IoT Penetration Testing Methodology

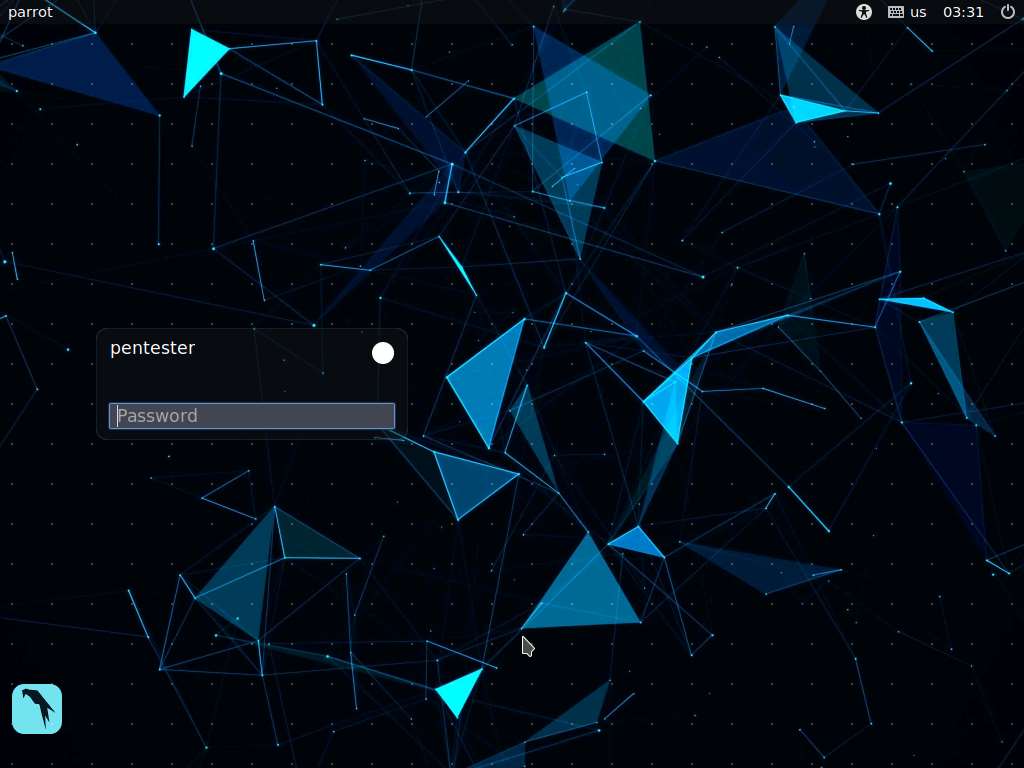
59 Minutes Remaining

Instructions Resources Help  100%

Exercise 1: IOT Firmware Acquisition, Extraction, Analysis and Emulation

**Lab Duration**: **10** Minutes

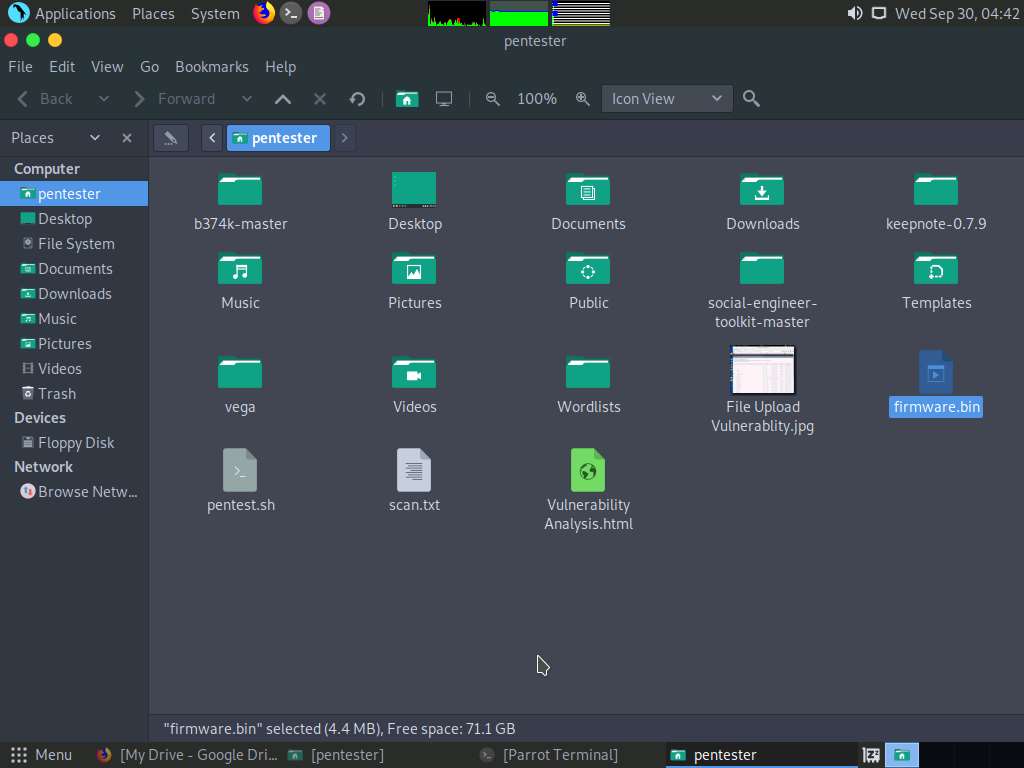
1. By default Parrot machine appears. In the Password field type **toor** and press **Enter** to login.



