Student L11A1

This activity focuses on hardening your Linux server to protect it in a basic sense. This focuses on Linux servers in general, whereas there are more steps to hardening your servers if you have different services on them, such as a web server, DNS server, or others.

1. Check Listening Network Ports

   With the help of 'netstat' networking command you can view all open ports and associated programs. Use the command: `netstat -tulpn` to see listening ports

2. Use Secure Shell(SSH)

   Telnet and rlogin protocols uses plain text, not encrypted format which is the security breaches. SSH is a secure protocol that use encryption technology during communication with server.

   Never login directly as root unless necessary. Use `sudo` to execute commands. Users who are allowed to use the sudo command are specified in `/etc/sudoers` The file can also be edited with the `visudo` utility which opens in VI editor.

   It's also recommended to change the default SSH 22 port number to some other higher level port number.

   Open the main SSH configuration file and make add the following parameters to restrict access.

   Open the following file with your favorite text editor: `/etc/ssh/sshd_config`

   To Disable root Login add the following line:

   ```
   PermitRootLogin no
   ```

   To Only allow Specific Users add the following line specifying a username:

```
AllowUsers username
```

To force the Use of SSH Protocol 2 Version add the following line:

```
Protocol 2
```

3. Keep System updated

Always keep your systems updated with latest releases patches, security fixes and kernel when it's available.

Use the following commands:

```
sudo apt-get update

sudo apt-get upgrade
```

NOTE: The update command, which updates the repositories, must be executed before the upgrade command in order to ensure you are getting the latest updates.

4. Remove KDE/GNOME Desktops (if you installed a unix distribution with a desktop interface)

There is no need to run X Window desktops like KDE or GNOME on your dedicated LAMP server. You can remove or disable them to increase security of server and performance.

To disable simple open the file `/etc/inittab` in your favorite editor and set run level to 3. If you wish to remove it completely from the system use the package manager's remove function.

5. Turn Off IPv6

If you're not using a IPv6 protocol, then you should disable it because most of the applications or policies don't require IPv6 protocol and currently it isn't required on the

server. Go to network configuration file and add followings lines to disable it.

To disable IPv6, you have to open `/etc/sysctl.conf` using your favorite text editor and insert the following lines at the end:

net.ipv6.conf.all.disable_ipv6 = 1

net.ipv6.conf.default.disable_ipv6 = 1

net.ipv6.conf.lo.disable_ipv6 = 1

After adding those lines and saving the file, run the command `sudo sysctl -p` and IPv6 should be disabled.

6. Lock and Unlock Account Manually

In a CDC where you are given servers that are already made for you, they may have accounts created on them ahead of time. The lock and unlock features are very useful, instead of removing an account from the system, you can lock it for a certain amount of time. To lock a specific user, you can use the follow command.

```
sudo passwd -l accountName
```

Note : The locked user is still available for root user only. The locking is performed by replacing encrypted password with an (!) string.

To unlock or enable access to an locked account, use the command as shown below. This will remove (!) string with encrypted password.

```
sudo passwd -u accountName
```

7. Change Default Passwords

If you are given accounts on a machine and are unable to disable or remove them because you need them, then you need to make sure to change the default password that they gave it to you with. This can be done with the following command:

```
sudo passwd {username}
```

You will be prompted for a new password. Be careful though, as setting a password in this way will bypass security policies for password selection. (Such as minimum length,

complexity, etc.)

8. Enable Iptables (Firewall)

   If you are not creating, or given, a firewall for your entire network, then it's highly recommended to enable the Linux firewall to secure unauthorized access to your servers. Apply rules in iptables to filter incoming, outgoing and forwarding packets. We can specify the source and destination address to allow and deny in specific udp/tcp port numbers.

9. Disable Ctrl+Alt+Delete in Inittab

   In most Linux distributions, pressing `CTRL-ALT-DELETE` will make your system reboot. So, it's not a good idea to have this option enabled, at least on production servers. Especially since someone could do this by mistake.

   This is defined in the `/etc/init/control-alt-delete.conf` file. If you look closely in that file you will see a line similar to below. By default the line is not commented out. We have to comment it out to disable what it does. To comment it out, add a "#" in front of the line

   ```
   #exec shutdown -r now "Control-Alt-Delete pressed"
   ```

10. Review Logs Regularly

    Move logs to a dedicated log server, this may prevent intruders from easily modifying local logs. Below are the Common Linux default log files name and their usage:

    /var/log/message – Where whole system logs or current activity logs are available.

    /var/log/auth.log – Authentication logs.

    /var/log/kern.log – Kernel logs.

    /var/log/cron.log – Crond logs (cron job).

    /var/log/maillog – Mail server logs.

    /var/log/boot.log – System boot log.

/var/log/mysqld.log – MySQL database server log file.

/var/log/secure – Authentication log.

/var/log/utmp or /var/log/wtmp : Login records file.

11. Backup Important Files

    In a production system, it is necessary to take important files backup and keep them in safety vault, remote site or offsite for Disasters recovery. This can be done in a number of ways. Secure Copy (SCP) or rsync are a couple of common ways.

12. Diable Ping Response

    You can disable responding to ping requests by your server. This can help to secure your server in the event that an attacker tries to ping a range of IP Addresses in order to find a target. However, if you try to ping it to test if it is up and running, it will also ignore your ping request.

    In order to disable ping, open the file `/etc/sysctl.conf` and add the line below:

    ```
    net.ipv4.icmp_echo_ignore_all=1
    ```

    You may need to restart your server for it to take effect. To re-enable responding to pings, either remove that line or just change the 1 to a 0 in that line

These are just some of the basic ways to harden your Linux server. You are encouraged to find more ways, and to research ways to secure specific services that you are going to be using. A good thing to do is to search the internet for vulnerabilities in any programs you use on your servers, as new vulnerabilities can be discovered often. The next activity will focus on basic hardening for Windows servers.
