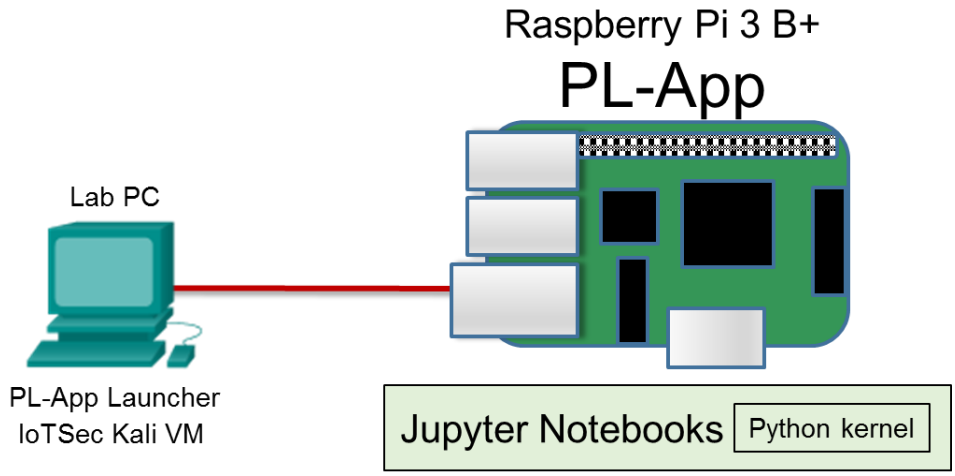
Lab - Harden a Raspberry Pi

Topology



Objectives

Part 1: Securing Remote Access

Part 2: Removing the Default Pi User Account

Part 3: Configuring the Uncomplicated Firewall (UFW)

Background/Scenario

The Raspberry Pi is a doorway to the Internet of Things (IoT). In this lab you will take a Raspberry Pi that is acting as an IoT gateway device and perform device hardening. You will harden the Raspberry Pi by following recommended security practices for the Raspbian OS. You will also limit the network protocols and services allowed to connect to the IoT gateway by activating and configuring the Uncomplicated Firewall (UFW). Finally, you will utilize a separate Kali VM with the Kali Linux OS, acting as a threat actor, to test the security of the IoT gateway.

Required Resources

- Raspberry Pi 3 Model B or later (with PL-App)
- 8GB Micro SD card (minimum required)
- PC with IoTSec Kali VM
- Network connectivity between PC and Raspberry Pi

Part 1: Securing Remote Access

The Raspberry Pi comes with a default user called “pi” with the default password of “raspberry” which is well known. While this makes it easy to use the system, it is not very secure. Anyone with network access to your Pi could login with these widely known credentials. Furthermore, because the SSH server and HTTP server on the Pi are enabled, unknown users on the network could attempt a connection using these default

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

credentials. While basic security would advise the changing of the password for the “pi” user, having a default username alone is a security risk.

Step 1: Access the Raspberry Pi remotely.

a. Connect the Raspberry Pi to the Kali VM host as shown in the topology.

b. Start the Kali VM and login with the username **root** and the password **toor**.

c. On the Kali VM, run the shell script to configure IP addressing. To run the script, at the terminal prompt type the following:

```
root@kali:~# ./lab_support_files/scripts/start_dhcp.sh
[ ok ] Starting isc-dhcp-server (via systemctl): isc-dhcp-server.service.
```

d. Use the PL-App or the command **fping -ag 203.0.113.0/24** in a terminal on the Kali VM to determine the IP address of your Raspberry Pi.

Record the IP address of your Raspberry Pi. _____

e. Every Raspberry Pi has the default username **pi** and password **raspberry**. Use SSH to remotely access the Raspberry Pi. The IP address used in this lab is only used as an example, the IP address for your Raspberry Pi may be different. The IP address for your Raspberry Pi will be in the subnet of 203.0.113.1/24.