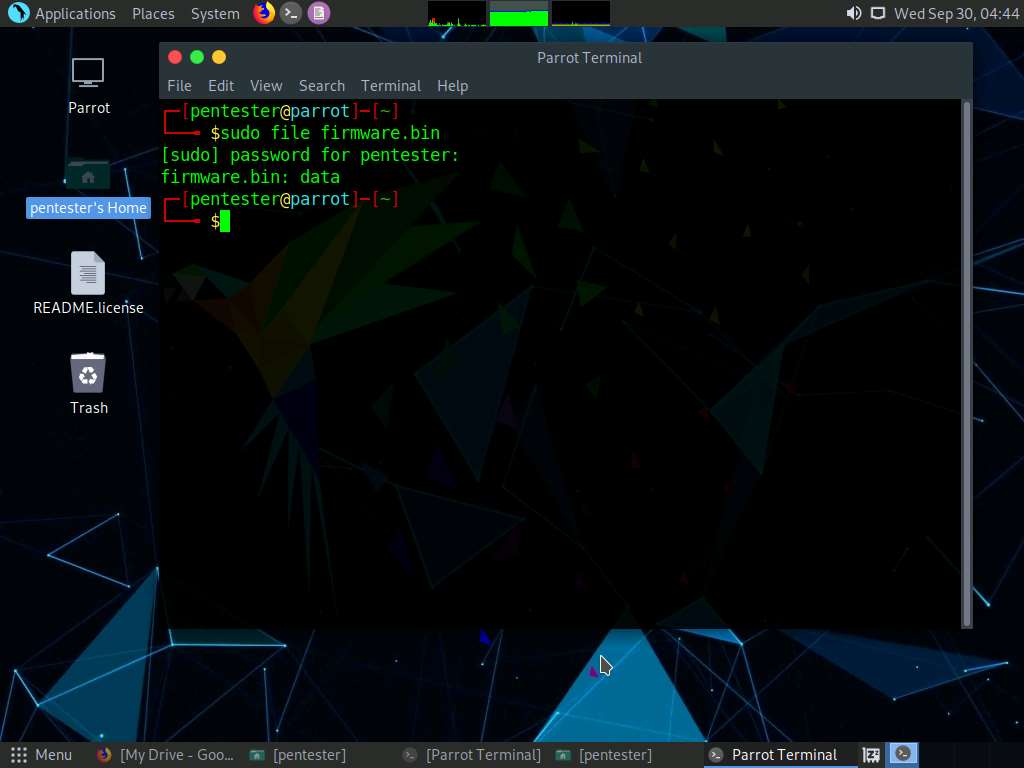
1. We have a sample firmware package binary for a **Netgear IOT** device that is often used for practicing IOT hacking. The device model number is **WNAP 320**, which is a wireless access point for a network. An example of the device is shown in the following image.



