Mode: adaptive load balancing   
Primary Slave: None   
Currently Active Slave: eth0   
MII Status: up   
MII Polling Interval (ms): 100   
Up Delay (ms): 0   
Down Delay (ms): 0*

*Slave Interface: eth0  
MII Status: up   
Link Failure Count: 0   
Permanent HW addr: aa:bb:cc:dd:ee:ff*

## Configurable Bonding Parameters

max\_bonds : specifies the number of bonding devices that can be configured on single bonding driver.

xmit\_hash\_policy : Slecect the transmit has policy to use for slave selection in balance-xor and 802.3d modevice. Possible values , layer2 (default) , larer2+3 or layer3+4

arp\_interval : Specifies the ARP link monitoring frequency in milli seconds

arp\_ip\_target : Specifies the IP addresses to use as ARP monitoring peers when arp\_interval is > 0. Multiple IP addresses must be seperated by comma. At least one IP addres must be given for ARP monitoring to work. Max. no. of target IPs are 16.

arp\_validate : Specifies Whether or not ARP probes and replies should be validated in the active-backup mode. This causes the ARP monitor to examine the incoming ARP requests and replies, and only consider a slave to be up if it is receiving the appropriate ARP traffic. Possible values

* none(0) – default value
* active(1) – validate only the active slave
* backup(2) – validate only the backup slave
* all(3) – validate all slaves

miimon : Specifies the MII link monitoring frequency in milliseconds. A value of 0 disables the MII link monitroing. A value of 100 is recommended.

updelay : specifies the time, in milliseconds, to wait before disabling a slave after a link recovery has been detected.

downdelay : specifies the time, in milliseconds, to wait before disabling a slava after a link failure has been detected.

use\_carrier : specifies whether or not miimon shoudl use MII/ETHTOO “ioctls for value of 0 ” or ” netif\_carrier\_ok() for value 1 ( default value)” functions to determine the link status.

## F AQ to Troubleshoot Network Bonding

### >>>> 1. /proc/net/bond0/bonding display the network interface card’s link is up when the cable is unplugged within a bonding environment. why ?

Bonding driver uses “use\_carrier”  with default value of 1 ( i.e. to use netif\_carrier\_ok() function), to identify the link status. Just incase if the bonding driver doesn’t support netif\_carrier\_ok() function, it cannot determine the link status properly. As a workaround we can set use\_carrier to value 0 ( i.e. to use  MII/ETHTOOL ioctl method) to determine the link status, as mentioned below

*# cat /etc/modprobe.conf  
alias bond0 bonding  
options bonding mode=1 miimon=100 use\_carrier=0*

if you are using mode-1 , we can also set the arp\_interval and arp\_ip\_target to determine the link status as given below

*# cat /etc/modprobe.conf  
alias bond0 bonding  
options bonding mode=1 miimon=100 arp\_interval=200 arp\_ip\_target=192.168.0.1,192.168.0.2*

### >>>> 2. Although the Bonding configured in balance-rr / mode-0 ( round-robin policy), outgoing network traffic doesn’t appear to go through all interfaces (RHEL5)

Possible cause could be that Bonded interfaces is predominantly communicating with a low number of IP address (for example, only NAS NFS traffic is used on the interface). To make sure that traffic was getting distributed to all available slave interfaces change the xmit\_hash\_policy value as given below:

*Option 1:  modify the /etc/modprobe.conf to add the xmit\_hash\_policy value*

*#vi /etc/modprobe.conf*

*:::::: output truncated::::*

*options bond0 mode=0 miimon=100 xmit\_hash\_policy=1*

*:::::: output truncated::::*

*Option 2:  Modify /etc/sysconfig/network-scripts/ifcfg-bondX and add/modify the following entry*

*#vi /etc/sysconfig/network-scripts/ifcfg-bondX*

*BONDING\_OPTS=”mode=0 miimon=100 xmit\_hash\_policy=1″*

*Then restart the bonding interface using*

*# ifdown bondX  
# ifup bondX*

For more reference to use there values you can check, in your linux box :  /usr/share/doc/kernel-doc-2.6.18/Documentation/networking/bonding.txt

### >>>> 3.  How to Gracefully Break the Bonded Network Interface?

In Sample Configuration is we have bond0 configure for two interface name eth0 and eht1, and bond running configuration looks as

*# cat /proc/net/bonding/bond0  
Ethernet Channel Bonding Driver: v3.4.0 Bonding Mode: load balancing (round-robin)  
MII Status: up  
MII Polling Interval (ms): 100  
Up Delay (ms): 0  
Down Delay (ms): 0*

Slave Interface: eth0  
MII Status: up  
Link Failure Count: 0  
Permanent HW addr: 53:44:00:43:91:04

Slave Interface: eth1  
MII Status: up  
Link Failure Count: 0  
Permanent HW addr: 52:33:00:6d:0e:70

The process to break the bonded interface is :

a.  Online remove one slave network interface which will be assigned with a new IP address from the bonding device. In this example, we are going to remove the eth1 from bond0. Execute the following command as root:

*# ifenslave -d bond0 eth1  
Or  
# echo -eth1 > /sys/class/net/bond0/bonding/slaves*

*After issue the command above , You will see the eth1 device has been remove from bond0# cat /proc/net/bonding/bond0  
Ethernet Channel Bonding Driver: v3.4.0*

Bonding Mode: load balancing (round-robin)  
MII Status: up  
MII Polling Interval (ms): 100  
Up Delay (ms): 0  
Down Delay (ms): 0

Slave Interface: eth0  
MII Status: up  
Link Failure Count: 0  
Permanent HW addr:  53:44:00:43:91:04

b. Create a new network configuration for eth1 via graphic tool or text tool and then using following command to start eth1 again. Make sure the physical link has been changed correctly to match the new network configuration of eth1.

*# ifup eth1  
During the above 2 steps, the network of bond0 will work fine without any break up.*

c. Create the same network configuration as bond0 for eth0

*— just copy the content of /etc/sysconfig/network-scripts/ifcfg-bondX to  /etc/sysconfig/network-scripts/ifcfg-eth0, except the interface name and HW MAC address.*

d. Remove the eth0 from bond0, destory the bond0 device and start eth0 immediately.

*Please note this disruptive operation, The network of bond0/eth0 will break shortly during this step.# ifenslave -d bond0 eth0; ifdown bond0; ifup eth0*

# cat /proc/net/bonding/bond0  
Ethernet Channel Bonding Driver: v3.4.0

Bonding Mode: load balancing (round-robin)  
MII Status: down  
MII Polling Interval (ms): 100  
Up Delay (ms): 0  
Down Delay (ms): 0

now ifconfig will show the configuration for only eth0 and eth1, no bonding interface appears.

e. Remove all bonding setting of bond0 in /etc/modprobe.conf and remove the ifcfg-bond0 in /etc/sysconfig/network-script directory. Then issue following command to remove the bond0 device from memory completely.

*# echo -bond0 > /sys/class/net/bonding\_masters*

### >>>> 4.  The interface eth0 configured with the multiple virtual interfaces like eth0:1 , eht0:2  and eth0:3.  When we disable the virtual interface eth0:0 , the other two interfaces also getting disabled.

In RHEL5, the lowest numbered virtual interface was treated as “primary” interface, and remaining all as secondary interfaces as “secondary” . If we disable primary interface , it will by default disable all the secondary interface.

And to avoid this default behavior, we should set a kernel parameter  “net.ipv4.conf.all.promote\_secondaries” to  value “1”, so that the secondary interfaces will be promoted as primary, in case of primary down.

*There are two ways to set this parameter:*

*Method-1*

*# Vi /etc/sysctl.conf*

*net.ipv4.conf.all.promote\_secondaries = 1*

*#  sysctl -p*

*Method-2*

*echo “1” > /proc/sys/net/ipv4/conf/eth1/promote\_secondaries*

### >>>> 5. Difference between arp monitoring and MII monitoring

ARP monitoring periodically sends an arp packet to the designated hosts. When the reply does not come back, then the bonding device considers the network on that specific device is down.

MII monitoring periodically monitors the information provided by MII (Media Independent Interface). When the interface of the driver detects the network failure, the bonding device considers the network on that specific device is down.

MII is a driver-dependent function. It cannot detect network failure if it occurred beyond the nearest switch, because the monitoring covers the link from the device to the nearest connected switch. Whenever emitting excessive network packets is accepted, it is safe to use ARP monitoring.

 LVM Administration

## ****1. How do I reduce LVM logical voulme in Red Hat Enterprise linux?****

*First of all make sure to have sufficient disk space available before reducing logical volume (otherwise it would result in data loss). Also, make sure to have valid data backup before going forward and making any changes.*

*It’s important to shrink the filesystem before reducing it to prevent the data loss/corruption. The resize2fs program will resize ext2, ext3 or ext4 file systems. It can be used to enlarge or shrink an unmounted file system located on device. Refer following steps to reduce the logical volume by 500GB for example,*

*1) Unmount the filesystem*

*# umount /dev/VG00/LV00  
2) Scan and check the filesystem to be at safer side:*

*# e2fsck /dev/VG00/LV00  
3) Shrink the filesystem with resize2fs as follows:*

*# resize2fs /dev/VG00/LV00 500M  
where 500M is amount of disk space to which you wish to shrink the disk data.*

*4) Reduce the logical volume by 500GB with lvreduce:*

*# lvreduce -L -500G VG00/LV00  
It will reduces the size of logical volume LV00 in volume group VG00 by 500GB.*

*5) Mount the filesystem and check the disk space with df -h command.*

## ****2. What is the difference between “Linux” and “Linux LVM” partition types?****

*It does not have any specific advantage. both partition type can work with LVM.*

*The type id is only for informative purposes. Logical volumes don’t have a concept of a “type”, they’re just block devices. They do not have a partition ID or type. They are composed of physical extents (PE) which may be spread over multiple physical volumes (PV), each of which could be a partition or a complete disk. LVM logical volumes are normally treated like individual partitions, not as disks, so there’s no partition table and therefore no partition type id to look for.*

## ****3. How do we Log all LVM commands, that we execute on the machine?****

*The default LVM configuration do not log the commands that are used in a shell or in a GUI (e.g system-config-lvm) environment. But it’s possible to active the log in the lvm.conf*

*To active the log follow the following steps.*

*Make a copy of the original lvm.conf file  
# cp /etc/lvm/lvm.conf /root  
Edit the lvm.conf file and find the log section. It starts as ‘log {‘. The default configuration comes like the following:  
log {  
verbose = 0   
syslog = 1   
#file = “/var/log/lvm2.log”  
overwrite = 0   
level = 0   
indent = 1   
command\_names = 0   
prefix = ” “  
# activation = 0  
}  
It’s necessary only 2 modifications to active the log of the LVM:*

*– Uncomment the line # file = “/var/log/lvm2”  
– Change the level = 0 to a value between 2 and 7.*

*Remind that the 7 is more verbose than 2.*

*Save and exit the file.  
It’s not necessary restart any service, the file /var/log/lvm2.log will be created when any command from lvm run (e.g lvs, lvextend, lvresize etc).*

## ****4. How do I create LVM-backed raw devices with udev in RHEL6?****

*Edit /etc/udev/rules.d/60-raw.rules and add lines similar to the following:*

*ACTION==”add”, ENV{DM\_VG\_NAME}==”VolGroup00″, ENV{DM\_LV\_NAME}==”LogVol00″, RUN+=”/bin/raw /dev/raw/raw1 %N”*

*where VolGroup00 is your Volume Group name, and LogVol00 is your Logical Volume name.*

*To set permissions on these devices, we can do so as per usual:*

*ACTION==”add”, KERNEL==”raw\*”, OWNER==”username”, GROUP==”groupname”, MODE==”0660″*

## ****5. What is optimal stripe count for better performance in LVM?****

*The maximum number of stripes in LVM is 128. The “optimal number of stripes” depends on the storage devices used for the LVM logical volume.*

*In the case of local physical disks connected via SAS or some other protocol, the optimal number of stripes is equal to the number of disks.*

*In the case of SAN storage presented to the Linux machine in the form of LUNs, there may be no advantage to a striped logical volume (a single LUN may provide optimal performance), or there may be an advantage. With a LUN coming from SAN storage, the LUN is often a chunk of storage which comes from a SAN volume, and the SAN volume often is a RAID volume with some sort of striping and/or parity. In this case there is no advantage.*

*In some cases where SAN characteristics are unknown or changing, performance testing of two LVM volumes with a differing number of stripes may be worthwhile. For simple sequential IO performance, “dd” can be used (for random another tool will be needed). Make the second LVM logical volume containing twice the number of stripes as the first, and compare performance. Continue increasing stripes and comparing performance in this manner until there is no noticeable performance improvement.*

## ****6. LVM commands are failing with this error: Can’t open exclusively. Mounted filesystem?****

*We often face this problem that we cannot create a logical volume on a disk that is part of multipathed storage, after creating a new disk partition created with parted.*

*Multipathing uses the mpath\* name to refer to storage rather than the sd\* disk name, since multiple sd\* names can refer to the same storage.*

*The below error message appears because the multipath daemon is accessing the sd\* device, and the lvm commands cannot open the device exclusively. Commands such as “pvcreate /dev/sd\*” fail with the*

*Error: Can’t open exclusively. Mounted Filesystem?*

*To resolve the issue:*

*Run “# fuser -m -v /dev/sd\*” to see what processes are accessing the device.*

*If multipathd appears, run “multipath -ll” to determine which mpath\* device maps to that disk.*

*Run the command “pvcreate /dev/mapper/mpath\*” to successfully create a physical volume on the device.  
Continue creating the volume group, logical volume, and filesystem using the correct path to the disk.*

## ****7. Recommended region size for mirrored lvm volumes ?****

*Region size can impact performance, generally for larger region sizes, there will be fewer writes to the log device – this could increase performance. Smaller region sizes lead to faster recovery times after a machine crashes. The default region size of 512KB balances these considerations pretty fairly.*

*A change in the region size for mirror lvm volume would not result in much performance gain, but if you could simulate the workload, then please try a few different variants for region size to ensure that.*

*Also, there is a limitation in the cluster infrastructure, cluster mirrors greater than 1.5TB cannot be created with the default region size of 512KB. Users that require larger mirrors should increase the region size from its default to something larger. Failure to increase the region size will cause LVM creation to hang and may hang other LVM commands as well.*

*As a general guideline for specifying the region size for mirrors that are larger than 1.5TB, you could take your mirror size in terabytes and round up that number to the next power of 2, using that number as the -R argument to the lvcreate command. For example, if your mirror size is 1.5TB, you could specify -R 2. If your mirror size is 3TB, you could specify -R 4. For a mirror size of 5TB, you could specify -R 8. For more information about the same please refer to 4.4.3. Creating Mirrored Volumes*

*Above calculation could be used to decide the region size for large mirror lvm volumes of size 16 – 20 TB also. For example, when creating the cluster mirror lvm volume with size of 20 TB, please set a region size of 32 using -R 32 argument with lvcreate command as shown below:*

*$ lvcreate -m1 -L 20T -R 32 -n mirror vol\_group*

## ****8. Recreating a partition table that was accidentally deleted that contains an LVM Physical Volume on Red Hat Enterprise Linux?****

NOTE: This is a very difficult procedure and does not guarantee that data can be recovered. You may wish to try this procedure on a snapshot of the data first (where possible). Alternatively, seek a data recovery company to assist you with restoring the data*.*

*The location of the LVM2 label can be found on the disk by using “hexedit -C /dev/ | grep LABELONE” (be sure to locate the correct label, not another one that might have been added by mistake). Using the location of the label, we can discover the cylinder where the partition that holds that LVM PV starts.*

*Recreating the partition at the correct location will allow LVM2 tools to find the LVM PV and the volume group can be reactivated. If you cannot locate the LVM2 label, this procedure will not be useful to you. It is possible to use the same procedure for other data located on the disk (such as ext3 filesystems).*

*If the following symptoms are observed, this solution may apply to you:*

*When scanning for the volume group, it is not found:*

*# vgchange -an vgtest  
Volume group “vgtest” not found*

*Looking through LVM volume group history (in /etc/lvm/archive/.-), the PV for this volume group used to contain partitions:*

*$ grep device /etc/lvm/archive/vgtest\_00004-313881633.vg   
device = “/dev/vdb5” # Hint only*

*Device that should contain the LVM PV now does not have any partitions:*

*Try using parted rescue first as it may be able to detect the start of other partitions on the device and restore the partition table.*

*If parted rescue does not work, the following procedure can help you restore the partition table*

*Using hexdump, try to locate the LVM label on the device that had the partition table removed (in the data below, the location of the LABELONE label is {hex} 0fc08000 bytes into the device):*

*# hexdump -C /dev/vdb | grep LABELONE  
0fc08000 4c 41 42 45 4c 4f 4e 45 01 00 00 00 00 00 00 00 |LABELONE……..|  
Converting the byte-address of the LVM2 label to decimal:  
0x0fc08000 = 264273920*

*Run fdisk -l against the device to find out how many bytes per cylinder:*

*# fdisk -l /dev/vdb*

*Disk /dev/vdb: 2113 MB, 2113929216 bytes  
16 heads, 63 sectors/track, 4096 cylinders  
Units = cylinders of 1008 \* 512 = 516096 bytes <— 516096 bytes per cylinder  
The byte location into the disk that the partition for the LVM PV starts is:  
(byte position of LVM label (decimal) = 264273920  
number of bytes per cylinder) = 516096  
(264273920 / 516096) = 512.063492063 <– round this down to cylinder 512*

*Add one cylinder because they start at 1, not zero: 512 + 1 = Starting cylinder: 513*

*Create partition table with a partition starting at cylinder 513:*

*# fdisk /dev/vdb  
…  
Command (m for help): n  
Command action  
e extended  
p primary partition (1-4)  
e  
Partition number (1-4): 4  
First cylinder (1-4096, default 1): 513  
Last cylinder or +size or +sizeM or +sizeK (513-4096, default 4096):   
Using default value 4096*

*Command (m for help): n  
Command action  
l logical (5 or over)  
p primary partition (1-4)  
l  
First cylinder (513-4096, default 513):   
Using default value 513  
Last cylinder or +size or +sizeM or +sizeK (513-4096, default 4096): 1024*

*Command (m for help): p*

*Disk /dev/vdb: 2113 MB, 2113929216 bytes  
16 heads, 63 sectors/track, 4096 cylinders  
Units = cylinders of 1008 \* 512 = 516096 bytes*

*Device Boot Start End Blocks Id System  
/dev/vdb4 513 4096 1806336 5 Extended  
/dev/vdb5 513 1024 258016+ 83 Linux*

*Command (m for help): w  
The partition table has been altered!*

*Calling ioctl() to re-read partition table.  
Syncing disks.*

*Rescan and activate the volume group:*

*# pvscan  
PV /dev/vdb5 VG vgtest lvm2 [248.00 MB / 0 free]  
Total: 1 [8.84 GB] / in use: 1 [8.84 GB] / in no VG: 0 [0 ]*

*# vgchange -ay vgtest  
1 logical volume(s) in volume group “vgtest” now active*

## ****9.****  ****LVM2 volume group in partial mode with physical volumes marked missing even though they are available in RHEL****

*Sometimes, attempting to modify a volume group or logical volume fails due to missing devices that are not actually missing:*

*# lvextend -l+100%PVS /dev/myvg/lv02 /dev/mapper/mpath80*

*WARNING: Inconsistent metadata found for VG myvg – updating to use version 89  
Missing device /dev/mapper/mpath73 reappeared, updating metadata for VG myvg to version 89.  
Device still marked missing because of alocated data on it, remove volumes and consider vgreduce –removemissing.*

*Any attempt to change a VG or LV claims PVs are missing:*

*Cannot change VG myvg while PVs are missing.  
Consider vgreduce –removemissing.*

*LVM physical volumes are marked with the missing (m) flag in pvs output even though they are healthy and available:*

*PV VG Fmt Attr PSize PFree   
/dev/mapper/mpath24 myvg lvm2 a-m 56.20G 0*

*Volume group is marked as ‘partial’ and causes lvm commands to fail:*

*VG #PV #LV #SN Attr VSize VFree   
myvg 42 10 0 wz-pn- 2.31T 777.11G*

*Restore each missing physical volume with:*

*# vgextend –restoremissing <volume group> <physical volume>  
# vgextend –restoremissing myvg /dev/mapper/mpath24  
Volume group “myvg” successfully extended*

## ****10. Mount LVM partitions on SAN storage connected to a newly-built server?****

Scan for Physical Volumes, scan those PVs for Volume Groups, scan those VGs for Logical Volumes, change Volume Groups to active.

*# pvscan   
# vgscan   
# lvscan   
# vgchange -ay*

*Volumes are now ready to mount as per usual with mount command and/or add to /etc/fstab file.*

### Check Multipath storage is actually available to host:

*[root@host ~]# multipath -l  
mpath1 (350011c600365270c) dm-8 HP 36.4G,ST336754LC  
[size=34G][features=1 queue\_if\_no\_path][hwhandler=0][rw]  
\_ round-robin 0 [prio=0][active]  
\_ 0:0:1:0 sda 8:0 [active][undef]  
mpath5 (3600508b4001070510000b00001610000) dm-11 HP,HSV300  
[size=15G][features=1 queue\_if\_no\_path][hwhandler=0][rw]  
\_ round-robin 0 [prio=0][active]  
\_ 4:0:0:3 sde 8:64 [active][undef]  
\_ 5:0:0:3 sdo 8:224 [active][undef]  
\_ round-robin 0 [prio=0][enabled]  
\_ 4:0:1:3 sdj 8:144 [active][undef]  
\_ 5:0:1:3 sdt 65:48 [active][undef]  
mpath11 (3600508b4000f314a0000400001600000) dm-13 HP,HSV300  
[size=500G][features=1 queue\_if\_no\_path][hwhandler=0][rw]  
\_ round-robin 0 [prio=0][active]  
\_ 4:0:0:5 sdg 8:96 [active][undef]  
\_ 5:0:0:5 sdq 65:0 [active][undef]  
\_ round-robin 0 [prio=0][enabled]  
\_ 4:0:1:5 sdl 8:176 [active][undef]  
\_ 5:0:1:5 sdv 65:80 [active][undef]  
mpath4 (3600508b4001070510000b000000d0000) dm-10 HP,HSV300  
[size=750G][features=1 queue\_if\_no\_path][hwhandler=0][rw]  
\_ round-robin 0 [prio=0][active]  
\_ 4:0:0:2 sdd 8:48 [active][undef]  
\_ 5:0:0:2 sdn 8:208 [active][undef]  
\_ round-robin 0 [prio=0][enabled]  
\_ 4:0:1:2 sdi 8:128 [active][undef]  
\_ 5:0:1:2 sds 65:32 [active][undef]  
mpath10 (3600508b4000f314a0000400001090000) dm-12 HP,HSV300  
[size=350G][features=1 queue\_if\_no\_path][hwhandler=0][rw]  
\_ round-robin 0 [prio=0][active]  
\_ 4:0:0:4 sdf 8:80 [active][undef]  
\_ 5:0:0:4 sdp 8:240 [active][undef]  
\_ round-robin 0 [prio=0][enabled]  
\_ 4:0:1:4 sdk 8:160 [active][undef]  
\_ 5:0:1:4 sdu 65:64 [active][undef]  
mpath3 (3600508b4001070510000b000000a0000) dm-9 HP,HSV300  
[size=750G][features=1 queue\_if\_no\_path][hwhandler=0][rw]  
\_ round-robin 0 [prio=0][active]  
\_ 4:0:0:1 sdc 8:32 [active][undef]  
\_ 5:0:0:1 sdm 8:192 [active][undef]  
\_ round-robin 0 [prio=0][enabled]  
\_ 4:0:1:1 sdh 8:112 [active][undef]  
\_ 5:0:1:1 sdr 65:16 [active][undef]*

### Check Physical Volumes are being scanned by LVM and seen:

*[root@host ~]# pvdisplay  
— Physical volume —  
PV Name /dev/dm-14  
VG Name VolG\_CFD  
PV Size 750.00 GB / not usable 4.00 MB  
Allocatable yes (but full)  
PE Size (KByte) 4096  
Total PE 191999  
Free PE 0  
Allocated PE 191999  
PV UUID 0POViC-2Pml-AmfI-W5Mh-s6fC-Ei18-hCOOoJ*

*— Physical volume —  
PV Name /dev/dm-13  
VG Name VolG\_CFD  
PV Size 500.00 GB / not usable 4.00 MB  
Allocatable yes  
PE Size (KByte) 4096  
Total PE 127999  
Free PE 1  
Allocated PE 127998  
PV UUID RcDER4-cUwa-sDGF-kieA-44q9-DLm2-1CMOh4*

*— Physical volume —  
PV Name /dev/dm-15  
VG Name VolG\_FEA  
PV Size 750.00 GB / not usable 4.00 MB  
Allocatable yes (but full)  
PE Size (KByte) 4096  
Total PE 191999  
Free PE 0  
Allocated PE 191999  
PV UUID 5DprQD-OOs9-2vxw-MGT1-13Nl-YTTt-BnxGhq*

*— Physical volume —  
PV Name /dev/dm-12  
VG Name VolG\_FEA  
PV Size 350.00 GB / not usable 4.00 MB  
Allocatable yes (but full)  
PE Size (KByte) 4096  
Total PE 89599  
Free PE 0  
Allocated PE 89599  
PV UUID uQIqyq-0PiC-XT2e-J90h-tRBk-Nb8L-MdleF5*

*— Physical volume —  
PV Name /dev/dm-11  
VG Name vgnbu  
PV Size 15.00 GB / not usable 4.00 MB  
Allocatable yes (but full)  
PE Size (KByte) 4096  
Total PE 3839  
Free PE 0  
Allocated PE 3839  
PV UUID aZNgCY-eRYe-3HmZ-bnAD-4kGN-9DhN-8I5R7D*

*— Physical volume —  
PV Name /dev/sdb4  
VG Name vg00  
PV Size 33.24 GB / not usable 16.86 MB  
Allocatable yes (but full)  
PE Size (KByte) 32768  
Total PE 1063  
Free PE 0  
Allocated PE 1063  
PV UUID PKvqoX-hWfx-dQUv-9NCL-Re78-LyIa-we69rm*

*— Physical volume —  
PV Name /dev/dm-8  
VG Name vg00  
PV Size 33.92 GB / not usable 12.89 MB  
Allocatable yes  
PE Size (KByte) 32768  
Total PE 1085  
Free PE 111  
Allocated PE 974  
PV UUID GnjUsb-NxJR-aLgC-fga8-Ct1q-cf89-xhaaFs*

## ****11. How to Grow an LVM Physical Volume after resizing the disk?****

*Note : This procedure does have the potential to lose data on the disk if done improperly, we strongly recommend a backup be performed before proceeding.*

*For example, we have resized  a disk from 50Gb to 120Gb*

*# pvs  
PV VG Fmt Attr PSize PFree   
/dev/sdb1 VolGroup02 lvm2 a– 50.00g 2.00g*

*Whilst the underlying storage (eg: /dev/sdb) may have been resized, the partition we are using as a physical volume (eg: /dev/sdb1) remains at the smaller size.*

*We’ll need to resize the partition, then resize the Physical Volume before you can proceed.*

*First we would confirm the actual storage size with “****fdisk -ul /dev/sdb****” and observe the increased disk size. Depending on how the storage is presented, we may need to reboot for this to appear.*

*We will then need to resize the partition on the disk.  We can achieve this by observing the starting sector in****fdisk -ul /dev/sdb****, then removing the partition with fdisk and re-creating it with the same starting sector but the (default) last sector of the drive as the ending sector. Then write the partition table and confirm the change (and the correct starting sector) with****fdisk -ul /dev/sdb****.*

*Now We are ready to****pvresize /dev/sdb1****to grow the PV onto the rest of the expanded partition.This will create free extents within the Volume Group which we can then grow a Logical Volume into.*

*If we run your LV resize with****lvresize -r****, it will grow the filesystem we have within the Logical Volume as well.*

## ****12. Delete a LVM partition****

*Delete LVM partition from the /etc/fstab For example:*

*/dev/sda2 / ext3 defaults 1 1  
/dev/sda1 /boot ext3 defaults 1 2  
tmpfs /dev/shm tmpfs defaults 0 0  
devpts /dev/pts devpts gid=5,mode=620 0 0  
sysfs /sys sysfs defaults 0 0  
proc /proc proc defaults 0 0  
/dev/sd3 swap swap defaults 0 0  
/dev/volumegroup/lvm /var ext3 defaults 0 0*

*umount LVM partition  
# umount /dev/volumegroup/lvm*

*Disable lvm  
# lvchange -an /dev/volumegroup/lvm*

*Delete lvm volume  
# lvremove /dev/volumegroup/lvm*

*Disable volume group  
# vgchange -an volumegroup*

*Delete volume group  
# vgremove volumegroup*

*Delete phisical Volume  
# pvremove /dev/sdc1 /dev/sdc2*

## ****13. Restore a volume group in Red Hat Enterprise Linux if one of the physical volumes that constitutes the volume group has failed.****

*NOTE :  These commands have the potential to corrupt data, should be executed at one’s own discretion*

*This procedure requires a recent backup of the LVM configuration. This can be generated with the command vgcfgbackup and is stored in the file /etc/lvm/backup/<volume group name>.*

*The /etc/lvm/archive directory also contains recent configurations that are created when modifications to the volume group metadata are made. It is recommended that these files be regularly backed up to a safe location so that they will be available if required for recovery purposes.*

*Assuming a physical volume has been lost that was a part of a volume group the following procedure may be followed. The procedure will replace the failed physical volume with a new disk rendering any remaining logical volumes accessible for recovery purposes.*

*The procedure for recovery is as follows:*

*1. Execute the following command to display information about the volume group in question:*

*# vgdisplay –partial –verbose*

*The output will be similar to the following (note that the –partial flag is required to activate or manipulate a volume group having one or more physical volumes missing and that use of this flag with LVM2 activation commands (vgchange -a) will force volumes to be activated in a read-only state):*

*Partial mode. Incomplete volume groups will be activated read-only.  
Finding all volume groups  
Finding volume group “volGroup00”  
Couldn’t find device with uuid ƏeWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j’.  
Couldn’t find device with uuid ƏeWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j’.  
Couldn’t find device with uuid ƏeWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j’.  
Couldn’t find device with uuid ƏeWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j’.  
— Volume group —  
VG Name volGroup00  
System ID  
Format lvm2  
Metadata Areas 4  
Metadata Sequence No 33  
VG Access read  
VG Status resizable  
MAX LV 0  
Cur LV 1  
Open LV 0  
Max PV 0  
Cur PV 5  
Act PV 5  
VG Size 776.00 MB  
PE Size 4.00 MB  
Total PE 194  
Alloc PE / Size 194 / 776.00 MB  
Free PE / Size 0 / 0  
VG UUID PjnqwZ-AYXR-BUyo-9VMN-uSRZ-AFlj-WOaA6z  
— Logical volume —  
LV Name /dev/volGroup00/myLVM  
VG Name volGroup00  
LV UUID az6REi-mkt5-sDpS-4TyH-GBj2-cisD-olf6SW  
LV Write Access read/write  
LV Status available  
# open 0  
LV Size 776.00 MB  
Current LE 194  
Segments 5  
Allocation inherit  
Read ahead sectors 0  
Block device 253:0  
— Physical volumes —  
PV Name /dev/hda8  
PV UUID azYDV8-e2DT-oxGi-5S9Q-yVsM-dxoB-DgC4qN  
PV Status allocatable  
Total PE / Free PE 48 / 0  
PV Name /dev/hda10  
PV UUID SWICqb-YIbb-g1MW-CY60-AkNQ-gNBu-GCMWOi  
PV Status allocatable  
Total PE / Free PE 48 / 0  
PV Name /dev/hda11  
PV UUID pts536-Ycd5-kNHR-VMZY-jZRv-nTx1-XZFrYy  
PV Status allocatable  
Total PE / Free PE 48 / 0  
PV Name /dev/hda14  
PV UUID OtIMPe-SZK4-arxr-jGlp-eiHY-2OA6-kyntME  
PV Status allocatable  
Total PE / Free PE 25 / 0  
PV Name unknown device  
PV UUID 9eWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j  
PV Status allocatable  
Total PE / Free PE 25 / 0*

*Note the PV UUID line:*

*PV UUID 9eWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j*

*This line contains the universally unique identifier (UUID) of the physical volume that failed and will be needed in the next step.*

*2. If the physical volume failed, it must be replaced with a disk or partition that is equal in size or larger than the failed volume. If the disk did not fail but was overwritten or corrupted, the same volume can be re-used. Run the following command to re-initialize the physical volume:*

*# pvcreate –restorefile /etc/lvm/backup/<volume group name> –uuid <UUID> <device>*

*In the above command the UUID is the value taken from the output in step 1. In this example the full command would be:*

*# pvcreate –restorefile /etc/lvm/backup/volGroup00 –uuid 9eWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j /dev/hda15*

*Couldn’t find device with uuid 9eWicl-1HSB-Fkcz-wrMf-DzMd-Dgx2-Kyc11j.  
Physical volume “/dev/hda15” successfully created*

*Note that when overwriting a previously-used LVM2 physical volume (for example when recovering from a situation where the volume had been inadvertently overwritten) the -ff option must be given to the pvcreate command.*

*3. Now the new physical volume has been initialized with the UUID of the old physical volume. The volume group metadata may be restored with the following command:*

*# vgcfgrestore –file /etc/lvm/backup/<volume group name > <volume group name>*

*Continuing the earlier example the exact command would be:*

*# vgcfgrestore –file /etc/lvm/backup/volGroup00 volGroup00  
Restored volume group volGroup00*

*4. To check that the new physical volume is intact and the volume group is functioning correctly execute vgdisplay -v.*

*Note: This procedure will not restore any data lost from a physical volume that has failed and been replaced. If a physical volume has been partially overwritten (for example, the label or metadata regions have been damaged or destroyed) then user data may still exist in the data area of the volume and this may be recovered using standard tools after restoring access to the volume group using these steps.*

## ****14. Logical Volume Manager (LVM) snapshot and how do we use it?****

*Logical Volume Manager (LVM) provides the ability to take a snapshot of any logical volume for the purpose of obtaining a backup of a partition in a consistent state. Traditionally the solution has been to mount the partition read-only, apply table-level write locks to databases or shut down the database engine etc.; all measures which adversely impact availability (but not as much as data loss without a backup will). With LVM snapshots it is possible to obtain a consistent backup without compromising availability.*

*The LVM snapshot works by logging the changes to the filesystem to the snapshot partition, rather than mirroring the partition. Thus when you create a snapshot partition you do not need to use space equal to the size of the partition that you are taking a snapshot of, but rather the amount of changes that it will undergo during the lifetime of the snapshot. This is a function of both how much data is being written to the partition and also how long you intend keeping the LVM snapshot.*

*Below example shows about LVM snapshot creation. Here we create a logical volume of 500MB to use to take a snapshot. This will allow 500MB of changes on the volume we are taking a snapshot of during the lifetime of the snapshot.*

*The following command will create /dev/ops/dbbackup as a snapshot of /dev/ops/databases.*

*# lvcreate -L500M -s -n dbbackup /dev/ops/databases  
lvcreate — WARNING: the snapshot must be disabled if it gets full  
lvcreate — INFO: using default snapshot chunk size of 64 KB for “/dev/ops/dbbackup”  
lvcreate — doing automatic backup of “ops”  
lvcreate — logical volume “/dev/ops/dbbackup” successfully created*

*Now we create the mount point and mount the snapshot.*

*# mkdir /mnt/ops/dbbackup  
# mount /dev/ops/dbbackup /mnt/ops/dbbackup  
mount: block device /dev/ops/dbbackup is write-protected, mounting read-only*

*After performing the backup of the snapshot partition we release the snapshot. The snapshot will be automatically released when it fills up, but maintaining incurs a system overhead in the meantime.*

*# umount /mnt/ops/dbbackup  
# lvremove /dev/ops/dbbackup  
lvremove — do you really want to remove “/dev/ops/dbbackup”? [y/n]: y  
lvremove — doing automatic backup of volume group “ops”  
lvremove — logical volume “/dev/ops/dbbackup” successfully removed*

## ****15. How do we Create New LVM volume from a LVM snapshot?****

*create a sparse file, providing room for the volumegroup*

*# dd if=/dev/zero of=file bs=1 count=1 seek=3G*

*setup the sparse file as block device*

*# losetup -f file*

*create a volumegroup and volume*

*# pvcreate /dev/loop0  
# vgcreate vgtest /dev/loop0   
# lvcreate -l 10 -n lvoriginal vg00*

*create a filesystem, create content on it*

*# mkdir /mnt/tmp /mnt/tmp2  
# mkfs.ext4 /dev/vg00/lvoriginal  
# mount /dev/vg00/lvoriginal /mnt/tmp  
# echo state1 >>/mnt/tmp/contents*

*create the mirror – the volumegroup has to have enough free PE’s*

*# lvconvert -m 1 /dev/vg00/lvoriginal  
# lvconvert –splitmirrors 1 -n lvclone /dev/vg00/lvoriginal*

*change the contents on the original volume*

*# echo state2 >>/mnt/tmp/contents*

*now access the clone volume and verify it represents the originals old state*

*# mount /dev/vgtest/lvolnew /mnt/tmp2  
# cat /mnt/tmp2/contents   
# cat /mnt/tmp/contents*

## ****16. How can I boot from an LVM snapshot on Red Hat Enterprise Linux?****

*The snapshot has to be in the same volume group as the original root logical volume.Often, other file systems should be snapshotted at the same time (eg. /var, /usr) if they are separate file systems to root.*

*Procedure:*

***Step 1 :****Create a snapshot of any local filesystems (for RHEL6, it is recommended that you do not put a ‘-‘ in the name as it makes addressing the volume more complicated):*

*# lvcreate -s -n varsnapshot -L 1G /dev/VolGroup00/var  
# lvcreate -s -n rootsnapshot -L 2G /dev/VolGroup00/root*

***Step 2 :****Mount the root snapshot so we can change the /etc/fstab of the snapshot version:*

*# mkdir /mnt/snapshot  
# mount /dev/VolGroup00/rootsnapshot /mnt/snapshot  
# vi /mnt/snapshot/etc/fstab*

***Step 3 :****Change the entries in /mnt/etc/fstab to point to the snapshot volumes rather than the original devices:*

*/dev/VolGroup00/rootsnapshot / ext3 defaults 1 1  
/dev/VolGroup00/varsnapshot /var ext3 defaults 1 2*