Open a terminal on the Kali VM and SSH into your Raspberry Pi using the pi account.

```
root@kali:~# ssh pi@203.0.113.12
```

If you were unsuccessful, you can start the service from the PL-App terminal through the web interface:

```
(pl-app) root@myPi:~# systemctl enable ssh
(pl-app) root@myPi:~# systemctl start ssh
```

The **enable** command allows the service to restart automatically after the system reloads. The **start** command only starts the service during this session, and the service will no longer be available after a reboot without starting it again manually.

For more information about enabling SSH, navigate to https://www.raspberrypi.org/documentation/remote-access/ssh/.

Step 2: Securing the user accounts

As we know, while basic security would advise the changing of the password for the **pi** user, just having a default username is a security risk. Instead, you will create a new user with sudo permissions. After the new user has the appropriate permissions, the default **pi** user can be deleted.

Note: The name of the Raspberry Pi that appears in the terminal will differ depending on the device name that was configured in PL-App launcher when the SD card was made.

a. Add a new user to the Raspberry Pi with the command **sudo adduser kingbob** in the terminal. Choose a secure password to use and provide any user information desired.

```
pi@IoTpi:~ $ sudo adduser kingbob
```

b. Give the kingbob user sudo permissions by adding it to the sudo group with the command **sudo adduser kingbob sudo**.

```
pi@IoTpi:~ $ sudo adduser kingbob sudo
Adding user `kingbob' to group `sudo' ...
Adding user kingbob to group sudo
Done.
```

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

c. To stop the Raspberry Pi from logging in to a user account automatically at boot, issue the **sudo raspi-config** terminal command.

```
pi@IoTpi:~ $ sudo raspi-config
```

d. Select **Boot Options** to configure options for start-up.

e. Select **Desktop / CLI** to choose whether to boot into a desktop environment or the command line.

f. Select **Console** to require users to log into text console.

g. Select **Finish** and click **Yes** to reboot when prompted.

h. After reboot, use the Kali VM to SSH into the **kingbob** account of the IoTpi.

```
root@kali:~# ssh kingbob@203.0.113.12
```

Step 3: Secure remote access.

The general implementation of SSH as the method for remote access in itself does not provide strong security. The default installation for SSH uses a single password, commonly with a default value. To implement stronger security when deploying an SSH server, a username and password combination should be implemented. This step includes the activation of the SSH service and restriction of SSH authentication attempts.

Note: The usernames are for example only. Choose your desired usernames. Create a new user account for specifically utilizing remote access connections.

a. You will create kevin, a standard user, with the command **sudo adduser kevin**. Provide the necessary information when prompted.

```
kingbob@IoTpi:~ $ sudo adduser kevin

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for kingbob:
```

b. Finally, you will edit the **sshd_config** file located in the **/etc/ssh** directory to limit the users that are allowed to access this device using SSH.

Edit the **/etc/ssh/sshd_config** file by adding the following lines to the end of the file:

```
AllowUsers kingbob
DenyUsers kevin
```

Edit the file by using a text editor, such as nano.

```
kingbob@IoTpi:~ $ sudo nano /etc/ssh/sshd_config
```

Predict the success of SSH access for the users **kingbob** and **kevin.** Record your predictions below.

_____

_____

_____

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

c. To force the SSH settings to take effect now, restart the SSH service using **sudo systemctl restart ssh**.

```
kingbob@IoTpi:~ $ sudo systemctl restart ssh
```

d. Verify that the **kevin** user account cannot be exploited by a threat actor via the SSH service. Issue the command **ssh kevin@203.0.113.12** to attempt an SSH connection with the **kevin** username.

```
root@kali:~# ssh kevin@203.0.113.12
```

e. After issuing the password for the **kevin** user account the IoT gateway reports SSH access is denied.

```
kevin@203.0.113.12's password:
Permission denied, please try again.
```

f. SSH into the Raspberry Pi using the **kingbob** account, which is permitted to access the IoTpi via SSH.

```
root@kali:~# ssh kingbob@203.0.113.12
```

Part 2: Removing the Default Pi User Account

Step 1: Open a terminal and remove the Pi account but leave the directory.

a. To remove the **pi** account from the IoT gateway, make sure you are logged into the “kingbob” account. Then, enter the **sudo deluser pi** command in the terminal.

```
kingbob@IoTpi:~ $ sudo deluser pi
Removing user `pi' ...
Warning: group `pi' has no more members.
Done.
```

Step 2: Require a password with the command sudo.