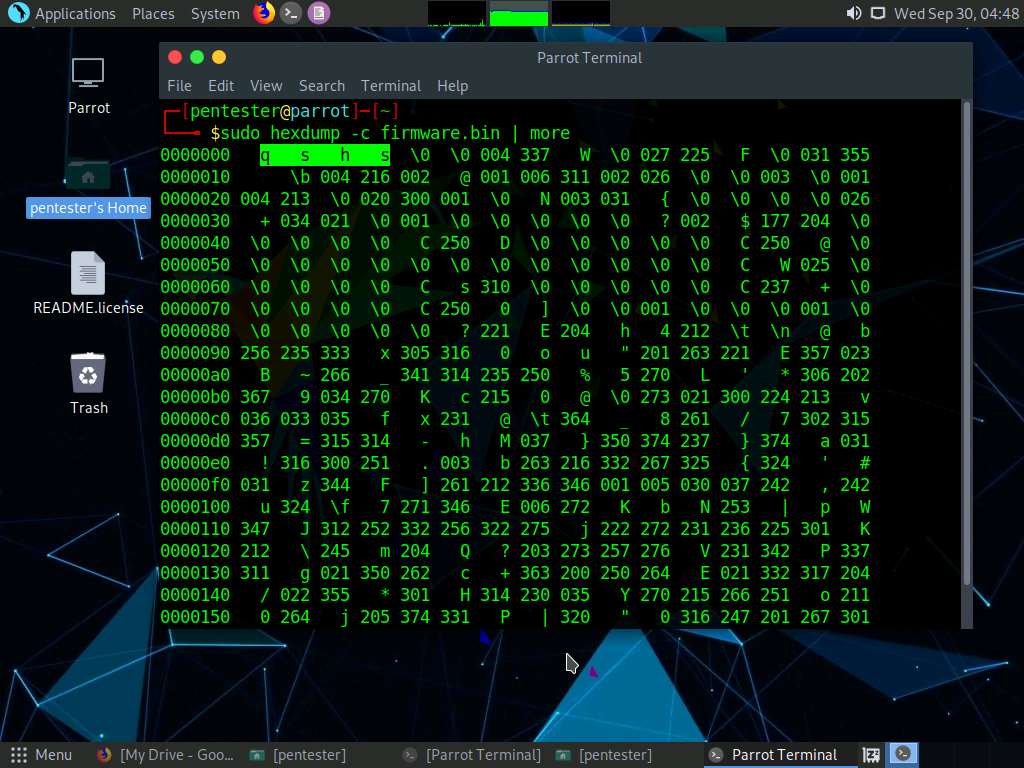
1. We have downloaded the firmware for you. It is located in the **home** folder and named **firmware.bin**.



