FreeBSD security exercises

1. Figure out what is running on your machine

There are quite a few ways to approach this, but very quickly here are some useful commands:

```
ps                        # processes
netstat                   # network status
sockstat                  # socket status
grep -i yes /etc/rc.conf
```

You should, as usual, read the man pages for "ps", "netstat", "sockstat". On other versions of Unix you may have the command "lsof" (List Open Files), and this can be installed as an optional extra package under FreeBSD too if you wish.

Now here are some good options:

```
$ ps -auxw | less
$ netstat -an -f inet
$ sockstat -4 -l
```

You should try all of these commands and analyse what is being shown. For example, the command "sockstat -4 -l" may produce:

```
$ sockstat -4 -l
USER     COMMAND    PID   FD PROTO  LOCAL ADDRESS         FOREIGN ADDRESS
root     syslogd    19072 5  udp4   *:514                 *:*
root     sshd       19196 3  tcp4   *:22                  *:*
root     sendmail   19202 4  tcp4   127.0.0.1:25          *:*
```

What you are seeing is that the syslog daemon is listening for UDP traffic on port 514; sshd is listening for TCP connections on port 22; and sendmail is listening for TCP connections on port 25, but is only listening on IP address 127.0.0.1. That's the loopback address, so will only accept mail generated within the same machine, not from outside.

An easy way to check what you've chosen to start at system boot is:

```
$ grep -i yes /etc/rc.conf
```

although some things may be started by default anyway. You can look through /etc/defaults/rc.conf for these, or try:

```
$ grep -i '="yes"' /etc/defaults/rc.conf
```

Finally, in a perfect world you should be able to type:

```
$ ps -auxw | less
```

and know what each and every item in the process list means and does. You can look for unusual items, and use available resources like "man" to try to figure out what all the items in the list mean.

2. Turn off any unnecessary services

This exercise is very simple and short. You just finished looking at your system in much more detail. Did you see anything that should not be running, starting, etc? This is a *very* subjective question. If you did find something, then the likely place to turn the item off will be in the file

```
/etc/rc.conf
```

You usually do this by immediately shutting it down with:

```
# /etc/rc.d/service stop
```

(or just by 'kill *pid*'), and then adding an entry to /etc/rc.conf that reads:

```
service_enable="NO"
```

As you are using a fresh install of FreeBSD and it is a fairly minimal install there is probably nothing that needs to be turned off right now, but if you think there is, call an instructor over to discuss the item.

3. Understand your logs

On a daily basis your server will receive two emails that are generated from a script that is run via the crontab facility. This script reads your log files, checks your system state, etc. and then generates the summary reports for you. The reports can be very useful, particularly if you do not have additional logging facilities in place to warn you immediately of potential attacks.

First, have a look at the file:

```
 $ less /etc/syslog.conf
```

to understand where logs are generated on your machine (full details of this file are in 'man syslog.conf', of course). Now go to the directory /var/log

```
$ cd /var/log
```

and look at all the log files available to you. These files can be invaluable when troubleshooting problems and checking for potential security issues.

The log you are probably going to look at the most is "messages".

Feel free to look at some/all of the log files. Some of the files are not text files, so you won't want to type them to the screen. And, some are empty. A quick way to check what files are text, binary, data, or empty is to do the following:

```
# cd /var/log
# file *
```

The 'file' command tells you what type of file a file is, and the shell expands '*' to all the filenames in the current directory. The output from this command may look something like this:

```
Xorg.0.log:        ASCII English text
Xorg.0.log.old:    ASCII English text
aculog:            ASCII text
auth.log:          ASCII text
connect-errors:    ASCII text
cron:              ASCII text
cron.0.bz2:        bzip2 compressed data, block size = 900k
cron.1.bz2:        bzip2 compressed data, block size = 900k
cron.2.bz2:        bzip2 compressed data, block size = 900k
debug.log:         ASCII text
debug.log.0.bz2:   bzip2 compressed data, block size = 900k
exim:              directory
lastlog:           data
lpd-errs:          ASCII text
maillog:           ASCII text
maillog.0.bz2:     bzip2 compressed data, block size = 900k
maillog.1.bz2:     bzip2 compressed data, block size = 900k
maillog.2.bz2:     bzip2 compressed data, block size = 900k
messages:          ASCII English text
messages.0.bz2:    bzip2 compressed data, block size = 900k
messages.1.bz2:    bzip2 compressed data, block size = 900k
messages.2.bz2:    bzip2 compressed data, block size = 900k
ppp.log:           empty
security:          empty
sendmail.st:       empty
sendmail.st.0:     empty
sendmail.st.1:     empty
sendmail.st.2:     empty
slip.log:          empty
userlog:           ASCII text
wtmp:              data
```

Files that are gzip compressed or bzip2 compressed can be read using one of the following commands:

```
# gzip -dc filename.gz | less
# bzip2 -dc filename.bz2 | less
```

On a daily basis your server will receive two emails that are generated from a script that is run via the crontab facility. This script reads your log files, checks your system state, etc. and then
generates the summary reports for you. The reports can be very useful, particularly if you do not have additional logging facilities in place to warn you immediately of potential attacks.

If you want to generate the daily summary of your server right no you can issue the following command:

```
# periodic daily
```

It might take a minute or so to complete. Now type:

```
# mail
Mail version 8.1 6/6/93.  Type ? for help.
"/var/mail/root": 2 messages 2 new
>N  1 root@localhost.local  Sun Jan  9 11:02  37/1039  "localhost.localdomain"
 N  2 root@localhost.local  Sun Jan  9 11:02  72/2440  "localhost.localdomain"
```

Press "1" to see the first message. Press spacebar to scroll. Press "2" to see the next message. Type "quit" to exit from mail.

Normally you'd set up a mail alias so that these system messages go somewhere other than to "root".

You can look at your crontab file, /etc/crontab, to see how and at what time the daily cron items are done. For more information see

```
$ man cron
$ man -a crontab
```

If you want to see how the periodic script works you can look at the file /etc/defaults/periodic.conf. It runs various scripts which are under the directory /etc/periodic:

```
$ cd /etc/periodic
$ ls
daily           monthly         security        weekly
```

Now look in one of the directories, like 'daily':

```
$ cd daily
$ ls
100.clean-disks         210.backup-aliases      430.status-rwho
110.clean-tmps          300.calendar            440.status-mailq
120.clean-preserve      310.accounting          450.status-security
130.clean-msgs          330.news                460.status-mail-rejects
140.clean-rwho          400.status-disks        470.status-named
150.clean-hoststat      405.status-ata-raid     500.queuerun
200.backup-passwd       420.status-network      999.local
```

These are all the scripts that can be run (if enabled in /etc/defaults/periodic.conf) each time the crontab calls the 'periodic daily' command.

This is quite an advanced and complex topic, but it's important to begin to understand what is going on in the background on your machine.