The Raspbian OS does not require a password when placing sudo in front of a command to run it as a superuser, by default. If your IoT gateway is exposed to the Internet and somehow becomes exploited (perhaps via a webpage exploit for example), the attacker will be able to change items that require superuser rights, unless you have set sudo to require a password.

a. To force sudo to require a password, you will edit the file **/etc/sudoers.d/010_pi-nopasswd**.

```
kingbob@IoTpi:~ $ sudo nano /etc/sudoers.d/010_pi-nopasswd
pi ALL=(ALL) NOPASSWD: ALL
```

List all the users who can access superuser privileges without inputting their password.

_____

b. Replace the pi entry with your username, kingbob, with superuser rights.

```
kingbob ALL=(ALL) PASSWD: ALL
```

c. Now save the file and reboot.

```
kingbob@IoTpi:~ $ sudo reboot
```

How would you verify that the pi user account has been deleted?

_____

_____

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

Part 3: Configuring the Uncomplicated Firewall (UFW)

Uncomplicated Firewall (UFW) provides a more user-friendly framework than the traditional iptables firewall. Iptables has handled the security for Linux for a long time but UFW provides a much simpler interface with a good amount of power. The UFW is already installed on the IoT gateway Pi, but it needs to be activated and configured to limit dangerous protocols and services, while allowing important connections to be available.

You will also use network mapper (Nmap) to rapidly scan the network. Nmap is a network scanning tool that allows you to discover network hosts and resources, including services, ports, operating systems, and other fingerprinting information. Nmap **should not** be used to scan networks without prior permission. The act of network scanning can be considered a form of network attack.

Nmap will test the firewall port/service restriction and IPS capabilities of the IoT gateway. You will run the scanning program from the Kali Linux VM and attempt to scan open ports on the IoT gateway before and after viewing the Uncomplicated Firewall (UFW) rules.

Step 1: Identify what network services are listening on the Pi.

a. Connect to the IoT gateway Pi over SSH as kingbob.

b. Start the Telnet and vsftpd servers. These services are started in this lab for demonstration purposes. They are only available until the Raspberry Pi is reloaded.

```
kingbob@IoTpi:~ $ sudo systemctl start openbsd-inetd.service
kingbob@IoTpi:~ $ sudo systemctl start vsftpd
```

c. Open a new terminal on the Kali VM and perform an Nmap scan targeting the TCP ports of the Pi. This scan checks all TCP ports in the range of 1-65535.

```
root@kali:~# nmap -p 1-65535 203.0.113.12

Starting Nmap 7.60 ( https://nmap.org ) at 2018-05-04 13:59 EDT
Nmap scan report for 203.0.113.12
Host is up (0.00043s latency).
Not shown: 65532 closed ports
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
23/tcp open  telnet
80/tcp open  http
MAC Address: B8:27:EB:AC:8E:B2 (Raspberry Pi Foundation)
Nmap done: 1 IP address (1 host up) scanned in 17.02 seconds
```

d. On the Pi, the currently open TCP sessions can be viewed by using **ss -t**:

```
kingbob@IoTpi:~ $ ss -t
State       Recv-Q Send-Q Local Address:Port               Peer Address:Port

ESTAB       0      168     203.0.113.12:ssh               203.0.113.11:34730
```

Step 2: Check the status of the UFW.

a. SSH into the kingbob account, which is the only account allowed to access the Pi.

```
root@kali:~# ssh kingbob@203.0.113.12
```

b. Enter **sudo ufw status** to view the current state of the UFW firewall.