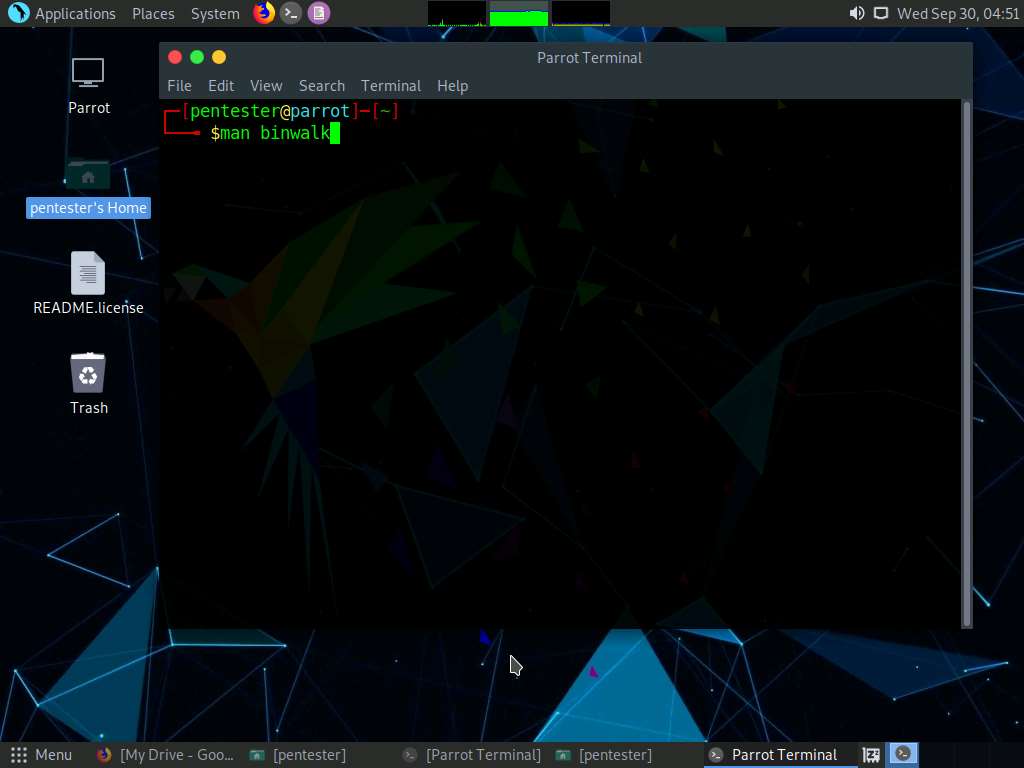
1. The first step when we have a file is to run the file command on it. Launch a Terminal and type **sudo file firmware.bin** and press **Enter**. Type **toor** when prompted for the password and press **Enter**.

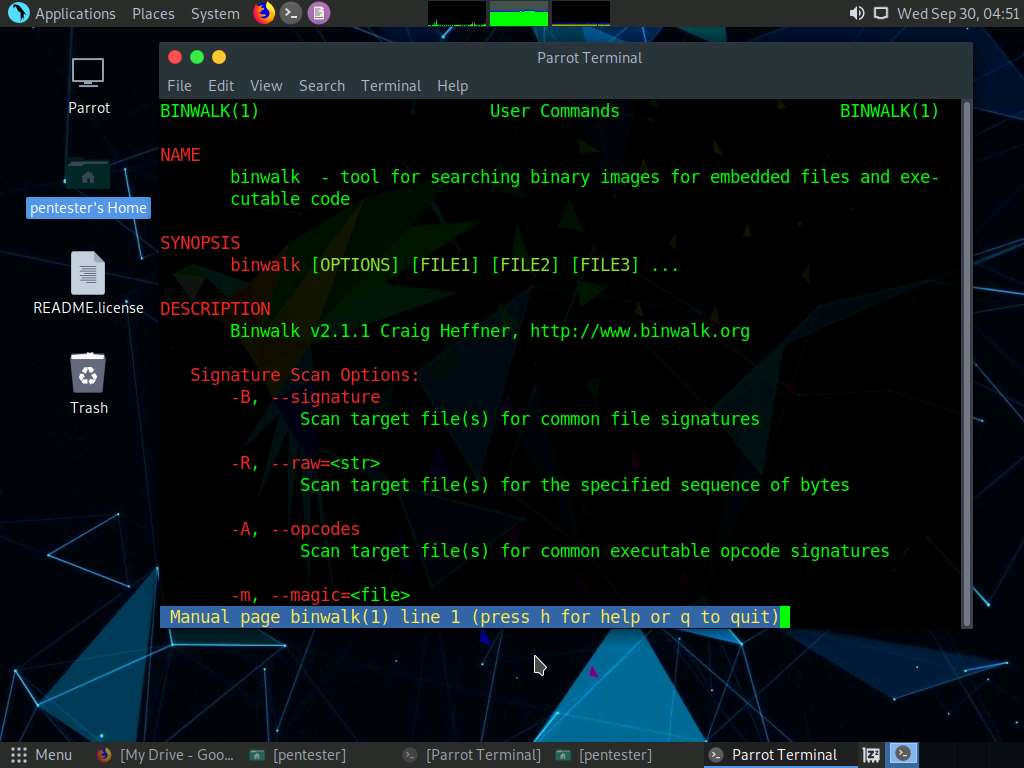


1. There is nothing useful for us here, so we continue trying. Type **sudo hexdump -c firmware.bin | more** and press **Enter**.
2. Now we have **qshs** that is identified in the hex trace. It is a known signature of the **Squashfs** file system, so this is a start.