***Step 4 :****Now unmount the snapshot:*

*# cd /tmp  
# umount /mnt/snapshot*

***Step 5 :****Add an entry in grub to boot into the snapshot:*

***Step 5a****For Red Hat Enterprise Linux 5, copy the current default grub.conf entry, and make a new entry pointing to the snapshot version:*

*/boot/grub/grub.conf entry before:  
…  
default=0  
…  
title Red Hat Enterprise Linux 5 (2.6.18-194.el5)  
root (hd0,0)  
kernel /vmlinuz-2.6.18-194.el5 ro root=/dev/VolGroup00/root  
initrd /initrd-2.6.18-194.el5.img*

*After:  
…  
default=0  
…  
title Snapshot (2.6.18-194.el5)  
root (hd0,0)  
kernel /vmlinuz-2.6.18-194.el5 ro root=/dev/VolGroup00/rootsnapshot  
initrd /initrd-2.6.18-194.el5.img  
title Red Hat Enterprise Linux 5 (2.6.18-194.el5)  
root (hd0,0)  
kernel /vmlinuz-2.6.18-194.el5 ro root=/dev/VolGroup00/root  
initrd /initrd-2.6.18-194.el5.img*

***Step 5b****For Red Hat Enterprise Linux 6, copy the default grub.conf entry, and maake a new entry pointing to the snapshot version:*

*/boot/grub/grub.conf before:  
…  
default=0  
…  
title Red Hat Enterprise Linux Server (2.6.32-279.9.1.el6.x86\_64)  
root (hd0,0)  
kernel /vmlinuz-2.6.32-279.9.1.el6.x86\_64 ro root=/dev/mapper/VolGroup00-rootvol rd\_NO\_LUKS LANG=en\_US.UTF-8 rd\_NO\_MD quiet SYSFONT=latarcyrheb-sun16 rhgb crashkernel=auto rd\_LVM\_LV=VolGroup00/rootvol KEYBOARDTYPE=pc KEYTABLE=us rd\_NO\_DM  
initrd /initramfs-2.6.32-279.9.1.el6.x86\_64.img*

*/boot/grub/grub.conf after:  
…  
default=0  
…  
title Snapshot (2.6.32-279.9.1.el6.x86\_64)  
root (hd0,0)  
kernel /vmlinuz-2.6.32-279.9.1.el6.x86\_64 ro root=/dev/mapper/VolGroup00-rootsnapshot rd\_NO\_LUKS LANG=en\_US.UTF-8 rd\_NO\_MD quiet SYSFONT=latarcyrheb-sun16 rhgb crashkernel=auto rd\_LVM\_LV=VolGroup00/rootvol KEYBOARDTYPE=pc KEYTABLE=us rd\_NO\_DM  
initrd /initramfs-2.6.32-279.9.1.el6.x86\_64.img  
title Red Hat Enterprise Linux Server (2.6.32-279.9.1.el6.x86\_64)  
root (hd0,0)  
kernel /vmlinuz-2.6.32-279.9.1.el6.x86\_64 ro root=/dev/mapper/VolGroup00-rootvol rd\_NO\_LUKS LANG=en\_US.UTF-8 rd\_NO\_MD quiet SYSFONT=latarcyrheb-sun16 rhgb crashkernel=auto rd\_LVM\_LV=VolGroup00/rootvol KEYBOARDTYPE=pc KEYTABLE=us rd\_NO\_DM  
initrd /initramfs-2.6.32-279.9.1.el6.x86\_64.img*

*NOTE: On the grub menu entry on RHEL6, change “root=” to point to the snapshot but DO NOT change rd\_LVM\_LV to point to the snapshot, because this will prevent both the real and snapshot devices from activating on boot. Snapshots cannot be activated without the real volume being activated as well.*

***Step 6 :****Now you can boot into the snapshot by choosing the correct grub menu entry. To boot back onto the real LVM device, just select the original grub menu entry.*

***Step 7 :****You can verify that you are booted into the snapshot version by checking which LVM device is mounted:*

*# mount | grep VolGroup00  
/dev/mapper/VolGroup00-rootsnapshot on / type ext4 (rw)  
/dev/mapper/VolGroup00-varsnapshot on /var type ext4 (rw)*

***You can remove the snapshot with the following procedure:***

*Step 1) Remove the grub entry from /boot/grub/grub.conf for your snapshot volume.*

*Step 2) Boot into (or ensure you are already booted into) the real LVM volume:*

*# mount | grep VolGroup00  
/dev/mapper/VolGroup00-root on / type ext4 (rw)  
/dev/mapper/VolGroup00-var on /var type ext4 (rw)*

*Step 3) Remove the snapshot volumes:*

*# lvremove /dev/VolGroup00/rootsnapshot  
# lvremove /dev/VolGroup00/varsnapshot*

***Summary***

*To boot into an LVM snapshot of the root filesystem, you must change only the following locations:*

*/etc/fstab on the LVM snapshot volume (do not change fstab on the real volume)  
/boot/grub/grub.conf to add an entry that points to the snapshot device as the root disk.  
There is no need to rebuild initrd to boot into the snapshot.*

## ****17.How much memory is consumed by an LVM snapshot?****

*An LVM snapshot implements a “clone” of an existing logical volume (LV) by tracking “exceptions (a write request to an area on the origin volume will trigger the creation of an exception on the snapshot)” to the data on the origin volume. An exception tracks one or more “chunks” of data that has changed from the time the snapshot was taken. The size of a “chunk” of data is determined at the time of snapshot creation (see lvcreate man page, “–chunksize” option), and must be a power of 2, from 4kB to 512kB. The default chunk size is 4k.*

*Each exception will consume a small amount of memory in the kernel. The memory consumed by one exception can be found by examining the dm-snapshot-ex slab cache statistics in /proc/slabinfo.A single object in this cache is a single exception.*

*The LVM snapshot implementation is efficient, and if possible, it stores more than one chunk of data for each exception. In the worst case, only one chunk of data will be stored for each exception. In the best case, 255 consecutive chunks of data will be stored for each exception. The write I/O patterns determine whether LVM can store consecutive chunks of data in a single exception. If the I/O patterns are mostly sequential, more than one sequential chunk of data will be stored for each exception, and the memory usage will tend towards the best case memory usage. However, if the I/O pattern is more random, the memory usage will tend towards the worst case memory usage.*

* *i386 (32-bit)                    : 145 exceptions / page (assuming 4k page size, approximately 28 bytes / exception)*
* *x86\_64 (64-bit)             : 112 exceptions / page (assuming 4k page size, approximately 36.5 bytes / exception)*

*Approximate Calculation for Memory usage:*

*Calculating memory needs of an LVM snapshot may be done by using a simple calculation which involves the number of exceptions (assume each exception stores only one chunk of data), the number of exceptions per page, and the page size, in particular:*

***W = (N / E) \* P***

*where*

*W is worst case memory overhead, in bytes  
N is the worst case number of exceptions  
E is the number of exceptions per page of memory  
P is the page size, in bytes*

*Calculating number of exceptions is based on the size of the snapshot logical volume, and the chunksize:*

*In worst Case : N = S / C  
In Best Case    : N = S / (C \* 255)*

*where :*

*S is the size of the snapshot logical volume  
C is the chunksize  
In the best case, 255 chunks can be stored per exception*

*For Example:*

*Architecture: i386  
Origin LV size: 100GB  
Snapshot LV size: 10GB  
Chunksize: 512k*

*Worst case memory overhead : 10\*1024\*1024\*1024 / 524288 / 145 \* 4096 = 578,524 bytes  
Best Case Memory Overhead  : 10\*1024\*1024\*1024 / (524288\*255) / 145 \* 4096 = 2,268 bytes*

 Kernel Panic and System Crash

## What is the meaning of a Linux System Crash?

Crash is a generic term used usually to say that the system has come to halt and no progress is observed. The system seems unresponsive or has already rebooted.

**Kernel Panic** – A voluntary halt to all system activity when an abnormal situation is detected by the kernel. A Kernel panic is an action taken by an operating system upon detecting an Internal fatal error from which it cannot safely recover. And in Linux these Kernel Panics can be caused by different reasons

* + Hardware: Machine Check Exceptions
  + Error Detection and Correction (EDAC)
  + Non-Maskable Interrupts (NMIs)
    - Hardware NMI Button
    - NMI Watch Dog
    - unknown\_nmi\_panic
    - panic\_on\_unrecovered\_nmi
    - panic\_on\_io\_nmi
  + Software related BUG() macro
  + Software related  Bad pointer handling
  + Software related Pseudo-hangs
  + Software related Out-of-Memory killer

## Hardware: Machine Check Exceptions

Hardware Machine Check Exceptions normally caused by the the Component failures detected and reported by the hardware via an exception, and they typically looks like:

kernel: CPU 0: Machine Check Exception: 4  
Bank 0: b278c00000000175  
kernel: TSC 4d9eab664a9a60  
kernel: Kernel panic – not syncing: Machine check

***Sample Scenario 1 :***

*System hangs or kernel panics with MCE (Machine Check Exception) in /var/log/messages file.  
System was not responding. Checked the messages in netdump server. Found the following messages …”Kernel panic – not syncing: Machine check”.  
System crashes under load.  
System crashed and rebooted.  
Machine Check Exception panic*

***Troubleshooting Procedure Posted here***[***Redhat Enterprise Linux – Troubleshooting Kernel Panic issues – Part 2***](http://gurkulindia.com/main/2013/09/redhat-enterprise-linux-troubleshooting-kernel-panic-issues-part-2/)

## Error Detection and Correction (EDAC)

Normally, EDAC errors caused by Hardware mechanism to detect and report memory chip and PCI transfer errors, and reported in /sys/devices/system/edac/{mc/,pci} and logged by the kernel as:

EDAC MC0: CE page 0x283, offset 0xce0, grain 8,  
syndrome 0x6ec3, row 0, channel 1 “DIMM\_B1”:  
amd76x\_edac

 All the Informational EDAC messages (such as a corrected ECC error) are printed to the system log, where as critical EDAC messages (such as exceeding a hardware-defined temperature threshold) trigger a kernel panic.

***Sample Scenario 2 :***

*Console Screen having the messages as below*

*Northbridge Error, node 1, core: -1  
K8 ECC error.  
EDAC amd64 MC1: CE ERROR\_ADDRESS= 0x101a793400  
EDAC MC1: INTERNAL ERROR: row out of range (-22 >= 8)  
EDAC MC1: CE – no information available: INTERNAL ERROR  
EDAC MC1: CE – no information available: amd64\_edacError Overflow*

**Troubleshooting Procedure Posted here**[**Redhat Enterprise Linux – Troubleshooting Kernel Panic issues – Part 2**](http://gurkulindia.com/main/2013/09/redhat-enterprise-linux-troubleshooting-kernel-panic-issues-part-2/)

## Non-Maskable Interrupts (NMIs)

A Non maskable interrupt (NMI) is an interrupt that is unable to be ignored/masked out by standard operating system mechanisms. A non-maskable interrupt (NMI) cannot be ignored, and is generally used only for critical hardware errors however recent changes in behavior has added additional functionality of:

### ****1) NMI button.****

The NMI This can be used to signal the operating system when other standard input mechanisms (keyboard, ssh, network) have ceased to function.  
It can be used to create an intentional panic for additional debugging. It may not always be a physical button.  
It may be presented through an iLO or Drac Interface.

Unknown NMIs – The kernel has mechanisms to handle certain known NMIs appropriately, unknown ones typically result in kernel log warnings such as:

Uhhuh. NMI received.  
Dazed and confused, but trying to continue  
You probably have a hardware problem with your RAM chips  
Uhhuh. NMI received for unknown reason 32.  
Dazed and confused, but trying to continue.  
Do you have a strange power saving mode enabled?

These unknown NMI messages can be produced by ECC and other hardware problems. The kernel can be configured to panic when these are received  
   
though this sysctl:

kernel.unknown\_nmi\_panic=1  
   
This is generally only enabled for troubleshooting

***Sample Scenario 3:***

*The following error message appearing in /var/log/messages*

*kernel: Dazed and confused, but trying to continue  
kernel: Do you have a strange power saving mode enabled?  
kernel: Uhhuh. NMI received for unknown reason 21 on CPU 0  
kernel: Dazed and confused, but trying to continue  
kernel: Do you have a strange power saving mode enabled?  
kernel: Uhhuh. NMI received for unknown reason 31 on CPU 0.*

**Troubleshooting Procedure Posted here**[**Redhat Enterprise Linux – Troubleshooting Kernel Panic issues – Part 2**](http://gurkulindia.com/main/2013/09/redhat-enterprise-linux-troubleshooting-kernel-panic-issues-part-2/)

### ****2) A Watchdog-like software on the system that monitors for perceived system hang****s

The NMI watchdog monitors system interrupts and sends an NMI if the system appears to have hung.  
On a normal system hundreds of device and timer interrupts are received per second. If there are no interrupts in a 30 second interval\*,  
the NMI watchdog assumes that the system has hung and sends an NMI to the system to trigger a kernel panic or restart.

### ****How an NMI watchdog works****

A standard system level watchdog waits for regular events to fire and reboots the machine if no event is received within a designated timeframe. The NMI watchdog is no different. When using the NMI watchdog the system generates periodic NMI interrupts, and the kernel can monitor whether any CPU has locked up and print out debugging messages if so.

### ****Enabling NMI Watchdog****

The Red Hat Enterprise Linux 6 kernel is built with NMI watchdog support on currently supported x86 and x86-64 platforms.

### ****Ensure NMI is being used:****

For SMP machines and Single processor systems with an IO-APIC use nmi\_watchdog=1.

For Single processor systems without an IO-APIC use            nmi\_watchdog=2.

### ****Verification to check NMI watchdog working****

Boot the system with the the parameter as stated above and check the /proc/interrupts file for the “NMI count” line. This value should be non zero and increase over time. If the value is zero and does not increase over time the wrong NMI watchdog parameter has been used, change

If it is still zero then log a problem, you probably have a processor that needs to be added to the nmi code.

Here is an example from /etc/grub.conf for systems which utilize the GRUB boot loader:

title Red Hat Enterprise Linux Server (2.6.32-358.6.1.el6.x86\_64)  
root (hd0,0)  
kernel /vmlinuz-2.6.32-358.6.1.el6.x86\_64 ro root=/dev/mapper/vg\_worklaptop-lv\_root crashkernel=auto rd\_LVM\_LV=vg\_worklaptop/lv\_root rhgb quiet nmi\_watchdog=1  
initrd /initramfs-2.6.32-358.6.1.el6.x86\_64.img

To determine if the NMI watchdog was properly activated, check the /proc/interrupts file. The NMI interrupt should display a non-zero value. If the NMI interrupt displays a zero, alter the nmi\_watchdog value, restart the system, and examine this file again. If a zero is still displayed, then the processor in the test system is not supported by the NMI watchdog code.

The output, when functioning correctly, should look similar to the following:

[root@work-laptop wmealing]# cat /proc/interrupts | grep ^NMI  
NMI: 861 636 377 357 Non-maskable interrupts

Each processor core has an NMI count. These should all be increasing over time. The above example is a quad core system.

### ****System wide NMI settings****

The NMI settings can be configured at runtime by using the sysctl interface.

In the /etc/sysctl.conf, to enable, set:

kernel.nmi\_watchdog = 1

To disable, set:

            kernel.nmi\_watchdog = 0

Note that this does not enable the functionality, the kernel parameter is required to correctly enable the NMI watchdog.

### ****unknown\_nmi\_panic****

A feature was introduced in kernel 2.6.9 which helps to make easier the process of diagnosing system hangs on specific hardware.  
The feature utilizes the kernels behavior when dealing with unknown NMI sources. The behavior is to allow it to panic, rather than handle the unknown nmi source. This feature cannot be utilized on systems that also use the NMI Watchdog or some oprofile (and other tools that use performance metric features as both of these also make use of the undefined NMI interrupt. If unknown\_nmi\_panic is activated with one of these features present, it will not work.

Note that this is a user-initiated interrupt which is really most useful for helping to diagnose a system that is experiencing system hangs for unknown reasons.

To enable this feature, set the following system control parameter in the /etc/sysctl.conf file as follows:

kernel.unknown\_nmi\_panic = 1

To disable, set:

          kernel.unknown\_nmi\_panic = 0

Once this change has taken effect, a panic can be forced by pushing the system’s NMI switch. Systems that do not have an NMI switch can still use the NMI Watchdog feature which will automatically generate an NMI if a system hang is detected.

### ****panic\_on\_unrecovered\_nmi****

Some systems may generate an NMI based on vendor configuration, such as power management, low battery etc. It may be important to set this if your system is generating NMI’s in a known-working environment.

To enable this feature, set the following system control parameter in the /etc/sysctl.conf file as follows:

kernel.panic\_on\_unrecovered\_nmi = 1

To disable, set:

              kernel.panic\_on\_unrecovered\_nmi = 0

### ****panic\_on\_io\_nmi****

This setting was only available in Red Hat Enterprise Linux 6. When set, this will cause a kernel panic when the kernel receives an NMI caused by an Input/Output error.

***Sample Scenario 4 :***

*Console Shows following Error Message*

*NMI: IOCK error (debug interrupt?)  
CPU 0  
Modules linked in: ipt\_MASQUERADE iptable\_nat ip\_nat xt\_state ip\_conntrack nfnetlink ipt\_REJECT xt\_tcpudp iptable\_filter ip\_tables x\_tables bridge mptctl mptbase bonding be2iscsi ib\_iser rdma\_cm ib\_cm iw\_cm ib\_sa ib\_mad ib\_core ib\_addr iscsi\_tcp bnx2i cnic ipv6 xfrm\_nalgo crypto\_api uio cxgb3i cxgb3 8021q libiscsi\_tcp libiscsi2 scsi\_transport\_iscsi2 scsi\_transport\_iscsi dm\_round\_robin dm\_multipath scsi\_dh video backlight sbs power\_meter hwmon i2c\_ec i2c\_core dell\_wmi wmi button battery asus\_acpi acpi\_memhotplug ac parport\_pc lp parport joydev sr\_mod cdrom hpilo bnx2 serio\_raw shpchp pcspkr sg dm\_raid45 dm\_message dm\_region\_hash dm\_mem\_cache dm\_snapshot dm\_zero dm\_mirror dm\_log dm\_mod usb\_storage qla2xxx scsi\_transport\_fc ata\_piix libata cciss sd\_mod scsi\_mod ext3 jbd uhci\_hcd ohci\_hcd ehci\_hcd  
Pid: 0, comm: swapper Not tainted 2.6.18-194.17.4.el5 #1  
RIP: 0010:[<ffffffff8019d550>] [<ffffffff8019d550>] acpi\_processor\_idle\_simple+0x14c/0x30e  
RSP: 0018:ffffffff803fbf58 EFLAGS: 00000046  
RAX: 0000000000d4d87e RBX: ffff81061e10a160 RCX: 0000000000000908  
RDX: 0000000000000915 RSI: 0000000000000003 RDI: 0000000000000000  
RBP: 0000000000d4d87e R08: ffffffff803fa000 R09: 0000000000000039  
R10: ffff810001005710 R11: 0000000000000000 R12: 0000000000000000  
R13: ffff81061e10a000 R14: 0000000000000000 R15: 0000000000000000  
FS: 0000000000000000(0000) GS:ffffffff803ca000(0000) knlGS:0000000000000000  
CS: 0010 DS: 0018 ES: 0018 CR0: 000000008005003b  
CR2: 0000000009013954 CR3: 000000060799d000 CR4: 00000000000006e0  
Process swapper (pid: 0, threadinfo ffffffff803fa000, task ffffffff80308b60)  
Stack: ffff81061e10a000 ffffffff8019d404 0000000000000000 ffffffff8019d404  
0000000000090000 0000000000000000 0000000000000000 ffffffff8004923a  
0000000000200800 ffffffff80405807 0000000000090000 0000000000000000  
Call Trace:  
[<ffffffff8019d404>] acpi\_processor\_idle\_simple+0x0/0x30e  
[<ffffffff8019d404>] acpi\_processor\_idle\_simple+0x0/0x30e  
[<ffffffff8004923a>] cpu\_idle+0x95/0xb8  
[<ffffffff80405807>] start\_kernel+0x220/0x225  
[<ffffffff8040522f>] \_sinittext+0x22f/0x236  
   
Code: 89 ca ed ed 41 89 c4 41 8a 45 1c 83 e0 30 3c 30 75 15 f0 ff*

**Troubleshooting Procedure Posted here**[**Redhat Enterprise Linux – Troubleshooting Kernel Panic issues – Part 2**](http://gurkulindia.com/main/2013/09/redhat-enterprise-linux-troubleshooting-kernel-panic-issues-part-2/)

## Software: The BUG() macro

This kind of  kernel panic normally caused by the kernel code when an abnormal situation is seen , that indicates a programming error . And normally the Output looks like:

Kernel BUG at spinlock:118  
invalid operand: 0000 [1] SMP  
CPU 0

***Sample Scenario 5:***

*NFS client kernel crash because async task already queued hitting BUG\_ON(RPC\_IS\_QUEUED(task)); in \_\_rpc\_execute  
kernel BUG at net/sunrpc/sched.c:616!  
invalid opcode: 0000 [#1] SMP  
last sysfs file: /sys/devices/system/cpu/cpu15/cache/index2/shared\_cpu\_map  
CPU 8  
Modules linked in: nfs lockd fscache nfs\_acl auth\_rpcgss pcc\_cpufreq sunrpc power\_meter hpilo  
hpwdt igb mlx4\_ib(U) mlx4\_en(U) raid0 mlx4\_core(U) sg microcode serio\_raw iTCO\_wdt  
iTCO\_vendor\_support ioatdma dca shpchp ext4 mbcache jbd2 raid1 sd\_mod crc\_t10dif mpt2sas  
scsi\_transport\_sas raid\_class ahci dm\_mirror dm\_region\_hash dm\_log dm\_mod  
[last unloaded: scsi\_wait\_scan]  
   
Pid: 2256, comm: rpciod/8 Not tainted 2.6.32-220.el6.x86\_64 #1 HP ProLiant SL250s Gen8/  
RIP: 0010:[<ffffffffa01fe458>] [<ffffffffa01fe458>] \_\_rpc\_execute+0x278/0x2a0 [sunrpc]  
…  
Process rpciod/8 (pid: 2256, threadinfo ffff882016152000, task ffff8820162e80c0)  
…  
Call Trace:  
[<ffffffffa01fe4d0>] ? rpc\_async\_schedule+0x0/0x20 [sunrpc]  
[<ffffffffa01fe4e5>] rpc\_async\_schedule+0x15/0x20 [sunrpc]  
[<ffffffff8108b2b0>] worker\_thread+0x170/0x2a0  
[<ffffffff81090bf0>] ? autoremove\_wake\_function+0x0/0x40  
[<ffffffff8108b140>] ? worker\_thread+0x0/0x2a0  
[<ffffffff81090886>] kthread+0x96/0xa0  
[<ffffffff8100c14a>] child\_rip+0xa/0x20  
Code: db df 2e e1 f6 05 e0 26 02 00 40 0f 84 48 fe ff ff 0f b7 b3 d4 00 00 00 48 c7  
c7 94 39 21 a0 31 c0 e8 b9 df 2e e1 e9 2e fe ff ff <0f> 0b eb fe 0f b7 b7 d4 00 00 00  
31 c0 48 c7 c7 60 63 21 a0 e8  
RIP [<ffffffffa01fe458>] \_\_rpc\_execute+0x278/0x2a0 [sunrpc]*

**Troubleshooting Procedure Posted here**[**Redhat Enterprise Linux – Troubleshooting Kernel Panic issues – Part 2**](http://gurkulindia.com/main/2013/09/redhat-enterprise-linux-troubleshooting-kernel-panic-issues-part-2/)

## Software: Bad pointer handling

This kind of kernel panics typically indicates a programming error and normally appear as below:

NULL pointer dereference at 0x1122334455667788 ..  
or  
Unable to handle kernel paging request at virtual address 0x11223344

One of the most common reason for this kind of error is possible memory corruption

***Sample Scenario 6 :***

* *NFS client kernel panics when doing an ls in the directory of a snapshot that has already been removed.*
* *NFS client kernel panics under certain conditions when connected to NFS server either NetApp or Solaris ZFS*
* *Kernel crashes with message*

*BUG: unable to handle kernel NULL pointer dereference at 0000000000000018  
IP: [<ffffffff81192957>] commit\_tree+0x77/0x100  
PGD 7ff2e69067 PUD 7feaf59067 PMD 0  
Oops: 0000 [#1] SMP  
last sysfs file: /sys/devices/pci0000:00/0000:00:03.0/0000:07:00.0/vendor  
CPU 64  
Modules linked in: nls\_utf8 fuse mptctl mptbase autofs4 nfs lockd fscache(T) nfs\_acl auth\_rpcgss bnx2fc cnic uio fcoe libfcoe libfc scsi\_transport\_fc scsi\_tgt 8021q garp stp llc smbus(U) ipmi\_devintf ipmi\_si ipmi\_msghandler sunrpc cpufreq\_ondemand acpi\_cpufreq freq\_table nf\_conntrack\_ftp ipt\_REJECT ipt\_LOG iptable\_filter ipt\_MASQUERADE iptable\_nat nf\_nat nf\_conntrack\_ipv4 nf\_defrag\_ipv4 ip\_tables ip6t\_REJECT nf\_conntrack\_ipv6 nf\_defrag\_ipv6 xt\_state nf\_conntrack ip6table\_filter ip6\_tables ipv6 vfat fat dm\_mirror dm\_region\_hash dm\_log microcode sg i2c\_i801 i2c\_core iTCO\_wdt iTCO\_vendor\_support ioatdma i7core\_edac edac\_core ixgbe mdio igb dca ext4 mbcache jbd2 sr\_mod cdrom sd\_mod crc\_t10dif ata\_generic pata\_acpi ata\_piix megaraid\_sas dm\_mod [last unloaded: scsi\_wait\_scan]  
   
Modules linked in: nls\_utf8 fuse mptctl mptbase autofs4 nfs lockd fscache(T) nfs\_acl auth\_rpcgss bnx2fc cnic uio fcoe libfcoe libfc scsi\_transport\_fc scsi\_tgt 8021q garp stp llc smbus(U) ipmi\_devintf ipmi\_si ipmi\_msghandler sunrpc cpufreq\_ondemand acpi\_cpufreq freq\_table nf\_conntrack\_ftp ipt\_REJECT ipt\_LOG iptable\_filter ipt\_MASQUERADE iptable\_nat nf\_nat nf\_conntrack\_ipv4 nf\_defrag\_ipv4 ip\_tables ip6t\_REJECT nf\_conntrack\_ipv6 nf\_defrag\_ipv6 xt\_state nf\_conntrack ip6table\_filter ip6\_tables ipv6 vfat fat dm\_mirror dm\_region\_hash dm\_log microcode sg i2c\_i801 i2c\_core iTCO\_wdt iTCO\_vendor\_support ioatdma i7core\_edac edac\_core ixgbe mdio igb dca ext4 mbcache jbd2 sr\_mod cdrom sd\_mod crc\_t10dif ata\_generic pata\_acpi ata\_piix megaraid\_sas dm\_mod [last unloaded: scsi\_wait\_scan]  
Pid: 79910, comm: ls Tainted: G —————- T 2.6.32-131.6.1.el6.x86\_64 #1 PRIMERGY RX900 S1  
RIP: 0010:[<ffffffff81192957>] [<ffffffff81192957>] commit\_tree+0x77/0x100  
RSP: 0018:ffff885f1484dab8 EFLAGS: 00010246  
RAX: ffff881f5f43d3e8 RBX: ffff885f1484dab8 RCX: ffff885f1484dab8  
RDX: ffff881f5f43d3e8 RSI: ffff881f5f43d3e8 RDI: ffff885f1484dab8  
RBP: ffff885f1484dae8 R08: ffff881f5f43d3e8 R09: 0000000000000000  
R10: ffff882080440a40 R11: 0000000000000000 R12: 0000000000000000  
R13: ffff881f5f43d380 R14: ffff881f5fcba2c0 R15: 0000000000000000  
FS: 00007f9b188177a0(0000) GS:ffff88011c700000(0000) knlGS:0000000000000000  
CS: 0010 DS: 0000 ES: 0000 CR0: 000000008005003b  
CR2: 0000000000000018 CR3: 0000007fecaf5000 CR4: 00000000000006e0  
DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000  
DR3: 0000000000000000 DR6: 00000000ffff0ff0 DR7: 0000000000000400  
Process ls (pid: 79910, threadinfo ffff885f1484c000, task ffff881fc4164b00)  
Stack:  
ffff881f5f43d3e8 ffff881f5f43d3e8 ffff881f5f43d380 ffff885f1484db08  
<0> ffff881f5fcba2c0 ffff885f1484ddd8 ffff885f1484db48 ffffffff81192c6f  
<0> ffff881c94a4d200 000000001484dbf8 ffff885f1484db08 ffff885f1484db08  
Call Trace:  
[<ffffffff81192c6f>] attach\_recursive\_mnt+0x28f/0x2a0  
[<ffffffff81192d80>] graft\_tree+0x100/0x140  
[<ffffffff814dc686>] ? down\_write+0x16/0x40  
[<ffffffff81192e5f>] do\_add\_mount+0x9f/0x160  
[<ffffffffa045ce2f>] nfs\_follow\_mountpoint+0x1bf/0x570 [nfs]  
[<ffffffff811810a0>] do\_follow\_link+0x120/0x440  
[<ffffffffa03112e0>] ? put\_rpccred+0x50/0x150 [sunrpc]  
[<ffffffff81180eeb>] \_\_link\_path\_walk+0x78b/0x820  
[<ffffffff8118164a>] path\_walk+0x6a/0xe0  
[<ffffffff8118181b>] do\_path\_lookup+0x5b/0xa0  
[<ffffffff811819a7>] user\_path\_at+0x57/0xa0  
[<ffffffff81041594>] ? \_\_do\_page\_fault+0x1e4/0x480  
[<ffffffff810ce97d>] ? audit\_filter\_rules+0x2d/0xa10  
[<ffffffff81177cac>] vfs\_fstatat+0x3c/0x80  
[<ffffffff81177d5e>] vfs\_lstat+0x1e/0x20  
[<ffffffff81177d84>] sys\_newlstat+0x24/0x50  
[<ffffffff810d1ad2>] ? audit\_syscall\_entry+0x272/0x2a0  
[<ffffffff814e054e>] ? do\_page\_fault+0x3e/0xa0  
[<ffffffff8100b172>] system\_call\_fastpath+0x16/0x1b  
Code: 83 e8 68 eb 12 0f 1f 80 00 00 00 00 4c 89 a0 c0 00 00 00 48 8d 42 98 48 8b 50 68 48 8d 48 68 48 39 cb 0f 18 0a 75 e5 48 8b 45 d0 <49> 8b 54 24 18 48 39 d8 74 15 48 8b 0a 48 8b 5d d8 48 89 50 08  
RIP [<ffffffff81192957>] c  
ommit\_tree+0x77/0x100  
RSP <ffff885f1484dab8>  
CR2: 0000000000000018*

**Troubleshooting Procedure Posted here**[**Redhat Enterprise Linux – Troubleshooting Kernel Panic issues – Part 2**](http://gurkulindia.com/main/2013/09/redhat-enterprise-linux-troubleshooting-kernel-panic-issues-part-2/)

## Software: Pseudo-hangs

This are the common situations, that we commonly encounter where the system appears to be hung, but some progress is being made, there are several reasons for this kind of behaviour, and they are

* **Livelock**  if running a realtime kernel, application load could be too high, leading the system into a state where it becomes effectively unresponsive in a “live lock/ busy wait” state. The system is not actually hung, but just moving so slowly that it appears to be hung.
* **Thrashing** – continuous swapping with close to no useful processing done
* **Lower zone starvation** – on i386 the low memory has a special significance and the system may “hang” even when there’s plenty of free memory
* **Memory starvation** in one node in a NUMA system

Normally, Hangs which are not detected by the hardware are trickier to debug:

* Use [sysrq + t] to collect process stack traces when possible
* Enable the NMI watchdog which should detect those situations
* Run hardware diagnostics when it’s a hard hang: memtest86, HP diagnostics

***Sample Scenario 7:***

*The system is frequently getting hung and following error messages are getting logged in /var/log/messages file while performing IO operations on the /dev/cciss/xx devices:*

*INFO: task cmaperfd:5628 blocked for more than 120 seconds.  
“echo 0 > /proc/sys/kernel/hung\_task\_timeout\_secs” disables this message.  
cmaperfd D ffff810009025e20 0 5628 1 5655 5577 (NOTLB)  
ffff81081bdc9d18 0000000000000082 0000000000000000 0000000000000000  
0000000000000000 0000000000000007 ffff81082250f040 ffff81043e100040  
0000d75ba65246a4 0000000001f4db40 ffff81082250f228 0000000828e5ac68  
Call Trace:  
[<ffffffff8803bccc>] :jbd2:start\_this\_handle+0x2ed/0x3b7  
[<ffffffff800a3c28>] autoremove\_wake\_function+0x0/0x2e  
[<ffffffff8002d0f4>] mntput\_no\_expire+0x19/0x89  
[<ffffffff8803be39>] :jbd2:jbd2\_journal\_start+0xa3/0xda  
[<ffffffff8805e7b0>] :ext4:ext4\_dirty\_inode+0x1a/0x46  
[<ffffffff80013deb>] \_\_mark\_inode\_dirty+0x29/0x16e  
[<ffffffff80041bf5>] inode\_setattr+0xfd/0x104  
[<ffffffff8805e70c>] :ext4:ext4\_setattr+0x2db/0x365  
[<ffffffff88055abc>] :ext4:ext4\_file\_open+0x0/0xf5  
[<ffffffff8002cf2b>] notify\_change+0x145/0x2f5  
[<ffffffff800e45fe>] sys\_fchmod+0xb3/0xd7*

**Troubleshooting Procedure Posted here**[**Redhat Enterprise Linux – Troubleshooting Kernel Panic issues – Part 2**](http://gurkulindia.com/main/2013/09/redhat-enterprise-linux-troubleshooting-kernel-panic-issues-part-2/)

## Software: Out-of-Memory killer

In certain memory starvation cases, the OOM killer is triggered to force the release of some memory by killing a “suitable” process.  In severe starvation cases, the OOM killer may have to panic the system when no killable processes are found:

Kernel panic – not syncing:  
Out of memory and no killable processes…

The kernel can also be configured to always panic during an OOM by setting the vm.panic\_on\_oom = 1 sysctl.

***Sample Scenario 8 :***

*When the system panics kdump starts, but kdump hangs and does not output a vmcore. I see following error messages on the console:*

*Kernel panic - not syncing: Out of memory and no killable processes...*

## What is Kdump?

Starting in Red Hat Enterprise Linux 5, kernel crash dumps are captured using the kdump mechanism.  Kexec is used to start another complete copy of the Linux kernel in a reserved area of memory. This secondary kernel takes over and copies the memory pages to the crash dump location.

## How do we configure the kdump ?

**In brief it has following steps:**

*Step 1. Install kexec-tools  
Step 2. Edit /etc/grub.conF, and add the ” crashkernel=<reservered-memory-setting> ” at the end of kernel line*

***Example for RHEL 5:***

*title Red Hat Enterprise Linux Client (2.6.17-1.2519.4.21.el5)  
root (hd0,0)  
kernel /boot/vmlinuz-2.6.17-1.2519.4.21.el5 ro root=LABEL=/ rhgb quiet crashkernel=128M@16M  
initrd /boot/initrd-2.6.17-1.2519.4.21.el5.img*

*crashkernel=memory@offset*

*+—————————————+  
| RAM | crashkernel | crashkernel |  
| size | memory | offset |  
|———–+————-+————-|  
| 0 – 2G | 128M | 16 |  
| 2G – 6G | 256M | 24 |  
| 6G – 8G | 512M | 16 |  
| 8G – 24G | 768M | 32 |  
+—————————————+*

***Example for RHEL 6:***

*title Red Hat Enterprise Linux Server (2.6.32-71.7.1.el6.x86\_64)  
root (hd0,0)  
kernel /vmlinuz-2.6.32-71.7.1.el6.x86\_64 ro root=/dev/mapper/vg\_example-lv\_root rd\_LVM\_LV=vg\_example/lv\_root rd\_LVM\_LV=vg\_example/lv\_swap rd\_NO\_LUKS rd\_NO\_MD rd\_NO\_DM LANG=en\_US.UTF-8 SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=us crashkernel=128M rhgb quiet  
initrd /initramfs-2.6.32-71.7.1.el6.x86\_64.img*

*Guidelines for Crash Kernel Reserved Memory Settings:*

*crashkernel=0M to 2G: 128M, 2G-6G:256M, 6G-8G:512M,8G-:768M*

*Ram Size CrashKernel  
>0GB 128MB  
>2GB 256MB  
>6GB 512MB  
>8GB 768MB*

*Step 3. configure /etc/kdump.conf*

*3A) to specify the destination to send the output of kexec,i.e. vmcore. Following destinations can be used*

*– raw device : raw /dev/sda4  
– file : ext3 /dev/sda3 , that will dump vmcore to /dev/sda3:/var/crash  
– NFS share : net nfs.example.com:/export/vmcores  
– Another system via SSH : net kdump@crash.example.com*

*3B) Configure Core Collector, to discard unnecessary memory pages and compress the only needed ones*

*Option    Discard  
1               Zero pages  
2              Cache pages  
4              Cache private  
8              User pages  
16            Free pages*

***To  discard all optional pages:***

*core\_collector makedumpfile -d 31 -c 4. reboot the server with crashkernel=<reserved-memory-setting>5. Start kdump service6. Enable the kdump for autostart on*

***Important Notes about Redhat Linux on HP ASR ( AUTOMATED SERVER RECOVERY ) FEATURE***

*According to the Automatic System Recovery definition on HP website, “If the internal health LED on a server is amber, the server might shutdown or reboot as soon as the hp Insight Manager agents are installed. This feature is called Automatic Server Recovery, and is enabled to ensure that the server can be recovered even though there is a major hardware problem.”*

*The Automatic System Recovery generally occurs during low server utilization for an extended period of time.*

*The Automatic System Recovery feature is a 10 minute timer. If the OS stops communicating the server reboots. It is implemented using a “heartbeat” timer that continually counts down. The hpasm driver frequently reloads the counter to prevent it from counting down to zero. If the Automatic System Recovery timer counts down to 0, it is assumed that the operating system is locked up and the system automatically attempts to reboot.*

*The logs of system lockup, the Automatic System Recovery and hardware failure are logged in IML . You can check the IML through the ILO interface.*

*To check whether ASR is enabled in the BIOS or not, we can use the below command.*

*# hpasmcli -s “show asr”*

*We have to make sure, that ASR timeout value should not interrupt the complete core collection. So it is often required to*

*Disable ASR using the command :*

*# hpasmcli -s ‘DISABLE ASR’*

*Or set longer timeout using the command:*

*# hpasmcli -s ‘SET ASR 30’*

## How do we analyse the VMCORE file collected using kdump?

If we assume that we configured that /var/crash as destination for vmcore in /etc/kdump.conf, we will see the file as below

# ls -l /var/crash/127.0.0.1-2013-09-21-19:45:17/vmcore  
-rw——-. 1 root root 490958682 Sep 21 18:46 /var/crash/127.0.0.1-2013-09-21-19:45:17/vmcore  
   
To Analyse the vmcore we need the “crash” utility. And we need to install following packages to get started with vmcore analysis

# yum install crash  
# yum install kernel-debuginfo-2.6.32-220.23.1.el6.x86\_64

Purpose of Debuginfo package : Debugging symbols are stripped out of the standard kernel for performance and size reasons. To analyze the vmcore, separate debugging information needs to be provided. This is specific to the exact revision of the kernel which crashed. Debuginfo package will help us to analyse these symbols from the vmcore.

