U.S. DEPARTMENT OF ENERGY'S
CYBERFORCE® PROGRAM

CyberForce® 101

Linux Hardening

October 2023

cyberforcecompetition@anl.gov

Linux Hardening 101

Pre-Requisites

- **Intro to Linux**
- **Useful Protocols**
- **Typical Services**

Mapping the Network

Using tools like nmap, we can see what services are running on the machine and what ports are open. The **Useful Protocols** and **Typical Services** went over which ports and services were vulnerable to attack. Services and ports that you are unfamiliar with should be researched to see if any vulnerabilities are known during this process.

```
# Install nmap
sudo apt-get install nmap
# or install
sudo yum install nmap
# Run nmap
nmap....
# find all open ports and associated programs
nmap -sT -O localhost
```

Services like FTP, Telnet, and Rlogin / Rsh services should not be running on your server. To remove these unwanted services you can use the commands below.

```
sudo apt-get --purge remove service_name
# or
yum erase service_name
```

You can also disable unnecessary services and daemons (services that run in the background). You need to remove all unwanted services from the system start-up.

```
# list all services started at boot time.
chkconfig --list | grep '3:on'
# disable a service
service serviceName stop
chkconfig serviceName off
```

```
# or
sudo systemctl stop serviceName
sudo systemctl disable serviceName
```

We also want to see what ports are open and listening. We can do that using `net-tools`.

```
sudo apt-get install net-tools
# list listening ports
sudo netstat -tunlp
# show all connections
sudo netstat -antup
```

Minimize Software

You want to avoid installing unnecessary software to avoid vulnerabilities in that software. You can use `yum`, `apt-get`, or `dpkg` to review all installed set of software packages on a system before deleting unwanted packages.

```
yum list installed
yum list packageName
yum remove packageName
# or
dpkg --list
dpkg --info packageName
apt-get remove packageName
```

Managing Users and Groups

When handed a box for the first time, you always want to ensure that the users on the account are needed users. You also want to make sure that the groups have the correct permissions.

First, you want to **list users** to verify any unknown accounts that need to be removed or disabled. To ensure that no accounts have empty passwords use the command below:

```
awk -F: '($2 == "") {print}' /etc/shadow
# lock all empty password accounts
passwd -l accountName
```

Password Policy

You can simply set a user password through the command below.

```
sudo passwd userName
```

However, you might want to enforce a password policy in general. Usually, the password and authentication-related configuration files are stored in `/etc/pam.d` and password policies are defined in the `/etc/pam.d/common-password` file. Backup this file just in case before making any changes.

To set a minimum password length, edit the `/etc/pam.d/common-password` file and modify the line below.

```
password [success=2 default=ignore] pam_unix.so obscure sha512
# add minlen=8 or a longer length to the end if desired
password [success=2 default=ignore] pam_unix.so obscure sha512 minlen=8
```

Locking User Accounts After Login Failures

You can use the `faillog` command to display `faillog` records or to set login failure limits. `faillog` formats the contents of the failure log from `/var/log/faillog` database or log file.

```
# see failed login attempts
faillog
# unlock account after login failures
faillog -r -u userName
# lock account
passwd -l userName
# unlock account
passwd -u userName
```

Ensure Non-Root Accounts Do Not Have UID Set to 0

Only the root account should have UID 0 with full permissions to access the system. The command below will display all accounts with UID set to 0.

```
awk -F: '($3 == "0") {print}' /etc/passwd
# desired output
root:x:0:0:root:/root:/bin/bash
```

If you see other lines, delete them or make sure other accounts are authorized by you to use UID 0.

Disable Root Login

You shouldn't login as the root user, but instead use `sudo` to execute root level commands as and when required. You can do this by changing its shell from `/bin/bash` to `/sbin/nologin` in the `/etc/passwd` file.

```
# Change the line below
root:x:0:0:root:/root:/bin/bash
# to
root:x:0:0:root:/root:/sbin/nologin
```

Harden SSH

SSH is already a pretty secure protocol by design, but you shouldn't leave it at just the the default configuration. SSH configuration files are located at `/etc/ssh/sshd_config`. You'll need to edit this config file for many of these hardening steps, so make sure to back up the original file.

First, make sure to disable empty passwords. In the `/etc/ssh/sshd_config` file, set the `PermitEmptyPasswords` option to `no`. You also want to make sure that root login via SSH is disabled. In that same file, modify the `PermitRootLogin` option to `no`. You also want to make sure that you have SSH protocol 2 enabled, as protocol 1 has known vulnerabilities and shouldn't be used.