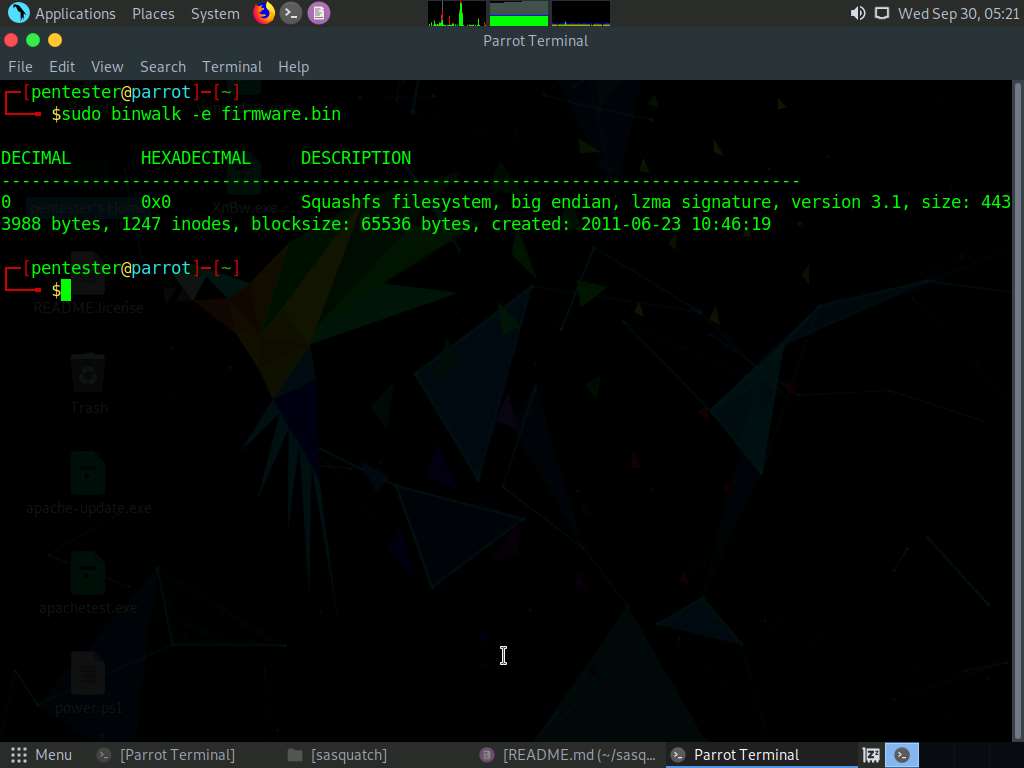


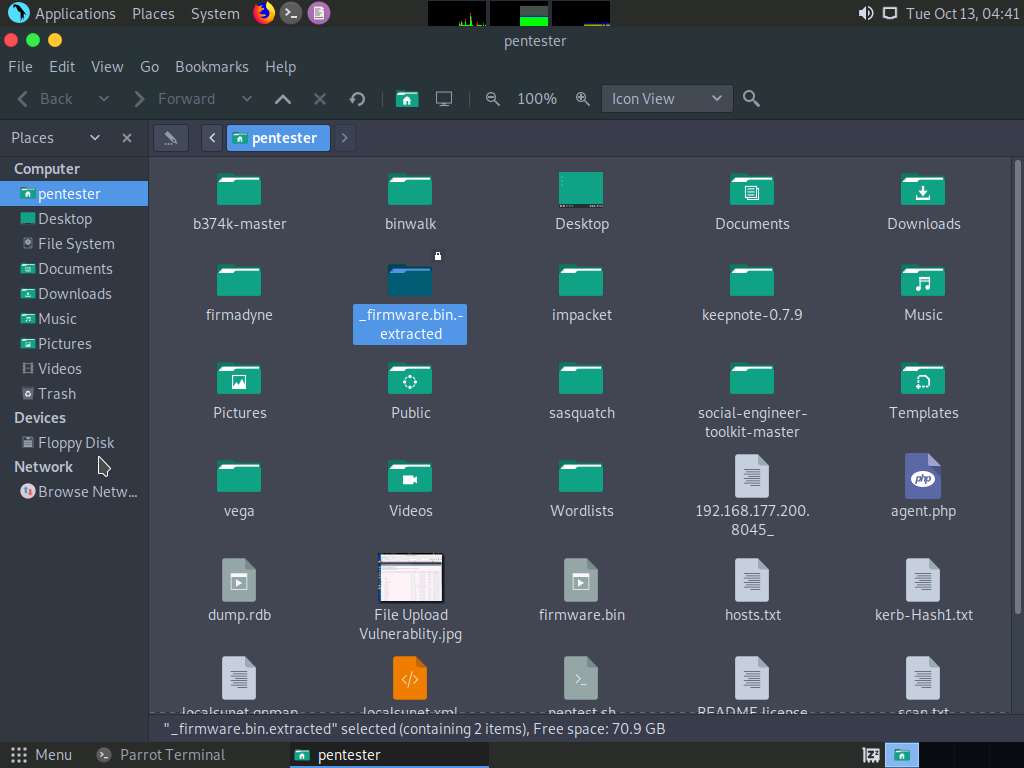
1. The next step in the process is to try and extract the file system from this firmware image. Our preferred tool is **binwalk**, so we will try this. In the terminal window, type **man binwalk** and press **Enter**. Take a few minutes to read about the tool. Once you are done, type **q** to exit the man page.