## How to Use Crash Utility ?

At this level We will only discuss about basic usage of crash utility from the linux administration point of view.

* **log** – Display the kernel ring buffer log. On a running system, dmesg also displays the kernel ring buffer log.

Often times this can capture log messages that were not written to disk due to the crash.

crash> log  
— snip —  
SysRq : Trigger a crash  
BUG: unable to handle kernel NULL pointer dereference at (null)  
IP: [<ffffffff8130e126>] sysrq\_handle\_crash+0x16/0x20  
PGD 7a602067 PUD 376ff067 PMD 0  
Oops: 0002 [#1] SMP

* **kmem -i** – Show available memory at time of crash
* **ps** – Show running processes at time of crash. Useful with grep
* **net** – Show configured network interfaces at time of crash
* **bt** – Backtraces are read upside-down, from bottom to top

crash> bt  
PID: 6875 TASK: ffff88007a3aaa70 CPU: 0 COMMAND: “bash”  
#0 [ffff88005f0f5de8] sysrq\_handle\_crash at ffffffff8130e126  
#1 [ffff88005f0f5e20] \_\_handle\_sysrq at ffffffff8130e3e2  
#2 [ffff88005f0f5e70] write\_sysrq\_trigger at ffffffff8130e49e  
#3 [ffff88005f0f5ea0] proc\_reg\_write at ffffffff811cfdce  
#4 [ffff88005f0f5ef0] vfs\_write at ffffffff8116d2e8  
#5 [ffff88005f0f5f30] sys\_write at ffffffff8116dd21  
#6 [ffff88005f0f5f80] system\_call\_fastpath at ffffffff81013172  
RIP: 00000037702d4230 RSP: 00007fff85b95f40 RFLAGS: 00010206  
RAX: 0000000000000001 RBX: ffffffff81013172 RCX: 0000000001066300  
RDX: 0000000000000002 RSI: 00007f04ae8d2000 RDI: 0000000000000001  
RBP: 00007f04ae8d2000 R8: 000000000000000a R9: 00007f04ae8c4700  
R10: 00000000ffffffff R11: 0000000000000246 R12: 0000000000000002  
R13: 0000003770579780 R14: 0000000000000002 R15: 0000003770579780  
ORIG\_RAX: 0000000000000001 CS: 0033 SS: 002b

* **sys** – Displays system data – same information displayed when crash starts

crash> sys  
DUMPFILE: /tmp/vmcore [PARTIAL DUMP]  
CPUS: 2  
DATE: Thu May 5 14:32:50 2011  
UPTIME: 00:01:15  
LOAD AVERAGE: 1.19, 0.34, 0.12  
TASKS: 252  
NODENAME: rhel6-desktop  
RELEASE: 2.6.32-220.23.1.el6.x86\_64  
VERSION: #1 SMP Mon Oct 29 19:45:17 EDT 2012  
MACHINE: x86\_64 (3214 Mhz)  
MEMORY: 2 GB  
PANIC: “Oops: 0002 [#1] SMP ” (check log for details)  
PID: 6875  
COMMAND: “bash”  
TASK: ffff88007a3aaa70 [THREAD\_INFO: ffff88005f0f4000]  
CPU: 0  
STATE: TASK\_RUNNING (PANIC)

* **dmesg** – To check the kernel log from vmcore output

crash> dmesg  
— snip —  
CPU 0: Machine Check Exception: 0000000000000004  
Kernel panic – not syncing: Unable to continue  
Redirect Crash output to Regular Commands

Example 1:                crash> log > log.txt  
Example 2:                crash> ps | fgrep bash | wc -l

## Sample Scenario 1 :

System hangs or kernel panics with MCE (Machine Check Exception) in /var/log/messages file.  
System was not responding. Checked the messages in netdump server. Found the following messages …”Kernel panic – not syncing: Machine check”.  
System crashes under load.  
System crashed and rebooted.  
Machine Check Exception panic

### Observations:

Look for the phrase “Machine Check Exception” in the log just before the panic message. If this message occurs, the rest of the panic message is of no interest.

Analyze vmcore

$crash /path/to/2.6.18-128.1.6.el5/vmlinux vmcore  
   
KERNEL: ./usr/lib/debug/lib/modules/2.6.18-128.1.6.el5/vmlinux  
DUMPFILE: 563523\_vmcore [PARTIAL DUMP]  
CPUS: 4  
DATE: Thu Feb 21 00:32:46 2011  
UPTIME: 14 days, 17:46:38  
LOAD AVERAGE: 1.14, 1.20, 1.18  
TASKS: 220  
NODENAME: gurkulnode1  
RELEASE: 2.6.18-128.1.6.el5  
VERSION: #1 SMP Tue Mar 24 12:05:57 EDT 2009  
MACHINE: x86\_64 (2599 Mhz)  
MEMORY: 7.7 GB  
PANIC: “Kernel panic – not syncing: Uncorrected machine check”  
PID: 0  
COMMAND: “swapper”  
TASK: ffffffff802eeae0 (1 of 4) [THREAD\_INFO: ffffffff803dc000]  
CPU: 0  
STATE: TASK\_RUNNING (PANIC)  
   
crash> log  
…  
CPU 0: Machine Check Exception:7 Bank 4: b40000000005001b  
RIP 10:<ffffffff8006b2b0> {default\_idle+0x29/0x50}  
TSC bc34c6f78de8f ADDR 17fe30000  
This is not a software problem!  
Run through mcelog –ascii to decode and contact your hardware vendor  
Kernel panic – not syncing: Uncorrected machine check

Process error through mcelog –ascii, specify –k8 for events that are for AMD processors and –p4 for a Pentium 4 or Xeon. This resulting information might be helpful to your hardware vendor.

$ cat > mcelog.txt  
CPU 0: Machine Check Exception:7 Bank 4: b40000000005001b  
RIP 10:<ffffffff8006b2b0> {default\_idle+0x29/0x50}  
TSC bc34c6f78de8f ADDR 17fe30000  
[ctrl]+[d]  
   
$ mcelog –ascii –k8 < mcelog.txt  
HARDWARE ERROR. This is \*NOT\* a software problem!  
Please contact your hardware vendor  
CPU 0 4 northbridge TSC bc34c6f78de8f  
RIP 10:ffffffff8006b2b0  
Northbridge GART error  
bit61 = error uncorrected  
TLB error ‘generic transaction, level generic’  
STATUS b40000000005001b MCGSTATUS 7  
RIP: default\_idle+0x29/0x50}

**Observations and Recommended Solution :** The information printed by the kernel (the line printed immediately before the panic message) comes from the hardware and should be provided to the hardware support person for analysis. This information should resemble the following:

CPU 10: Machine Check Exception: 4 Bank 0: b200000410000800

Machine Check Exception (MCE) is an error that occurs when a computer’s CPU detects a hardware problem. Typically, the impending hardware failure will cause the kernel to panic in order to protect against data corruption.

Normally at this level we do engage the hardware vendor for further troubleshooting and diagnosis; the message present in the logs just before the kernel panic should be given to hardware support.

**How to run a memory test ?**

Red Hat Enterprise Linux ships a memory test tool called memtest86+. It is a bootable utility that tests physical memory by writing various patterns to it and reading them back. Since memtest86+ runs directly off the hardware it does not require any operating system support for execution.

This tool is available as an RPM package from Red Hat Network (RHN) as well as a boot option from the Red Hat Enterprise Linux rescue disk.

To boot memtest86+ from the rescue disk, you will need to boot your system from CD 1 of the Red Hat Enterprise Linux installation media, and type the following at the boot prompt (before the Linux kernel is started):

boot: memtest86

If you would rather install memtest86+ on the system, here is an example of how to do it on a Red Hat Enterprise Linux 5 machine registered to RHN:

# yum install memtest86+

For the Red Hat Enterprise Linux version 4, perform the following command to install memtest86+. Make sure current system has been registered to RHN:

# up2date -i memtest86+

Then you will have to configure it to run on next reboot:

# memtest-setup

After reboot, the GRUB menu will list memtest. Select this item and it will start testing the memory.

Please note that once memtest86+ is running it will never stop unless you interrupt it by pressing the Esc key. It is usually a good idea to let it run for a few hours so it has time to test each block of memory several times.

memtest86+ may not always find all memory problems. It is possible that the system memory can have a fault that memtest86+ does not detect.

## Sample Scenario 2

Console Screen having the messages as below  
   
Northbridge Error, node 1, core: -1  
K8 ECC error.  
EDAC amd64 MC1: CE ERROR\_ADDRESS= 0x101a793400  
EDAC MC1: INTERNAL ERROR: row out of range (-22 >= 8)  
EDAC MC1: CE – no information available: INTERNAL ERROR  
EDAC MC1: CE – no information available: amd64\_edacError Overflow

**Observations :**

check for any other related Kernel Errors from /var/log/messages

kernel: [Hardware Error]: Northbridge Error (node 1): DRAM ECC error detected on the NB.  
kernel: [Hardware Error]: cache level: L3/GEN, mem/io: MEM, mem-tx: RD, part-proc: RES (no timeout)  
kernel: [Hardware Error]: CPU:2 MC4\_STATUS[Over|CE|MiscV|-|AddrV|CECC]: 0xdc2c410000000a13  
kernel: [Hardware Error]: MC4\_ADDR: 0x0000000210b67d90

When the EDAC core monitoring module and various supported chipset drivers are loaded, whenever an error has been detected, an error message will get logged in the syslog message file. Based on the Keywords used in the Error message we can decode the criticality of the error, below are the guideline

– “Non-Fatal Error” – Recoverable error  
– “DRAM Controller” – In memory controller module  
– MC0 – Memory controller 0  
– CE – Correctable Error Page – in memory page  
– 0xc6397 Offset – offset into that page at 0x0  
– Grain – accuracy of reporting  
– Syndrome – error bits from controller (specific)  
– Channel – which memory channel (often only channel 0 on machines)  
– Row – which DIMM row. How that maps to a chip is vendor specific but often as simple as row 0/1 -> DIMM0 row 2/3 > DIMM1 etc  
– label “” – description of this DIMM (NULL string in U3)  
– e752x – chip type (eg e7501, AMD76x)

Memory error checking on a memory module used to be accomplished with a parity checking bit that was attached to each byte of memory. The parity bit was calculated when each byte of memory was written, and then verified when each byte of memory was read. If the stored parity bit didn’t match the calculated parity bit on a read, that byte of memory was known to have changed. Parity checking is known to be a reasonably effective method for detecting a single bit change in a byte of memory.

**Recommended Solution**

— EDAC messages, errors, and warnings are often indicative of a hardware problem such as a memory module failure or memory controller failure.  
If EDAC errors are encountered running a hardware diagnostic and contacting your hardware vendor are advised.

— In some cases EDAC errors can be thrown due to a bug in the EDAC kernel module, the kernel, or due to an incompatibility between the system’s chipset and the

## Sample Scenario 3

The following error message appearing in /var/log/messages  
   
kernel: Dazed and confused, but trying to continue  
kernel: Do you have a strange power saving mode enabled?  
kernel: Uhhuh. NMI received for unknown reason 21 on CPU 0  
kernel: Dazed and confused, but trying to continue  
kernel: Do you have a strange power saving mode enabled?  
kernel: Uhhuh. NMI received for unknown reason 31 on CPU 0.

**Observations :**

The above message is typically output when system hardware has generated a non-maskable interrupt not recognized by the kernel. If there are any codes associated with the fault these may be trapped by hpasm software or equivalent HW Vendor monitoring software and logged there, but the fact that the NMI is not known to the kernel suggests the problem is a fundamental hardware issue. NMI 21 and 31 events typically indicate faulty RAM or perhaps CPU. However, hardware issues relating to the motherboard cannot be ruled out.

Schedule some downtime in order to run hardware diagnostics. A standard memtest86 could be run, but this is not guaranteed to find all possible memory issues. memtest86 can be initiated on a system by booting from a RHEL 5 DVD. Following this, any available manufacturer specific hardware diagnostics tools should be run.

**Recommeded Solution:**

If the issue happens frequenly on a production machine and that causes a system crash or panic, then first to with a BIOS firmware upgrade for the hardware. Should that not resolve the issue, have the support vendor replace faulty hardware, such as memory, CPU or motherboard.

## ****Sample Scenario 4****

Console Shows following Error Message

NMI: IOCK error (debug interrupt?)  
CPU 0  
Modules linked in: ipt\_MASQUERADE iptable\_nat ip\_nat xt\_state ip\_conntrack nfnetlink ipt\_REJECT xt\_tcpudp iptable\_filter ip\_tables x\_tables bridge mptctl mptbase bonding be2iscsi ib\_iser rdma\_cm ib\_cm iw\_cm ib\_sa ib\_mad ib\_core ib\_addr iscsi\_tcp bnx2i cnic ipv6 xfrm\_nalgo crypto\_api uio cxgb3i cxgb3 8021q libiscsi\_tcp libiscsi2 scsi\_transport\_iscsi2 scsi\_transport\_iscsi dm\_round\_robin dm\_multipath scsi\_dh video backlight sbs power\_meter hwmon i2c\_ec i2c\_core dell\_wmi wmi button battery asus\_acpi acpi\_memhotplug ac parport\_pc lp parport joydev sr\_mod cdrom hpilo bnx2 serio\_raw shpchp pcspkr sg dm\_raid45 dm\_message dm\_region\_hash dm\_mem\_cache dm\_snapshot dm\_zero dm\_mirror dm\_log dm\_mod usb\_storage qla2xxx scsi\_transport\_fc ata\_piix libata cciss sd\_mod scsi\_mod ext3 jbd uhci\_hcd ohci\_hcd ehci\_hcd  
Pid: 0, comm: swapper Not tainted 2.6.18-194.17.4.el5 #1  
RIP: 0010:[<ffffffff8019d550>] [<ffffffff8019d550>] acpi\_processor\_idle\_simple+0x14c/0x30e  
RSP: 0018:ffffffff803fbf58 EFLAGS: 00000046  
RAX: 0000000000d4d87e RBX: ffff81061e10a160 RCX: 0000000000000908  
RDX: 0000000000000915 RSI: 0000000000000003 RDI: 0000000000000000  
RBP: 0000000000d4d87e R08: ffffffff803fa000 R09: 0000000000000039  
R10: ffff810001005710 R11: 0000000000000000 R12: 0000000000000000  
R13: ffff81061e10a000 R14: 0000000000000000 R15: 0000000000000000  
FS: 0000000000000000(0000) GS:ffffffff803ca000(0000) knlGS:0000000000000000  
CS: 0010 DS: 0018 ES: 0018 CR0: 000000008005003b  
CR2: 0000000009013954 CR3: 000000060799d000 CR4: 00000000000006e0  
Process swapper (pid: 0, threadinfo ffffffff803fa000, task ffffffff80308b60)  
Stack: ffff81061e10a000 ffffffff8019d404 0000000000000000 ffffffff8019d404  
0000000000090000 0000000000000000 0000000000000000 ffffffff8004923a  
0000000000200800 ffffffff80405807 0000000000090000 0000000000000000  
Call Trace:  
[<ffffffff8019d404>] acpi\_processor\_idle\_simple+0×0/0x30e  
[<ffffffff8019d404>] acpi\_processor\_idle\_simple+0×0/0x30e  
[<ffffffff8004923a>] cpu\_idle+0×95/0xb8  
[<ffffffff80405807>] start\_kernel+0×220/0×225  
[<ffffffff8040522f>] \_sinittext+0x22f/0×236  
   
Code: 89 ca ed ed 41 89 c4 41 8a 45 1c 83 e0 30 3c 30 75 15 f0 ff  
   
**Observations**

In these scenarios we normally check dmesg and lspci output to figure out who the culprit might be

Parity and uncorrectable hardware errors are examples of why an IOCHK error could be raised.

Most hardware errors should however be reported through the MCE (Machine Check Exception) mechanism. An MCE indicates that the CPU detected an internal machine error or a bus error, or that an external agent detected a bus error. Normally the hardware manufacturer will be able to provide further details.

**Recommended Solution**

Use vendor hardware diagnostics software to analyse system health.  
Contact the hardware manufacturer for further assistance.  
Under RHEL6, the kernel.panic\_on\_io\_nmi = 1 sysctl can be set to have the system panic when an I/O NMI is received.

NTP Services in Linux

The NTP service is responsible for maintaining the system time, which is highly important in any  environment because so many things rely on time (logs, error messages, applications, and so on). There are  
a few different roles that your system can have when using NTP. They include

* Primary NTP Server Provides time to secondary NTP servers or clients
* Secondary NTP Server Provides time to clients; helps load balance primary NTP servers
* NTP Peer Provides and receives time
* NTP Client Receives time from primary or secondary NTP servers

According to the Red Hat Exam Prep Guide, you need to be able to synchronize clients with a higher-stratum server. The term stratum is used to define different levels, from 1 to 15, of time servers that are available to sync with. A stratum 1 time server is the most accurate. For load balancing and redundancy, you would probably want to configure a primary NTP server and a secondary NTP server to sync with a stratum 1 time server (which is located off your network). Now let’s look more closely at both the server and client side of NTP.

## Task 1 : Installing a Time Server

Usually, the NTP package comes installed as part of your Red Hat system, but it is always good to know how to install it should you ever need to.

Step 1. If you need to, install the package:

*# yum install -y ntp*

Step 2. Verify that the package has been installed:

*# rpm -qa | grep ntp  
ntp-4.2.4p8-2.el6.x86\_64  
ntpdate-4.2.4p8-2.el6.x86\_64*

Step 3. Make sure that the service will start when the system boots up:

*# chkconfig ntpd on*

Step 4. Verify the service starts on boot:

*# chkconfig ntpd –list  
ntpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off*

Before starting the service, let’s configure the system first.

## Task 2: Configuring NTP Server

To configure the primary NTP server, let’s look at part of the config file that comes with the system:

*# cat /etc/ntp.conf  
…  
server 0.rhel.pool.ntp.org  
server 1.rhel.pool.ntp.org  
server 2.rhel.pool.ntp.org  
…  
#server 127.127.1.0  
#fudge 127.127.1.0 stratum 10*

The first three lines shown here are the Internet (public) NTP servers that you sync with for the correct time. The fourth server option, which is commented out, defines a local clock driver that you can use to update the time. This is usually based on the BIOS click if used.

The final option, fudge, defines the stratum level to which your server is set. The server here is set at a stratum level of 10; a local system clock is also defined, and there are three public Internet servers you can sync with for accurate time.

For now, just let the system sync with the public Internet time servers.

Step 1. Start the NTP service:

*# service ntpd start  
Starting ntpd: [ OK ]*

With the server configured, you can turn your attention over to one of the client systems. For this example,  use the Client01 system to sync  with the RHEL01 primary NTP server.On Client01, you need to verify  
that the NTP package is already installed:

*# rpm -qa | grep ntp  
ntp-4.2.4p8-2.el6.x86\_64*

With the client installed, again look at the config file. Instead of public servers, you can set the RHEL01 system as the primary server to sync with.

Step 2. Define RHEL01 as the primary time server and make sure the driftfile line is uncommented:

*# cat /etc/ntp.conf  
…  
Server 172.168.1.1  
driftfile /var/lib/ntp/drift*

Now the client is looking at RHEL01 to sync its time. The additional option, driftfile, is a scratch place for the NTP service to calculate time checks and errors for accuracy. It is recommended for better results  
but not required.

Step 3. Now that the client is configured, you just need to start the service:

*# service ntpd start  
Starting ntpd: [ OK ]*

Securing NTP with iptables

Like the DHCP server, NTP requires only firewall rules to operate (no SELinux configuration is necessary).

The NTP service uses TCP and UDP port 123, so you need to open both on the firewall.

Step 1. Use iptables to create the required firewall rules:

*# iptables -I INPUT 5 -p udp -m udp –dport 123 -j ACCEPT  
# iptables -I INPUT 5 -p tcp -m tcp –dport 123 -j ACCEPT*

Step 2. Save the firewall rules you just created:

*# service iptables save  
Saving firewall rules to /etc/sysconfig/iptables: [ OK ]*

Step 3. Then restart the iptables service:

*# service iptables restart  
iptables: Flushing firewall rules: [ OK ]  
iptables: Setting chains to policy ACCEPT: filter [ OK ]  
iptables: Unloading modules: [ OK ]  
iptables: Applying firewall rules: [ OK ]*

When you want to troubleshoot the NTP service, there are a few commands that can help you out. First, if you want to update the time manually, you can use the ntpdate command.

*Syntax: ntpdate <server>*

Step 1. Stop the NTP service:

*# service ntpd stop  
Shutting down ntpd: [ OK ]*

Step 2. Specify the upstream server that you want to sync against:

*# ntpdate 0.rhel.pool.ntp.org  
23 Aug 16:29:15 ntpdate[7047]: adjust time server 209.234.249.11 offset 0.031852 sec*

Step 3. Start the service again:

*# service ntpd start  
Starting ntpd: [ OK ]*

Step 4. Verify that the time is accurate with the date command:

*# date  
Mon Aug 23 16:29:34 GMT 2010*

The next troubleshooting command is ntpq, which enables you to query for other NTP servers. Here, the only concern is the -p option, which polls for other NTP servers:

*# ntpq -p  
remote refid st t when poll reach delay offset jitter  
====================================================================  
pool-test.ntp.o .INIT. 16 u – 64 0 0.000 0.000 0.061  
knowledge.globa .INIT. 16 u – 64 0 0.000 0.000 0.061  
bindcat.fhsu.ed .INIT. 16 u – 64 0 0.000 0.000 0.061  
LOCAL(0) .LOCL. 10 l – 64 0 0.000 0.000 0.061*

Finally, you can also go really in depth by tracing where the NTP server is getting its time sync from. To do this, you can use the ntptrace command.

*Syntax: ntptrace <server>*

You can specify the public Internet server from before to determine how high in the stratum the server is on other various sync statistics:

*# ntptrace 0.rhel.pool.ntp.org  
0.rhel.pool.ntp.org: stratum 2, offset -0.000210, synch distance 0.*

### Spread a word

RHEL6 Installation based on Security BenchMarking Recommendations – Part1

# 1 Install Updates, Patches and Additional Security Software

## 1.1 Filesystem Configuration

Directories that are used for system-wide functions can be further protected by placing them on separate partitions. This provides protection for resource exhaustion and enables the use of mounting options that are applicable to the directory’s intended use. User’s data can be stored on separate partitions and have stricter mount options. A user partition is a filesystem that has been established for use by the users and does not contain software for system operations. The directives in this section are easier to perform during initial system installation. If the system is already installed, it is recommended that a full backup be performed before repartitioning the system.

Note: If you are repartitioning a system that has already been installed, make sure the data has been copied over to the new partition, unmount it and then remove the data from the directory that was in the old partition. Otherwise it will still consume space in the old partition that will be masked when the new filesystem is mounted. For example, if a system is in single-user mode with no filesystems mounted and the administrator adds a lot of data to the /tmp directory, this data will still consume space in / once the /tmp filesystem is mounted unless it is removed first.

### 1.1.1 Create Separate Partition for /tmp

The /tmp directory is a world-writable directory used for temporary storage by all users and some applications.

Rationale:

Since the /tmp directory is intended to be world-writable, there is a risk of resource exhaustion if it is not bound to a separate partition. In addition, making /tmp its own file system allows an administrator to set the noexec option on the mount, making /tmp useless for an attacker to install executable code. It would also prevent an attacker from establishing a hardlink to a system setuid program and wait for it to be updated. Once the program was updated, the hardlink would be broken and the attacker would have his own copy of the program. If the program happened to have a security vulnerability, the attacker could continue to exploit the known flaw.

Audit:

Verify that there is a /tmp file partition in the /etc/fstab file.

# grep “[[:space:]]/tmp[[:space:]]” /etc/fstab

Remediation:

For new installations, check the box to “Review and modify partitioning” and create a separate partition for /tmp.

For systems that were previously installed, use the Logical Volume Manager (LVM) to create partitions.

References: “[LVM HOWTO”](http://unixadminschool.com/blog/2012/11/redhat-linux-lvm-troubleshooting-reference/)

### 1.1.2 Set nodev option for /tmp Partition

The nodev mount option specifies that the filesystem cannot contain special devices.

Rationale:

Since the /tmp filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices in /tmp.

Audit:

Run the following commands to determine if the system is configured as recommended.

# grep /tmp /etc/fstab | grep nodev

# mount | grep /tmp | grep nodev

If either command emits no output then the system is not configured as recommended.

Remediation:

Edit the /etc/fstab file and add nodev to the fourth field (mounting options). See the fstab(5) manual page for more information.

# mount -o remount,nodev /tmp

### 1.1.3 Set nosuid option for /tmp Partition

The nosuid mount option specifies that the filesystem cannot contain set userid files.

Rationale:

Since the /tmp filesystem is only intended for temporary file storage, set this option to ensure that users cannot create set userid files in /tmp.

# mount -o remount,nosuid /tmp

Audit:

Run the following commands to determine if the system is configured as recommended.

# grep /tmp /etc/fstab | grep nosuid

# mount | grep /tmp | grep nosuid

If either command emits no output then the system is not configured as recommended. Remediation:

Edit the /etc/fstab file and add nosuid to the fourth field (mounting options). See the fstab(5) manual page for more information.

### 1.1.4 Set noexec option for /tmp Partition

The noexec mount option specifies that the filesystem cannot contain executable binaries.

Rationale:

Since the /tmp filesystem is only intended for temporary file storage, set this option to ensure that users cannot run executable binaries from /tmp.

Audit:

Run the following commands to determine if the system is configured as recommended.

# grep /tmp /etc/fstab | grep noexec

# mount | grep /tmp | grep noexec

If either command emits no output then the system is not configured as recommended.

Remediation:

Edit the /etc/fstab file and add noexec to the fourth field (mounting options). See the fstab(5) manual page for more information.

# mount -o remount,noexec /tmp

### 1.1.5 Create Separate Partition for /var

The /var directory is used by daemons and other system services to temporarily store dynamic data. Some directories created by these processes may be world-writable.

Rationale:

Since the /var directory may contain world-writable files and directories, there is a risk of resource exhaustion if it is not bound to a separate partition.

Audit:

#grep /var /etc/fstab <volume> /var ext3 <options>

Remediation:

For new installations, check the box to “Review and modify partitioning” and create a separate partition for /var.

For systems that were previously installed, use the Logical Volume Manager (LVM) to create partitions.

References : “[LVM HOWTO](http://unixadminschool.com/blog/2012/11/redhat-linux-lvm-troubleshooting-reference/)“,

### 1.1.6 Bind Mount the /var/tmp directory to /tmp

The /var/tmp directory is normally a standalone directory in the /var file system. Binding /var/tmp to /tmp establishes an unbreakable link to /tmp that cannot be removed (even by the root user). It also allows /var/tmp to inherit the same mount options that /tmp owns, allowing /var/tmp to be protected in the same /tmp is protected. It will also prevent /var from filling up with temporary files as the contents of /var/tmp will actually reside in the file system containing /tmp.

Rationale:

All programs that use /var/tmp and /tmp to read/write temporary files will always be written to the /tmp file system, preventing a user from running the /var file system out of space or trying to perform operations that have been blocked in the /tmp filesystem.

Audit:

Perform the following to determine if the system is configured as recommended:

# grep -e “^/tmp” /etc/fstab | grep /var/tmp

/tmp /var/tmp none none 0 0

# mount | grep -e “^/tmp” | grep /var/tmp

/tmp on /var/tmp type none (rw,bind)

If the above commands emit no output then the system is not configured as recommended.

Remediation:

# mount –bind /tmp /var/tmp

and edit the /etc/fstab file to contain the following line:

/tmp /var/tmp none bind 0 0

### 1.1.7 Create Separate Partition for /var/log

The /var/log directory is used by system services to store log data .

Rationale:

There are two important reasons to ensure that system logs are stored on a separate partition: protection against resource exhaustion (since logs can grow quite large) and protection of audit data.

Audit:

# grep /var/log /etc/fstab <volume> /var/log ext3 <options>

Remediation:

For new installations, check the box to “Review and modify partitioning” and create a separate partition for /var/log.

For systems that were previously installed, use the Logical Volume Manager (LVM) to create partitions.

References: “[LVM HOWTO](http://unixadminschool.com/blog/2012/11/redhat-linux-lvm-troubleshooting-reference/)”

### 1.1.8 Create Separate Partition for /var/log/audit

The auditing daemon, auditd, stores log data in the /var/log/audit directory.

Rationale:

There are two important reasons to ensure that data gathered by auditd is stored on a separate partition: protection against resource exhaustion (since the audit.log file can grow quite large) and protection of audit data. The audit daemon calculates how much free space is left and performs actions based on the results. If other processes (such as syslog) consume space in the same partition as auditd, it may not perform as desired.

Audit:

# grep /var/log/audit /etc/fstab <volume> /var/log/audit ext3 <options>

Remediation:

For new installations, check the box to “Review and modify partitioning” and create a separate partition for /var/log/audit. For systems that were previously installed, use the Logical Volume Manager (LVM) to create partitions.

References: “[LVM HOWTO](http://unixadminschool.com/blog/2012/11/redhat-linux-lvm-troubleshooting-reference/)“

### 1.1.9 Create Separate Partition for /home

The /home directory is used to support disk storage needs of local users.

Rationale:

If the system is intended to support local users, create a separate partition for the /home directory to protect against resource exhaustion and restrict the type of files that can be stored under /home.

Audit:

# grep /home /etc/fstab <volume> /home ext3 <options>

Remediation:

For new installations, check the box to “Review and modify partitioning” and create a separate partition for /home.

For systems that were previously installed, use the Logical Volume Manager (LVM) to create partitions.

References: “[LVM HOWTO](http://unixadminschool.com/blog/2012/11/redhat-linux-lvm-troubleshooting-reference/)“

### 1.1.10 Add nodev Option to /home

When set on a file system, this option prevents character and block special devices from being defined, or if they exist, from being used as character and block special devices.

Rationale:

Since the user partitions are not intended to support devices, set this option to ensure that users cannot attempt to create block or character special devices.

Note: The actions in the item refer to the /home partition, which is the default user partition that is defined in RHEL6. If you have created other user partitions, it is recommended that the Remediation and Audit steps be applied to these partitions as well.

Audit:

# grep /home /etc/fstab

Verify that nodev is an option

# mount | grep /home

<each user partition> on <mount point> type <fstype> (nodev)

Note: There may be other options listed for this filesystem

Remediation:

Edit the /etc/fstab file and add nodev to the fourth field (mounting options). See the fstab(5) manual page for more information.

# mount -o remount,nodev /home

### 1.1.11 Add nodev Option to Removable Media Partitions

Set nodev on removable media to prevent character and block special devices that are present on the removable be treated as these device files.

Rationale:

Removable media containing character and block special devices could be used to circumvent security controls by allowing non-root users to access sensitive device files such as /dev/kmem or the raw disk partitions.

Audit:

# grep <each removable media mountpoint> /etc/fstab Verify that nodev is an option

Remediation:

Edit the /etc/fstab file and add “nodev” to the fourth field (mounting options). Look for entries that have mount points that contain words such as floppy or cdrom. See the fstab(5) manual page for more information.

### 1.1.12 Add noexec Option to Removable Media Partitions

Set noexec on removable media to prevent programs from executing from the removable media.

Rationale:

Setting this option on a file system prevents users from executing programs from the removable. This deters users from being to introduce potentially malicious software on the system.

Audit:

# grep <each removable media mountpoint> /etc/fstab

Note: Verify that noexec is an option

Remediation:

Edit the /etc/fstab file and add noexec to the fourth field (mounting options). Look for entries that have mount points that contain words such as floppy or cdrom. See the fstab(5) manual page for more information.

### 1.1.13 Add nosuid Option to Removable Media Partitions

Set nosuid on removable media to prevent setuid and setgid executable files that are on that media from being executed as setuid and setgid.

Rationale:

Setting this option on a file system prevents users from introducing privileged programs onto the system and allowing non-root users to execute them.

Audit:

# grep <each removable media mountpoint> /etc/fstab Verify that nosuid is an option

Remediation:

Edit the /etc/fstab file and add nosuid to the fourth field (mounting options). Look for entries that have mount points that contain words such as floppy or cdrom. See the fstab(5) manual page for more information.

### 1.1.14 Add nodev Option to /dev/shm Partition

The nodev mount option specifies that the /dev/shm (temporary filesystem stored in memory) cannot contain block or character special devices.

Rationale:

Since the /dev/shm filesystem is not intended to support devices, set this option to ensure that users cannot attempt to create special devices in /dev/shm partitions.

Audit:

Run the following commands to determine if the system is in configured as recommended:

# grep /dev/shm /etc/fstab | grep nodev

# mount | grep /dev/shm | grep nodev

If either command emits no output then the system is not configured as recommended. Remediation:

Edit the /etc/fstab file and add nodev to the fourth field (mounting options of entries that have mount points that contain /dev/shm. See the fstab(5) manual page for more information.

# mount -o remount,nodev /dev/shm

### 1.1.15 Add nosuid Option to /dev/shm Partition

The nosuid mount option specifies that the /dev/shm (temporary filesystem stored in memory) will not execute setuid and setgid on executable programs as such, but rather execute them with the uid and gid of the user executing the program.

Rationale:

Setting this option on a file system prevents users from introducing privileged programs onto the system and allowing non-root users to execute them.

Audit:

Run the following commands to determine if the system is in configured as recommended:

# grep /dev/shm /etc/fstab | grep nosuid

# mount | grep /dev/shm | grep nosuid

If either command emits no output then the system is not configured as recommended.

Remediation:

Edit the /etc/fstab file and add nosuid to the fourth field (mounting options). Look for entries that have mount points that contain /dev/shm. See the fstab(5) manual page for more information.

# mount -o remount,nosuid /dev/shm

### 1.1.16 Add noexec Option to /dev/shm Partition

Set noexec on the shared memory partition to prevent programs from executing from there.

Rationale:

Setting this option on a file system prevents users from executing programs from shared memory. This deters users from introducing potentially malicious software on the system.

Audit:

Run the following commands to determine if the system is in configured as recommended:

# grep /dev/shm /etc/fstab | grep noexec

# mount | grep /dev/shm | grep noexec

If either command emits no output then the system is not configured as recommended.

Remediation:

Edit the /etc/fstab file and add noexec to the fourth field (mounting options). Look for entries that have mount points that contain /dev/shm. See the fstab(5) manual page for more information.

# mount -o remount,noexec /dev/shm

### 1.1.17 Set Sticky Bit on All World-Writable Directories

Setting the sticky bit on world writable directories prevents users from deleting or renaming files in that directory that are not owned by them.

Rationale:

This feature prevents the ability to delete or rename files in world writable directories (such as /tmp) that are owned by another user.

Audit:

# df –local -P | awk {‘if (NR!=1) print $6’} | xargs -I ‘{}’ find ‘{}’ -xdev -type d

\( -perm -0002 -a ! -perm -1000 \) 2>/dev/null

Remediation:

# df –local -P | awk {‘if (NR!=1) print $6’} | xargs -I ‘{}’ find ‘{}’ -xdev -type d

\( -perm -0002 -a ! -perm -1000 \) 2>/dev/null | xargs chmod a+t

### 1.1.18 Disable Mounting of cramfs Filesystems

Level 2

The cramfs filesystem type is a compressed read-only Linux filesystem embedded in small footprint systems. A cramfs image can be used without having to first decompress the image.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

Audit:

# /sbin/modprobe -n -v cramfs install /bin/true

# /sbin/lsmod | grep cramfs <No output>

Remediation:

Edit or create the file /etc/modprobe.d/CIS.conf and add the following line:

install cramfs /bin/true

### 1.1.19 Disable Mounting of freevxfs Filesystems

Level 2

The freevxfs filesystem type is a free version of the Veritas type filesystem. This is the primary filesystem type for HP-UX operating systems.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

Audit:

# /sbin/modprobe -n -v freevxfs install /bin/true

# /sbin/lsmod | grep freevxfs <No output>

Remediation:

Edit or create the file /etc/modprobe.d/CIS.conf and add the following line:

install freevxfs /bin/true

### 1.1.20 Disable Mounting of jffs2 Filesystems

Level 2

The jffs2 (journaling flash filesystem 2) filesystem type is a log-structured filesystem used in flash memory devices.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

Audit:

# /sbin/modprobe -n -v jffs2 install /bin/true # /sbin/lsmod | grep jffs2 <No output>

Remediation:

Edit or create the file /etc/modprobe.d/CIS.conf and add the following line:

install jffs2 /bin/true

### 1.1.21 Disable Mounting of hfs Filesystems

Level 2

The hfs filesystem type is a hierarchical filesystem that allows you to mount Mac OS filesystems.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

Audit:

# /sbin/modprobe -n -v hfs install /bin/true # /sbin/lsmod | grep hfs <No output>

Remediation:

Edit or create the file /etc/modprobe.d/CIS.conf and add the following line:

install hfs /bin/true

### 1.1.22 Disable Mounting of hfsplus Filesystems

Level 2

The hfsplus filesystem type is a hierarchical filesystem designed to replace hfs that allows you to mount Mac OS filesystems.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

Audit:

# /sbin/modprobe -n -v hfsplus install /bin/true

# /sbin/lsmod | grep hfsplus <No output>

Remediation:

Edit or create the file /etc/modprobe.d/CIS.conf and add the following line:

install hfsplus /bin/true

### 1.1.23 Disable Mounting of squashfs Filesystems

Level 2

The squashfs filesystem type is a compressed read-only Linux filesystem embedded in small footprint systems (similar to cramfs). A squashfs image can be used without having to first decompress the image.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

Audit:

# /sbin/modprobe -n -v squashfs install /bin/true

# /sbin/lsmod | grep squashfs <No output>

Remediation:

Edit or create the file /etc/modprobe.d/CIS.conf and add the following line:

install squashfs /bin/true

### 1.1.24 Disable Mounting of udf Filesystems

Level 2

The udf filesystem type is the universal disk format used to implement ISO/IEC 13346 and ECMA-167 specifications. This is an open vendor filesystem type for data storage on a broad range of media. This filesystem type is necessary to support writing DVDs and newer optical disc formats.