You also want to configure the idle timeout interval, which is the amount of time an SSH connection can remain active without any activity. These idle sessions are a security risk, so you want to change the default from 0 seconds. In that same file, change it to your desired time. For instance if you want 5 minutes, change the `ClientAliveInternal` to `300`. After this interval, the SSH server will send an alive message to the client and will close the connection, logging the user out, if it does not receive a response. If you want multiple alive messages before disconnection, change the `ClientAliveCountMax` to your desired number.

Following the principle of least privilege, you should allow SSH access to a selected few users and restrict it for all other users. You can do this by changing the line below in the `sshd_config` file.

```
AllowUsers User1 User2
# or by allowing a group access
AllowGroups ssh_group
```

You can also use `DenyUsers` and `DenyGroups` to deny SSH access to certain users and groups.

SSH can be configured to only allow for key-based login, ensuring that you do not have to deal with brute force password attacks. You can add the public key of the remote client systems to the known keys list on the SSH server, allowing client machines access to SSH without entering the user account password. Then you can disable password based SSH login. Make sure that you have added your own public key to the server and it works before going this approach.

You'll need to restart SSH to apply the configuration changes you make.

Set Files as Immutable

You may want to make certain files immutable in order to secure them from accidental removal or tamper. You however only want to do this after Users, Groups, and SSH Config have been changed accordingly. It's important to do on files such as `passwd` and `shadow`.

```
# list files with their corresponding attributes
lsattr

# make a file immutable
chattr +i <FILE>
```

Enable UFW

UFW, or uncomplicated firewall, can be used to manage a Linux firewall, providing an easy to use interface for the user, developers, and system admins.

```
# install ufw
sudo apt-get install ufw
# allow ssh
sudo ufw allow ssh
# or
sudo ufw allow 22/tcp
# view status
sudo ufw status
# view ufw default
grep 'DEFAULT_' /etc/default/ufw
# set policy examples
sudo ufw default allow outgoing
sudo ufw default deny incoming
# enable the firewall
sudo ufw enable
# disable the firewall
sudo ufw disable
```

Install and Setup Fail2ban

`Fail2ban` or `denyhost` scans the log files for too many failed login attempts and blocks the IP address which is showing malicious signs. You can install `fail2ban` with the command `sudo apt-get install fail2ban` or `sudo yum install fail2ban`.

To edit the config file as you need, use a text editor on the `/etc/fail2ban/jail.conf` file. Then restart the service with the command `sudo systemctl restart fail2ban.service`.

Updates

Security patches need to be applied to ensure that your system isn't vulnerable. You can use `yum` or `apt-get` to apply security updates.

```
yum update
# or
apt-get update
apt-get upgrade
```

Backups

You should always make a backup of your system and any servers hosted on your system. You can use the `cp` command to backup an existing file. There are other ways and commands that perform backups besides the method we have provided here.

```
cp --backup fileName destinationDirectory
```

Check for CronJobs

CronJobs are scheduled tasks. Sometimes these can be used as backdoors or targeted for attacks so you should regularly check these. You want to check the following locations:

- `/etc/crontab`
- `/etc/cron.d/*`
- `/etc/cron.{hourly,daily,weekly,monthly}/*`
- `/var/spool/cron/*`
- `/var/spool/cron/crontabs/*`

You then want to find where the cron is running and stop or disable it.

Get and Manage Processes

Processes can also be targeted by attackers, so it's important to monitor them and stop any suspicious ones.

```
# list running processes
sudo ps aux
# or use lsof
lsof -i tcp:80 -P -R # -P shows port numbers and -R shows Parent Process ID (who
initiated the process)
# get process id or name
ps aux | grep processName
```

```
# get the path to the process running
readlink /proc/processNameOrID/exe
# kill a process
sudo pkill processNameOrPID
# or
sudo killall processNameOrPID
```

Check /etc Permissions

You want to check `/etc` for incorrect permissions. Any backup files present will be listed, and you want to ensure that only root has read/write permissions.

```
# check permissions
ls -l /etc/*-
```

Check for ShellShock

If the machine is vulnerable, "vulnerable" will be sent back to the terminal. If it does, bash needs to be updated.

```
env x='() { :;}; echo vulnerable' bash -c 'echo hello'
```

Security Audits

Security audits help you find vulnerabilities you may miss manually. There are many tools for this, including Lynis, Root Kit Hunter, Chrootkit, ClamAV, and LinPEAS, to name a few. These can be installed via the command line.

Sources

1. 40 Linux Server Hardening Security Tips 2023 edition
2. 10 Actionable SSH Hardening Tips to Secure Your Linux Server
3. 4 Ways to Disable Root Account in Linux
4. How to set up a UFW firewall on Ubuntu 16.04 LTS server
5. How to Identify the Bad Processes on a Hacked Linux Box
6. How to Set Password Policies in Linux