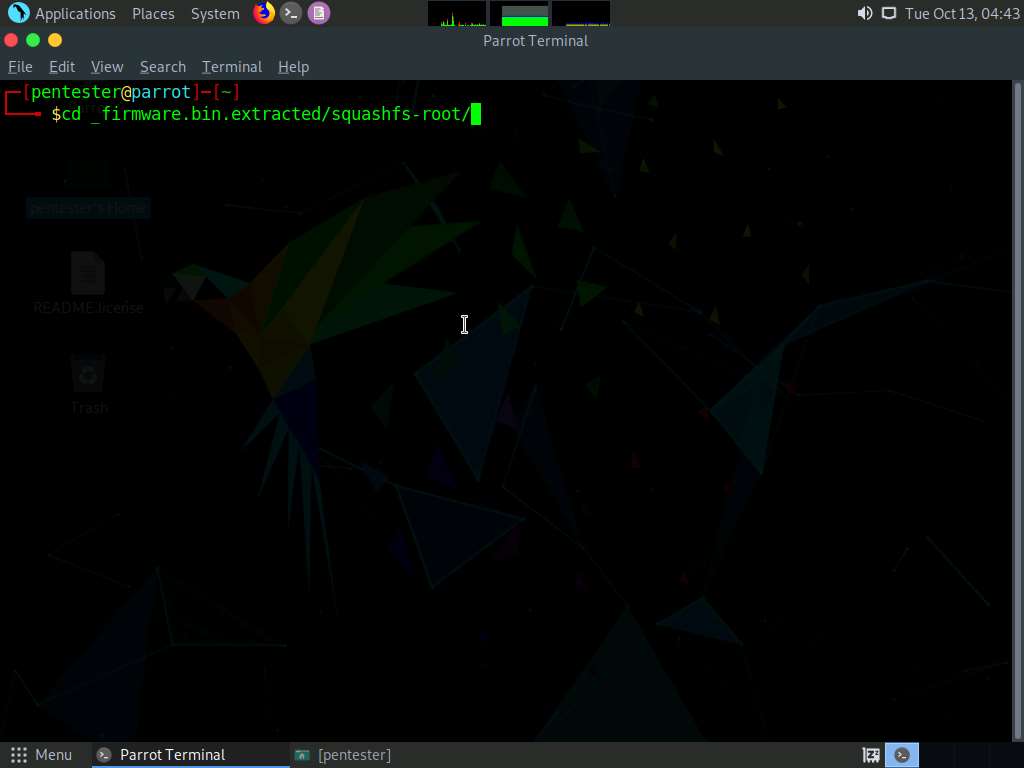


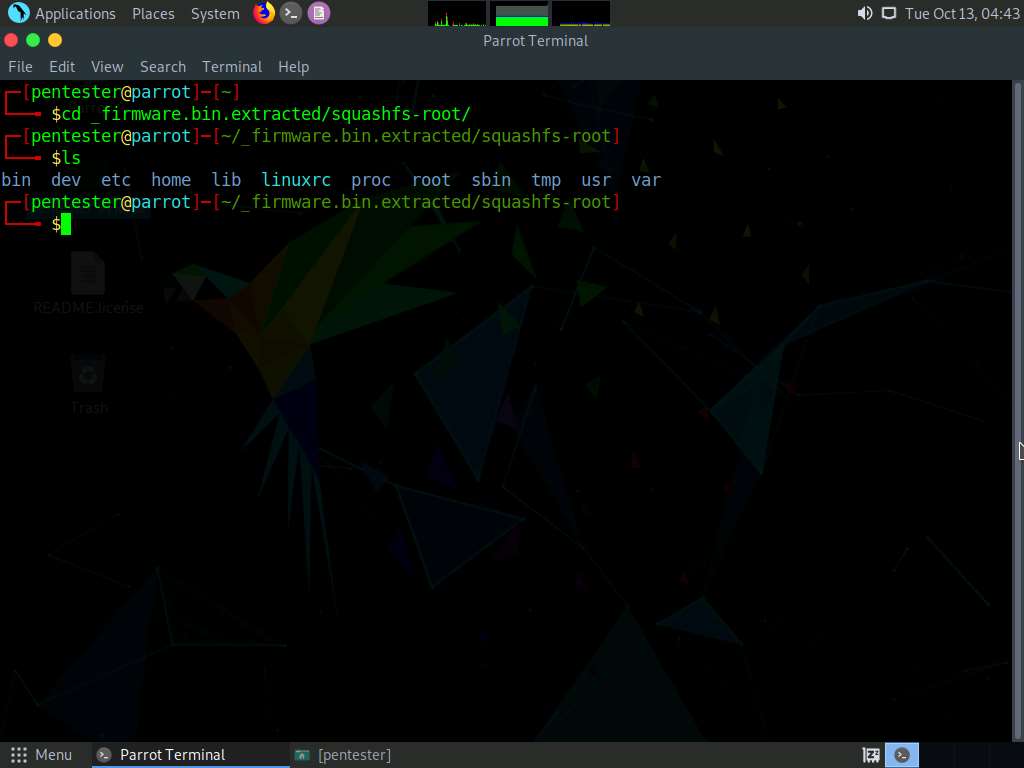
1. In the terminal type **sudo binwalk -e firmware.bin** and press **Enter**. However, the main thing is the message that the image has been extracted. We also see that the filesystem is confirmed as Squashfs. Once the command is completed, you will have a directory that represents the extracted filesystem in the **Home** folder.