Rationale:

Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

Audit:

# /sbin/modprobe -n -v udf install /bin/true # /sbin/lsmod | grep udf <No output>

Remediation:

Edit or create the file /etc/modprobe.d/CIS.conf and add the following line:

install udf /bin/true

## 1.2 Configure Software Updates

Red Hat uses the yum command line tool to install and update software packages. Updating the RHEL6 software via the Internet requires a valid Red Hat subscription. Patch management procedures may vary widely between enterprises. Large enterprises may choose to install a Red Hat Satellite Update server that can be used in place of Red Hat’s servers, whereas a single deployment of a Red Hat system may prefer to get updates from Red Hat’s servers. Updates can be performed automatically or manually, depending on the site’s policy for patch management. Many large enterprises prefer to test patches on a nonproduction system before rolling out to production.

For the purpose of this benchmark, the requirement is to ensure that a patch management system is configured and maintained. The specifics on patch update procedures are left to the organization.

### 1.2.1 Configure Connection to the RHN RPM Repositories

Systems need to be registered with the Red Hat Network (RHN) to receive patch updates.

This is usually configured during initial installation.

Rationale:

It is important to register with the Red Hat Network to make sure that patches are updated on a regular basis. This helps to reduce the exposure time as new vulnerabilities are discovered.

Audit:

Verify that the system is registered by executing the following command:

# yum check-update

Remediation:

If your system is not listed on the RHN site as a registered system, run the Red Hat Network Registration tool as follows:

# rhn\_register

Follow the prompts on the screen. If successful, the system will appear on the RHN web site and be subscribed to one or more software update channels. Additionally, a new daemon, rhnsd, will be enabled.

### 1.2.2 Verify Red Hat GPG Key is Installed

Red Hat cryptographically signs updates with a GPG key to verify that they are valid.

Rationale:

It is important to ensure that updates are obtained from a valid source to protect against spoofing that could lead to the inadvertent installation of malware on the system.

Audit:

Run the following command to ensure that the system has the Red Hat GPG key properly installed:

# rpm -q –queryformat “%{SUMMARY}\n” gpg-pubkey

Remediation:

Compare the GPG fingerprint with the one from Red Hat’s web site at[http://www.redhat.com/security/team/key.](http://www.redhat.com/security/team/key) The following command can be used to print the installed release key’s fingerprint, which is actually contained in the file referenced below:

# gpg –quiet –with-fingerprint /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release

More information on package signing is also available at [https://access.redhat.com/security/team/key.](https://access.redhat.com/security/team/key)

References:

1. More information on package signing is also available at <https://access.redhat.com/security/team/key>

#### 1.2.3 Verify that gpgcheck is Globally Activated

The gpgcheck option, found in the main section of the /etc/yum.conf file determines if an RPM package’s signature is always checked prior to its installation.

Rationale:

It is important to ensure that an RPM’s package signature is always checked prior to installation to ensure that the software is obtained from a trusted source.

Audit:

Run the following command to verify that gpgcheck is set to 1 in all occurrences of the /etc/yum.conf file:

# grep gpgcheck /etc/yum.conf gpgcheck=1

Remediation:

Edit the /etc/yum.conf file and set the gpgcheck to 1 as follows:

gpgcheck=1

### 1.2.4 Disable the rhnsd Daemon

Level 2

The rhnsd daemon polls the Red Hat Network web site for scheduled actions and, if there are, executes those actions.

Rationale:

Patch management policies may require that organizations test the impact of a patch before it is deployed in a production environment. Having patches automatically deployed could have a negative impact on the environment. It is best to not allow an action by default but only after appropriate consideration has been made. It is recommended that the service be disabled unless the risk is understood and accepted or you are running your own satellite . This item is not scored because organizations may have addressed the risk.

Audit:

# chkconfig –list rhnsd rhnsd: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

Remediation:

Disable the rhnsd daemon by running the following command:

# chkconfig rhnsd off

### 1.2.5 Obtain Software Package Updates with yum

The yum update utility performs software updates, including dependency analysis, based on repository metadata and can be run manually from the command line, invoked from one of the provided front-end tools, or configured to run automatically at specified intervals.

Rationale:

The yum update utility is the preferred method to update software since it checks for dependencies and ensures that the software is installed correctly. Refer to your local patch management procedures for the method used to perform yum updates.

Audit:

Perform the following command to determine if there are any packages that need to be updated:

# yum check-update

Remediation:

# yum update

### 1.2.6 Verify Package Integrity Using RPM

RPM has the capability of verifying installed packages by comparing the installed files against the file information stored in the package.

Rationale:

Verifying packages gives a system administrator the ability to detect if package files were changed, which could indicate that a valid binary was overwritten with a trojaned binary.

Audit:

Perform the following to verify integrity of installed packages.

# rpm -qVa | awk ‘$2 != “c” { print $0}’

If any output shows up, you may have an integrity issue with that package

Note: Actions in other areas of the benchmark change permissions on some files to make them more secure than the default, which would cause this check to fail. It is important to validate the packages either have the permissions they were intended to have, or have been intentionally altered. It is recommended that any output generated in the audit step be investigated to justify the discrepancy.

Remediation:

Address unexpected discrepancies identified in the audit step.

## 1.3 Advanced Intrusion Detection Environment (AIDE)

AIDE is a file integrity checking tool, similar in nature to Tripwire. While it cannot prevent intrusions, it can detect unauthorized changes to configuration files by alerting when the files are changed. When setting up AIDE, decide internally what the site policy will be concerning integrity checking. Review the AIDE quick start guide and AIDE documentation before proceeding.

### 1.3.1 Install AIDE

Level 2

In some installations, AIDE is not installed automatically.

Rationale:

Install AIDE to make use of the file integrity features to monitor critical files for changes that could affect the security of the system.

Audit:

Perform the following to determine if AIDE is installed.

# rpm -q aide aide.<package version>.<hardware platform>

Remediation:

Use yum to install AIDE (-y option may be used to assume yes at all prompts):

# yum install aide

<Output messages from Yum install>

aide.<hardware platform> <package version> installed

Initialize AIDE:

# /usr/sbin/aide –init -B ‘database\_out=file:/var/lib/aide/aide.db.gz’

Note: The prelinking feature can interfere with AIDE because it alters binaries to speed up their start up times. Set PRELINKING=no in /etc/sysconfig/prelink and run

/usr/sbin/prelink -ua to restore the binaries to their prelinked state, thus avoiding false positives from AIDE.

### 1.3.2 Implement Periodic Execution of File Integrity

Level 2

Implement periodic file checking, in compliance with site policy.

Rationale:

Periodic file checking allows the system administrator to determine on a regular basis if critical files have been changed in an unauthorized fashion.

Audit:

Perform the following to determine if there is a cron job scheduled to run the aide check.

# crontab -u root -l | grep aide 0 5 \* \* \* /usr/sbin/aide –check

Remediation:

Execute the following command:

# crontab -u root -e

Add the following line to the crontab:

0 5 \* \* \* /usr/sbin/aide –check

Note: The checking in this instance occurs every day at 5am. Alter the frequency and time of the checks in compliance with site policy.

## 1.4 Configure SELinux

SELinux provides a Mandatory Access Control (MAC) system that greatly augments the default Discretionary Access Control (DAC) model. Under SELinux, every process and every object (files, sockets, pipes) on the system is assigned a security context, a label that includes detailed type information about the object. The kernel allows processes to access objects only if that access is explicitly allowed by the policy in effect. The policy defines transitions, so that a user can be allowed to run software, but the software can run under a different context than the user’s default. This automatically limits the damage that the software can do to files accessible by the calling user. The user does not need to take any action to gain this benefit. For an action to occur, both the traditional DAC permissions must be satisfied as well as the SELinux MAC rules. The action will not be allowed if either one of these models does not permit the action. In this way, SELinux rules can only make a system’s permissions more restrictive and secure. SELinux requires a complex policy to allow all the actions required of a system under normal operation. Three such policies have been designed for use with RHEL6 and are included with the system: targeted, strict, and mls. These are described as follows:

* targeted: consists mostly of Type Enforcement (TE) rules, and a small number of Role-Based Access Control (RBAC) rules. Targeted restricts the actions of many types of programs, but leaves interactive users largely unaffected.
* strict: also uses TE and RBAC rules, but on more programs and more aggressively.
* mls: implements Multi-Level Security (MLS), which introduces even more kinds of labels (sensitivity and category) and rules that govern access based on these.

This section provides guidance for the configuration of the targeted policy.

### 1.4.1 Enable SELinux in /etc/grub.conf

Level 2

Configure SELINUX to be enabled at boot time and verify that it has not been overwritten by the grub boot parameters

Rationale:

SELinux must be enabled at boot time in /etc/grub.conf to ensure that the controls it provides are not overwritten.

Audit:

Perform the following to verify that SELinux is enabled at boot time:

# grep selinux=0 /etc/grub.conf

[no output produced]

# grep enforcing=0 /etc/grub.conf [no output produced]

Remediation:

Remove all instances of selinux=0 and enforcing=0 from /etc/grub.conf.

### 1.4.2 Set the SELinux State

Level 2

Set SELinux to enable when the system is booted.

Rationale:

SELinux must be enabled at boot time in to ensure that the controls it provides are in effect at all times.

Audit:

Perform the following to determine if SELinux is enabled at boot time.

# grep SELINUX=enforcing /etc/selinux/config

SELINUX=enforcing

# /usr/sbin/sestatus

SELinux status: enabled

Current mode: enforcing

Mode from config file: enforcing Policy from config file: targeted

Remediation:

Edit the /etc/selinux/config file to set the SELINUX parameter:

SELINUX=enforcing

### 1.4.3 Set the SELinux Policy

Level 2

Configure SELinux to meet or exceed the default targeted policy, which constrains daemons and system software only.

Rationale:

Security configuration requirements vary from site to site. Some sites may mandate a policy that is stricter than the default policy, which is perfectly acceptable. This item is intended to ensure that least the default recommendations are met.

Audit:

Perform the following to determine if the targeted policy is selected in the /etc/selinux/config file.

# grep SELINUXTYPE=targeted /etc/selinux/config

SELINUXTYPE=targeted

# /usr/sbin/sestatus

SELinux status: enabled

Current mode: enforcing

Mode from config file: enforcing

Policy from config file: targeted

Note: If your organization requires stricter policies, verify that they are selected by using the “grep” command on the /etc/selinux/config file.

Remediation:

Edit the /etc/selinux/config file to set the SELINUXTYPE parameter:

SELINUXTYPE=targeted

Note: If your organization requires stricter policies, make sure they are added to the /etc/selinux/config file.

### 1.4.4 Remove SETroubleshoot

Level 2

The SETroubleshoot service notifies desktop users of SELinux denials through a userfriendly interface. The service provides important information around configuration errors, unauthorized intrusions, and other potential errors.

Rationale:

The SETroubleshoot service is an unnecessary daemon to have running on a server, especially if X Windows is disabled.

Audit:

# rpm -q setroubleshoot setroubleshoot.<package version>.<hardware platform>

Remediation:

# yum erase setroubleshoot

### 1.4.5 Remove MCS Translation Service (mcstrans)

Level 2

The mcstransd daemon provides category label information to client processes requesting information. The label translations are defined in /etc/selinux/targeted/setrans.conf

Rationale:

Since this service is not used very often, disable it to reduce the amount of potentially vulnerable code running on the system.

Audit:

Perform the following to determine if mctrans is disabled:

# rpm -q mcstrans mcstrans.<package version>.<hardware platform>

Remediation:

# yum erase mcstrans

### 1.4.6 Check for Unconfined Daemons

Level 2

Daemons that are not defined in SELinux policy will inherit the security context of their parent process.

Rationale:

Since daemons are launched and descend from the init process, they will inherit the security context label initrc\_t. This could cause the unintended consequence of giving the process more permission than it requires.

Audit:

Perform the following to determine if unconfined daemons are running on the system.

# ps -eZ | egrep “initrc” | egrep -vw “tr|ps|egrep|bash|awk” | tr ‘:’ ‘ ‘ | awk ‘{ print $NF }’ [no output produced]

Remediation:

Investigate any unconfined daemons found during the audit action.

## 1.5 Secure Boot Settings

### 1.5.1 Set User/Group Owner on /etc/grub.conf

Set the owner and group of /etc/grub.conf to the root user.

Rationale:

Setting the owner and group to root prevents non-root users from changing the file.

Audit:

Perform the following to determine if the /etc/grub.conf file has the correct ownership:

# stat -L -c “%u %g” /etc/grub.conf | egrep “0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/grub.conf

### 1.5.2 Set Permissions on /etc/grub.conf

Set permission on the /etc/grub.conf file to read and write for root only.

Rationale:

Setting the permissions to read and write for root only prevents non-root users from seeing the boot parameters or changing them. Non-root users who read the boot parameters may be able to identify weaknesses in security upon boot and be able to exploit them.

Audit:

Perform the following to determine if the /etc/grub.conf file permissions are correct:

# stat -L -c “%a” /etc/grub.conf | egrep “.00”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chmod og-rwx /etc/grub.conf

### 1.5.3 Set Boot Loader Password

Setting the boot loader password will require that the person who is rebooting the must enter a password before being able to set command line boot parameters

Rationale:

Requiring a boot password upon execution of the boot loader will prevent an unauthorized user from entering boot parameters or changing the boot partition. This prevents users from weakening security (e.g. turning off SELinux at boot time).

Audit:

Perform the following to determine if a password is required to set command line boot parameters:

# grep “^password” /etc/grub.conf password –md5 [Encrypted Password]

Note: Requirement is only that a password is set, other encryption options are available.

Remediation:

Use grub-md5-crypt to produce an encrypted password:

# grub-md5-crypt

Password:

Retype password:

[Encrypted Password]

Set the password parameter to[Encrypted Password] in /etc/grub.conf:

password –md5 [Encrypted Password]

### 1.5.4 Require Authentication for Single-User Mode

Since /etc/init determines what run state the system is in, setting the entry in /etc/sysconfig/init will force single user authentication.

Rationale:

Requiring authentication in single user mode prevents an unauthorized user from rebooting the system into single user to gain root privileges without credentials.

Audit:

Perform the following to determine if /etc/sysconfig/init is configured correctly:

# grep SINGLE /etc/sysconfig/init SINGLE=/sbin/sulogin

Remediation:

Run the following to edit /etc/sysconfig/init:

sed -i “/SINGLE/s/sushell/sulogin/” /etc/sysconfig/init

### 1.5.5 Disable Interactive Boot

The PROMPT option provides console users the ability to interactively boot the system and select which services to start on boot .

Rationale:

Turn off the PROMPT option on the console to prevent console users from potentially overriding established security settings.

Audit:

Perform the following to determine if PROMPT is disabled:

# grep “^PROMPT=” /etc/sysconfig/init PROMPT=no

Remediation:

Set the PROMPT parameter in /etc/sysconfig/init to no.

PROMPT=no

## 1.6 Additional Process Hardening

### 1.6.1 Restrict Core Dumps

A core dump is the memory of an executable program. It is generally used to determine why a program aborted. It can also be used to glean confidential information from a core file. The system provides the ability to set a soft limit for core dumps, but this can be overridden by the user.

Rationale:

Setting a hard limit on core dumps prevents users from overriding the soft variable. If core dumps are required, consider setting limits for user groups (see limits.conf(5)). In addition, setting the fs.suid\_dumpable variable to 0 will prevent setuid programs from dumping core.

Audit:

Perform the following to determine if core dumps are restricted.

|  |
| --- |
| # grep “hard core” /etc/security/limits.conf  \* hard core 0  # sysctl fs.suid\_dumpable fs.suid\_dumpable = 0 |

Remediation:

Add the following line to the /etc/security/limits.conf file.

\* hard core 0

Add the following line to the /etc/sysctl.conf file.

fs.suid\_dumpable = 0

### 1.6.2 Configure ExecShield

Execshield is made up of a number of kernel features to provide protection against buffer overflow attacks. These features include prevention of execution in memory data space, and special handling of text buffers.

Rationale:

Enabling any feature that can protect against buffer overflow attacks enhances the security of the system.

Audit:

Perform the following to determine if ExecShield is enabled.

# sysctl kernel.exec-shield kernel.exec-shield = 1

Remediation:

Add the following line to the /etc/sysctl.conf file.

kernel.exec-shield = 1

### 1.6.3 Enable Randomized Virtual Memory Region Placement

Set the system flag to force randomized virtual memory region placement.

Rationale:

Randomly placing virtual memory regions will make it difficult for to write memory page exploits as the memory placement will be consistently shifting.

Audit:

Perform the following to determine if virtual memory is randomized.

# sysctl kernel.randomize\_va\_space kernel.randomize\_va\_space = 2

Remediation:

Add the following line to the /etc/sysctl.conf file.

kernel.randomize\_va\_space = 2

### 1.7 Use the Latest OS Release

Periodically, Red Hat releases updates to the Red Hat operating system to support new hardware platforms, deliver new functionality as well as the bundle together a set of patches that can be tested as a unit.

Rationale:

Newer updates may contain security enhancements that would not be available through the standard patching process. As a result, it is recommended that the latest update of the Red Hat software be used to take advantage of the latest functionality. As with any software installation, organizations need to determine if a given update meets their requirements and verify the compatibility and supportability of any additional software against the update revision that is selected.

Audit:

Run the following command to determine the current OS level:

# uname -r

or

# cat /etc/redhat-release

Remediation:

Use the latest update when installing new systems and upgrade to or reinstall with the latest update as appropriate for existing systems.

## 2.1 Remove Legacy Services

The items in this section are intended to ensure that legacy services are not installed on the system. Some guidance includes directives to both disable and remove the service. There is no good reason to have these services on the system, even in a disabled state.

Note: The audit items in the section check to see if the packages are listed in the yum database and installed using rpm. It could be argued that someone may have installed them separately. However, this is also true for any other type of rogue software. It is beyond the scope of this benchmark to address software that is installed using non-standard methods and installation directories.

### 2.1.1 Remove telnet-server

The telnet-server package contains the telnetd daemon, which accepts connections from users from other systems via the telnet protocol.

Rationale:

The telnet protocol is insecure and unencrypted. The use of an unencrypted transmission medium could allow a user with access to sniff network traffic the ability to steal credentials. The ssh package provides an encrypted session and stronger security and is included in most Red Hat Linux distributions.

Audit:

Perform the following to determine if the telnet-server package is on the system.

# rpm -q telnet-server package telnet-server is not installed

Remediation:

# yum erase telnet-server

### 2.1.2 Remove telnet Clients

The telnet package contains the telnet client, which allows users to start connections to other systems via the telnet protocol.

Rationale:

The telnet protocol is insecure and unencrypted. The use of an unencrypted transmission medium could allow an authorized user to steal credentials. The ssh package provides an encrypted session and stronger security and is included in most Red Hat Linux distributions.

Audit:

Perform the following to determine if the telnet package is on the system.

# rpm -q telnet package telnet is not installed

Remediation:

# yum erase telnet

### 2.1.3 Remove rsh-server

The Berkeley rsh-server (rsh, rlogin, rcp) package contains legacy services that exchange credentials in clear-text.

Rationale:

These legacy service contain numerous security exposures and have been replaced with the more secure SSH package.

Audit:

Perform the following to determine if rsh-server is installed on the system.

# rpm -q rsh-server package rsh-server is not installed

Remediation:

# yum erase rsh-server

### 2.1.4 Remove rsh

The rsh package contains the client commands for the rsh services.

Rationale:

These legacy clients contain numerous security exposures and have been replaced with the more secure SSH package. Even if the server is removed, it is best to ensure the clients are also removed to prevent users from inadvertently attempting to use these commands and therefore exposing their credentials. Note that removing the rsh package removes the clients for rsh, rcp and rlogin.

Audit:

Perform the following to determine if rsh is installed on the system.

# rpm -q rsh package rsh is not installed

Remediation:

# yum erase rsh

### 2.1.5 Remove NIS Client

The Network Information Service (NIS), formerly known as Yellow Pages, is a client-server directory service protocol used to distribute system configuration files. The NIS client (ypbind) was used to bind a machine to an NIS server and receive the distributed configuration files.

Rationale:

The NIS service is inherently an insecure system that has been vulnerable to DOS attacks, buffer overflows and has poor authentication for querying NIS maps. NIS generally has been replaced by such protocols as Lightweight Directory Access Protocol (LDAP). It is recommended that the service be removed.

Audit:

Perform the following to determine if ypbind is installed on the system.

# rpm -q ypbind package ypbind is not installed

Remediation:

# yum erase ypbind

### 2.1.6 Remove NIS Server

The Network Information Service (NIS) (formally known as Yellow Pages) is a client-server directory service protocol for distributing system configuration files. The NIS server is a collection of programs that allow for the distribution of configuration files.

Rationale:

The NIS service is inherently an insecure system that has been vulnerable to DOS attacks, buffer overflows and has poor authentication for querying NIS maps. NIS generally been replaced by such protocols as Lightweight Directory Access Protocol (LDAP). It is recommended that the service be disabled and other, more secure services be used

Audit:

Perform the following to determine if ypserv is installed on the system.

# rpm -q ypserv package ypserv is not installed

Remediation:

# yum erase ypserv

### 2.1.7 Remove tftp

Trivial File Transfer Protocol (TFTP) is a simple file transfer protocol, typically used to automatically transfer configuration or boot files between machines. TFTP does not support authentication and can be easily hacked. The package tftp is a client program that allows for connections to a tftp server.

Rationale:

It is recommended that TFTP be removed, unless there is a specific need for TFTP (such as a boot server). In that case, use extreme caution when configuring the services.

Audit:

Perform the following to determine if tftp is installed on the system.

# rpm -q tftp package tftp is not installed

Remediation:

# yum erase tftp

### 2.1.8 Remove tftp-server

Trivial File Transfer Protocol (TFTP) is a simple file transfer protocol, typically used to automatically transfer configuration or boot machines from a boot server. The package tftp-server is the server package used to define and support a TFTP server.

Rationale:

TFTP does not support authentication nor does it ensure the confidentiality of integrity of data. It is recommended that TFTP be removed, unless there is a specific need for TFTP. In that case, extreme caution must be used when configuring the services.

Audit:

Perform the following to determine if tftp-server is installed on the system.

# rpm -q tftp-server package tftp-server is not installed

Remediation:

# yum erase tftp-server

### 2.1.9 Remove talk

The talk software makes it possible for users to send and receive messages across systems through a terminal session. The talk client (allows initialization of talk sessions) is installed by default.

Rationale:

The software presents a security risk as it uses unencrypted protocols for communication.

Audit:

Perform the following to determine if talk is installed on the system.

# rpm -q talk

package talk is not installed

Remediation:

# yum erase talk

### 2.1.10 Remove talk-server

The talk software makes it possible for users to send and receive messages across systems through a terminal session. The talk client (allows initiate of talk sessions) is installed by default.

Rationale:

The software presents a security risk as it uses unencrypted protocols for communication.

Audit:

Perform the following to determine if talk-server is installed on the system:

# rpm -q talk-server package talk-server is not installed

Remediation:

# yum erase talk-server

### 2.1.11 Remove xinetd

Level 2

The eXtended InterNET Daemon (xinetd) is an open source super daemon that replaced the original inetd daemon. The xinetd daemon listens for well known services and dispatches the appropriate daemon to properly respond to service requests.

Rationale:

If there are no xinetd services required, it is recommended that the daemon be deleted from the system.

Audit:

Perform the following to determine if xinetd is installed on the system.

# rpm -q xinetd package xinetd is not installed

Remediation:

# yum erase xinetd

### 2.1.12 Disable chargen-dgram

chargen-dram is a network service that responds with 0 to 512 ASCII characters for each datagram it receives. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.

Rationale:

Disabling this service will reduce the remote attack surface of the system.

Audit:

# chkconfig –list chargen-dgram chargen-dgram: off

Remediation:

Disable the chargen-dgram service by running the following command:

# chkconfig chargen-dgram off

### 2.1.13 Disable chargen-stream

chargen-stream is a network service that responds with 0 to 512 ASCII characters for each connection it receives. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.

Rationale:

Disabling this service will reduce the remote attack surface of the system.

Audit:

# chkconfig –list chargen-stream chargen-stream: off

Remediation:

Disable the chargen-stream service by running the following command:

# chkconfig chargen-stream off

### 2.1.14 Disable daytime-dgram

daytime-dgram is a network service that responds with the server’s current date and time. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.

Rationale:

Disabling this service will reduce the remote attack surface of the system.

Audit:

# chkconfig –list daytime-dgram daytime-dgram: off

Remediation:

Disable the daytime-dgram service by running the following command:

# chkconfig daytime-dgram off

### 2.1.15 Disable daytime-stream

daytime-stream is a network service that responds with the server’s current date and time. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.

Rationale:

Disabling this service will reduce the remote attack surface of the system.

Audit:

# chkconfig –list daytime-stream daytime-stream: off

Remediation:

Disable the daytime-stream service by running the following command:

# chkconfig daytime-stream off

### 2.1.16 Disable echo-dgram

echo-dgram is a network service that responds to clients with the data sent to it by the client. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.

Rationale:

Disabling this service will reduce the remote attack surface of the system.

Audit:

# chkconfig –list echo-dgram echo-dgram: off

Remediation:

Disable the echo-dgram service by running the following command:

# chkconfig echo-dgram off

### 2.1.17 Disable echo-stream

echo-stream is a network service that responds to clients with the data sent to it by the client. This service is intended for debugging and testing purposes. It is recommended that this service be disabled.

Rationale:

Disabling this service will reduce the remote attack surface of the system.

Audit:

# chkconfig –list echo-stream echo-stream: off

Remediation:

Disable the echo-stream service by running the following command:

# chkconfig echo-stream off

### 2.1.18 Disable tcpmux-server

tcpmux-server is a network service that allows a client to access other network services running on the server. It is recommended that this service be disabled.

Rationale: tcpmux-server can be abused to circumvent the server’s host based firewall. Additionally, tcpmux-server can be leveraged by an attacker to effectively port scan the server.

Audit:

# chkconfig –list tcpmux-server tcpmux-server: off

Remediation:

Disable the tcpmux-server service by running the following command:

# chkconfig tcpmux-server off

### 3.1 Set Daemon umask

Set the default umask for all processes started at boot time. The settings in umask selectively turn off default permission when a file is created by a daemon process.

Rationale:

Setting the umask to 027 will make sure that files created by daemons will not be readable, writable or executable by any other than the group and owner of the daemon process and will not be writable by the group of the daemon process. The daemon process can manually override these settings if these files need additional permission.

Audit:

Perform the following to determine if the daemon umask is set.

# grep umask /etc/sysconfig/init umask 027

Remediation:

Add the following line to the /etc/sysconfig/init file.

umask 027

### 3.2 Remove the X Window System

The X Window system provides a Graphical User Interface (GUI) where users can have multiple windows in which to run programs and various add on. The X Window system is typically used on desktops where users login, but not on servers where users typically do not login.

Rationale:

Unless your organization specifically requires graphical login access via the X Window System, remove the server to reduce the potential attack surface.

Audit:

Perform the following to determine if the X Window server is installed on the system:

# grep “^id:” /etc/inittab id:3:initdefault

# rpm -q xorg-x11-server-common

Remediation:

Edit the /etc/inittab file to set the default runlevel as follows:

id:3:initdefault

Uninstall the X Window Server:

# yum remove xorg-x11-server-common

### 3.3 Disable Avahi Server

Avahi is a free zeroconf implementation, including a system for multicast DNS/DNS-SD service discovery. Avahi allows programs to publish and discover services and hosts running on a local network with no specific configuration. For example, a user can plug a computer into a network and Avahi automatically finds printers to print to, files to look at and people to talk to, as well as network services running on the machine.

Rationale:

Since servers are not normally used for printing, this service is not needed unless dependencies require it. If this is the case, disable the service to reduce the potential attack surface. If for some reason the service is required on the server, follow the recommendations in sub-sections 3.2.1 – 3.2.5 to secure it.

Audit:

Perform the following to determine if Avahi is disabled.

# chkconfig –list avahi-daemon avahi-daemon: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

Remediation:

# chkconfig avahi-daemon off

In addition, edit the /etc/sysconfig/network file and remove zeroconf.

### 3.4 Disable Print Server – CUPS

The Common Unix Print System (CUPS) provides the ability to print to both local and network printers. A system running CUPS can also accept print jobs from remote systems and print them to local printers. It also provides a web based remote administration capability.

Rationale:

If the system does not need to print jobs or accept print jobs from other systems, it is recommended that CUPS be disabled to reduce the potential attack surface.

Audit:

Perform the following to determine if CUPS is disabled.

# chkconfig –list cups chkconfig: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

Remediation:

# chkconfig cups off

References:

1. More detailed documentation on CUPS is available at the project homepage at [http://www.cups.org.](http://www.cups.org./)

### 3.5 Remove DHCP Server

The Dynamic Host Configuration Protocol (DHCP) is a service that allows machines to be dynamically assigned IP addresses.

Rationale:

Unless a server is specifically set up to act as a DHCP server, it is recommended that this service be deleted to reduce the potential attack surface.

Audit:

Perform the following to determine if DHCP is disabled.

# rpm -q dhcp package dhcp is not installed

Remediation:

# yum erase dhcp

References:

1. More detailed documentation on DHCP is available at <http://www.isc.org/software/dhcp.>

### 3.6 Configure Network Time Protocol (NTP)

The Network Time Protocol (NTP) is designed to synchronize system clocks across a variety of systems and use a source that is highly accurate. The version of NTP delivered with Red Hat can be found at [http://www.ntp.org.](http://www.ntp.org/) NTP can be configured to be a client and/or a server.

Rationale:

It is recommended that physical systems and virtual guests lacking direct access to the physical host’s clock be configured as NTP clients to synchronize their clocks (especially to support time sensitive security mechanisms like Kerberos). This also ensures log files have consistent time records across the enterprise, which aids in forensic investigations.

Audit:

The following script checks for the correct parameters on restrict default and restrict -6 default:

# grep “restrict default” /etc/ntp.conf

restrict default kod nomodify notrap nopeer noquery # grep “restrict -6 default” /etc/ntp.conf restrict -6 default kod nomodify notrap nopeer noquery

Perform the following to determine if the system is configured to use an NTP Server and that the ntp daemon is running as an unprivileged user.

|  |
| --- |
| # grep “^server” /etc/ntp.conf server  # grep “ntp:ntp” /etc/sysconfig/ntpd  OPTIONS=”-u ntp:ntp -p /var/run/ntpd.pid” |

Remediation:

NTP is configured by default in RHEL6. If for some reason, it is not configured on your system, set the following restrict parameters in /etc/ntp.conf:

restrict default kod nomodify notrap nopeer noquery restrict -6 default kod nomodify notrap nopeer noquery

Also, make sure /etc/ntp.conf has an NTP server specified:

server <ntp-server>

Note: <ntp-server> is the IP address or hostname of a trusted time server. Configuring an NTP server is outside the scope of this benchmark.

References:  For more information on configuring NTP servers, go to the NTP homepage at [http://www.ntp.org.](http://www.ntp.org./)

### 3.7 Remove LDAP

The Lightweight Directory Access Protocol (LDAP) was introduced as a replacement for NIS/YP. It is a service that provides a method for looking up information from a central database. The default client/server LDAP application for Red Hat is OpenLDAP.

Rationale:

If the server will not need to act as an LDAP client or server, it is recommended that the software be disabled to reduce the potential attack surface.

Audit:

Perform the following to determine if LDAP is running.

# rpm -q openldap-servers

package openldap-servers is not installed

# rpm -q openldap-clients package openldap-clients is not installed

Remediation:

If LDAP is running on the system and is not needed, remove it as follows:

# yum erase openldap-servers # yum erase openldap-clients

References:

1. For more detailed documentation on OpenLDAP, go to the project homepage at [http://www.openldap.org.](http://www.openldap.org./)

### 3.8 Disable NFS and RPC

The Network File System (NFS) is one of the first and most widely distributed file systems in the UNIX environment. It provides the ability for systems to mount file systems of other servers through the network.

Rationale:

If the server does not export NFS shares or act as an NFS client, it is recommended that these services be disabled to reduce remote attack surface.

Audit:

Perform the following to determine if NFS is disabled.

# chkconfig –list nfslock

nfslock: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

# chkconfig –list rpcgssd

rpcgssd: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

# chkconfig –list rpcbind

rpcbind: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

# chkconfig –list rpcidmapd

rpcidmapd: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

# chkconfig –list rpcsvcgssd rpcsvcgssd: 0:off 1:off 2:off 3:off 4:off 5:off 6:off

Remediation:

# chkconfig nfslock off

# chkconfig rpcgssd off

# chkconfig rpcbind off

# chkconfig rpcidmapd off

# chkconfig rpcsvcgssd off

### 3.9 Remove DNS Server

The Domain Name System (DNS) is a hierarchical naming system that maps names to IP addresses for computers, services and other resources connected to a network.

Rationale:

Unless a server is specifically designated to act as a DNS server, it is recommended that the package be deleted to reduce the potential attack surface.

Audit:

Perform the following to determine if DNS is disabled on the system.

# rpm -q bind package bind is not installed

Remediation:

# yum erase bind

### 3.10 Remove FTP Server

The File Transfer Protocol (FTP) provides networked computers with the ability to transfer files.

Rationale:

FTP does not protect the confidentiality of data or authentication credentials. It is recommended sftp be used if file transfer is required. Unless there is a need to run the system as a FTP server (for example, to allow anonymous downloads), it is recommended that the package be deleted to reduce the potential attack surface.

Audit:

Perform the following to determine if FTP is disabled.

# rpm -q vsftpd package vsftpd is not installed

Remediation:

# yum erase vsftpd

### 3.11 Remove HTTP Server

HTTP or web servers provide the ability to host web site content. The default HTTP server shipped with Red Hat Linux is Apache.

Rationale:

Unless there is a need to run the system as a web server, it is recommended that the package be deleted to reduce the potential attack surface.

Audit:

Perform the following to determine if apache is disabled.

# rpm -q httpd package httpd is not installed

Remediation:

# yum erase httpd

### 3.12 Remove Dovecot (IMAP and POP3 services)

Dovecot is an open source IMAP and POP3 server for Linux based systems.

Rationale:

Unless POP3 and/or IMAP servers are to be provided to this server, it is recommended that the service be deleted to reduce the potential attack surface.

Audit:

Perform the following to determine if dovecot is installed on the system.

# rpm -q dovecot package dovecot is not installed

Remediation:

# yum erase dovecot

References: . [http://www.dovecot.org](http://www.dovecot.org/)

### 3.13 Remove Samba

The Samba daemon allows system administrators to configure their Linux systems to share file systems and directories with Windows desktops. Samba will advertise the file systems and directories via the Small Message Block (SMB) protocol. Windows desktop users will be able to mount these directories and file systems as letter drives on their systems.

Rationale:

If there is no need to mount directories and file systems to Windows systems, then this service can be deleted to reduce the potential attack surface.

Audit:

Perform the following to determine if samba is installed on the system.

# rpm -q samba package samba is not installed

Remediation:

# yum erase samba

### 3.14 Remove HTTP Proxy Server

The default HTTP proxy package shipped with Red Hat Linux is squid.

Rationale:

If there is no need for a proxy server, it is recommended that the squid proxy be deleted to reduce the potential attack surface.

Audit:

Perform the following to determine if squid is installed on the system.

# rpm -q squid package squid is not installed

Remediation:

# yum erase squid

### 3.15 Remove SNMP Server

The Simple Network Management Protocol (SNMP) server is used to listen for SNMP commands from an SNMP management system, execute the commands or collect the information and then send results back to the requesting system.

Rationale:

The SNMP server communicates using SNMP v1, which transmits data in the clear and does not require authentication to execute commands. Unless absolutely necessary, it is recommended that the SNMP service not be used.

Audit:

Perform the following to determine if net-snmp is installed on the system.

# rpm -q net-snmp package net-snmp is not installed

Remediation:

# yum erase net-snmp

### 3.16 Configure Mail Transfer Agent for Local-Only Mode

Mail Transfer Agents (MTA), such as sendmail and Postfix, are used to listen for incoming mail and transfer the messages to the appropriate user or mail server. If the system is not intended to be a mail server, it is recommended that the MTA be configured to only process local mail. By default, the MTA is set to loopback mode on RHEL5 and RHEL6.

Rationale:

The software for all Mail Transfer Agents is complex and most have a long history of security issues. While it is important to ensure that the system can process local mail messages, it is not necessary to have the MTA’s daemon listening on a port unless the server is intended to be a mail server that receives and processes mail from other systems.

Note: RHEL5 uses sendmail as the default MTA while RHEL6 uses Postfix. The recommendation to set the default MTA to local-only mode applies regardless of the MTA that is used.

Audit:

Perform the following command and make sure that the MTA is listening on the loopback address (127.0.0.1):

# netstat -an | grep LIST | grep “:25[[:space:]]” tcp 0 0 127.0.0.1:25 0.0.0.0:\* LISTEN

Remediation:

Edit /etc/postfix/main.cf and add the following line to the RECEIVING MAIL section. If the line already exists, change it to look like the line below.

inet\_interfaces = localhost

 Execute the following command to restart postfix

# service postfix restart

## Network Configuration and Firewalls

This section provides guidance for secure network and firewall configuration.

## 4.1 Modify Network Parameters (Host Only)

The following network parameters determine if the system is to act as a host only. A system is considered host only if the system has a single interface, or has multiple interfaces but will not be configured as a router.

### 4.1.1 Disable IP Forwarding

The net.ipv4.ip\_forward flag is used to tell the server whether it can forward packets or not. If the server is not to be used as a router, set the flag to 0.

Rationale:

Setting the flag to 0 ensures that a server with multiple interfaces (for example, a hard proxy), will never be able to forward packets, and therefore, never serve as a router.

Audit:

Perform the following to determine if net.ipv4.ip\_forward is enabled on the system.

# /sbin/sysctl net.ipv4.ip\_forward net.ipv4.ip\_forward = 0

Remediation:

Set the net.ipv4.ip\_forward parameter to 0 in /etc/sysctl.conf:

net.ipv4.ip\_forward=0

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.ip\_forward=0

# /sbin/sysctl -w net.ipv4.route.flush=1

### 4.1.2 Disable Send Packet Redirects

ICMP Redirects are used to send routing information to other hosts. As a host itself does not act as a router (in a host only configuration), there is no need to send redirects.