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

```
kingbob@IoTpi:~ $ sudo ufw status
Status: inactive
```

c. If you'd like to specifically view rules added to the UFW, enter **sudo ufw show added**.

```
kingbob@IoTpi:~ $ sudo ufw show added
Added user rules (see 'ufw status' for running firewall):
(None)
```

Step 3: Configure firewall rules for the Uncomplicated Firewall (UFW).

a. At this time there are no rules configured in the UFW. However, when the UFW service is started, the it includes a hidden implicit deny rule to block any traffic not explicitly allowed. While this means all traffic is dropped by default, we'd like to ensure that specific insecure protocols are denied regardless of future configured allow statements.

b. Configure a statement to explicitly deny Web traffic

```
kingbob@IoTpi:~ $ sudo ufw deny http
Rules updated
Rules updated (v6)
```

c. Configure a statement to explicitly deny telnet traffic and FTP.

```
kingbob@IoTpi:~ $ sudo ufw deny telnet
Rules updated
Rules updated (v6)
```

d. Configure a statement to explicitly deny ftp traffic

```
kingbob@IoTpi:~ $ sudo ufw deny ftp
Rules updated
Rules updated (v6)
```

Step 4: View the configured firewall rules for the UFW.

a. View the rules added to the UFW, enter **sudo ufw show added**

```
kingbob@IoTpi:~ $ sudo ufw show added
Added user rules (see 'ufw status' for running firewall):
ufw deny 80/tcp
ufw deny 23/tcp
ufw deny 21/tcp
```

b. At this point, we are NOT ready to activate (enable) the UFW service. Because the UFW includes an implicit deny rule for any traffic not explicitly permitted, we could have our SSH session terminated and be locked out of the Pi.

c. Configure a statement to explicitly permit secure remote access (SSH).

```
kingbob@IoTpi:~ $ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

d. To activate the UFW firewall, which is disabled by default, enter **sudo ufw enable**. Ensure that you have added the allow ssh statement to the UFW before activating the UFW firewall.

```
kingbob@IoTpi:~ $ sudo ufw enable
```

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

```
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
```

e. Observe the UFW firewall rules.

You should see the UFW rules that you have created using the **sudo ufw status** command.

```
kingbob@IoTpi:~ $ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
80/tcp                     DENY        Anywhere
23/tcp                     DENY        Anywhere
21/tcp                     DENY        Anywhere
22/tcp                     ALLOW       Anywhere
80/tcp (v6)                DENY        Anywhere (v6)
23/tcp (v6)                DENY        Anywhere (v6)
21/tcp (v6)                DENY        Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)
```

The Allow statements are showing the permitted connections available.

Step 5: Decrease the threat surface of SSH.

At this time, even though the only accessible service is SSH, SSH can become vulnerable to a brute force user/password attack. To prevent or better mitigate this threat, a limitation on the number of login attempts in a specific period of time should be applied.

a. Limit the login attempts for the SSH service using the **sudo ufw limit ssh/tcp** command.

```
kingbob@IoTpi:~ $ sudo ufw limit ssh/tcp
Rules updated
Rules updated (v6)
```

Note: The UFW limit command denies future connections if the same IP address has attempted to connect six or more times in the last 30 seconds. (This acts as a basic intrusion prevention system (IPS) for SSH to prevent brute force attacks.)

b. View the UFW rules again using **sudo ufw status**. What action is now in place for SSH connections destined for the Pi?

_____

_____

c. According to the UFW rules, if a permit HTTPS rule is NOT created, what will happen to HTTPS traffic that arrives at the IoTpi interface?

_____

_____

Step 6: Run Nmap and set scanning options.

a. Open a terminal on the Kali Linux VM.

b. Perform another Nmap scan targeting the TCP ports of the Pi.

```
root@kali:~# nmap -p 1-65535 203.0.113.12
```

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

```
Starting Nmap 7.60 ( https://nmap.org ) at 2018-05-04 14:12 EDT
Nmap scan report for 203.0.113.12
Host is up (0.00087s latency).
Not shown: 65534 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: B8:27:EB:AC:8E:B2 (Raspberry Pi Foundation)

Nmap done: 1 IP address (1 host up) scanned in 121.88 seconds
```

c. The Uncomplicated Firewall (UFW) on the Pi is preventing the scan from accessing the denied services.

Compare the above output with the previous Nmap scan. What protocols are no longer reachable on the Pi from the Kali VM?

_____

_____

How many open ports did Nmap find on the IoT Gateway?

_____

_____

What are the associated port numbers and services?

_____

_____

d. The Telnet and vsftp services could also be stopped so these services are no longer enabled.