Rationale:

An attacker could use a compromised host to send invalid ICMP redirects to other router devices in an attempt to corrupt routing and have users access a system set up by the attacker as opposed to a valid system.

Audit:

Perform the following to determine if send packet redirects is disabled.

# /sbin/sysctl net.ipv4.conf.all.send\_redirects net.ipv4.conf.all.send\_redirects = 0

# /sbin/sysctl net.ipv4.conf.default.send\_redirects net.ipv4.conf.default.send\_redirects = 0

Remediation:

Set the net.ipv4.conf.all.send\_redirects and net.ipv4.conf.default.send\_redirects parameters to 0 in /etc/sysctl.conf:

net.ipv4.conf.all.send\_redirects=0 net.ipv4.conf.default.send\_redirects=0

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.conf.all.send\_redirects=0

# /sbin/sysctl -w net.ipv4.conf.default.send\_redirects=0

# /sbin/sysctl -w net.ipv4.route.flush=1

## 4.2 Modify Network Parameters (Host and Router)

The following network parameters determine if the system is to act as a router. A system acts as a router if it has at least two interfaces and is configured to perform routing functions.

### 4.2.1 Disable Source Routed Packet Acceptance

In networking, source routing allows a sender to partially or fully specify the route packets take through a network. In contrast, non-source routed packets travel a path determined by routers in the network. In some cases, systems may not be routable or reachable from some locations (e.g. private addresses vs. Internet routable), and so source routed packets would need to be used.

Rationale:

Setting net.ipv4.conf.all.accept\_source\_route and

net.ipv4.conf.default.accept\_source\_route to 0 disables the system from accepting source routed packets. Assume this server was capable of routing packets to Internet routable addresses on one interface and private addresses on another interface. Assume that the private addresses were not routable to the Internet routable addresses and vice versa. Under normal routing circumstances, an attacker from the Internet routable addresses could not use the server as a way to reach the private address servers. If, however, source routed packets were allowed, they could be used to gain access to the private address systems as the route could be specified, rather than rely on routing protocols that did not allow this routing.

Audit:

Perform the following to determine if accepting source routed packets is disabled.

# /sbin/sysctl net.ipv4.conf.all.accept\_source\_route net.ipv4.conf.all.accept\_source\_route = 0

# /sbin/sysctl net.ipv4.conf.default.accept\_source\_route net.ipv4.conf.default.accept\_source\_route = 0

Remediation:

Set the net.ipv4.conf.all.accept\_source\_route and net.ipv4.conf.default.accept\_source\_route parameters to 0 in /etc/sysctl.conf:

net.ipv4.conf.all.accept\_source\_route=0 net.ipv4.conf.default.accept\_source\_route=0

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.conf.all.accept\_source\_route=0

# /sbin/sysctl -w net.ipv4.conf.default.accept\_source\_route=0 # /sbin/sysctl -w net.ipv4.route.flush=1

### 4.2.2 Disable ICMP Redirect Acceptance

ICMP redirect messages are packets that convey routing information and tell your host (acting as a router) to send packets via an alternate path. It is a way of allowing an outside routing device to update your system routing tables. By setting net.ipv4.conf.all.accept\_redirects to 0, the system will not accept any ICMP redirect messages, and therefore, won’t allow outsiders to update the system’s routing tables.

Rationale:

Attackers could use bogus ICMP redirect messages to maliciously alter the system routing tables and get them to send packets to incorrect networks and allow your system packets to be captured.

Audit:

Perform the following to determine if ICMP redirect messages will be rejected.

# /sbin/sysctl net.ipv4.conf.all.accept\_redirects net.ipv4.conf.all.accept\_redirects = 0

# /sbin/sysctl net.ipv4.conf.default.accept\_redirects net.ipv4.conf.default.accept\_redirects = 0

Remediation:

Set the net.ipv4.conf.all.accept\_redirects and net.ipv4.conf.default.accept\_redirects parameters to 0 in /etc/sysctl.conf:

net.ipv4.conf.all.accept\_redirects=0 net.ipv4.conf.default.accept\_redirects=0

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.conf.all.accept\_redirects=0

# /sbin/sysctl -w net.ipv4.conf.default.accept\_redirects=0

# /sbin/sysctl -w net.ipv4.route.flush=1

### 4.2.3 Disable Secure ICMP Redirect Acceptance

Level 2

Secure ICMP redirects are the same as ICMP redirects, except they come from gateways listed on the default gateway list. It is assumed that these gateways are known to your system, and that they are likely to be secure.

Rationale:

It is still possible for even known gateways to be compromised. Setting net.ipv4.conf.all.secure\_redirects to 0 protects the system from routing table updates by possibly compromised known gateways.

Audit:

Perform the following to determine if ICMP redirect messages will be rejected from known gateways.

# /sbin/sysctl net.ipv4.conf.all.secure\_redirects net.ipv4.conf.all.secure\_redirects = 0

# /sbin/sysctl net.ipv4.conf.default.secure\_redirects net.ipv4.conf.default.secure\_redirects = 0

Remediation:

Set the net.ipv4.conf.all.secure\_redirects and net.ipv4.conf.default.secure\_redirects parameters to 0 in /etc/sysctl.conf:

net.ipv4.conf.all.secure\_redirects=0 net.ipv4.conf.default.secure\_redirects=0

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.conf.all.secure\_redirects=0

# /sbin/sysctl -w net.ipv4.conf.default.secure\_redirects=0

# /sbin/sysctl -w net.ipv4.route.flush=1 4.2.4 Log Suspicious Packets

When enabled, this feature logs packets with un-routable source addresses to the kernel log.

Rationale:

Enabling this feature and logging these packets allows an administrator to investigate the possibility that an attacker is sending spoofed packets to their server.

Audit:

Perform the following to determine if suspicious packets are logged.

# /sbin/sysctl net.ipv4.conf.all.log\_martians net.ipv4.conf.all.log\_martians = 1

# /sbin/sysctl net.ipv4.conf.default.log\_martians net.ipv4.conf.default.log\_martians = 1

Remediation:

Set

the net.ipv4.conf.all.log\_martians and net.ipv4.conf.default.log\_martians parameters to 1 in /etc/sysctl.conf:

net.ipv4.conf.all.log\_martians=1 net.ipv4.conf.default.log\_martians=1 net.ipv4.route.flush=1

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.conf.all.log\_martians=1

# /sbin/sysctl -w net.ipv4.conf.default.log\_martians=1

# /sbin/sysctl -w net.ipv4.route.flush=1

### 4.2.5 Enable Ignore Broadcast Requests

Setting net.ipv4.icmp\_echo\_ignore\_broadcasts to 1 will cause the system to ignore all ICMP echo and timestamp requests to broadcast and multicast addresses.

Rationale:

Accepting ICMP echo and timestamp requests with broadcast or multicast destinations for your network could be used to trick your host into starting (or participating) in a Smurf attack. A Smurf attack relies on an attacker sending large amounts of ICMP broadcast messages with a spoofed source address. All hosts receiving this message and responding would send echo-reply messages back to the spoofed address, which is probably not routable. If many hosts respond to the packets, the amount of traffic on the network could be significantly multiplied.

Audit:

Perform the following to determine if all ICMP echo and timestamp requests to broadcast and multicast addresses will be ignored.

# /sbin/sysctl net.ipv4.icmp\_echo\_ignore\_broadcasts net.ipv4.icmp\_echo\_ignore\_broadcasts = 1

Remediation:

Set the net.ipv4.icmp\_echo\_ignore\_broadcasts parameter to 1 in /etc/sysctl.conf:

net.ipv4.icmp\_echo\_ignore\_broadcasts=1

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.icmp\_echo\_ignore\_broadcasts=1

# /sbin/sysctl -w net.ipv4.route.flush=1

### 4.2.6 Enable Bad Error Message Protection

Setting icmp\_ignore\_bogus\_error\_responses to 1 prevents the kernel from logging bogus responses (RFC-1122 non-compliant) from broadcast reframes, keeping file systems from filling up with useless log messages.

Rationale:

Some routers (and some attackers) will send responses that violate RFC-1122 and attempt to fill up a log file system with many useless error messages.

Audit:

Perform the following to determine if bogus messages will be ignored.

# /sbin/sysctl net.ipv4.icmp\_ignore\_bogus\_error\_responses net.ipv4.icmp\_ignore\_bogus\_error\_responses = 1

Remediation:

Set the net.ipv4.icmp\_ignore\_bogus\_error\_responses parameter to 1 in /etc/sysctl.conf:

net.ipv4.icmp\_ignore\_bogus\_error\_responses=1

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.icmp\_ignore\_bogus\_error\_responses=1

# /sbin/sysctl -w net.ipv4.route.flush=1

### 4.2.7 Enable RFC-recommended Source Route Validation

Level 2

Setting net.ipv4.conf.all.rp\_filter and net.ipv4.conf.default.rp\_filter to 1 forces the Linux kernel to utilize reverse path filtering on a received packet to determine if the packet was valid. Essentially, with reverse path filtering, if the return packet does not go out the same interface that the corresponding source packet came from, the packet is dropped (and logged if log\_martians is set).

Rationale:

Setting these flags is a good way to deter attackers from sending your server bogus packets that cannot be responded to. One instance where this feature breaks down is if asymmetrical routing is employed. This is would occur when using dynamic routing protocols (bgp, ospf, etc) on your system. If you are using asymmetrical routing on your server, you will not be able to enable this feature without breaking the routing.

Audit:

Perform the following to determine if RFC-recommended source route validation is enabled.

# /sbin/sysctl net.ipv4.conf.all.rp\_filter net.ipv4.conf.all.rp\_filter = 1

# /sbin/sysctl net.ipv4.conf.default.rp\_filter net.ipv4.conf.default.rp\_filter = 1

Remediation:

Set the net.ipv4.conf.all.rp\_filter and net.ipv4.conf.default.rp\_filter parameters to 1 in /etc/sysctl.conf:

net.ipv4.conf.all.rp\_filter=1 net.ipv4.conf.default.rp\_filter=1

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.conf.all.rp\_filter=1

# /sbin/sysctl -w net.ipv4.conf.default.rp\_filter=1

# /sbin/sysctl -w net.ipv4.route.flush=1 4.2.8 Enable TCP SYN Cookies

When tcp\_syncookies is set, the kernel will handle TCP SYN packets normally until the half-open connection queue is full, at which time, the SYN cookie functionality kicks in. SYN cookies work by not using the SYN queue at all. Instead, the kernel simply replies to the SYN with a SYN|ACK, but will include a specially crafted TCP sequence number that encodes the source and destination IP address and port number and the time the packet was sent. A legitimate connection would send the ACK packet of the three way handshake with the specially crafted sequence number. This allows the server to verify that it has received a valid response to a SYN cookie and allow the connection, even though there is no corresponding SYN in the queue.

Rationale:

Attackers use SYN flood attacks to perform a denial of service attacked on a server by sending many SYN packets without completing the three way handshake. This will quickly use up slots in the kernel’s half-open connection queue and prevent legitimate connections from succeeding. SYN cookies allow the server to keep accepting valid connections, even if under a denial of service attack.

Audit:

Perform the following to determine if TCP SYN Cookies is enabled.

# /sbin/sysctl net.ipv4.tcp\_syncookies net.ipv4.tcp\_syncookies = 1

Remediation:

Set the net.ipv4.tcp\_syncookies parameter to 1 in /etc/sysctl.conf: net.ipv4.tcp\_syncookies=1

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv4.tcp\_syncookies=1

# /sbin/sysctl -w net.ipv4.route.flush=1

## 4.3 Wireless Networking

### 4.3.1 Deactivate Wireless Interfaces

Wireless networking is used when wired networks are unavailable. Red Hat contains a wireless tool kit to allow system administrators to configure and use wireless networks.

Rationale:

If wireless is not to be used, wireless devices can be disabled to reduce the potential attack surface.

Audit:

Perform the following to determine if wireless interfaces are active.

# ifconfig -a

Validate that all interfaces using wireless are down.

Remediation:

Use the following commands to list all interfaces and identify devices with wireless interfaces. Once identified, shutdown the interface and remove it.

# ifconfig -a

# iwconfig

# ifdown <interface>

# rm /etc/sysconfig/network-scripts/ifcfg-<interface>

## 4.4 IPv6

IPv6 is a networking protocol that supersedes IPv4. It has more routable addresses and has built in security

### 4.4.1 Configure IPv6

If IPv6 is to be used, follow this section of the benchmark to configure IPv6.

### 4.4.1.1 Disable IPv6 Router Advertisements

This setting disables the systems ability to accept router advertisements

Rationale:

It is recommended that systems not accept router advertisements as they could be tricked into routing traffic to compromised machines. Setting hard routes within the system (usually a single default route to a trusted router) protects the system from bad routes.

Audit:

Perform the following to determine if the system is disabled from accepting router advertisements:

# /sbin/sysctl net.ipv6.conf.all.accept\_ra net.ipv6.conf.all.accept\_ra = 0

# /sbin/sysctl net.ipv6.conf.default.accept\_ra net.ipv6.conf.default.accept\_ra = 0

Remediation:

Set the net.ipv6.conf.all.accept\_ra and net.ipv6.conf.default.accept\_ra parameter to 0 in /etc/sysctl.conf:

net.ipv6.conf.all.accept\_ra=0 net.ipv6.conf.default.accept\_ra=0

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv6.conf.all.accept\_ra=0

# /sbin/sysctl -w net.ipv6.conf.default.accept\_ra=0

# /sbin/sysctl -w net.ipv6.route.flush=1

### 4.4.1.2 Disable IPv6 Redirect Acceptance

This setting prevents the system from accepting ICMP redirects. ICMP redirects tell the system about alternate routes for sending traffic.

Rationale:

It is recommended that systems not accept ICMP redirects as they could be tricked into routing traffic to compromised machines. Setting hard routes within the system (usually a single default route to a trusted router) protects the system from bad routes.

Audit:

Perform the following to determine if IPv6 redirects are disabled.

# /sbin/sysctl net.ipv6.conf.all.accept\_redirects net.ipv6.conf.all.accept\_redirect = 0

# /sbin/sysctl net.ipv6.conf.default.accept\_redirects net.ipv6.conf.default.accept\_redirect = 0

Remediation:

Set the net.ipv6.conf.all.accept\_redirects and net.ipv6.conf.default.accept\_redirects parameters to 0 in /etc/sysctl.conf:

net.ipv6.conf.all.accept\_redirects=0 net.ipv6.conf.default.accept\_redirects=0

Modify active kernel parameters to match:

# /sbin/sysctl -w net.ipv6.conf.all.accept\_redirects=0

# /sbin/sysctl -w net.ipv6.conf.default.accept\_redirects=0

# /sbin/sysctl -w net.ipv6.route.flush=1

### 4.4.2 Disable IPv6

Although IPv6 has many advantages over IPv4, few organizations have implemented IPv6.

Rationale:

If IPv6 is not to be used, it is recommended that it be disabled to reduce the attack surface of the system.

Audit:

Perform the following to determine if IPv6 is enabled

# grep NETWORKING\_IPV6 /etc/sysconfig/network

NETWORKING\_IPV6=no

# grep IPV6INIT /etc/sysconfig/network

IPV6INIT=no

# grep ipv6 /etc/modprobe.d/ipv6.conf options ipv6 disable=1

Remediation:

Edit /etc/sysconfig/network, and add the following line:

NETWORKING\_IPV6=no IPV6INIT=no

Create the file /etc/modprobe.d/ipv6.conf and add the following lines:

options ipv6 disable=1

Perform the following command to turn ip6tables off:

# /sbin/chkconfig ip6tables off

## 4.5 Install TCP Wrappers

### 4.5.1 Install TCP Wrappers

TCP Wrappers provides a simple access list and standardized logging method for services capable of supporting it. In the past, services that were called from inetd and xinetd supported the use of tcp wrappers. As inetd and xinetd have been falling in disuse, any service that can support tcp wrappers will have the libwrap.so library attached to it.

Rationale:

TCP Wrappers provide a good simple access list mechanism to services that may not have that support built in. It is recommended that all services that can support TCP Wrappers, use it.

Audit:

Perform the following to determine if TCP Wrappers is enabled.

# yum list tcp\_wrappers tcp\_wrappers.<hardware platform> <release> <installed>

Remediation:

# yum install tcp\_wrappers

To verify if a service supports TCP Wrappers, run the following command:

# ldd <path-to-daemon> | grep libwrap.so

If there is any output, then the service supports TCP Wrappers.

### 4.5.2 Create /etc/hosts.allow

The /etc/hosts.allow file specifies which IP addresses are permitted to connect to the host. It is intended to be used in conjunction with the /etc/hosts.deny file.

Rationale:

The /etc/hosts.allow file supports access control by IP and helps ensure that only authorized systems can connect to the server.

Audit:

Run the following command to verify the contents of the /etc/hosts.allow file.

# cat /etc/hosts.allow

[contents will vary, depending on your network configuration]

Remediation:

Create /etc/hosts.allow:

# echo “ALL: <net>/<mask>, <net>/<mask>, …” >/etc/hosts.allow

where each <net>/<mask> combination (for example, “192.168.1.0/255.255.255.0”) represents one network block in use by your organization that requires access to this system.

### 4.5.3 Verify Permissions on /etc/hosts.allow

The /etc/hosts.allow file contains networking information that is used by many applications and therefore must be readable for these applications to operate.

Rationale:

It is critical to ensure that the /etc/hosts.allow file is protected from unauthorized write access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to determine the permissions on the /etc/hosts.allow file.

# /bin/ls -l /etc/hosts.allow

-rw-r–r– 1 root root 2055 Jan 30 16:30 /etc/hosts.allow

Remediation:

If the permissions of the /etc/hosts.allow file are incorrect, run the following command to correct them:

# /bin/chmod 644 /etc/hosts.allow

### 4.5.4 Create /etc/hosts.deny

The /etc/hosts.deny file specifies which IP addresses are not permitted to connect to the host. It is intended to be used in conjunction with the /etc/hosts.allow file.

Rationale:

The /etc/hosts.deny file serves as a failsafe so that any host not specified in /etc/hosts.allow is denied access to the server.

Audit:

Verify that /etc/hosts.deny exists and is configured to deny all hosts not explicitly listed in /etc/hosts.allow:

# grep “ALL: ALL” /etc/hosts.deny ALL: ALL

Remediation:

Create /etc/hosts.deny:

# echo “ALL: ALL” >> /etc/hosts.deny

### 4.5.5 Verify Permissions on /etc/hosts.deny

The /etc/hosts.deny file contains network information that is used by many system applications and therefore must be readable for these applications to operate.

Rationale:

It is critical to ensure that the /etc/hosts.deny file is protected from unauthorized write access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to determine the permissions on the /etc/hosts.deny file.

# /bin/ls -l /etc/hosts.deny

-rw-r–r– 1 root root 2055 Jan 30 16:30 /etc/hosts.deny

Remediation:

If the permissions of the /etc/hosts.deny file are incorrect, run the following command to correct them:

# /bin/chmod 644 /etc/hosts.deny

## 4.6 Uncommon Network Protocols

Red Hat Linux supports several network protocols that are not commonly used. If these protocols are not needed, it is recommended that they be disabled in the kernel.

### 4.6.1 Disable DCCP

The Datagram Congestion Control Protocol (DCCP) is a transport layer protocol that supports streaming media and telephony. DCCP provides a way to gain access to congestion control, without having to do it at the application layer, but does not provide insequence delivery.

Rationale:

If the protocol is not required, it is recommended that the drivers not be installed to reduce the potential attack surface.

Audit:

Perform the following to determine if DCCP is disabled.

# grep “install dccp /bin/true” /etc/modprobe.d/CIS.conf install dccp /bin/true

Remediation:

# echo “install dccp /bin/true” >> /etc/modprobe.d/CIS.conf

### 4.6.2 Disable SCTP

The Stream Control Transmission Protocol (SCTP) is a transport layer protocol used to support message oriented communication, with several streams of messages in one connection. It serves a similar function as TCP and UDP, incorporating features of both. It is message-oriented like UDP, and ensures reliable in-sequence transport of messages with congestion control like TCP.

Rationale:

If the protocol is not being used, it is recommended that kernel module not be loaded, disabling the service to reduce the potential attack surface.

Audit:

Perform the following to determine if SCTP is disabled.

# grep “install sctp /bin/true” /etc/modprobe.d/CIS.conf install sctp /bin/true

Remediation:

# echo “install sctp /bin/true” >> /etc/modprobe.d/CIS.conf

### 4.6.3 Disable RDS

The Reliable Datagram Sockets (RDS) protocol is a transport layer protocol designed to provide low-latency, high-bandwidth communications between cluster nodes. It was developed by the Oracle Corporation.

Rationale:

If the protocol is not being used, it is recommended that kernel module not be loaded, disabling the service to reduce the potential attack surface.

Audit:

Perform the following to determine if RDS is disabled.

# grep “install rds /bin/true” /etc/modprobe.d/CIS.conf install rds /bin/true

Remediation:

# echo “install rds /bin/true” >> /etc/modprobe.d/CIS.conf

### 4.6.4 Disable TIPC

The Transparent Inter-Process Communication (TIPC) protocol is designed to provide communication between cluster nodes.

Rationale:

If the protocol is not being used, it is recommended that kernel module not be loaded, disabling the service to reduce the potential attack surface.

Audit:

Perform the following to determine if TIPC is disabled.

# grep “install tipc /bin/true” /etc/modprobe.d/CIS.conf install tipc /bin/true

Remediation:

# echo “install tipc /bin/true” >> /etc/modprobe.d/CIS.conf

### 4.7 Enable IPtables

IPtables is an application that allows a system administrator to configure the IPv4 tables, chains and rules provided by the Linux kernel firewall.

Rationale:

IPtables provides extra protection for the Linux system by limiting communications in and out of the box to specific IPv4 addresses and ports.

Audit:

Perform the following to determine if IPtables is enabled:

# chkconfig –list iptables iptables 0:off 1:off 2:on 3:on 4:on 5:on 6:off

Remediation:

# service iptables restart

# chkconfig iptables on

### 4.8 Enable IP6tables

IP6tables is an application that allows a system administrator to configure the IPv6 tables, chains and rules provided by the Linux kernel firewall.

Rationale:

IP6tables provides extra protection for the Linux system by limiting communications in and out of the box to specific IPv6 addresses and ports.

Note: IP6Tables should only be enabled if IPv6 has been enabled on your system.

Audit:

Perform the following to determine if IP6Tables is enabled:

# chkconfig –list ip6tables ip6tables 0:off 1:off 2:on 3:on 4:on 5:on 6:off

Remediation:

# service ip6tables restart # chkconfig ip6tables on

# Logging and Auditing

The items in this section describe how to configure logging, log monitoring, and auditing, using tools included with RHEL6.

It is recommended that rsyslog be used for logging (with logwatch providing summarization) and auditd be used for auditing (with aureport providing summarization) to automatically monitor logs for intrusion attempts and other suspicious system behavior.

In addition to the local log files created by the steps in this section, it is also recommended that sites collect copies of their system logs on a secure, centralized log server via an encrypted connection. Not only does centralized logging help sites correlate events that may be occurring on multiple systems, but having a second copy of the system log information may be critical after a system compromise where the attacker has modified the local log files on the affected system(s). If a log correlation system is deployed, configure it to process the logs described in this section.

Because it is often necessary to correlate log information from many different systems (particularly after a security incident) it is recommended that the time be synchronized among systems and devices connected to the local network. The standard Internet protocol for time synchronization is the Network Time Protocol (NTP), which is supported by most network-ready devices. See the ntpd(8) manual page for more information on configuring NTP.

It is important that all logs described in this section be monitored on a regular basis and correlated to determine trends. A seemingly innocuous entry in one log could be more significant when compared to an entry in another log.

Note on log file permissions: There really isn’t a “one size fits all” solution to the permissions on log files. Many sites utilize group permissions so that administrators who are in a defined security group, such as “wheel” do not have to elevate privileges to root in order to read log files. Also, if a third party log aggregation tool is used, it may need to have group permissions to read the log files, which is preferable to having it run setuid to root. Therefore, there are two remediation and audit steps for log file permissions. One is for systems that do not have a secured group method implemented that only permits root to read the log files (root:root 600). The other is for sites that do have such a setup and are designated as root:<securegrp> 640 where <securegrp> is the defined security group (in some cases wheel).

## 5.1 Configure rsyslog

The rsyslog software is recommended as a replacement for the default syslogd daemon and provides improvements over syslogd, such as connection-oriented (i.e. TCP) transmission of logs, the option to log to database formats, and the encryption of log data en route to a central logging server.

### 5.1.1 Install the rsyslog package

The rsyslog package is a third party package that provides many enhancements to syslog, such as multi-threading, TCP communication, message filtering and data base support. As of RHEL 5.2, rsyslog is available as part of the core distribution.

Rationale:

The security enhancements of rsyslog such as connection-oriented (i.e. TCP) transmission of logs, the option to log to database formats, and the encryption of log data en route to a central logging server) justify installing and configuring the package.

Audit:

Perform the following command to verify that rsyslog is installed.

# rpm -q rsyslog rsyslog.<package version>.<hardware platform>

Remediation:

# yum install rsyslog

### 5.1.2 Activate the rsyslog Service

The chkconfig command can be used to ensure that the syslog service is turned off and that the rsyslog service is turned on.

Rationale:

It is important to ensure that syslog is turned off so that it does not interfere with the rsyslog service.

Audit:

# chkconfig –list syslog

syslog 0:off 1:off 2:off 3:off 4:off 5:off 6:off

# chkconfig –list rsyslog rsyslog 0:off 1:off 2:on 3:on 4:on 5:on 6:off

Remediation:

# chkconfig syslog off

# chkconfig rsyslog on

### 5.1.3 Configure /etc/rsyslog.conf

The /etc/rsyslog.conf file specifies rules for logging and which files are to be used to log certain classes of messages.

Rationale:

A great deal of important security-related information is sent via rsyslog (e.g., successful and failed su attempts, failed login attempts, root login attempts, etc.).

Audit:

Review the contents of the /etc/rsyslog.conf file to ensure appropriate logging is set. In addition, perform the following command and ensure that the log files are logging information:

# ls -l /var/log/

Remediation:

Edit the following lines in the /etc/rsyslog.conf file as appropriate for your environment:

|  |
| --- |
| auth,user.\* /var/log/messages kern.\* /var/log/kern.log daemon.\* /var/log/daemon.log syslog.\* /var/log/syslog  lpr,news,uucp,local0,local1,local2,local3,local4,local5,local6.\* /var/log/unused.log  # Execute the following command to restart rsyslogd # pkill -HUP rsyslogd |

References:  See the rsyslog.conf(5) man page for more information.

### 5.1.4 Create and Set Permissions on rsyslog Log Files

A log file must already exist for rsyslog to be able to write to it.

Rationale:

It is important to ensure that log files exist and have the correct permissions to ensure that sensitive rsyslog data is archived and protected.

Audit:

For each <logfile> listed in the /etc/rsyslog.conf file, perform the following command and verify that the <owner>:<group> is root:root and the permissions are 0600 (for sites that have not implemented a secure group) and root:securegrp with permissions of 0640 (for sites that have implemented a secure group):

# ls -l <logfile>

Remediation:

For sites that have not implemented a secure admin group:

Create the /var/log/ directory and for each <logfile> listed in the /etc/rsyslog.conf file, perform the following commands:

# touch <logfile>

# chown root:root <logfile>

# chmod og-rwx <logfile>

For sites that have implemented a secure admin group:

Create the /var/log/ directory and for each <logfile> listed in the /etc/rsyslog.conf file, perform the following commands (where is the name of the security group):

# touch <logfile>

# chown root:<securegrp> <logfile> # chmod g-wx,o-rwx<logfile>

References:

1. See the rsyslog.conf(5) man page for more information.

### 5.1.5 Configure rsyslog to Send Logs to a Remote Log Host

The rsyslog utility supports the ability to send logs it gathers to a remote log host running syslogd(8) or to receive messages from remote hosts, reducing administrative overhead.

Rationale:

Storing log data on a remote host protects log integrity from local attacks. If an attacker gains root access on the local system, they could tamper with or remove log data that is stored on the local system

Audit:

Review the /etc/rsyslog.conf file and verify that logs are sent to a central host (where logfile.example.com is the name of your central log host).

# grep “^\*.\*[^I][^I]\*@” /etc/rsyslog.conf \*.\* @@loghost.example.com

Remediation:

Edit the /etc/rsyslog.conf file and add the following line (where logfile.example.com is the name of your central log host).

\*.\* @@loghost.example.com

# Execute the following command to restart rsyslogd

# pkill -HUP rsyslogd

Note: The double “at” sign (@@) directs rsyslog to use TCP to send log messages to the server, which is a more reliable transport mechanism than the default UDP protocol.

References:

1. See the rsyslog.conf(5) man page for more information.

### 5.1.6 Accept Remote rsyslog Messages Only on Designated Log Hosts

By default, rsyslog does not listen for log messages coming in from remote systems. The

ModLoad tells rsyslog to load the imtcp.so module so it can listen over a network via TCP. The InputTCPServerRun option instructs rsyslogd to listen on the specified TCP port.

Rationale:

The guidance in the section ensures that remote log hosts are configured to only accept rsyslog data from hosts within the specified domain and that those systems that are not designed to be log hosts do not accept any remote rsyslog messages. This provides protection from spoofed log data and ensures that system administrators are reviewing reasonably complete syslog data in a central location.

Audit:

Run the following to determine if rsyslog is listening for remote messages:

# grep ‘$ModLoad imtcp.so’ /etc/rsyslog.conf

$ModLoad imtcp.so

# grep ‘$InputTCPServerRun’ /etc/rsyslog.conf $InputTCPServerRun 514

Remediation:

On hosts that are designated as log hosts edit the /etc/rsyslog.conf file and un-comment the following lines:

$ModLoad imtcp.so

$InputTCPServerRun 514

Note: On hosts that are not designated log hosts these lines should be commented out instead.

Execute the following command to restart rsyslogd:

# pkill -HUP rsyslogd

References:

1. See the rsyslog(8) man page for more information.

## 5.2 Configure System Accounting (auditd)

System auditing, through auditd, allows system administrators to monitor their systems such that they can detect unauthorized access or modification of data. By default, auditd will audit SELinux AVC denials, system logins, account modifications, and authentication events. Events will be logged to /var/log/audit/audit.log. The recording of these events will use a modest amount of disk space on a system. If significantly more events are captured, additional on system or off system storage may need to be allocated.

Note: For 64 bit systems that have arch as a rule parameter, you will need two rules: one for 64 bit and one for 32 bit systems. For 32 bit systems, only one rule is needed.

### 5.2.1 Configure Data Retention

When auditing, it is important to carefully configure the storage requirements for audit logs. By default, auditd will max out the log files at 5MB and retain only 4 copies of them. Older versions will be deleted. It is possible on a system that the 20 MBs of audit logs may fill up the system causing loss of audit data. While the recommendations here provide guidance, check your site policy for audit storage requirements. 5.2.1.1 Configure Audit Log Storage Size

Level 2

Configure the maximum size of the audit log file. Once the log reaches the maximum size, it will be rotated and a new log file will be started.

Rationale:

It is important that an appropriate size is determined for log files so that they do not impact the system and audit data is not lost.

Audit:

Perform the following to determine the maximum size of the audit log files.

# grep max\_log\_file /etc/audit/auditd.conf max\_log\_file = <MB>

Remediation:

Set the max\_log\_file parameter in /etc/audit/auditd.conf

max\_log\_file = <MB>

Note: MB is the number of MegaBytes the file can be.

**5.2.1.2 Disable System on Audit Log Full**

Level 2

The auditd daemon can be configured to halt the system when the audit logs are full.

Rationale:

In high security contexts, the risk of detecting unauthorized access or nonrepudiation exceeds the benefit of the system’s availability.

Audit:

Perform the following to determine if auditd is configured to notify the administrator and halt the system when audit logs are full.

# grep space\_left\_action /etc/audit/auditd.conf space\_left\_action = email

# grep action\_mail\_acct /etc/audit/auditd.conf action\_mail\_acct = root

# grep admin\_space\_left\_action /etc/audit/auditd.conf admin\_space\_left\_action = halt

Remediation:

Add the following lines to the /etc/audit/auditd.conf file.

space\_left\_action = email action\_mail\_acct = root admin\_space\_left\_action = halt

**5.2.1.3 Keep All Auditing Information**

Level 2

Normally, auditd will hold 4 logs of maximum log file size before deleting older log files.

Rationale:

In high security contexts, the benefits of maintaining a long audit history exceed the cost of storing the audit history.

Audit:

Perform the following to determine if audit logs are retained.

# grep max\_log\_file\_action /etc/audit/auditd.conf max\_log\_file\_action = keep\_logs

Remediation:

Add the following line to the /etc/audit/auditd.conf file.

max\_log\_file\_action = keep\_logs

### 5.2.2 Enable auditd Service

Level 2

Turn on the auditd daemon to record system events.

Rationale:

The capturing of system events provides system administrators with information to allow them to determine if unauthorized access to their system is occurring.

Audit:

Perform the following to determine if auditd is enabled.

# chkconfig –list auditd auditd: 0: off 1: off 2: on 3: on 4: on 5: on 6: off

Remediation:

# chkconfig auditd on

### 5.2.3 Enable Auditing for Processes That Start Prior to auditd

Level 2

Configure grub so that processes that are capable of being audited can be audited even if they start up prior to auditd startup.

Rationale:

Audit events need to be captured on processes that start up prior to auditd, so that potential malicious activity cannot go undetected.

Audit:

Perform the following to determine if /etc/grub.conf is configured to log processes that start prior to auditd.

# grep “kernel” /etc/grub.conf

Make sure each line that starts with kernel has the audit=1 parameter set.

Remediation:

# ed /etc/grub.conf << END g/audit=1/s///g g/kernel/s/$/ audit=1/ w q END

### 5.2.4 Record Events That Modify Date and Time Information

Level 2

Capture events where the system date and/or time has been modified. The parameters in this section are set to determine if the adjtimex (tune kernel clock), settimeofday (Set time, using timeval and timezone structures) stime (using seconds since 1/1/1970) or clock\_settime (allows for the setting of several internal clocks and timers) system calls have been executed and always write an audit record to the /var/log/audit.log file upon exit, tagging the records with the identifier “time-change”

Rationale:

Unexpected changes in system date and/or time could be a sign of malicious activity on the system.

Audit:

Perform the following to determine if events where the system date and/or time has been modified are captured.

On a 64 bit system, perform the following command and ensure the output is as shown. Note: “-a always,exit” may be specified as “-a exit,always”.

|  |
| --- |
| # grep time-change /etc/audit/audit.rules  -a always,exit -F arch=b64 -S adjtimex -S settimeofday -k time-change  -a always,exit -F arch=b32 -S adjtimex -S settimeofday -S stime -k time-change  -a always,exit -F arch=b64 -S clock\_settime -k time-change  -a always,exit -F arch=b32 -S clock\_settime -k time-change  -w /etc/localtime -p wa -k time-change  # Execute the following command to restart auditd # pkill -P 1-HUP auditd |

On a 32 bit system, perform the following command and ensure the output is as shown. Note: “-a always,exit” may be specified as “-a exit,always”.

|  |
| --- |
| # grep time-change /etc/audit/audit.rules  -a always,exit -F arch=b32 -S adjtimex -S settimeofday -S stime -k time-change  -a always,exit -F arch=b32 -S clock\_settime -k time-change  -w /etc/localtime -p wa -k time-change  # Execute the following command to restart auditd # pkill -P 1-HUP auditd |

Remediation:

For 64 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b64 -S adjtimex -S settimeofday -k time-change  -a always,exit -F arch=b32 -S adjtimex -S settimeofday -S stime -k time-change  -a always,exit -F arch=b64 -S clock\_settime -k time-change  -a always,exit -F arch=b32 -S clock\_settime -k time-change  -w /etc/localtime -p wa -k time-change  # Execute the following command to restart auditd # pkill -P 1-HUP auditd |

For 32 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b32 -S adjtimex -S settimeofday -S stime -k time-change  -a always,exit -F arch=b32 -S clock\_settime -k time-change  -w /etc/localtime -p wa -k time-change  # Execute the following command to restart auditd # pkill -P 1-HUP auditd |

### 5.2.5 Record Events That Modify User/Group Information

Level 2

Record events affecting the group, passwd (user IDs), shadow and gshadow (passwords) or

/etc/security/opasswd (old passwords, based on remember parameter in the PAM configuration) files. The parameters in this section will watch the files to see if they have been opened for write or have had attribute changes (e.g. permissions) and tag them with the identifier “identity” in the audit log file.

Rationale:

Unexpected changes to these files could be an indication that the system has been compromised and that an unauthorized user is attempting to hide their activities or compromise additional accounts.

Audit:

Perform the following to determine if events that modify user/group information are recorded.

# grep identity /etc/audit/audit.rules

-w /etc/group -p wa -k identity

-w /etc/passwd -p wa -k identity

-w /etc/gshadow -p wa -k identity

-w /etc/shadow -p wa -k identity

-w /etc/security/opasswd -p wa -k identity

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

-w /etc/group -p wa -k identity

-w /etc/passwd -p wa -k identity

-w /etc/gshadow -p wa -k identity

-w /etc/shadow -p wa -k identity

-w /etc/security/opasswd -p wa -k identity

# Execute the following command to restart auditd

# pkill -P 1-HUP auditd

### 5.2.6 Record Events That Modify the System’s Network Environment

Level 2

Record changes to network environment files or system calls. The below parameters monitor the sethostname (set the system’s host name) or setdomainname (set the system’s domainname) system calls, and write an audit event on system call exit. The other parameters monitor the /etc/issue and /etc/issue.net files (messages displayed prelogin), /etc/hosts (file containing host names and associated IP addresses) and

/etc/sysconfig/network (directory containing network interface scripts and configurations) files.

Rationale:

Monitoring sethostname and setdomainname will identify potential unauthorized changes to host and domainname of a system. The changing of these names could potentially break security parameters that are set based on those names. The /etc/hosts file is monitored for changes in the file that can indicate an unauthorized intruder is trying to change machine associations with IP addresses and trick users and processes into connecting to unintended machines. Monitoring /etc/issue and /etc/issue.net is important, as intruders could put disinformation into those files and trick users into providing information to the intruder. Monitoring /etc/sysconfig/network is important as it can show if network interfaces or scripts are being modified in a way that can lead to the machine becoming unavailable or compromised. All audit records will be tagged with the identifier “system-locale.”

Audit:

On a 64 bit system, perform the following command and ensure the output is as shown to determine if events that modify the system’s environment are recorded. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep system-locale /etc/audit/audit.rules

-a always,exit -F arch=b64 -S sethostname -S setdomainname -k system-locale

-a always,exit -F arch=b32 -S sethostname -S setdomainname -k system-locale

-w /etc/issue -p wa -k system-locale

-w /etc/issue.net -p wa -k system-locale

-w /etc/hosts -p wa -k system-locale

-w /etc/sysconfig/network -p wa -k system-locale

For 32 bit systems, perform the following command and ensure the output is as shown to determine if events that modify the system’s environment are recorded. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep system-locale /etc/audit/audit.rules

-a always,exit -F arch=b32 -S sethostname -S setdomainname -k system-locale

-w /etc/issue -p wa -k system-locale

-w /etc/issue.net -p wa -k system-locale

-w /etc/hosts -p wa -k system-locale

-w /etc/sysconfig/network -p wa -k system-locale

Remediation:

For 64 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b64 -S sethostname -S setdomainname -k system-locale  -a always,exit -F arch=b32 -S sethostname -S setdomainname -k system-locale  -w /etc/issue -p wa -k system-locale  -w /etc/issue.net -p wa -k system-locale  -w /etc/hosts -p wa -k system-locale  -w /etc/sysconfig/network -p wa -k system-locale  # Execute the following command to restart auditd # pkill -P 1-HUP auditd |

For 32 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b32 -S sethostname -S setdomainname -k system-locale  -w /etc/issue -p wa -k system-locale  -w /etc/issue.net -p wa -k system-locale  -w /etc/hosts -p wa -k system-locale  -w /etc/sysconfig/network -p wa -k system-locale  # Execute the following command to restart auditd # pkill -P 1-HUP auditd |

### 5.2.7 Record Events That Modify the System’s Mandatory Access

Controls

Level 2

Monitor SELinux mandatory access controls. The parameters below monitor any write access (potential additional, deletion or modification of files in the directory) or attribute changes to the /etc/selinux directory.

Rationale:

Changes to files in this directory could indicate that an unauthorized user is attempting to modify access controls and change security contexts, leading to a compromise of the system.

Audit:

Perform the following to determine if events that modify the system’s mandatory access controls are recorded

# grep MAC-policy /etc/audit/audit.rules -w /etc/selinux/ -p wa -k MAC-policy

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

Add the following lines to /etc/audit/audit.rules -w /etc/selinux/ -p wa -k MAC-policy

# Execute the following command to restart auditd # pkill -P 1-HUP auditd

### 5.2.8 Collect Login and Logout Events

Level 2

Monitor login and logout events. The parameters below track changes to files associated with login/logout events. The file /var/log/faillog tracks failed events from login. The file /var/log/lastlog maintain records of the last time a user successfully logged in. The file /var/log/btmp keeps track of failed login attempts and can be read by entering the command /usr/bin/last -f /var/log/btmp. All audit records will be tagged with the identifier “logins.”

Rationale:

Monitoring login/logout events could provide a system administrator with information associated with brute force attacks against user logins.

Audit:

Perform the following to determine if login and logout events are recorded.

# grep logins /etc/audit/audit.rules

-w /var/log/faillog -p wa -k logins

-w /var/log/lastlog -p wa -k logins -w /var/log/tallylog -p wa -k logins

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

-w /var/log/faillog -p wa -k logins

-w /var/log/lastlog -p wa -k logins

-w /var/log/tallylog -p wa -k logins

# Execute the following command to restart auditd

# pkill -HUP -P 1 auditd

### 5.2.9 Collect Session Initiation Information

Level 2

Monitor session initiation events. The parameters in this section track changes to the files associated with session events. The file /var/run/utmp file tracks all currently logged in users. The /var/log/wtmp file tracks logins, logouts, shutdown and reboot events. All audit records will be tagged with the identifier “session.”

Rationale:

Monitoring these files for changes could alert a system administrator to logins occurring at unusual hours, which could indicate intruder activity (i.e. a user logging in at a time when they do not normally log in).

Audit:

Perform the following to determine if session initiation information is collected.

# grep session /etc/audit/audit.rules

-w /var/run/utmp -p wa -k session

-w /var/log/wtmp -p wa -k session

-w /var/log/btmp -p wa -k session

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

-w /var/run/utmp -p wa -k session

-w /var/log/wtmp -p wa -k session

-w /var/log/btmp -p wa -k session

# Execute the following command to restart auditd

# pkill -HUP -P 1 auditd

Note: Use the last command to read /var/log/wtmp (last with no parameters) and

/var/run/utmp (last -f /var/run/utmp)

### 5.2.10 Collect Discretionary Access Control Permission Modification

Events

Level 2

Monitor changes to file permissions, attributes, ownership and group. The parameters in this section track changes for system calls that affect file permissions and attributes. The chmod, fchmod and fchmodat system calls affect the permissions associated with a file. The chown, fchown, fchownat and lchown system calls affect owner and group attributes on a file. The setxattr, lsetxattr, fsetxattr (set extended file attributes) and removexattr, lremovexattr, fremovexattr (remove extended file attributes) control extended file attributes. In all cases, an audit record will only be written for non-system userids (auid >= 500) and will ignore Daemon events (auid = 4294967295). All audit records will be tagged with the identifier “perm\_mod.”

Rationale:

Monitoring for changes in file attributes could alert a system administrator to activity that could indicate intruder activity or policy violation.

Audit:

For 64 bit systems, perform the following command and ensure the output is as shown to determine if permission modifications are being recorded. Note: “-a always,exit” may be specified as “-a exit,always”.

|  |
| --- |
| # grep perm\_mod /etc/audit/audit.rules  -a always,exit -F arch=b64 -S chmod -S fchmod -S fchmodat -F auid>=500 \ -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -F auid>=500 \  -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b64 -S chown -S fchown -S fchownat -S lchown -F auid>=500 \ -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b32 -S chown -S fchown -S fchownat -S lchown -F auid>=500 \ -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b64 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S \ lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b32 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S \ lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm\_mod |

For 32 bit systems, perform the following command and ensure the output is as shown to determine if permission modifications are being recorded. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep perm\_mod /etc/audit/audit.rules

-a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -F auid>=500 \

-F auid!=4294967295 -k perm\_mod

-a always,exit -F arch=b32 -S chown -S fchown -S fchownat -S lchown -F auid>=500 \ -F auid!=4294967295 -k perm\_mod

-a always,exit -F arch=b32 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S \ lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm\_mod

Remediation:

For 64 bit systems, add the following lines to the /etc/audit/audit.rules file.

-a always,exit -F arch=b64 -S chmod -S fchmod -S fchmodat -F auid>=500 \ -F auid!=4294967295 -k perm\_mod

|  |
| --- |
| -a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -F auid>=500 \  -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b64 -S chown -S fchown -S fchownat -S lchown -F auid>=500 \ -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b32 -S chown -S fchown -S fchownat -S lchown -F auid>=500 \ -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b64 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S \ lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b32 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S \ lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm\_mod  # Execute the following command to restart auditd # pkill -HUP -P 1 auditd |

For 32 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b32 -S chmod -S fchmod -S fchmodat -F auid>=500 \  -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b32 -S chown -S fchown -S fchownat -S lchown -F auid>=500 \ -F auid!=4294967295 -k perm\_mod  -a always,exit -F arch=b32 -S setxattr -S lsetxattr -S fsetxattr -S removexattr -S \ lremovexattr -S fremovexattr -F auid>=500 -F auid!=4294967295 -k perm\_mod  # Execute the following command to restart auditd # pkill -HUP -P 1 auditd |

### 5.2.11 Collect Unsuccessful Unauthorized Access Attempts to Files

Level 2

Monitor for unsuccessful attempts to access files. The parameters below are associated with system calls that control creation (creat), opening (open, openat) and truncation (truncate, ftruncate) of files. An audit log record will only be written if the user is a nonprivileged user (auid > = 500), is not a Daemon event (auid=4294967295) and if the system call returned EACCES (permission denied to the file) or EPERM (some other permanent error associated with the specific system call). All audit records will be tagged with the identifier “access.”

Rationale:

Failed attempts to open, create or truncate files could be an indication that an individual or process is trying to gain unauthorized access to the system.

Audit:

On 64 bit systems, perform the following command and ensure the output is as shown to determine if there are unsuccessful attempts to access files. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep access /etc/audit/audit.rules

-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access

-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access

-a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access

-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \

-F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access

On 32 bit systems, perform the following command and ensure the output is as shown to determine if there are unsuccessful attempts to access files. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep access /etc/audit/audit.rules

-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access

-a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access

Remediation:

For 64 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access  -a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access  -a always,exit -F arch=b64 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access  -a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \  -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access  # Execute the following command to restart auditd # pkill -HUP -P 1 auditd |

For 32 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \ -F exit=-EACCES -F auid>=500 -F auid!=4294967295 -k access  -a always,exit -F arch=b32 -S creat -S open -S openat -S truncate -S ftruncate \  -F exit=-EPERM -F auid>=500 -F auid!=4294967295 -k access  # Execute the following command to restart auditd # pkill -HUP -P 1 auditd |

### 5.2.12 Collect Use of Privileged Commands

Level 2

Monitor privileged programs (those that have the setuid and/or setgid bit set on execution) to determine if unprivileged users are running these commands.

Rationale:

Execution of privileged commands by non-privileged users could be an indication of someone trying to gain unauthorized access to the system.

Audit:

Verify that an audit line for each setuid/setgid program identified in the find command appears in the audit file with the above attributes.

Remediation:

To remediate this issue, the system administrator will have to execute a find command to locate all the privileged programs and then add an audit line for each one of them. The audit parameters associated with this are as follows:

-F path=” $1 ” – will populate each file name found through the find command and processed by awk.

-F perm=x – will write an audit record if the file is executed.

-F auid>=500 – will write a record if the user executing the command is not a privileged user.

-F auid!= 4294967295 – will ignore Daemon events

All audit records will be tagged with the identifier “privileged.”

# find PART -xdev \( -perm -4000 -o -perm -2000 \) -type f | awk ‘{print \ “-a always,exit -F path=” $1 ” -F perm=x -F auid>=500 -F auid!=4294967295 \ -k privileged” }’

Next, add those lines to the /etc/audit/audit.rules file.

### 5.2.13 Collect Successful File System Mounts

Level 2

Monitor the use of the mount system call. The mount (and umount) system call controls the mounting and unmounting of file systems. The parameters below configure the system to create an audit record when the mount system call is used by a non-privileged user Rationale:

It is highly unusual for a non privileged user to mount file systems to the system. While tracking mount commands gives the system administrator evidence that external media may have been mounted (based on a review of the source of the mount and confirming it’s an external media type), it does not conclusively indicate that data was exported to the media. System administrators who wish to determine if data were exported, would also have to track successful open, creat and truncate system calls requiring write access to a file under the mount point of the external media file system. This could give a fair indication that a write occurred. The only way to truly prove it, would be to track successful writes to the external media. Tracking write system calls could quickly fill up the audit log and is not recommended. Recommendations on configuration options to track data export to media is beyond the scope of this document.

Note: This tracks successful and unsuccessful mount commands. File system mounts do not have to come from external media and this action still does not verify write (e.g. CD ROMS)

Audit:

For 64 bit systems perform the following command and ensure the output is as shown to determine if filesystem mounts are recorded. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep mounts /etc/audit/audit.rules

-a always,exit -F arch=b64 -S mount -F auid>=500 -F auid!=4294967295 -k mounts

-a always,exit -F arch=b32 -S mount -F auid>=500 -F auid!=4294967295 -k mounts

For 32 bit systems perform the following command and ensure the output is as shown to determine if filesystem mounts are recorded. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep mounts /etc/audit/audit.rules

-a always,exit -F arch=b32 -S mount -F auid>=500 -F auid!=4294967295 -k mounts

Remediation:

For 64 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b64 -S mount -F auid>=500 -F auid!=4294967295 -k mounts  -a always,exit -F arch=b32 -S mount -F auid>=500 -F auid!=4294967295 -k mounts  # Execute the following command to restart auditd # pkill -HUP -P 1 auditd |

For 32 bit systems, add the following lines to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b32 -S mount -F auid>=500 -F auid!=4294967295 -k mounts  # Execute the following command to restart auditd # pkill -HUP -P 1 auditd |

### 5.2.14 Collect File Deletion Events by User

Level 2

Monitor the use of system calls associated with the deletion or renaming of files and file attributes. This configuration statement sets up monitoring for the unlink (remove a file), unlinkat (remove a file attribute), rename (rename a file) and renameat (rename a file attribute) system calls and tags them with the identifier “delete”.

Rationale:

Monitoring these calls from non-privileged users could provide a system administrator with evidence that inappropriate removal of files and file attributes associated with protected files is occurring. While this audit option will look at all events, system administrators will want to look for specific privileged files that are being deleted or altered.

Audit:

For 64 bit systems, perform the following command and ensure the output is as shown to determine if file deletion events by user are recorded.

# grep delete /etc/audit/audit.rules

-a always,exit -F arch=b64 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 \

-F auid!=4294967295 -k delete -a always,exit -F arch=b32 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 \ -F auid!=4294967295 -k delete

For 32 bit systems, perform the following command and ensure the output is as shown to determine if file deletion events by user are recorded.

# grep delete /etc/audit/audit.rules

-a always,exit -F arch=b32 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 \

-F auid!=4294967295 -k delete

Remediation:

At a minimum, configure the audit system to collect file deletion events for all users and root.

For 64 bit systems, add the following to the /etc/audit/audit.rules file.

-a always,exit -F arch=b64 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 \

-F auid!=4294967295 -k delete

|  |
| --- |
| -a always,exit -F arch=b32 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 \  -F auid!=4294967295 -k delete  # Execute the following command to restart auditd # pkill -HUP -P 1 auditd |

For 32 bit systems, add the following to the /etc/audit/audit.rules file.

|  |
| --- |
| -a always,exit -F arch=b32 -S unlink -S unlinkat -S rename -S renameat -F auid>=500 \  -F auid!=4294967295 -k delete  # Execute the following command to restart auditd # pkill -P 1-HUP auditd |

### 5.2.15 Collect Changes to System Administration Scope (sudoers)

Level 2

Monitor scope changes for system administrations. If the system has been properly configured to force system administrators to log in as themselves first and then use the sudo command to execute privileged commands, it is possible to monitor changes in scope.

The file /etc/sudoers will be written to when the file or its attributes have changed. The audit records will be tagged with the identifier “scope.”

Rationale:

Changes in the /etc/sudoers file can indicate that an unauthorized change has been made to scope of system administrator activity.

Audit:

Perform the following to determine if changes to /etc/sudoers are recorded.

# grep scope /etc/audit/audit.rules -w /etc/sudoers -p wa -k scope

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

-w /etc/sudoers -p wa -k scope

# Execute the following command to restart auditd

# pkill -HUP -P 1 auditd

### 5.2.16 Collect System Administrator Actions (sudolog)

Level 2

Monitor the sudo log file. If the system has been properly configured to disable the use of the su command and force all administrators to have to log in first and then use sudo to execute privileged commands, then all administrator commands will be logged to

/var/log/sudo.log. Any time a command is executed, an audit event will be triggered as the /var/log/sudo.log file will be opened for write and the executed administration command will be written to the log.

Rationale:

Changes in /var/log/sudo.log indicate that an administrator has executed a command or the log file itself has been tampered with. Administrators will want to correlate the events written to the audit trail with the records written to /var/log/sudo.log to verify if unauthorized commands have been executed.

Audit:

Perform the following to determine if administrator activity is recorded.

# grep actions /etc/audit/audit.rules

-w /var/log/sudo.log -p wa -k actions

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

-w /var/log/sudo.log -p wa -k actions

# Execute the following command to restart auditd

# pkill -HUP -P 1 auditd

Note: The system must be configured with su disabled (See Item 7.6 Restrict Access to the su Command) to force all command execution through sudo. This will not be effective on the console, as administrators can log in as root.

### 5.2.17 Collect Kernel Module Loading and Unloading

Level 2

Monitor the loading and unloading of kernel modules. The programs insmod (install a kernel module), rmmod (remove a kernel module), and modprobe (a more sophisticated program to load and unload modules, as well as some other features) control loading and unloading of modules. The init\_module (load a module) and delete\_module (delete a module) system calls control loading and unloading of modules. Any execution of the loading and unloading module programs and system calls will trigger an audit record with an identifier of “modules”.

Rationale:

Monitoring the use of insmod, rmmod and modprobe could provide system administrators with evidence that an unauthorized user loaded or unloaded a kernel module, possibly compromising the security of the system. Monitoring of the init\_module and delete\_module system calls would reflect an unauthorized user attempting to use a different program to load and unload modules.

Audit:

Perform the following to determine if kernel module loading and unloading is recorded. Note: “-a always,exit” may be specified as “-a exit,always”.

# grep modules /etc/audit/audit.rules

-w /sbin/insmod -p x -k modules

-w /sbin/rmmod -p x -k modules

-w /sbin/modprobe -p x -k modules

For 32 bit systems

-a always,exit -F arch=b32 -S init\_module -S delete\_module -k modules For 64 bit systems

-a always,exit -F arch=b64 -S init\_module -S delete\_module -k modules

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

-w /sbin/insmod -p x -k modules

-w /sbin/rmmod -p x -k modules

-w /sbin/modprobe -p x -k modules

For 32 bit systems, add

-a always,exit -F arch=b32 -S init\_module -S delete\_module -k modules For 64 bit systems, add

-a always,exit -F arch=b64 -S init\_module -S delete\_module -k modules

### 5.2.18 Make the Audit Configuration Immutable

Level 2

Set system audit so that audit rules cannot be modified with auditctl. Setting the flag “-e 2” forces audit to be put in immutable mode. Audit changes can only be made on system reboot.

Rationale:

In immutable mode, unauthorized users cannot execute changes to the audit system to potential hide malicious activity and then put the audit rules back. Users would most likely notice a system reboot and that could alert administrators of an attempt to make unauthorized audit changes.

Audit:

Perform the following to determine if the audit configuration is immutable.

# grep “^-e 2” /etc/audit/audit.rules -e 2

Remediation:

Add the following lines to the /etc/audit/audit.rules file.

-e 2

Note: This must be the last entry in the /etc/audit/audit.rules file

## 5.3 Configure logrotate

The system includes the capability of rotating log files regularly to avoid filling up the system with logs or making the logs unmanageable large. The file

/etc/logrotate.d/syslog is the configuration file used to rotate log files created by syslog or rsyslog. These files are rotated on a weekly basis via a cron job and the last 4 weeks are kept.

Rationale:

By keeping the log files smaller and more manageable, a system administrator can easily archive these files to another system and spend less time looking through inordinately large log files.

Audit:

Perform the following to determine if the appropriate system logs are rotated.

# grep ‘{‘ /etc/logrotate.d/syslog

/var/log/messages /var/log/secure /var/log/maillog /var/log/spooler /var/log/boot.log

/var/log/cron {

Remediation:

Edit the /etc/logrotate.d/syslog file to include appropriate system logs:

/var/log/messages /var/log/secure /var/log/maillog /var/log/spooler /var/log/boot.log /var/log/cron {

## 6.1 Configure cron and anacron

### 6.1.1 Enable anacron Daemon

The anacron daemon is used on systems that are not up 24×7. The anacron daemon will execute jobs that would have normally been run had the system not been down.

Rationale:

Cron jobs may include critical security or administrative functions that need to run on a regular basis. Use this daemon on machines that are not up 24×7, or if there are jobs that need to be executed after the system has been brought back up after a maintenance window.

Audit:

Perform the following to determine if anacron is enabled.

# rpm -q cronie-anacron cronie-anacron.<package version>.<hardware platform>

Remediation:

# yum install cronie-anacron

Note: NSA Guidance recommends disabling anacron for systems that are intended to be up 24X7, with the rationale that unnecessary software should be disabled to reduce risk. However, even systems that are designed to be up at all times can experience downtime that could prevent important system maintenance jobs from running. Review the requirements for your site to determine your appropriate risk level.

### 6.1.2 Enable crond Daemon

The crond daemon is used to execute batch jobs on the system.

Rationale:

While there may not be user jobs that need to be run on the system, the system does have maintenance jobs that may include security monitoring that have to run and crond is used to execute them.

Audit:

Perform the following to determine if cron is enabled.

# chkconfig –list crond crond: 0:off 1:off 2:on 3:on 4:on 5:on 6:off

Remediation:

# chkconfig crond on

### 6.1.3 Set User/Group Owner and Permission on /etc/anacrontab

The /etc/anacrontab file is used by anacron to control its own jobs. The commands in this item make sure that root is the user and group owner of the file and is the only user that can read and write the file.

Rationale:

This file contains information on what system jobs are run by anacron. Write access to these files could provide unprivileged users with the ability to elevate their privileges. Read access to these files could provide users with the ability to gain insight on system jobs that run on the system and could provide them a way to gain unauthorized privileged access.

Audit:

Perform the following to determine if the /etc/anacrontab file has the correct permissions.

# stat -L -c “%a %u %g” /etc/anacrontab | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/anacrontab

# chmod og-rwx /etc/anacrontab

### 6.1.4 Set User/Group Owner and Permission on /etc/crontab

The /etc/crontab file is used by cron to control its own jobs. The commands in this item make here sure that root is the user and group owner of the file and is the only user that can read and write the file.

Rationale:

This file contains information on what system jobs are run by cron. Write access to these files could provide unprivileged users with the ability to elevate their privileges. Read access to these files could provide users with the ability to gain insight on system jobs that run on the system and could provide them a way to gain unauthorized privileged access.

Audit:

Perform the following to determine if the /etc/crontab file has the correct permissions.

# stat -L -c “%a %u %g” /etc/crontab | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/crontab

# chmod og-rwx /etc/crontab

### 6.1.5 Set User/Group Owner and Permission on /etc/cron.hourly

This directory contains system cron jobs that need to run on an hourly basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Perform the following to determine if the /etc/cron.hourly file has the correct permissions.

# stat -L -c “%a %u %g” /etc/cron.hourly | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/cron.hourly

# chmod og-rwx /etc/cron.hourly

### 6.1.6 Set User/Group Owner and Permission on /etc/cron.daily

The /etc/cron.daily directory contains system cron jobs that need to run on a daily basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Perform the following to determine if the /etc/cron.daily directory has the correct permissions.

# stat -L -c “%a %u %g” /etc/cron.daily | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/cron.daily

# chmod og-rwx /etc/cron.daily

### 6.1.7 Set User/Group Owner and Permission on /etc/cron.weekly

The /etc/cron.weekly directory contains system cron jobs that need to run on a weekly basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Perform the following to determine if the /etc/cron.weekly directory has the correct permissions.

# stat -L -c “%a %u %g” /etc/cron.weekly | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/cron.weekly

# chmod og-rwx /etc/cron.weekly

### 6.1.8 Set User/Group Owner and Permission on /etc/cron.monthly

The /etc/cron.monthly directory contains system cron jobs that need to run on a monthly basis. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Perform the following to determine if the /etc/cron.monthly directory has the correct permissions.

# stat -L -c “%a %u %g” /etc/cron.monthly | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/cron.monthly

# chmod og-rwx /etc/cron.monthly

### 6.1.9 Set User/Group Owner and Permission on /etc/cron.d

The /etc/cron.d directory contains system cron jobs that need to run in a similar manner to the hourly, daily weekly and monthly jobs from /etc/crontab, but require more granular control as to when they run. The files in this directory cannot be manipulated by the crontab command, but are instead edited by system administrators using a text editor. The commands below restrict read/write and search access to user and group root, preventing regular users from accessing this directory.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means for gaining unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls.

Audit:

Perform the following to determine if the /etc/cron.d directory has the correct permissions.

# stat -L -c “%a %u %g” /etc/cron.d | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# chown root:root /etc/cron.d

# chmod og-rwx /etc/cron.d

### 6.1.10 Restrict at Daemon

The at daemon works with the cron daemon to allow non-privileged users to submit one time only jobs at their convenience. There are two files that control at: /etc/at.allow and /etc/at.deny. If /etc/at.allow exists, then users listed in the file are the only ones that can create at jobs. If /etc/at.allow does not exist and /etc/at.deny does exist, then any user on the system, with the exception of those listed in /etc/at.deny, are allowed to execute at jobs. An empty /etc/at.deny file allows any user to create at jobs. If neither

/etc/at.allow nor /etc/at.deny exist, then only superuser can create at jobs. The commands below remove the /etc/at.deny file and create an empty /etc/at.allow file that can only be read and modified by user and group root.

Rationale:

Granting write access to this directory for non-privileged users could provide them the means to gain unauthorized elevated privileges. Granting read access to this directory could give an unprivileged user insight in how to gain elevated privileges or circumvent auditing controls. In addition, it is a better practice to create a white list of users who can execute at jobs versus a blacklist of users who can’t execute at jobs as a system administrator will always know who can create jobs and does not have to worry about remembering to add a user to the blacklist when a new user id is created.

Audit:

Perform the following to determine if at jobs are restricted.

# stat -L /etc/at.deny > /dev/null

# stat -L -c “%a %u %g” /etc/at.allow | egrep “.00 0 0”

If the above command emits no output then the system is not configured as recommended.

Remediation:

# rm /etc/at.deny

# touch /etc/at.allow

# chown root:root /etc/at.allow

# chmod og-rwx /etc/at.allow

### 6.1.11 Restrict at/cron to Authorized Users

Configure /etc/cron.allow and /etc/at.allow to allow specific users to use these services. If /etc/cron.allow or /etc/at.allow do not exist, then /etc/at.deny and /etc/cron.deny are checked. Any user not specifically defined in those files is allowed to use at and cron. By removing the files, only users in /etc/cron.allow and /etc/at.allow are allowed to use at and cron. Note that even though a given user is not listed in cron.allow, cron jobs can still be run as that user. The cron.allow file only controls administrative access to the crontab command for scheduling and modifying cron jobs.

Rationale:

On many systems, only the system administrator is authorized to schedule cron jobs. Using the cron.allow file to control who can run cron jobs enforces this policy. It is easier to manage an allow list than a deny list. In a deny list, you could potentially add a user ID to the system and forget to add it to the deny files.

Audit:

Perform the following to determine if the remediation in the section has been performed:

# ls -l /etc/cron.deny

[no output returned]

# ls -l /etc/at.deny

[no output returned]

# ls -l /etc/cron.allow

-rw——- 1 root root /etc/cron.allow

# ls -l /etc/at.allow

-rw——- 1 root root /etc/at.allow

Remediation:

If /etc/cron.allow or /etc/at.allow do not exist on your system create them.

Run the following to ensure cron.deny and at.deny are removed and permissions are set correctly:

# /bin/rm /etc/cron.deny

# /bin/rm /etc/at.deny

# chmod og-rwx /etc/cron.allow

# chmod og-rwx /etc/at.allow

# chown root:root /etc/cron.allow

# chown root:root /etc/at.allow

## 6.2 Configure SSH

SSH is a secure, encrypted replacement for common login services such as telnet, ftp, rlogin, rsh, and rcp.

Rationale: It is strongly recommended that sites abandon older clear-text login protocols and use SSH to prevent session hijacking and sniffing of sensitive data off the network.

### 6.2.1 Set SSH Protocol to 2

SSH supports two different and incompatible protocols: SSH1 and SSH2. SSH1 was the original protocol and was subject to security issues. SSH2 is more advanced and secure.

Rationale:

SSH v1 suffers from insecurities that do not affect SSH v2.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^Protocol” /etc/ssh/sshd\_config Protocol 2

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

Protocol 2

### 6.2.2 Set LogLevel to INFO

The INFO parameter specifies that login and logout activity will be logged.

Rationale:

SSH provides several logging levels with varying amounts of verbosity. DEBUG is specifically not recommended other than strictly for debugging SSH communications since it provides so much data that it is difficult to identify important security information. INFO level is the basic level that only records login activity of SSH users. In many situations, such as Incident Response, it is important to determine when a particular user was active on a system. The logout record can eliminate those users who disconnected, which helps narrow the field.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^LogLevel” /etc/ssh/sshd\_config LogLevel INFO

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

LogLevel INFO

### 6.2.3 Set Permissions on /etc/ssh/sshd\_config

The /etc/ssh/sshd\_config file contains configuration specifications for sshd. The command below sets the owner and group of the file to root.

Rationale:

The /etc/ssh/sshd\_config file needs to be protected from unauthorized changes by nonprivileged users, but needs to be readable as this information is used with many nonprivileged programs.

Audit:

Run the following command to determine the user and group ownership on the /etc/ssh/sshd\_config file.

# /bin/ls -l /etc/ssh/sshd\_config -rw——- 1 root root 762 Sep 23 002 /etc/ssh/sshd\_config

Remediation:

If the user and group ownership of the /etc/ssh/sshd\_config file are incorrect, run the following command to correct them:

# chown root:root /etc/ssh/sshd\_config

If the permissions are incorrect, run the following command to correct them:

# chmod 600 /etc/ssh/sshd\_config

### 6.2.4 Disable SSH X11 Forwarding

The X11Forwarding parameter provides the ability to tunnel X11 traffic through the connection to enable remote graphic connections.

Rationale:

Disable X11 forwarding unless there is an operational requirement to use X11 applications directly. There is a small risk that the remote X11 servers of users who are logged in via SSH with X11 forwarding could be compromised by other users on the X11 server. Note that even if X11 forwarding is disabled, users can always install their own forwarders.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^X11Forwarding” /etc/ssh/sshd\_config X11Forwarding no

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

X11Forwarding no

### 6.2.5 Set SSH MaxAuthTries to 4 or Less

The MaxAuthTries parameter specifies the maximum number of authentication attempts permitted per connection. When the login failure count reaches half the number, error messages will be written to the syslog file detailing the login failure.

Rationale:

Setting the MaxAuthTries parameter to a low number will minimize the risk of successful brute force attacks to the SSH server. While the recommended setting is 4, it is set the number based on site policy.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^MaxAuthTries” /etc/ssh/sshd\_config MaxAuthTries 4

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

MaxAuthTries 4

### 6.2.6 Set SSH IgnoreRhosts to Yes

The IgnoreRhosts parameter specifies that .rhosts and .shosts files will not be used in RhostsRSAAuthentication or HostbasedAuthentication.

Rationale:

Setting this parameter forces users to enter a password when authenticating with ssh.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^IgnoreRhosts” /etc/ssh/sshd\_config IgnoreRhosts yes

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

IgnoreRhosts yes

### 6.2.7 Set SSH HostbasedAuthentication to No

The HostbasedAuthentication parameter specifies if authentication is allowed through trusted hosts via the user of .rhosts, or /etc/hosts.equiv, along with successful public key client host authentication. This option only applies to SSH Protocol Version 2.

Rationale:

Even though the .rhosts files are ineffective if support is disabled in /etc/pam.conf, disabling the ability to use .rhosts files in SSH provides an additional layer of protection .

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^HostbasedAuthentication” /etc/ssh/sshd\_config HostbasedAuthentication no

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

HostbasedAuthentication no

### 6.2.8 Disable SSH Root Login

The PermitRootLogin parameter specifies if the root user can log in using ssh(1). The default is no.

Rationale:

Disallowing root logins over SSH requires server admins to authenticate using their own individual account, then escalating to root via sudo or su. This in turn limits opportunity for non-repudiation and provides a clear audit trail in the event of a security incident

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^PermitRootLogin” /etc/ssh/sshd\_config PermitRootLogin no

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

PermitRootLogin no

### 6.2.9 Set SSH PermitEmptyPasswords to No

The PermitEmptyPasswords parameter specifies if the server allows login to accounts with empty password strings.

Rationale:

Disallowing remote shell access to accounts that have an empty password reduces the probability of unauthorized access to the system

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^PermitEmptyPasswords” /etc/ssh/sshd\_config PermitEmptyPasswords no

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

PermitEmptyPasswords no

### 6.2.10 Do Not Allow Users to Set Environment Options

The PermitUserEnvironment option allows users to present environment options to the ssh daemon.

Rationale:

Permitting users the ability to set environment variables through the SSH daemon could potentially allow users to bypass security controls (e.g. setting an execution path that has ssh executing trojaned programs).

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep PermitUserEnvironment /etc/ssh/sshd\_config PermitUserEnvironment no

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

PermitUserEnvironment no

### 6.2.11 Use Only Approved Cipher in Counter Mode

This variable limits the types of ciphers that SSH can use during communication.

Rationale:

Based on research conducted at various institutions, it was determined that the symmetric portion of the SSH Transport Protocol (as described in RFC 4253) has security weaknesses that allowed recovery of up to 32 bits of plaintext from a block of ciphertext that was encrypted with the Cipher Block Chaining (CBC) method. From that research, new Counter mode algorithms (as described in RFC4344) were designed that are not vulnerable to these types of attacks and these algorithms are now recommended for standard use.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep -v “Ciphers” /etc/ssh/sshd\_config

Ciphers aes128-ctr,aes192-ctr,aes256-ctr

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

Ciphers aes128-ctr,aes192-ctr,aes256-ctr

References:

1. For more information on the Counter mode algorithms, read RFC4344 at<http://www.ietf.org/rfc/rfc4344.txt.>

### 6.2.12 Set Idle Timeout Interval for User Login

The two options ClientAliveInterval and ClientAliveCountMax control the timeout of ssh sessions. When the ClientAliveInterval variable is set, ssh sessions that have no activity for the specified length of time are terminated. When the ClientAliveCountMax variable is set, sshd will send client alive messages at every ClientAliveInterval interval. When the number of consecutive client alive messages are sent with no response from the client, the ssh session is terminated. For example, if the ClientAliveInterval is set to 15 seconds and the ClientAliveCountMax is set to 3, the client ssh session will be terminated after 45 seconds of idle time.

Rationale:

Having no timeout value associated with a connection could allow an unauthorized user access to another user’s ssh session (e.g. user walks away from their computer and doesn’t lock the screen). Setting a timeout value at least reduces the risk of this happening..

While the recommended setting is 300 seconds (5 minutes), set this timeout value based on site policy. The recommended setting for ClientAliveCountMax is 0. In this case, the client session will be terminated after 5 minutes of idle time and no keepalive messages will be sent.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^ClientAliveInterval” /etc/ssh/sshd\_config ClientAliveInterval 300

# grep “^ClientAliveCountMax” /etc/ssh/sshd\_config ClientAliveCountMax 0

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

ClientAliveInterval 300

ClientAliveCountMax 0

### 6.2.13 Limit Access via SSH

There are several options available to limit which users and group can access the system via SSH. It is recommended that at least of the following options be leveraged:

AllowUsers

The AllowUsers variable gives the system administrator the option of allowing specific users to ssh into the system. The list consists of comma separated user names. Numeric userIDs are not recognized with this variable. If a system administrator wants to restrict user access further by only allowing the allowed users to log in from a particular host, the entry can be specified in the form of user@host.

AllowGroups

The AllowGroups variable gives the system administrator the option of allowing specific groups of users to ssh into the system. The list consists of comma separated user names. Numeric groupIDs are not recognized with this variable.

DenyUsers

The DenyUsers variable gives the system administrator the option of denying specific users to ssh into the system. The list consists of comma separated user names. Numeric userIDs are not recognized with this variable. If a system administrator wants to restrict user access further by specifically denying a user’s access from a particular host, the entry can be specified in the form of user@host.

DenyGroups

The DenyGroups variable gives the system administrator the option of denying specific groups of users to ssh into the system. The list consists of comma separated group names.

Numeric groupIDs are not recognized with this variable.

Rationale:

Restricting which users can remotely access the system via SSH will help ensure that only authorized users access the system.

Audit:

To verify the correct SSH setting, run the following command and verify that the output is as shown:

# grep “^AllowUsers” /etc/ssh/sshd\_config

AllowUsers <userlist>

# grep “^AllowGroups” /etc/ssh/sshd\_config

AllowGroups <grouplist>

# grep “^DenyUsers” /etc/ssh/sshd\_config

DenyUsers <userlist>

# grep “^DenyGroups” /etc/ssh/sshd\_config DenyGroups <grouplist>

Remediation:

Edit the /etc/ssh/sshd\_config file to set one or more of the parameter as follows:

AllowUsers <userlist>

AllowGroups <grouplist>

DenyUsers <userlist>

DenyGroups <grouplist>

### 6.2.14 Set SSH Banner

The Banner parameter specifies a file whose contents must be sent to the remote user before authentication is permitted. By default, no banner is displayed.

Rationale:

Banners are used to warn connecting users of the particular site’s policy regarding connection. Consult with your legal department for the appropriate warning banner for your site.

Audit:

To verify the correct SSH setting, run the following command and verify that <bannerfile> is either /etc/issue or /etc/issue.net:

# grep “^Banner” /etc/ssh/sshd\_config Banner <bannerfile>

Remediation:

Edit the /etc/ssh/sshd\_config file to set the parameter as follows:

Banner /etc/issue.net

## 6.3 Configure PAM

PAM (Pluggable Authentication Modules) is a service that implements modular authentication modules on UNIX systems. PAM is implemented as a set of shared objects that are loaded and executed when a program needs to authenticate a user. Files for PAM are typically located in the /etc/pam.d directory. PAM must be carefully configured to secure system authentication. While this section covers some of PAM, please consult other PAM resources to fully understand the configuration capabilities.

### 6.3.1 Upgrade Password Hashing Algorithm to SHA-512

The SHA-512 encryption has been available since Red Hat release 5.2,. The commands below change password encryption from md5 to sha512 (a much stronger hashing algorithm). All existing accounts will need to perform a password change to upgrade the stored hashes to the new algorithm.

Rationale:

The SHA-512 algorithm provides much stronger hashing than MD5, thus providing additional protection to the system by increasing the level of effort for an attacker to successfully determine passwords.

Note that these change only apply to accounts configured on the local system.

Audit:

Perform the following to determine if the password-hashing algorithm is set to SHA-512:

# authconfig –test | grep hashing | grep sha512

If the above command emits no output then the system is not configured as recommended

Remediation:

Perform the following to configure the system as recommended:

# authconfig –passalgo=sha512 –update

Note: If it is determined that the password algorithm being used is not SHA-512, once it is changed, it is recommended that all userID’s be immediately expired and forced to change their passwords on next login. To accomplish that, the following commands can be used. Any system accounts that need to be expired should be carefully done separately by the system administrator to prevent any potential problems.