```
kingbob@IoTpi:~ $ sudo systemctl stop openbsd-inetd.service
kingbob@IoTpi:~ $ sudo systemctl stop vsftpd
```

Part 4: Cleanup

It is very important to harden any IoT gateway device to protect against cybersecurity incidents. However, for classroom purposes, the Raspberry Pi device should be returned to its original configuration so that other users can access it as required for other courses.

Step 1: Restore the default user account.

The default user account **pi** is restored with all the default privileges.

a. Add the user account **pi** with superuser privileges. Use the default password **raspberry** for the user **pi**.

```
kingbob@IoTpi:~ $ sudo adduser pi
Adding user `pi' ...
Adding new group `pi' (1000) ...
Adding new user `pi' (1000) with group `pi' ...
The home directory `/home/pi' already exists.  Not copying from `/etc/skel'.
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for pi
```

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

```
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
kingbob@IoTpi:~ $ sudo adduser pi sudo
```

b. Restore the SSH access for all the users by deleting the last two lines in the file **/etc/ssh/sshd_config**. Delete the lines: **AllowUsers kingbob**, **DenyUsers kevin**. Save the file.

```
kingbob@IoTpi:~ $ sudo nano /etc/ssh/sshd_config
```

c. Edit the file **/etc/sudoers.d/010_pi-nopasswd** is returned to the default state. Change **kingbob** to **pi**.

```
kingbob@IoTpi:~ $ sudo nano /etc/sudoers.d/010_pi-nopasswd
pi ALL=(ALL) NOPASSWD: ALL
```

Step 2: Disable UFW and restore defaults.

In this step, you will disable and reset UFW.

a. Enter the command **sudo ufw reset** to disable and UFW.

```
kingbob@IoTpi:~ $ sudo ufw reset
[sudo] password for kingbob:
Resetting all rules to installed defaults. This may disrupt existing ssh
connections. Proceed with operation (y|n)? y
Backing up 'after6.rules' to '/etc/ufw/after6.rules.20181127_000859'
Backing up 'user6.rules' to '/etc/ufw/user6.rules.20181127_000859'
Backing up 'before6.rules' to '/etc/ufw/before6.rules.20181127_000859'
Backing up 'after.rules' to '/etc/ufw/after.rules.20181127_000859'
Backing up 'before.rules' to '/etc/ufw/before.rules.20181127_000859'
Backing up 'user.rules' to '/etc/ufw/user.rules.20181127_000859'
```

b. Verify the status of UFW is inactive.

```
kingbob@IoTpi:~ $ sudo ufw status
```

c. Before you reboot the Raspberry Pi device, verify that the user **pi** can log into the device via SSH. Restart the SSH service and log in as **pi**.

```
kingbob@IoTpi:~ $ sudo systemctl restart ssh
kingbob@IoTpi:~ $ ssh pi@localhost
```

If you are unable to log into the Pi device using the user **pi**, verify that you have configured the user account and SSH configurations correctly from the previous steps.

d. Reboot the Raspberry Pi device.

Step 3: Remove unnecessary users.

In this step, you will remove the unnecessary users and their home directories.

a. Log into the Raspberry Pi device as the default user **pi**.

b. Remove the users **kevin** and **kingbob** and their home directories.

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public

```
pi@IoTpi:~ $ sudo deluser --remove-home kevin
Looking for files to backup/remove ...
Removing files ...
Removing user `kevin' ...
Warning: group `kevin' has no more members.
Done.
pi@IoTpi:~ $ sudo deluser --remove-home kingbob
Looking for files to backup/remove ...
Removing files ...
Removing user `kingbob' ...
Warning: group `kingbob' has no more members.
Done.
```

Reflection

1. What are some common ports and services that may exist on an IoT device and should be locked down that were not mentioned in this lab?

   _____

   _____

   _____

2. What type of device could be used to recognize an Nmap scan and actively prevent the port scan and host sweep?

   _____

   _____

   _____

© 2018 - 2019 Cisco and/or its affiliates. All rights reserved. Cisco Public