# cat /etc/passwd | awk -F: ‘( $3 >=500 && $1 != “nfsnobody” ) { print $1 }’ | xargs n 1 chage -d 0

### 6.3.2 Set Password Creation Requirement Parameters Using pam\_cracklib

* Level 1

The pam\_cracklib module checks of the strength of passwords. It performs checks such as making sure a password is not a dictionary word, it is a certain length, contains a mix of characters (e.g. alphabet, numeric, other) and more. The following are definitions of the pam\_cracklib.so options.

* try\_first\_pass – retrieve the password from a previous stacked PAM module. If not available, then prompt the user for a password.
* retry=3 – Allow 3 tries before sending back a failure.
* minlen=14 – password must be 14 characters or more dcredit=-1 – provide at least 1 digit
* ucredit=-1 – provide at least one uppercase character
* ocredit=-1 – provide at least one special character
* lcredit=-1 – provide at least one lowercase character

The setting shown above is one possible policy. Alter these values to conform to your own organization’s password policies.

Rationale:

Strong passwords protect systems from being hacked through brute force methods.

Audit:

Perform the following to determine the current settings in the pam\_cracklib.so file.

# grep pam\_cracklib.so /etc/pam.d/system-auth

password required pam\_cracklib.so try\_first\_pass retry=3 minlen=14 dcredit=-1 ucredit=-1 ocredit=-1 lcredit=-1

Remediation:

Set the pam\_cracklib.so parameters as follows in /etc/pam.d/system-auth:

password required pam\_cracklib.so try\_first\_pass retry=3 minlen=14 dcredit=-1 ucredit=-1 ocredit=-1 lcredit=-1

### 6.3.3 Set Lockout for Failed Password Attempts

Lock out users after n unsuccessful consecutive login attempts. The first sets of changes are made to the PAM configuration file /etc/pam.d/system-auth. The second set of changes are applied to the program specific PAM configuration file. The second set of changes must be applied to each program that will lock out users. Check the documentation for each secondary program for instructions on how to configure them to work with PAM.

Set the lockout number to the policy in effect at your site.

Rationale:

Locking out userIDs after n unsuccessful consecutive login attempts mitigates brute force password attacks against your systems.

Audit:

Perform the following to determine the current settings for user lockout.

# grep “pam\_tally2” /etc/pam.d/system-auth auth required pam\_tally2.so onerr=fail audit silent deny=5 unlock\_time=900

Remediation:

Edit the /etc/pam.d/system-auth file and add the “auth” line as highlighted below:

auth required pam\_env.so

auth required pam\_tally2.so onerr=fail audit silent deny=5 unlock\_time=900 auth required pam\_deny.so

Note: If a user has been locked out because they have reached the maximum consecutive failure count defined by deny= in the pam\_tally2.so module, the user can be unlocked by issuing the command /sbin/pam\_tally2 -u <username> –reset. This command sets the failed count to 0, effectively unlocking the user.

### 6.3.4 Limit Password Reuse

The /etc/security/opasswd file stores the users’ old passwords and can be checked to ensure that users are not recycling recent passwords.

Rationale:

Forcing users not to reuse their past 5 passwords make it less likely that an attacker will be able to guess the password.

Note that these change only apply to accounts configured on the local system.

Audit:

Perform the following to determine the current setting for reuse of older passwords:

# grep “remember” /etc/pam.d/system\_auth password sufficient pam\_unix.so remember=5

Remediation:

Set the pam\_unix.so remember parameter to 5 in /etc/pam.d/system\_auth:

password sufficient pam\_unix.so remember=5

## 6.4 Restrict root Login to System Console

The file /etc/securetty contains a list of valid terminals that may be logged in directly as root.

Rationale:

Since the system console has special properties to handle emergency situations, it is important to ensure that the console is in a physically secure location and that unauthorized consoles have not been defined.

Audit:

# cat /etc/securetty

Remediation:

Remove entries for any consoles that are not in a physically secure location.

## 6.5 Restrict Access to the su Command

The su command allows a user to run a command or shell as another user. The program has been superseded by sudo, which allows for more granular control over privileged access. Normally, the su command can be executed by any user. By uncommenting the pam\_wheel.so statement in /etc/pam.d/su, the su command will only allow users in the wheel group to execute su.

Rationale:

Restricting the use of su, and using sudo in its place, provides system administrators better control of the escalation of user privileges to execute privileged commands. The sudo utility also provides a better logging and audit mechanism, as it can log each command executed via sudo, whereas su can only record that a user executed the su program.

Audit:

# grep pam\_wheel.so /etc/pam.d/su auth required pam\_wheel.so use\_uid

# grep wheel /etc/group wheel:x:10:root, <user list>

Remediation:

Set the pam\_wheel.so parameters as follows in /etc/pam.d/su:

auth required pam\_wheel.so use\_uid

Set the proper list of users to be included in the wheel group in /etc/groups.

# Configuring User Accounts and Environment

This section provides guidance on setting up secure defaults for system and user accounts and their environment. Guidance for monitoring these settings and others that may change over time is provided in Section 10 System Maintenance.

## 7.1 Set Shadow Password Suite Parameters (/etc/login.defs)

While a majority of the password control parameters have been moved to PAM, some parameters are still available through the shadow password suite. Any changes made to

/etc/login.defs will only be applied if the usermod command is used. If userIDs are added a different way, use the chage command to effect changes to individual userIDs.

### 7.1.1 Set Password Expiration Days

The PASS\_MAX\_DAYS parameter in /etc/login.defs allows an administrator to force passwords to expire once they reach a defined age. It is recommended that the PASS\_MAX\_DAYS parameter be set to less than or equal to 90 days.

Rationale:

The window of opportunity for an attacker to leverage compromised credentials or successfully compromise credentials via an online brute force attack is limited by the age of the password. Therefore, reducing the maximum age of a password also reduces an attacker’s window of opportunity.

Audit:

# grep PASS\_MAX\_DAYS /etc/login.defs

PASS\_MAX\_DAYS 90

# chage –list <user>

Maximum number of days between password change: 90

Remediation:

Set the PASS\_MAX\_DAYS parameter to 90 in /etc/login.defs:

PASS\_MAX\_DAYS 90

Modify active user parameters to match:

# chage –maxdays 90 <user>

### 7.1.2 Set Password Change Minimum Number of Days

The PASS\_MIN\_DAYS parameter in /etc/login.defs allows an administrator to prevent users from changing their password until a minimum number of days have passed since the last time the user changed their password. It is recommended that PASS\_MIN\_DAYS parameter be set to 7 or more days.

Rationale:

By restricting the frequency of password changes, an administrator can prevent users from repeatedly changing their password in an attempt to circumvent password reuse controls.

Audit:

# grep PASS\_MIN\_DAYS /etc/login.defs

PASS\_MAX\_DAYS 7

# chage –list <user>

Minimum number of days between password change: 7

Remediation:

Set the PASS\_MIN\_DAYS parameter to 7 in /etc/login.defs:

PASS\_MIN\_DAYS 7

Modify active user parameters to match:

# chage –mindays 7 <user>

### 7.1.3 Set Password Expiring Warning Days

The PASS\_WARN\_AGE parameter in /etc/login.defs allows an administrator to notify users that their password will expire in a defined number of days. It is recommended that the PASS\_WARN\_AGE parameter be set to 7 or more days.

Rationale:

Providing an advance warning that a password will be expiring gives users time to think of a secure password. Users caught unaware may choose a simple password or write it down where it may be discovered.

Audit:

# grep PASS\_WARN\_AGE /etc/login.defs

PASS\_WARN\_AGE 7

# chage –list <user>

Number of days of warning before password expires: 7

Remediation:

Set the PASS\_WARN\_AGE parameter to 7 in /etc/login.defs:

PASS\_WARN\_AGE 7

Modify active user parameters to match:

# chage –warndays 7 <user>

### 7.2 Disable System Accounts

There are a number of accounts provided with the Red Hat that are used to manage applications and are not intended to provide an interactive shell.

Rationale:

It is important to make sure that accounts that are not being used by regular users are locked to prevent them from being used to provide an interactive shell. By default, Red Hat sets the password field for these accounts to an invalid string, but it is also recommended that the shell field in the password file be set to /sbin/nologin. This prevents the account from potentially being used to run any commands.

Audit:

Run the following script to determine if any system accounts can be accessed:

egrep -v “^\+” /etc/passwd | awk -F: ‘($1!=”root” && $1!=”sync” && $1!=”shutdown” && $1!=”halt” && $3<500 && $7!=”/sbin/nologin”) {print}’

There should be no results returned.

Remediation:

Accounts that have been locked are prohibited from running commands on the system. Such accounts are not able to login to the system nor are they able to use scheduled execution facilities such as cron. To make sure system accounts cannot be accessed, using the following script:

#!/bin/bash

for user in `awk -F: ‘($3 < 500) {print $1 }’ /etc/passwd`; do if [ $user != “root” ] then

/usr/sbin/usermod -L $user

|  |
| --- |
| if [ $user != “sync” ] && [ $user != “shutdown” ] && [ $user != “halt” ] then  /usr/sbin/usermod -s /sbin/nologin $user fi fi done |

### 7.3 Set Default Group for root Account

The usermod command can be used to specify which group the root user belongs to. This affects permissions of files that are created by the root user.

Rationale:

Using GID 0 for the root account helps prevent root-owned files from accidentally becoming accessible to non-privileged users.

Audit:

# grep “^root:” /etc/passwd | cut -f4 -d: 0

Remediation:

# usermod -g 0 root

### 7.4 Set Default umask for Users

The default umask determines the permissions of files created by users. The user creating the file has the discretion of making their files and directories readable by others via the chmod command. Users who wish to allow their files and directories to be readable by others by default may choose a different default umask by inserting the umask command into the standard shell configuration files (.profile, .cshrc, etc.) in their home directories.

Rationale:

Setting a very secure default value for umask ensures that users make a conscious choice about their file permissions. A default umask setting of 077 causes files and directories created by users to not be readable by any other user on the system. A umask of 027 would make files and directories readable by users in the same Unix group, while a umask of 022 would make files readable by every user on the system.

Note: The directives in this section apply to bash and shell. If other shells are supported on the system, it is recommended that their configuration files also are checked.

Audit:

# grep “^umask 077” /etc/bashrc umask 077

# grep “^umask 077” /etc/profile.d/\* umask 077

Remediation:

Edit the /etc/bashrc and /etc/profile.d/cis.sh files (and the appropriate files for any other shell supported on your system) and add the following the UMASK parameter as shown:

umask 77

### 7.5 Lock Inactive User Accounts

User accounts that have been inactive for over a given period of time can be automatically disabled. It is recommended that accounts that are inactive for 35 or more days be disabled.

Rationale:

Inactive accounts pose a threat to system security since the users are not logging in to notice failed login attempts or other anomalies.

Audit:

# useradd -D | grep INACTIVE

Remediation:

# useradd -D -f 35

## 8.1 Set Warning Banner for Standard Login Services

The contents of the /etc/issue file are displayed prior to the login prompt on the system’s console and serial devices, and also prior to logins via telnet. The contents of the

/etc/motd file is generally displayed after all successful logins, no matter where the user is logging in from, but is thought to be less useful because it only provides notification to the user after the machine has been accessed.

Rationale:

Warning messages inform users who are attempting to login to the system of their legal status regarding the system and must include the name of the organization that owns the system and any monitoring policies that are in place. Consult with your organization’s legal counsel for the appropriate wording for your specific organization.

Audit:

Run the following commands and ensure that the files exist and have the correct permissions.

# /bin/ls -l /etc/motd

-rw-r–r– 1 root root 2055 Jan 30 16:30 /etc/motd

# ls /etc/issue

-rw-r–r– 1 root root 2055 Jan 30 16:30 /etc/issue

# ls /etc/issue.net

-rw-r–r– 1 root root 2055 Jan 30 16:30 /etc/issue.net

The commands above simply validate the presence of the /etc/motd, /etc/issue and /etc/issue.net files. Review the contents of these files with the “cat” command and ensure that it is appropriate for your organization.

Remediation:

# touch /etc/motd

# echo “Authorized uses only. All activity may be \ monitored and reported.” > /etc/issue

# echo “Authorized uses only. All activity may be \ monitored and reported.” > /etc/issue.net

# chown root:root /etc/motd

# chmod 644 /etc/motd

# chown root:root /etc/issue

# chmod 644 /etc/issue

# chown root:root /etc/issue.net

# chmod 644 /etc/issue.net

## 8.2 Remove OS Information from Login Warning Banners

Unix-based systems have typically displayed information about the OS release and patch level upon logging in to the system. This information can be useful to developers who are developing software for a particular OS platform. If mingetty(8) supports the following options, they display operating system information:

\m – machine architecture (uname -m)

\r – operating system release (uname -r)

\s – operating system name

\v – operating system version (uname -v)

Rationale:

Displaying OS and patch level information in login banners also has the side effect of providing detailed system information to attackers attempting to target specific exploits of a system. Authorized users can easily get this information by running the “uname -a” command once they have logged in.

Audit:

Perform the following commands to check if OS information is set to be displayed in the system login banners:

# egrep ‘(\\v|\\r|\\m|\\s)’ /etc/issue

# egrep ‘(\\v|\\r|\\m|\\s)’ /etc/motd # egrep'(\\v|\\r|\\m|\\s)’ /etc/issue.net

Remediation:

Edit the /etc/motd, /etc/issue and /etc/issue.net files and remove any lines containing \m, \r, \s or \v.

### 8.3 Set GNOME Warning Banner

The GNOME Display Manager is used for login session management. See the manual page gdm(1) for more information. The remediation action for this item sets a warning message for GDM users before they log in.

Rationale:

Warning messages inform users who are attempting to login to the system of their legal status regarding the system and must include the name of the organization that owns the system and any monitoring policies that are in place. Consult with your organization’s legal counsel for the appropriate wording for your specific organization.

Audit:

# gconftool-2 -get /apps/gdm/simple-greeter/banner\_message\_text

Remediation:

# /bin/su -s /bin/sh gdm

# gconftool-2 -direct -config-source=xml:readwrite:$HOME/.gconf -type bool -set /apps/gdm/simple-greeter/banner\_message\_enable true

# gconftool-2 -direct -config-source=xml:readwrite:$HOME/.gconf -t string -s /apps/gdm/simple-greeter/banner\_message\_text “Your-Login-Banner”

Restart gdm for these settings to take effect.

# Part-9 : System Maintenance Activities

No matter how securely a system has been installed and hardened, administrator and user activity over time can introduce security exposures. The section describes tasks to be performed on a regular, ongoing basis – perhaps in an automated fashion via the cron utility.

Note: unlike other sections, the items in this section specify an Audit action followed by a Remediation action since it is necessary to determine what the current setting is before determining remediation measures, which will vary depending on the site’s policy.

## 9.1 Verify System File Permissions

### 9.1.1 Verify System File Permissions

Level 2

The RPM package manager has a number of useful options. One of these, the -V (or verify) option, can be used to verify that system packages are correctly installed. The -V option can be used to verify a particular package or to verify all system packages (-Va). If no output is returned, the package is installed correctly. The following table describes the meaning of output from the verify option:

Code Meaning

S File size differs.

M File mode differs (includes permissions and file type).

5 The MD5 checksum differs.

D The major and minor version numbers differ on a device file.

L A mismatch occurs in a link.

U The file ownership differs.

G The file group owner differs.

T The file time (mtime) differs.

The rpm -qf command can be used to determine which package a particular file belongs to. For example the following command determines which package the /etc/passwd file belongs to:

# rpm -qf /etc/passwd setup-2.5.58-7.el5

To verify the settings for the package that controls the /etc/passwd file, run the following:

# rpm -V setup-2.5.58-7.el5

.M…… c /etc/passwd

S.5….T c /etc/printcap

Note that you can feed the output of the rpm -qf command to the rpm -V command:

# rpm -V `rpm -qf /etc/passwd`

.M…… c /etc/passwd S.5….T c /etc/printcap

Rationale:

It is important to confirm that packaged system files and directories are maintained with the permissions they were intended to have from the OS vendor.

Note: Since packages and important files may change with new updates and releases, it is recommended to verify everything, not just a finite list of files. This can be a time consuming task and is therefore not a scored benchmark item, but is provided for those interested in additional security measures.

Audit:

Run the following command to review all installed packages. Note that this may be very time consuming and may be best scheduled via the cron utility. It is recommended that the output of this command be redirected to a file that can be reviewed later.

# rpm -Va –nomtime –nosize –nomd5 –nolinkto > <filename>

Remediation:

Correct any discrepancies found and rerun the command until output is clean or risk is mitigated or accepted.

### 9.1.2 Verify Permissions on /etc/passwd

The /etc/passwd file contains user account information that is used by many system utilities and therefore must be readable for these utilities to operate.

Rationale:

It is critical to ensure that the /etc/passwd file is protected from unauthorized write access. Although it is protected by default, the file permissions could be changed either inadvertently or through malicious actions.

Audit:

Run the following command to determine the permissions on the /etc/passwd file.

# /bin/ls -l /etc/passwd

-rw-r–r– 1 root root 2055 Jan 30 16:30 /etc/passwd

Remediation:

If the permissions of the /etc/passwd file are incorrect, run the following command to correct them:

# /bin/chmod 644 /etc/passwd

### 9.1.3 Verify Permissions on /etc/shadow

The /etc/shadow file is used to store the information about user accounts that is critical to the security of those accounts, such as the hashed password and other security information.

Rationale:

If attackers can gain read access to the /etc/shadow file, they can easily run a password cracking program against the hashed password to break it. Other security information that is stored in the /etc/shadow file (such as expiration) could also be useful to subvert the user accounts.

Audit:

Run the following command to determine the permissions on the /etc/shadow file.

# /bin/ls -l /etc/shadow

———- 1 root root 633 Sep 23 2002 /etc/shadow

Remediation:

If the permissions of the /etc/shadow file are incorrect, run the following command to correct them:

# /bin/chmod 000 /etc/shadow

### 9.1.4 Verify Permissions on /etc/gshadow

The /etc/gshadow file contains information about group accounts that is critical to the security of those accounts, such as the hashed password and other security information.

Rationale:

If attackers can gain read access to the /etc/gshadow file, they can easily run a password cracking program against the hashed password to break it. Other security information that is stored in the /etc/gshadow file (such as expiration) could also be useful to subvert the group accounts.

Audit:

Run the following command to determine the permissions on the /etc/gshadow file.

# /bin/ls -l /etc/gshadow

———- 1 root root 633 Sep 23 2002 /etc/gshadow

Remediation:

If the permissions of the /etc/gshadow file are incorrect, run the following command to correct them:

# /bin/chmod 000 /etc/gshadow

### 9.1.5 Verify Permissions on /etc/group

The /etc/group file contains a list of all the valid groups defined in the system. The command below allows read/write access for root and read access for everyone else.

Rationale:

The /etc/group file needs to be protected from unauthorized changes by non-privileged users, but needs to be readable as this information is used with many non-privileged programs.

Audit:

Run the following command to determine the permissions on the /etc/group file.

# /bin/ls -l /etc/group

-rw-r–r– 1 root root 762 Sep 23 002 /etc/group

Remediation:

If the permissions of the /etc/group file are incorrect, run the following command to correct them:

# /bin/chmod 644 /etc/group

### 9.1.6 Verify User/Group Ownership on /etc/passwd

The /etc/passwd file contains a list of all the valid userIDs defined in the system, but not the passwords. The command below sets the owner and group of the file to root.

Rationale:

The /etc/passwd file needs to be protected from unauthorized changes by non-privileged users, but needs to be readable as this information is used with many non-privileged programs.

Audit:

Run the following command to determine the user and group ownership on the /etc/passwd file.

# /bin/ls -l /etc/passwd

-rw-r–r– 1 root root 762 Sep 23 002 /etc/passwd

Remediation:

If the user and group ownership of the /etc/passwd file are incorrect, run the following command to correct them:

# /bin/chown root:root /etc/passwd

### 9.1.7 Verify User/Group Ownership on /etc/shadow

The /etc/shadow file contains the one-way cipher text passwords for each user defined in the /etc/passwd file. The command below sets the user and group ownership of the file to root.

Rationale:

If attackers can gain read access to the /etc/shadow file, they can easily run a password cracking program against the hashed password to break it. Other security information that is stored in the /etc/shadow file (such as expiration) could also be useful to subvert the user accounts.

Audit:

Run the following command to determine the permissions on the /etc/shadow file.

# /bin/ls -l /etc/shadow

———- 1 root root 762 Sep 23 2002 /etc/shadow

Remediation:

If the ownership of the /etc/shadow file are incorrect, run the following command to correct them:

# /bin/chown root:root /etc/shadow

### 9.1.8 Verify User/Group Ownership on /etc/gshadow

The /etc/gshadow file contains information about group accounts that is critical to the security of those accounts, such as the hashed password and other security information.

Rationale:

If attackers can gain read access to the /etc/gshadow file, they can easily run a password cracking program against the hashed password to break it. Other security information that is stored in the /etc/gshadow file (such as expiration) could also be useful to subvert the group accounts.

Audit:

Run the following command to determine the permissions on the /etc/gshadow file.

# /bin/ls -l /etc/gshadow

———- 1 root root 633 Sep 23 2002 /etc/gshadow

Remediation:

If the ownership of the /etc/gshadow file are incorrect, run the following command to correct them:

# /bin/chown root:root /etc/gshadow

### 9.1.9 Verify User/Group Ownership on /etc/group

The /etc/group file contains a list of all the valid groups defined in the system. The command below allows read/write access for root and read access for everyone else.

Rationale:

The /etc/group file needs to be protected from unauthorized changes by nonprivileged users, but needs to be readable as this information is used with many nonprivileged programs.

Audit:

Run the following command to determine the permissions on the /etc/group file.

# /bin/ls -l /etc/group

-rw-r–r– 1 root root 762 Sep 23 002 /etc/group

Remediation:

If the ownership of the /etc/group file are incorrect, run the following command to correct them:

# /bin/chown root:root /etc/group

### 9.1.10 Find World Writable Files

Unix-based systems support variable settings to control access to files. World writable files are the least secure. See the chmod(2) man page for more information.

Rationale:

Data in world-writable files can be modified and compromised by any user on the system. World writable files may also indicate an incorrectly written script or program that could potentially be the cause of a larger compromise to the system’s integrity.

Audit:

#!/bin/bash

df –local -P | awk {‘if (NR!=1) print $6’} | xargs -I ‘{}’ find ‘{}’ -xdev -type f perm -0002

Remediation:

Removing write access for the “other” category (chmod o-w <filename>) is advisable, but always consult relevant vendor documentation to avoid breaking any application dependencies on a given file.

### 9.1.11 Find Un-owned Files and Directories

Sometimes when administrators delete users from the password file they neglect to remove all files owned by those users from the system.

Rationale:

A new user who is assigned the deleted user’s user ID or group ID may then end up “owning” these files, and thus have more access on the system than was intended.

Audit:

#!/bin/bash

df –local -P | awk {‘if (NR!=1) print $6’} | xargs -I ‘{}’ find ‘{}’ -xdev -nouser ls

Remediation:

Locate files that are owned by users or groups not listed in the system configuration files, and reset the ownership of these files to some active user on the system as appropriate.

### 9.1.12 Find Un-grouped Files and Directories

Sometimes when administrators delete users from the password file they neglect to remove all files owned by those users from the system.

Rationale:

A new user who is assigned the deleted user’s user ID or group ID may then end up “owning” these files, and thus have more access on the system than was intended.

Audit:

#!/bin/bash

df –local -P | awk {‘if (NR!=1) print $6’} | xargs -I ‘{}’ find ‘{}’ -xdev -nogroup ls

Remediation:

Locate files that are owned by users or groups not listed in the system configuration files, and reset the ownership of these files to some active user on the system as appropriate.

### 9.1.13 Find SUID System Executables

The owner of a file can set the file’s permissions to run with the owner’s or group’s permissions, even if the user running the program is not the owner or a member of the group. The most common reason for a SUID program is to enable users to perform functions (such as changing their password) that require root privileges.

Rationale:

There are valid reasons for SUID programs, but it is important to identify and review such programs to ensure they are legitimate.

Audit:

#!/bin/bash

df –local -P | awk {‘if (NR!=1) print $6’} | xargs -I ‘{}’ find ‘{}’ -xdev -type f perm -4000 -print

Remediation:

Ensure that no rogue set-UID programs have been introduced into the system. Review the files returned by the action in the Audit section and confirm the integrity of these binaries as described below:

# rpm -V `rpm -qf /usr/bin/sudo`

…….T /usr/bin/sudo

SM5….T /usr/bin/sudoedit

### 9.1.14 Find SGID System Executables

The owner of a file can set the file’s permissions to run with the owner’s or group’s permissions, even if the user running the program is not the owner or a member of the group. The most common reason for a SGID program is to enable users to perform functions (such as changing their password) that require root privileges.

Rationale:

There are valid reasons for SGID programs, but it is important to identify and review such programs to ensure they are legitimate. Review the files returned by the action in the audit section and check to see if system binaries have a different md5 checksum than what from the package. This is an indication that the binary may have been replaced. The following is an example of checking the “sudo” executable:

# rpm -V `rpm -qf /usr/bin/sudo`

…….T /usr/bin/sudo SM5….T /usr/bin/sudoedit

Audit:

#!/bin/bash

df –local -P | awk {‘if (NR!=1) print $6’} | xargs -I ‘{}’ find ‘{}’ -xdev -type f perm -2000 -print

Remediation:

Ensure that no rogue set-GID programs have been introduced into the system

# /bin/rpm -V `/bin/rpm -qf sudo`

## 9.2 Review User and Group Settings

This section provides guidance on securing aspects of the users and groups.

### 9.2.1 Ensure Password Fields are Not Empty

An account with an empty password field means that anybody may log in as that user without providing a password.

Rationale:

All accounts must have passwords or be locked to prevent the account from being used by an unauthorized user.

Audit:

Run the following command and verify that no output is returned:

# /bin/cat /etc/shadow | /bin/awk -F: ‘($2 == “” ) { print $1 ” does not have a password “}’

Remediation:

If any accounts in the /etc/shadow file do not have a password, run the following command to lock the account until it can be determined why it does not have a password:

# /usr/bin/passwd -l <username>

Also, check to see if the account is logged in and investigate what it is being used for to determine if it needs to be forced off.

### 9.2.2 Verify No Legacy “+” Entries Exist in /etc/passwd File

The character + in various files used to be markers for systems to insert data from NIS maps at a certain point in a system configuration file. These entries are no longer required on RHEL6 systems, but may exist in files that have been imported from other platforms.

Rationale:

These entries may provide an avenue for attackers to gain privileged access on the system.

Audit:

Run the following command and verify that no output is returned:

# /bin/grep ‘^+:’ /etc/passwd

Remediation:

Delete these entries if they exist.

### 9.2.3 Verify No Legacy “+” Entries Exist in /etc/shadow File

The character + in various files used to be markers for systems to insert data from NIS maps at a certain point in a system configuration file. These entries are no longer required on RHEL6 systems, but may exist in files that have been imported from other platforms.

Rationale:

These entries may provide an avenue for attackers to gain privileged access on the system.

Audit:

Run the following command and verify that no output is returned:

# /bin/grep ‘^+:’ /etc/shadow

Remediation:

Delete these entries if they exist.

### 9.2.4 Verify No Legacy “+” Entries Exist in /etc/group File

The character + in various files used to be markers for systems to insert data from NIS maps at a certain point in a system configuration file. These entries are no longer required on RHEL6 systems, but may exist in files that have been imported from other platforms.

Rationale:

These entries may provide an avenue for attackers to gain privileged access on the system.

Audit:

Run the following command and verify that no output is returned:

# /bin/grep ‘^+:’ /etc/group

Remediation:

Delete these entries if they exist.

### 9.2.5 Verify No UID 0 Accounts Exist Other Than root

Any account with UID 0 has superuser privileges on the system.

Rationale:

This access must be limited to only the default root account and only from the system console. Administrative access must be through an unprivileged account using an approved mechanism as noted in Item 7.5 Restrict root Login to System Console.

Audit:

Run the following command and verify that only the word “root” is returned:

# /bin/cat /etc/passwd | /bin/awk -F: ‘($3 == 0) { print $1 }’ root

Remediation:

Delete any other entries that are displayed. 9.2.6 Ensure root PATH Integrity

The root user can execute any command on the system and could be fooled into executing programs unemotionally if the PATH is not set correctly.

Rationale:

Including the current working directory (.) or other writable directory in root’s executable path makes it likely that an attacker can gain superuser access by forcing an administrator operating as root to execute a Trojan horse program.

Audit:

|  |
| --- |
| #!/bin/bash  if [ “`echo $PATH | /bin/grep :: `” != “” ]; then echo “Empty Directory in PATH (::)” fi  if [ “`echo $PATH | bin/grep :$`” != “” ]; then echo “Trailing : in PATH” fi  p=`echo $PATH | /bin/sed -e ‘s/::/:/’ -e ‘s/:$//’ -e ‘s/:/ /g’` set — $p  while [ “$1” != “” ]; do if [ “$1” = “.” ]; then echo “PATH contains .” shift continue fi  if [ -d $1 ]; then  dirperm=`/bin/ls -ldH $1 | /bin/cut -f1 -d” “` if [ `echo $dirperm | /bin/cut -c6 ` != “-” ]; then echo “Group Write permission set on directory $1” fi  if [ `echo $dirperm | /bin/cut -c9 ` != “-” ]; then echo “Other Write permission set on directory $1” fi  dirown=`ls -ldH $1 | awk ‘{print $3}’` if [ “$dirown” != “root” ] ; then echo $1 is not owned by root fi else  echo $1 is not a directory fi shift done |

Remediation:

Correct or justify any items discovered in the Audit step.

### 9.2.7 Check Permissions on User Home Directories

While the system administrator can establish secure permissions for users’ home directories, the users can easily override these.

Rationale:

Group or world-writable user home directories may enable malicious users to steal or modify other users’ data or to gain another user’s system privileges.

Audit:

|  |
| --- |
| #!/bin/bash  for dir in `/bin/cat /etc/passwd | /bin/egrep -v ‘(root|halt|sync|shutdown)’ |\ /bin/awk -F: ‘($8 == “PS” && $7 != “/sbin/nologin”) { print $6 }’`; do dirperm=`/bin/ls -ld $dir | /bin/cut -f1 -d” “` if [ `echo $dirperm | /bin/cut -c6 ` != “-” ]; then echo “Group Write permission set on directory $dir” fi  if [ `echo $dirperm | /bin/cut -c8 ` != “-” ]; then echo “Other Read permission set on directory $dir” fi  if [ `echo $dirperm | /bin/cut -c9 ` != “-” ]; then echo “Other Write permission set on directory $dir” fi  if [ `echo $dirperm | /bin/cut -c10 ` != “-” ]; then echo “Other Execute permission set on directory $dir” fi done |

Remediation:

Making global modifications to user home directories without alerting the user community can result in unexpected outages and unhappy users. Therefore, it is recommended that a monitoring policy be established to report user file permissions and determine the action to be taken in accordance with site policy.

### 9.2.8 Check User Dot File Permissions

While the system administrator can establish secure permissions for users’ “dot” files, the users can easily override these.

Rationale:

Group or world-writable user configuration files may enable malicious users to steal or modify other users’ data or to gain another user’s system privileges.

Audit:

|  |
| --- |
| #!/bin/bash  for dir in `/bin/cat /etc/passwd | /bin/egrep -v ‘(root|sync|halt|shutdown)’ |  /bin/awk -F: ‘($7 != “/sbin/nologin”) { print $6 }’`; do for file in $dir/.[A-Za-z0-9]\*; do if [ ! -h “$file” -a -f “$file” ]; then  fileperm=`/bin/ls -ld $file | /bin/cut -f1 -d” “` if [ `echo $fileperm | /bin/cut -c6 ` != “-” ]; then echo “Group Write permission set on file $file” fi  if [ `echo $fileperm | /bin/cut -c9 ` != “-” ]; then echo “Other Write permission set on file $file” fi fi done done |

Remediation:

Making global modifications to users’ files without alerting the user community can result in unexpected outages and unhappy users. Therefore, it is recommended that a monitoring policy be established to report user dot file permissions and determine the action to be taken in accordance with site policy.

### 9.2.9 Check Permissions on User .netrc Files

While the system administrator can establish secure permissions for users’ .netrc files, the users can easily override these.

Rationale:

.netrc files may contain unencrypted passwords that may be used to attack other systems.

Audit:

|  |
| --- |
| #!/bin/bash  for dir in `/bin/cat /etc/passwd | /bin/egrep -v ‘(root|sync|halt|shutdown)’ |\  /bin/awk -F: ‘($7 != “/sbin/nologin”) { print $6 }’`; do for file in $dir/.netrc; do  if [ ! -h “$file” -a -f “$file” ]; then  fileperm=`/bin/ls -ld $file | /bin/cut -f1 -d” “` if [ `echo $fileperm | /bin/cut -c5 ` != “-” ] then  echo “Group Read set on $file” |
| fi  if [ `echo $fileperm | /bin/cut -c6 ` != “-” ] then  echo “Group Write set on $file” fi  if [ `echo $fileperm | /bin/cut -c7 ` != “-” ] then  echo “Group Execute set on $file” fi  if [ `echo $fileperm | /bin/cut -c8 ` != “-” ] then  echo “Other Read set on $file” fi  if [ `echo $fileperm | /bin/cut -c9 ` != “-” ] then  echo “Other Write set on $file” fi  if [ `echo $fileperm | /bin/cut -c10 ` != “-” ] then  echo “Other Execute set on $file” fi fi done done |

Remediation:

Making global modifications to users’ files without alerting the user community can result in unexpected outages and unhappy users. Therefore, it is recommended that a monitoring policy be established to report user .netrc file permissions and determine the action to be taken in accordance with site policy.

### 9.2.10 Check for Presence of User .rhosts Files

While no .rhosts files are shipped with RHEL6, users can easily create them.

Rationale:

This action is only meaningful if .rhosts support is permitted in the file /etc/pam.conf. Even though the .rhosts files are ineffective if support is disabled in /etc/pam.conf, they may have been brought over from other systems and could contain information useful to an attacker for those other systems.

Audit:

|  |
| --- |
| #!/bin/bash  for dir in `/bin/cat /etc/passwd | /bin/egrep -v ‘(root|halt|sync|shutdown)’ |\  /bin/awk -F: ‘($7 != “/sbin/nologin”) { print $6 }’`; do for file in $dir/.rhosts; do  if [ ! -h “$file” -a -f “$file” ]; then echo “.rhosts file in $dir” fi done done |

Remediation:

If any users have .rhosts files determine why they have them.

### 9.2.11 Check Groups in /etc/passwd

Over time, system administration errors and changes can lead to groups being defined in /etc/passwd but not in /etc/group.

Rationale:

Groups defined in the /etc/passwd file but not in the /etc/group file pose a threat to system security since group permissions are not properly managed.

Audit:

Create a script as shown below and run it:

|  |
| --- |
| #!/bin/bash  for i in $(cut -s -d: -f4 /etc/passwd | sort -u ); do grep -q -P “^.\*?:x:$i:” /etc/group if [ $? -ne 0 ]; then  echo “Group $i is referenced by /etc/passwd but does not exist in /etc/group” fi done |

Remediation:

Analyze the output of the Audit step above and perform the appropriate action to correct any discrepancies found.

### 9.2.12 Check That Users Are Assigned Valid Home Directories

Users can be defined in /etc/passwd without a home directory or with a home directory does not actually exist.

Rationale:

If the user’s home directory does not exist or is unassigned, the user will be placed in “/” and will not be able to write any files or have local environment variables set.

Audit:

This script checks to make sure that home directories assigned in the /etc/passwd file exist.

#!/bin/bash

cat /etc/passwd | awk -F: ‘{ print $1 ” ” $3 ” ” $6 }’ | while read user uid dir; do if [ $uid -ge 500 -a ! -d “$dir” -a $user != “nfsnobody” ]; then echo “The home directory ($dir) of user $user does not exist.” fi done

Remediation:

If any users’ home directories do not exist, create them and make sure the respective user owns the directory. Users without assigned should be removed or assigned a home directory as appropriate.

### 9.2.13 Check User Home Directory Ownership

The user home directory is space defined for the particular user to set local environment variables and to store personal files.

Rationale:

Since the user is accountable for files stored in the user home directory, the user must be the owner of the directory.

Audit:

This script checks to make sure users own the home directory they are assigned to in the /etc/passwd file.

|  |
| --- |
| #!/bin/bash  cat /etc/passwd | awk -F: ‘{ print $1 ” ” $3 ” ” $6 }’ | while read user uid dir; do if [ $uid -ge 500 -a -d “$dir” -a $user != “nfsnobody” ]; then owner=$(stat -L -c “%U” “$dir”) if [ “$owner” != “$user” ]; then  echo “The home directory ($dir) of user $user is owned by $owner.” fi fi done |

Remediation:

Change the ownership any home directories that are not owned by the defined user to the correct user.

### 9.2.14 Check for Duplicate UIDs

Although the useradd program will not let you create a duplicate User ID (UID), it is possible for an administrator to manually edit the /etc/passwd file and change the UID field.

Rationale:

Users must be assigned unique UIDs for accountability and to ensure appropriate access protections.

Audit:

This script checks to make sure all UIDs in the /etc/passwd file are unique.

|  |
| --- |
| #!/bin/bash  echo “The Output for the Audit of Control 9.2.15 – Check for Duplicate UIDs is” /bin/cat /etc/passwd | /bin/cut -f3 -d”:” | /bin/sort -n | /usr/bin/uniq -c |\ while read x ; do [ -z “${x}” ] && break set – $x  if [ $1 -gt 1 ]; then  users=`/bin/gawk -F: ‘($3 == n) { print $1 }’ n=$2 \  /etc/passwd | /usr/bin/xargs` echo “Duplicate UID ($2): ${users}” fi done |

Remediation:

Based on the results of the script, establish unique UIDs and review all files owned by the shared UID to determine which UID they are supposed to belong to.

### 9.2.15 Check for Duplicate GIDs

Although the groupadd program will not let you create a duplicate Group ID (GID), it is possible for an administrator to manually edit the /etc/group file and change the GID field.

Rationale:

User groups must be assigned unique GIDs for accountability and to ensure appropriate access protections.

Note: In the case of extremely large groups it can become necessary to split a GID across group names due to character limits per line. Any such instances should be carefully audited, unless absolutely necessary such instances should be avoided.

Audit:

This script checks to make sure all GIDs in the /etc/group file are unique. You can also use the /usr/sbin/grpck command to check for other inconsistencies in the /etc/group file.

|  |
| --- |
| #!/bin/bash  echo “The Output for the Audit of Control 9.2.16 – Check for Duplicate GIDs is” /bin/cat /etc/group | /bin/cut -f3 -d”:” | /bin/sort -n | /usr/bin/uniq -c |\ while read x ; do [ -z “${x}” ] && break set – $x  if [ $1 -gt 1 ]; then  grps=`/bin/gawk -F: ‘($3 == n) { print $1 }’ n=$2 \  /etc/group | xargs`  echo “Duplicate GID ($2): ${grps}” fi done |

Remediation:

Based on the results of the script, establish unique GIDs and review all files owned by the shared GID to determine which group they are supposed to belong to.

### 9.2.16 Check for Duplicate User Names

Although the useradd program will not let you create a duplicate user name, it is possible for an administrator to manually edit the /etc/passwd file and change the user name.

Rationale:

If a user is assigned a duplicate user name, it will create and have access to files with the first UID for that username in /etc/passwd. For example, if “test4” has a UID of 1000 and a subsequent “test4” entry has a UID of 2000, logging in as “test4” will use UID 1000.

Effectively, the UID is shared, which is a security problem.

Audit:

This script checks to make sure all user names in the /etc/passwd file are unique.

|  |
| --- |
| #!/bin/bash  echo “The Output for the Audit of Control 9.2.18 – Check for Duplicate User Names is” cat /etc/passwd | cut -f1 -d”:” | /bin/sort -n | /usr/bin/uniq -c |\ while read x ; do [ -z “${x}” ] && break set – $x  if [ $1 -gt 1 ]; then  uids=`/bin/gawk -F: ‘($1 == n) { print $3 }’ n=$2 \  /etc/passwd | xargs`  echo “Duplicate User Name ($2): ${uids}” fi done |

Remediation:

Based on the results of the script, establish unique user names for the users. File ownerships will automatically reflect the change as long as the users have unique UIDs.

### 9.2.17 Check for Duplicate Group Names

Although the groupadd program will not let you create a duplicate group name, it is possible for an administrator to manually edit the /etc/group file and change the group name.

Rationale:

If a group is assigned a duplicate group name, it will create and have access to files with the first GID for that group in /etc/group. Effectively, the GID is shared, which is a security problem.

Audit:

This script checks to make sure all group names in the /etc/group file are unique.

|  |
| --- |
| #!/bin/bash  echo “The Output for the Audit of Control 9.2.19 – Check for Duplicate Group Names is” cat /etc/group | cut -f1 -d”:” | /bin/sort -n | /usr/bin/uniq -c |\ while read x ; do [ -z “${x}” ] && break set – $x  if [ $1 -gt 1 ]; then  gids=`/bin/gawk -F: ‘($1 == n) { print $3 }’ n=$2 \  /etc/group | xargs`  echo “Duplicate Group Name ($2): ${gids}” fi done |

Remediation:

Based on the results of the script, establish unique names for the user groups. File group ownerships will automatically reflect the change as long as the groups have unique GIDs.

### 9.2.18 Check for Presence of User .netrc Files

The .netrc file contains data for logging into a remote host for file transfers via FTP.

Rationale:

The .netrc file presents a significant security risk since it stores passwords in unencrypted form. Even if FTP is disabled, user accounts may have brought over .netrc files from other systems which could pose a risk to those systems.

Audit:

|  |
| --- |
| #!/bin/bash  for dir in `/bin/cat /etc/passwd |\ /bin/awk -F: ‘{ print $6 }’`; do  if [ ! -h “$dir/.netrc” -a -f “$dir/.netrc” ]; then echo “.netrc file $dir/.netrc exists” fi done |

Remediation:

Making global modifications to users’ files without alerting the user community can result in unexpected outages and unhappy users. Therefore, it is recommended that a monitoring policy be established to report user .netrc files and determine the action to be taken in accordance with site policy.

### 9.2.19 Check for Presence of User .forward Files

The .forward file specifies an email address to forward the user’s mail to.

Rationale:

Use of the .forward file poses a security risk in that sensitive data may be inadvertently transferred outside the organization. The .forward file also poses a risk as it can be used to execute commands that may perform unintended actions.

Audit:

This script checks for the presence of .forward files that may be in violation of the site security policy.

#!/bin/bash

for dir in `/bin/cat /etc/passwd |\ /bin/awk -F: ‘{ print $6 }’`; do

if [ ! -h “$dir/.forward” -a -f “$dir/.forward” ]; then echo “.forward file $dir/.forward exists” fi done

Remediation:

Making global modifications to users’ files without alerting the user community can result in unexpected outages and unhappy users. Therefore, it is recommended that a monitoring policy be established to report user .forward files and determine the action to be taken in accordance with site policy.

### ****1. Red Hat Enterprise Linux 5 supports LUKS encryption. True or False?****

1A. False. Red Hat Enterprise Linux 6 is the only version that officially supports LUKS encryption.

### ****2. In Red Hat Enterprise Linux 6, SELinux is set to Enforcing by default during the installation. True or False?****

2A. True. In Red Hat Enterprise Linux 6, the default for SELinux is Enforcing duringinstallation (which can be changed after the installation completes).  For Red Hat Enterprise Linux 5, you were able to choose what mode you wanted SELinux to start in.

### ****3. Which remote management service is installed by default? Can you name the port that it uses?****

3A. The SSH service is almost always installed by default in Red Hat. It uses TCP port 22, and this port is open on the default firewall rules.

### ****4. You can install software packages only after Red Hat Enterprise Linux is installed. True or False?****

4A. False. The package selection screen allows you to install any software you want during the installation process as long as you have access to the correct packages or repositories.

### ****5. The default partition layout includes only basic partitions. True or False?****

5A. False. The default partition layout for Red Hat Enterprise Linux 5 or 6 includes the use of LVM.

### ****6. Which file contains all messages generated during installation that can be used for troubleshooting if the need arises?****

6A. The install.log.syslog file contains messages that are generated during the installation. If you run into trouble during the installation, this is a good place to start.

### 1. GRUB has three stages. Can you name them? Ans:

* Stage 1: During this stage, the primary bootloader is read into memory by the BIOS from the MBR.
* Stage 1.5: During this stage, the bootloader is read into memory by the stage 1 bootloader (only if necessary).
* Stage 2: During this stage, the bootloader reads the operating system or kernel.

### 2. What option at the GRUB boot menu can you use to append something to a kernel?

2A. By entering the GRUB boot menu, you can choose the a option to append something to the kernel command-line options.

### 3. The old SysInit scripts have been replaced in Red Hat Enterprise Linux 6 for what new boot utility?

3A. Upstart. The Upstart utility is now used in the boot process for Red Hat Enterprise Linux 6.

### 4. Runlevel 0 reboots the system. True or False?

4A. False. Runlevel 0 halts the system. Runlevel 6 reboots the system.

### 5. If your system crashes and becomes unbootable, you have to reinstall the whole operating system. True or False?

5A. False. Most boot issues can be resolved by entering rescue mode and repairing the problem.

### 6. What command can you use to manage system services?

6A. The service command is used to start, stop, and manage system services.

### 7. What command disables the SSH service from running when the system boots?

7A. chkconfig sshd off

### 8. How can you list all services on the system to tell whether they will boot during startup?

8A. chkconfig —list

### 9. What does S12rsyslog in the /etc/rc.d/rc2.d directory mean?

9A. When the system enters into runlevel 2, the rsyslog service has a priority of 12 when starting. Anything with a lower number (or the same number and lower first letter) starts before the rsyslog service.

### 10. How can you verify the status of the SSH service after the system has booted?

10A. service sshd statuts

### 11. What command and option can you use to enable a service to start on boot?

11A. Use the chkconfig command to enable or disable services during the boot process. The on option enables the service and off disables it.

## Understanding Linux Boot Process

Linux Boot Process Involves multiple Steps, let’s explore the entire boot process:

**Step 1:** Power on the Machine

**Step 2:** Run POST and Identify the  boot Device

**Step 3**: Read the GRUB Boot loader from the MBR of Boot device

The Grand Unified Bootloader has become the default bootloader for Red Hat, Ubuntu, and many otherversions of Linux as well.

The bootloader can be used to boot into different operating systems (usually called a multiboot system), for system recovery, and to boot the kernel using special arguments.

**Step 4:** When GRUB loads, you are given a list of kernels and additional operating systems (if available) from which you can choose to boot

By default, there is a configurable 5-second timeout value that chooses the default kernel if you don’t make a selection and the timeout threshold is reached

**Step 5:** After GRUB loads the kernel, it passes control over to the kernel, which in turn begins to initialize and configure the computer’s hardware.

When the system’s drivers are in place, the kernel executes the /sbin/init program.

**Note 1:** During the boot process, everything is logged to the /var/log/dmesg file.

**Note 2:** You can also use the dmesg command to query information about the boot process after the system has booted.

**Step 6:** The init program is the first process created by the kernel. It is responsible for the rest of the boot process and setting up the environment for the user.

First, it consults the /etc/inittab file, which defines how the rest of the boot process will go. The /etc/inittab file lists the default runlevel to boot into and the system initialization script (/etc/rc.d/rc.sysinit).

Let’s look at the /etc/inittab file to see what the init process goes through:

*#****cat /etc/inittab*** *#  
# inittab This file describes how the INIT process should set up  
# the system in a certain run-level.  
#  
# Author: Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>  
# Modified for RHS Linux by Marc Ewing and Donnie Barnes  
#  
# Default runlevel. The runlevels used by RHS are:  
# 0 – halt (Do NOT set initdefault to this)  
# 1 – Single user mode  
# 2 – Multiuser, without NFS (The same as 3, if you do not have networking)  
# 3 – Full multiuser mode  
# 4 – unused  
# 5 – X11  
# 6 – reboot (Do NOT set initdefault to this)  
#  
id:5:initdefault:  
# System initialization.  
si::sysinit:/etc/rc.d/rc.sysinit  
l0:0:wait:/etc/rc.d/rc 0  
l1:1:wait:/etc/rc.d/rc 1  
l2:2:wait:/etc/rc.d/rc 2  
l3:3:wait:/etc/rc.d/rc 3  
l4:4:wait:/etc/rc.d/rc 4  
l5:5:wait:/etc/rc.d/rc 5  
l6:6:wait:/etc/rc.d/rc 6*

*# Trap CTRL-ALT-DELETE  
ca::ctrlaltdel:/sbin/shutdown -t3 -r now  
# When our UPS tells us power has failed, assume we have a few minutes  
# of power left. Schedule a shutdown for 2 minutes from now.  
# This does, of course, assume you have power installed and your  
# UPS connected and working correctly.  
pf::powerfail:/sbin/shutdown -f -h +2 “Power Failure; System Shutting Down”  
# If power was restored before the shutdown kicked in, cancel it.  
pr:12345:powerokwait:/sbin/shutdown -c “Power Restored; Shutdown Cancelled”  
# Run gettys in standard runlevels  
1:2345:respawn:/sbin/mingetty tty1  
2:2345:respawn:/sbin/mingetty tty2  
3:2345:respawn:/sbin/mingetty tty3  
4:2345:respawn:/sbin/mingetty tty4  
5:2345:respawn:/sbin/mingetty tty5  
6:2345:respawn:/sbin/mingetty tty6  
# Run xdm in runlevel 5  
x:5:respawn:/etc/X11/prefdm -nodaemon*

From ther above file, you can see that the default runlevel is set to 5, although six different runlevels are listed. This default runlevel is passed to the /etc/rc.d/rc script, which calls all the programs in the /etc/rc.d/rc#.d directory.

The /etc/inittab file also defines how to handle power failures and virtual terminals.

**Step 7 :** After the init process is done consulting the /etc/inittab file, the /etc/rc.d/rc.sysinit script is run, which handles setting the system clock, networking, setting up the user environment, and more.

Step 8 : The last thing that you should see is the login prompt.

If you have a desktop manager installed such as Gnome, you should see a GUI login screen where you can log in to the system; otherwise, you see a text mode login. This completes the boot process.

Grub Boot Options:

* Press e      Edit the commands before booting
* Press a      Modify or append the kernel arguments before booting
* Press c      Open the GRUB command line

you can modify kernel entry from the above image . Here are the different modes that you can boot into:

* **Single-User Mode**       Used to perform maintenance tasks or if you forget the root password
* **Runlevel 2 or 3**        Used to load only partial services during the boot process
* **Emergency Mode**        Used to perform tasks on an unbootable system
* **Rescue Mode**           Used to fix boot issues or reinstall GRUB

### 2.GRUB Configuration File :

GRUB Boot loader works based on the configuration mentioned in /etc/grub.conf ( actually a link file to /boot/grub/grub.conf).

**Sample grub.conf file**

*# cat grub.conf  
default=0  
timeout=5  
splashimage=(hd0,0)/grub/splash.xpm.gz  
hiddenmenu  
title Red Hat Enterprise Linux (2.6.32-71.el6.x86\_64)  
root (hd0,0)  
kernel /vmlinuz-2.6.32-71.el6.x86\_64 ro root=/dev/mapper/vg\_rhel01-lv\_root  
rd\_LVM\_LV=vg\_rhel01/lv\_root rd\_LVM\_LV=vg\_rhel01/lv\_swap rd\_NO\_LUKS rd\_NO\_MD  
rd\_NO\_DM LANG=en\_US.UTF-8 SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc  
KEYTABLE=us crashkernel=auto rhgb quiet  
initrd /initramfs-2.6.32-71.el6.x86\_64.im*

**Explanation about the grub.conf file:**

* In the preceding example, you can see that there is a single option to boot from, as defined by the title entry. This is also known as entry 0, so it becomes the default kernel to boot from.
* You can also see the version of the kernel that you are using and the place where the root partition will be mounted from during the boot process.
* If you want to add additional options to boot into, you could append another title with options for a second boot choice.
* This grub.conf file contains everything you need to make sure the boot process works smoothly.
* the file also defines additional options such as encryption, RAID, and use of the /dev/mapper for LVM management

### 3. GRUB Command line

One task that is common in the real world and also a possible exam question is how to repair a broken MBR. For this to happen, you need to make use of the rescue environment, which can be found by booting from the RHEL installation DVD. When you are in the rescue environment, you can repair your broken MBR.

Step 1. Load up the GRUB command line to find the disk and partition that contain the grub.conf file using the find command:

*grub> find /grub/grub.conf  
(hd0,0)*

You could also run:

*grub> root  
(hd0,0): Filesystem type is unknown, partition type 0x8e*

Step 2. Install GRUB on the drive that is returned:

*grub> setup (hd0)  
Checking if “/boot/grub/stage1” exists… no  
Checking if “/grub/stage1” exists… yes  
Checking if “/grub/stage2” exists… yes  
Checking if “/grub/e2fs\_stage1\_5” exists… yes  
Running “embed /grub/e2fs\_stage1\_5 (hd0)”… 26 sectors are embedded.  
Running “install /grub/stage1 (hd0) (hd0)1+26 p  
(hd0,0)/grub/stage2  
/grub/grub.conf”… succeeded  
Done.*

Now that your MBR is fixed, you should be able to boot into the system once again.

The preceding example also shows where the boot partition is located (partition 0 on disk 0), as well as the file system type. It is important that you also back up the grub.conf file in case something happens to your system; that way, you can easily restore it.

## Shutting Down a Linux System for routine maintenance Tasks

Let’s look at some examples for a better understanding the procedure of System Shutdown

you can use the shutdown command to bring the system to a powered-off state or reboot the system. The shutdown command also has options to alert users on the system when it is going down and to set up a shutdown at a particular time.

*Syntax: shutdown [options] time*

*Options:*

*-k Doesn’t shut down; just warns  
-h Halts the system after shutdown  
-r Reboots instead of turning off the system  
-F Forces a file system check on reboot  
-n Kills all processes quickly (not recommended)  
-t SECS Sends a shutdown message but delays shutdown by x seconds*

1. Command to Just turn off the system

*# shutdown -h now*

2. This time instead of a complete power off, reboot the system as follows:

*# shutdown -r now*

3. You could also use the reboot command to achieve the same effect:

*# reboot*

4. As a final example, delay the shutdown by 2 minutes:

*# shutdown -h 120*

Similar to the shutdown command is the halt command. The halt command brings the system to a powered-off state like shutdown.

5. To turn off the system, you use this command:

*# shutdown now  
You could also use  
# halt*

Coinciding with the halt command are poweroff and reboot. The reason these commands all go hand in hand is that poweroff and reboot are actually both links to the halt command. You can verify this by using the following example:

*# cd /sbin  
# ll | grep halt*

*-rwxr-xr-x 1 root root 16152 Jan 21 2009 halt  
lrwxrwxrwx 1 root root 4 Jul 22 2010 poweroff -> halt  
lrwxrwxrwx 1 root root 4 Jul 22 2010 reboot -> halt*

The difference is that each one calls the halt command with a different argument, yielding different results. The poweroff command, like halt, stops the system, whereas the reboot command restarts the  system…obviously.

### 1. What option is used with both the fdisk and parted commands to display the current partition tables?

1A. The print option is used with both the fdisk and parted commands to display the current partition tables.

### 2. What does the partprobe command do?

2A. The partprobe command forces the kernel to reread the partition table. You should always call it after making any changes to your system partitions.

### 3. Do you need to write changes to the disk when using the parted command? What about fdisk?

3A. When you exit the parted utility, all your changes are automatically written to disk. With the fdisk command, you need to manually write your changes to disk for them to take effect.

### 4. What are the three different types of RAID described in this chapter?

4A. RAID 0 (Striping), RAID 1 (Mirror), and RAID 5 (Striping with parity).

### 5. What command can you use to query information from the kernel about RAID arrays?

5A. cat /proc/mdstat

### 6. Can you put your /boot partition on a RAID 5 array?

6A. This is actually a trick question because the answer could go either way. In case of  software RAID setups,  the answer is no. If, however, you were using a hardware RAID 5, the answer would be yes.

### 7. What are the three items that make up LVM?

7A. Physical volumes, volume groups, and logical volumes.

### 8. What are the side effects of shrinking a volume group or logical volume?

8A. If you shrink a volume group or logical volume, there is a chance you could lose data depending on how much you shrink the volume.

### 9. What is the biggest benefit to using LVM over basic partitions?

9A. You have the flexibility to resize and add new volumes on the fly. With basic partitions, any time that you want to make a change, you need to destroy the  
partition and create it again.

### 10. What command can you use to get information about logical volumes?

10A. lvdisplay

If the system doesn’t even get that far,

* -> you can boot up
* -> enter a at the GRUB boot menu
* -> and append the word single to the kernel that you want to boot into.

This approach has the same effect as booting you into a recovery runlevel. After you get into the recovery runlevel, you can perform maintenance or repairs on your system (such as figuring out what is preventing the system from booting).

## Tasks you can do at recovery/rescue runlevel:

* ■ You can reset or change the root user’s password.
* ■ You can adjust system files or partitions that are normally locked when the system is in use.
* ■ You can repair system files by replacing them with working copies from a backup or the Red Hat installation CD.

After you finish making any changes or repairs, you can reboot the system and see whether it returns to its normal working state. If it doesn’t, you can re-enter the recovery runlevel to try again.

You can also access this rescue environment by booting the first Red Hat installation CD.

Now that you have an understanding of runlevels and how to move between them, we can shift focus to service management, which deals with different system services starting and stopping at the different runlevels

## Troubleshooting – Recovering Root Password in Linux

If you forget your root user password or you are taking over a system where the root password isn’t documented, you can still get into the system. You need to perform the following actions at the physical console of the system.

Step 1: Highlight you operating system line on the first GRUB screen

Step 2: Press “e” , so that grub will show the configuration lines for root hard disk, kernel and initrd image.

Step 3. Move your cursor to end of the the kerne line and append the following keyword at the end of the line

*single*

Step 4: Press “b” , to boot the linux system using modified kernel option, so that system will enter into single user mode of maintenance mode.

Step 5. Now you can reset the root password using the passwd command as mentionedi n Step 5a.

Just incase if the passwd command doesn’t able to set the password because of missing shadow file, then follow the procedure mentioned in the Step5B

Step 5a. When you are presented with a command prompt, change the root user password:

*# passwd root  
Changing password for user root.  
New password:  
Retype new password:  
Passwd: all authentication tokens updated successfully.*

Step 5b. Verify the existence of the /etc/shadow file:

*# ls /etc | grep shadow*

*If the /etc/shadow file doesn’t exist (which would be the cause of the error in this case), use the pwconv command to re-create the /etc/shadow file:*

*# pwconv*

*Now execute the passwd command to reset or change the root user’s  password:*

*# passwd root  
Changing password for user root.  
New password:  
Retype new password:  
Passwd: all authentication tokens updated successfully.*

Step 6. Reboot the system and validate that the new root password works correctly:

*# reboot*

## Troubleshooting – Recovering Linux System from Corrupted MBR issue

If you are having trouble booting the system and you have determined that the master boot record (MBR) is corrupt, you need to boot into rescue mode.  Use the Red Hat DVD, boot from it, and choose the option to enter rescue mode.

Step 1. After you boot, enter the GRUB shell:

*# grub*

Probing devices to guess BIOS drives. This may take a long time. GNU GRUB version 0.97 (640K lower / 3072K upper memory)  
[ Minimal BASH-like line editing is supported.

For the first word, TAB lists possible command completions. Anywhere else TAB lists the possible completions of a  device/filename.]

*grub>*

Step 2. Locate the root drive:

*grub> root  
(hd0,0): Filesystem type is unknown, partition type 0x8e*

Step 3. Reinstall the MBR from the GRUB shell:

*grub> setup (hd0)  
Checking if “/boot/grub/stage1” exists… no  
Checking if “/grub/stage1” exists… yes  
Checking if “/grub/stage2” exists… yes  
Checking if “/grub/e2fs\_stage1\_5” exists… yes  
Running “embed /grub/e2fs\_stage1\_5 (hd0)”… 26 sectors are embedded.  
Running “install /grub/stage1 (hd0) (hd0)1+26 p (hd0,0)/grub/stage2  
/grub/grub.conf”… succeeded  
Done.*

Step 4. Reboot the system to validate that the system boots properly:

*# reboot*

### 1. What is the difference between an ext2 and ext3 file system?

1A. An ext3 file system has journaling built in to it, whereas the ext2 file system doesn’t.

### 2. What command can you use to create a file system?

2a. mkfs.ext4

### 3. Can you grow a file system?

3A. Yes. Use the resize2fs command to grow a file system.

### 4. What is the superblock used for?

4a. The superblock is a structure that contains metadata of the file system. If this becomes corrupt, you are in trouble.

### 5. What is a swap? Is it created as a partition or device file?

5a. A swap is scratch space on your file system used as virtual memory. A swap can be created as a partition or a device file.

### 6. How can you check the currently mounted file systems?

6a. The mount command lists all currently mounted file systems.

### 7. What file needs to be edited so that the system will mount a file system at boot time?

7a. The /etc/fstab file.

### 8. Before you work with quotas, what do you need to do to the file system?

8a. The file system where quotas will be implemented must be mounted with the usrquota and grpquota options before quotas will work properly.

### 9. What command do you use to change the permissions on a file or directory? To change ownership?

9a. The chmod command is used to change the permissions of files and directories.  
The chown command is used to change the ownership of files and directories.

### 10. Explain the difference between soft and hard limits in quotas.

10a. A soft limit acts like an alarm, signaling you when you are reaching your limit.  
If you don’t specify a grace period, the soft limit is the max. A hard limit is required only when a grace period exists. It is the max limit you can hit before  
your grace period expires.

### 11. What command do you use to report information on quota usage?

11a. repquota

### 12. Before you work with ACLs, what do you need to do to the file system?

12a. The file system where ACLs will be implemented must be mounted with the acl option before ACLs will work properly.

### 13. What command can you use to view the current ACL on a file?

13a. getfacl

### 1. What command displays your current interfaces and IP address?

1a. ifconfig

### 2. What does ifconfig 172.168.1.100 netmask 255.255.255.0 eth1 do?

2a. This command sets the eth1 interface to have a static IP address of  
172.168.1.100 with a netmask of 255.255.255.0.

### 3. What command can you use to test connectivity to another host?

3a. The ping command can be used to verify connectivity to another host.

### 4. What does it mean if you ping a host and you receive the response Destination Unreachable?

4a. The gateway is incorrectly set, and the subnet of the host you are trying to reach is inaccessible.

### 5. What is a gateway used for on a network?

5a. A gateway is used as an entry and exit point for a subnet on a network. To contact hosts outside your subnet, you need to pass through a gateway.

### 6. How would you go about creating a static route?

6a. Use the route command with the add option to create a static route.

### 7. What command can you use to monitor and troubleshoot network connections?

7a. The tcpdump command is used to monitor network connections on different interfaces.

### 8. Can you name three utilities that can be used for network or DNS client troubleshooting?

8a. The three utilities are: route, ping, and nslookup.

### 9. What is the /etc/hosts file used for?

9a. The /etc/hosts file is a local lookup file used to map IP addresses to hostnames if a DNS server isn’t available.

### 1. What two commands are used for package management?

1a. The yum and rpm commands are used for package management.

### 2. What are the three modes in which the rpm command can operate?

2a. The rpm command can operate in install, query, or verify modes.

### 3. What option would you use to query an installed packages using the rpm command?

3a. You can use the -q option to query an installed package. Combining grep and the -qa options, you can search among all installed packages on the system.

### 4. How would you install a group of packages all at a single time?

4a. Use the yum groupinstall command to install multiple packages in a single group at once.

### 5. What options with the yum command would you use to remove a package?

5a. You can use the remove or erase options with yum to remove a package.

### 6. Where are Yum repository config files located?

6a. Yum repository config files (.repo files) are located in the /etc/yum.repos.d directory. You can also make direct entries into the main /etc/yum.conf file.

### 7. What command can you use to create your own repositories?

7a. createrepo

### 8. What command is used to create an RPM package?

8a. rpmbuild

### 9. What are the five required directories when building RPMS?

9a. The five directories are BUILD, RPMS, SOURCES, SPECS, and SRPMS.

### 10. If a package is built on an RHEL6 system and deployed to a custom RHEL5 repository, are RHEL5 systems able to use it?

10a. No. Red Hat Enterprise Linux 6 uses a different key to sign its packages.

### 1. What command (and options) can be used to create a user named JSmith with the description “Jr Admin”?

1a. useradd –c “Jr Admin” JSmith

### 2. What is the format of the /etc/shadow file?

2a. The format of the /etc/shadow file is <username>:<encrypted password>:<last passwd change>:<min>:<max>:<warn>:<inactive>:<expires>:<not used>.

### 3. What command would you use to create a group? How about to add the user JSmith to the group?

3a. Use the groupadd command to create a group. You can then add user JSmith with the following: usermod –G <group name> JSmith

### 4. How do you run a command with elevated privileges?

4a. You can use the sudo command to run a command with elevated privileges provided you have the rights in the /etc/sudoers file.

### 5. Is it possible to share files among groups? What permissions would you set on the directory to accomplish file sharing if possible?

5a. Yes. You can use the setgid flag to create the appropriate permissions (chmod 2770).

### 6. If you want a specific action to take place when user01 logs in to the system, which file would you edit?

6a. You add your action to the end of the /home/user01/.bashrc file.

### 7. You can add files to a user’s home directory during creation. True or False?

7a. True. Place all files you want added to a user’s home directory in the /etc/skel directory.

### 8. By default, what is the path to a user’s home directory?

8a. A user’s home directory is created under the /home directory.

### 9. What is the benefit to using centralized authentication?

9a. By using centralized authentication, you don’t need to re-create or maintain multiple accounts across every system in your organization.

### 10. What commands can you use to add a client machine to an LDAP server?

10a. You can use the authconfig-tui command or the authconfig command.

### 1. What option can you change in the rsyslog config file to accept remote logs (acting as a centralized logging server)?

1a. Uncomment the following line in the /etc/rsyslog.conf file:  
#$ModLoad imudp.so  
#$UDPServerRun 514

### 2. What two commands are special for dealing with user login events?

2a. The lastlog and faillog commands are used to view user login–related events?

### 3. Can you name the two commands that can be used to view the free space on the system?

3a. The du and df commands are used to view available space on the system.

### 4. What command can you use to view system processes and their CPU usage?

4a. Use the ps command to view processes and their CPU usage.

### 5. The at command is used to schedule reoccurring system jobs. True or False?

5a. False. The at command is used to schedule one-time-only jobs. The cron service handles reoccurring system jobs.

### 6. What happens to jobs that are scheduled to run while the system is off?

6a. When the system starts up again, the cron service will run any jobs that were missed while the system was off. On Red Hat Enterprise Linux 5, the anacron  
service handles this functionality.

### 7. What command can be used to view the queue for at service jobs?

7a. atq

### 8. What is the top command used for?

8a. Use the top command to view CPU and memory usage.

### 1. What is the difference between the update (-U) and install (-i) options when using rpm to update the kernel?

1a. You should never use the –U option because it erases the prior kernel when updating.  This leaves you with no fallback kernel should your system not boot properly.

### 2. What directory is used to represent the virtual file system created by the kernel?

2a. The /proc directory. The /proc/sys directory is the place where you actually tune kernel parameters.

### 3. What file is used to maintain custom parameters for the kernel during system boot?

3a. The /etc/sysctl.conf file maintains a list of custom kernel parameters that should be applied during system boot.

### 4. What critical step must you take after updating the kernel to a newer version?

4a. You must ensure that the /boot/grub/grub.conf file has the new entry for your newly updated kernel to be able to boot into it.

### 1. What is the point of using SELinux?

1a. SELinux provides enhanced granular security for the Linux operating system.

### 2. What are SELinux Booleans?

2a. SELinux Booleans provide restrictions to different aspects of a service.

### 3. What command can you use to change the context of files?

3a. chcon

### 4. What command can you use to query Boolean values?

4a. getsebool

### 5. What command and option do you use to view the description of Boolean values?

5a. semanage boolean -l

### 6. How would you view all Boolean options for the HTTP service?

6a. getsebool –a | grep http

### 7. Which log file is used to keep track of policy violations?

7a. The /var/log/audit/audit.log file contains all policy violations.

### 8. How would you disable SELinux protection for NFS, allowing shares to be read/write?

8a. setsebool –P nfs\_export\_all\_rw=1

### 1. All services in Red Hat can use TCP Wrappers. True or False?

1a. False. A service needs to have support for TCP Wrappers to be able to use it.

### 2. The firewall service is disabled by default. True or False?

2a. False. The firewall service is enabled by default.

### 3. When Red Hat Enterprise Linux 6 is installed, SELinux is set to enforcing by default. True or False?

3a. True. For Red Hat Enterprise Linux 6, SELinux comes set up in enforcing mode. When installing Red Hat Enterprise Linux 5, you have the option to  
choose which mode you’d like it to operate in.

### 4. What does the following firewall rule accomplish: iptables -I INPUT 5 -p tcp -m tcp –dport 80 -j ACCEPT?

4a. The firewall rule is inserted into the fifth line of the iptables rules and opens up TCP port 80 (for the HTTP service) on the firewall to allow incoming connections.

### 5. What is the last rule in the iptables file?

5a. The last rule is always an implicit deny statement rejecting anything that wasn’t matched by previous rules.

### 6. What is PAM used for?

6a. PAM is used as a modular way to leverage security. In this post, the main function we discussed was for managing password policies and complexity.

### 7. What is NAT? How is it used?

7a. Network address translation (NAT) maintains a table that allows the use of multiple internal IP addresses to a multiple public IP address (called address translation). It can also be used for one-to-one or one-to-many relationships as well.

### 8.  iptables –I RH-Firewall-1-INPUT –p tcp –m tcp -–dport 22 –j ACCEPT. Is this valid rule under RHEL6?  True or False?

8a. False. The INPUT chain being used is the default under Red Hat Enterprise Linux 5. This rule generates errors under Red Hat Enterprise Linux 6.

### 1. What is SSH used for?

1a. SSH is used for secure remote management of Linux systems.

### 2. Should you allow remote root access? Why or why not?

2a. You should never allow remote root access. Should your root account become compromised and you use the same password, someone could gain access to all  
your systems. You also don’t want the most powerful user of your system (with no accountability) logging in and making changes.

### 3. What happens if a host changes its IP address and the keys don’t match?

3a. A large warning message appears indicating that the key doesn’t match the host you are connecting to. You have to remove the key/host pair from the  
known\_hosts file to proceed.

### 4. Which version of SSH should you use?

4a. Version 2 is the latest and most secure version of SSH.

### 5. SSH can run only on TCP port 22. True or False?

5a. False. Through its main config file, SSH can be configured to run on any port you’d like (provided that port is available).

### 6. TCP Wrappers can be used with SSH. True or False?

6a. True. SSH does support TCP Wrappers.

### 7. What is the benefit of using public/private key authentication?

7a. Public/private key authentication provides an additional layer of security because you need the correct key instead of just knowing someone’s password. Passwords combined with public/private keys take the security one additional step.

### 8. What is VNC?

8a. VNC is used to control a remote desktop session.

### 9. What is the name of the package that allows you to install VNC?

9a. The VNC package for Red Hat Enterprise Linux 6 is tiger-vnc.

### 1. What port does the Apache web server run on? What about HTTPS?

1a. Apache uses port 80 for HTTP and port 443 for HTTPS.

### 2. Are additional packages besides httpd required for a secure website? If so, what are they?

2a. Yes. For Apache to run a secure site, the mod\_ssl package is required to be installed.

### 3. What command can you use to create a password-protected page?

3a. The htpasswd command can be used to password-protect a web page.

### 4. What happens if you call the command from question 3 with the –c option after it has already been run?

4a. The file that stores usernames and their encrypted passwords is erased and replaced with a new file.

### 5. What command can you use to validate the syntax of the Apache config file?

5a. service httpd configtest

### 6. What is a virtual host?

6a. A virtual host is a way of hosting multiple sites using a single IP address.

### 7. How can virtual hosts be used?

7a. If you are a hosting company or want to run multiple websites/applications from a single IP address, you can use virtual hosts to accomplish this.

### 8. How can you change the security context of a directory to work with Apache?

8a. Using the chcon command, you can change the context of a file to work with Apache.

### 9. What benefit does a web proxy provide to your network?

9a. A web proxy can provide multiple benefits, including security, web page caching for faster load times, and load balancing.

### 10. What is the default port that Squid runs on?

10a. By default, Squid runs on port 3128.

### 1. Red Hat Enterprise Linux 6 uses the portmap service alongside the NFS service. True or False?

1a. False. The portmap service was replaced by the rpcbind service in Red Hat Enterprise Linux 6.

### 2. What is the /var/lib/nfs/etab file used for?

2a. The /var/lib/nfs/etab file is used to keep track of currently exported resources.

### 3. To export directories as resources, what file needs to be edited?

3a. The /etc/exports file needs to contain any directory that you want to export, including which options you’d like to use as well.

### 4. The exportfs command is used to view currently exported resources. True or False?

4a. False. The exportfs command is used to export one or all resources.

### 5. What port does NFS use by default?

5a. The NFS service uses TCP port 2049 by default.

### 6. What command can you use to view NFS statistics?

6a. The nfsstat command can be used to view NFS statistics.

### 7. What option needs to be used with the mount command to mount an NFS resource?

7a. The –t nfs option is used with the mount command to mount NFS resources

### 1.What are the two types of back-end authentication mechanisms discussed in this post?

1a. This post described tdbsm and smbpasswd back-end authentication.

### 2. When you’re creating a share, there is no option to make it browseable. True or False?

2a. False. The browseable option is available for each share.

### 3. What does the testparm command do?

3a. The testparm command allows you to check for syntax errors in the /etc/samba/smb.conf file.

### 4. What command is used to create a user for Samba?

4a. The smbpasswd command is used to create Samba users.

### 5. What command can you use to view mounted Samba shares?

5a. The smbstatus command shows you currently mounted Samba resources.

### 6. Samba servers can serve files only to users. True or False?

6a. False. Samba servers can serve both files and printers to users

### 1. What is the name of the package used to install an FTP?

1a. The vsftpd package is used to install an FTP server.

### 2. What command can be used to easily list all the options in the FTP main config file?

2a. The grep –v ^# /etc/vsftpd/vsftpd.conf command shows you all the options currently being used with the FTP server.

### 3. What option is used to allow anonymous uploads to the FTP server?

3a. The anonymous\_enable=YES option in the main config file allows anonymous uploads.

### 4. What two ports should be opened on the firewall for the FTP server to function properly?

4a. You must open TCP ports 20 and 21 for the FTP server to function properly.

### 5. What is the /etc/vsftpd/user\_list file used for?

5a. The user\_list file can be used to limit which users have access to the FTP server.

### 6. What client-side command can you use to test your FTP server connection?

6a. The lftp command can be used on a client to test FTP server connections.

### 7. FTP is one of three protocols that can be used for kickstart and/or network installs. True or False?

7a. True. FTP is one of three protocols used for kickstart and/or network installs. The other two protocols are NFS and HTTP.

### 1. There are four types of DNS servers. Name them.

1a. The four types of DNS servers are master, slave, caching-only, and forwarding.

### 2. What is the difference between an A record and a PTR record?

2a. An A record translates translated hostnames to IP addresses, and a PTR record works the other way around.

### 3. A slave DNS server offers no additional benefits. True or False?

3a. False. A slave DNS server provides load balancing and redundancy benefits.

### 4. What is the rndc utility used for?

4a. The rndc utility provides a method for managing the DNS server remotely and securely.

### 5. What command can you use to test the config files before starting the DNS service?

5a. service named configtest

### 6. What three commands can you use to help verify that the DNS server is functioning properly?

6a. The dig, host, and ping commands can be used to test DNS server functionality.

### 7. What port must be opened on the firewall for DNS?

7a. DNS uses port 53.

### 8. What does the rdnc-confgen command do?

8a. The rdnc-confgen command generates an rndc.conf file for the rndc utility.

### 9. What port does the rndc utility listen on by default?

9a. By default, the rndc utility listens on port 953.

### 1. What port does the SMTP service run on?

1a. The SMTP service runs on TCP port 25 and pop3 service runs on 110.

### 2. What command is used to change the mail server from Sendmail to Postfix (useful only if you have both installed)?

2a. alternatives –config mta

### 3. What are the two config files for Postfix called?

3a. The two main config files are master.cf and main.cf.

### 4. Sendmail is the default SMTP service for Red Hat Enterprise Linux 6. True or False?

4a. False. The default SMTP service has been changed to Postfix in Red Hat Enterprise Linux 6.

### 5. What protocols are used with Dovecot?

5a. Dovecot supports the POP3, POP3S, IMAP, and IMAPS protocols.

### 6. What command-line program can you use to check your mail?

6a. Typing the mail or mutt command allows you to read mail from the command line.

### 7. What command can you use to create a new SSL certificate?

7a. Use the /usr/share/doc/dovecot-1.0.7/examples/mkcert.sh command after editing the /etc/pki/dovecot/dovecot-openssl.cnf file.