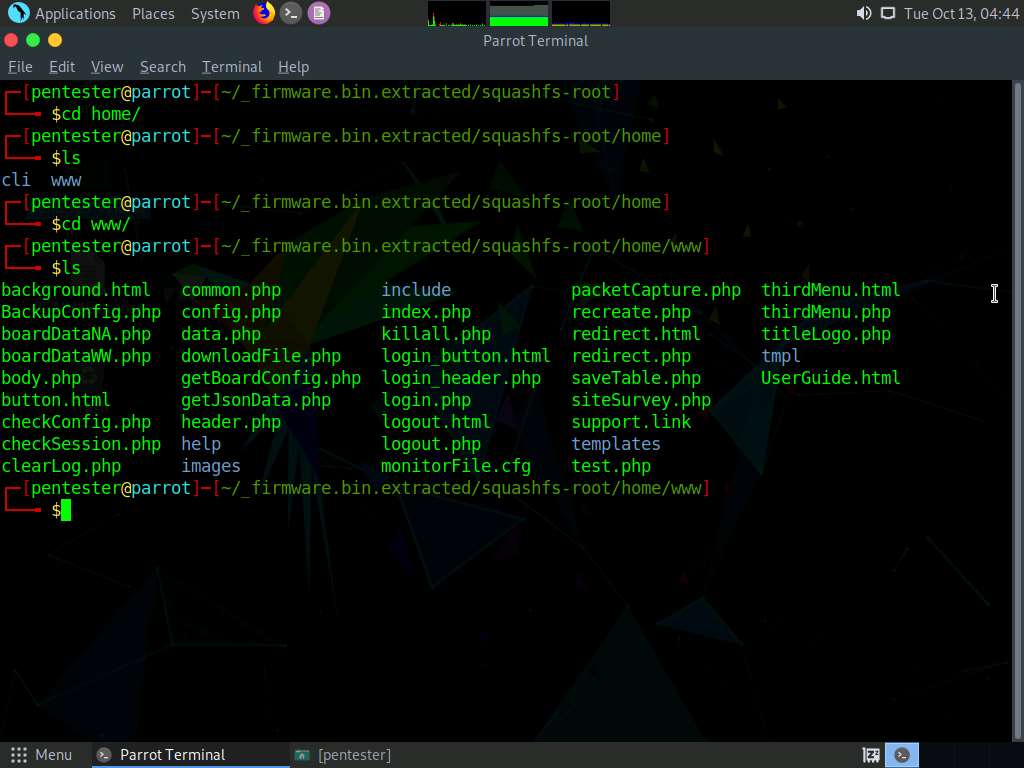


1. In the terminal type **cd \_firmware.bin.extracted/squashfs-root** and press **Enter**. Once you are in the directory type **ls** and press **Enter**. You will see that the extracted file system is in a squashfs-root directory. We now have a complete and extracted file system, which as you can see, is a Linux variant.

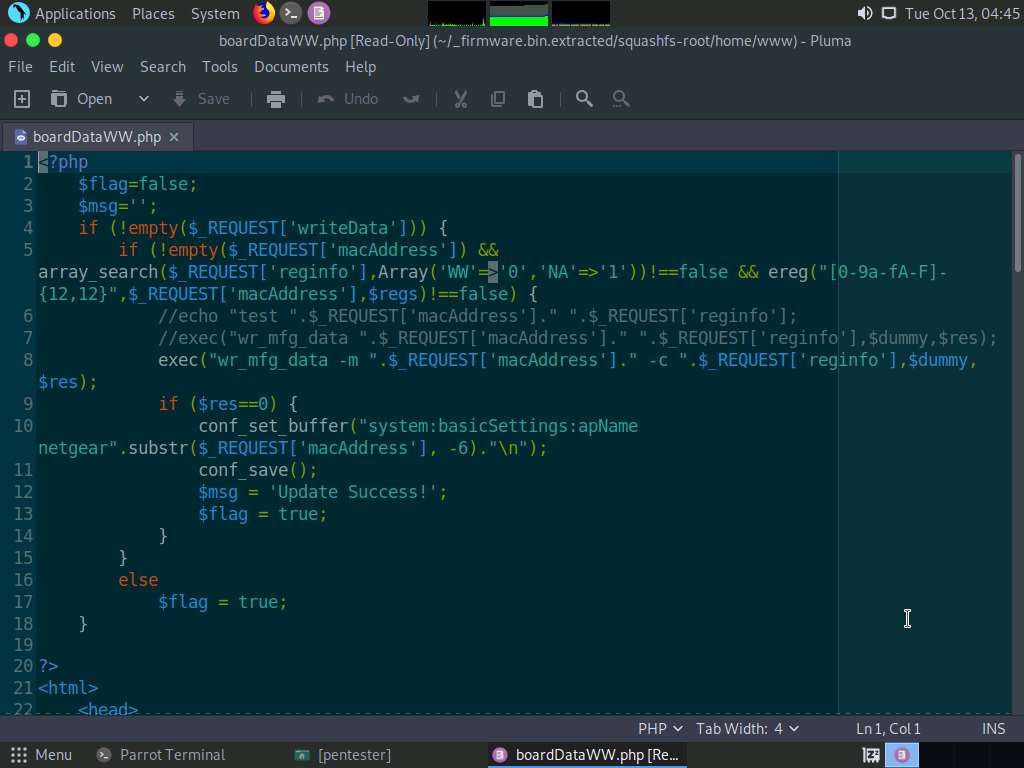




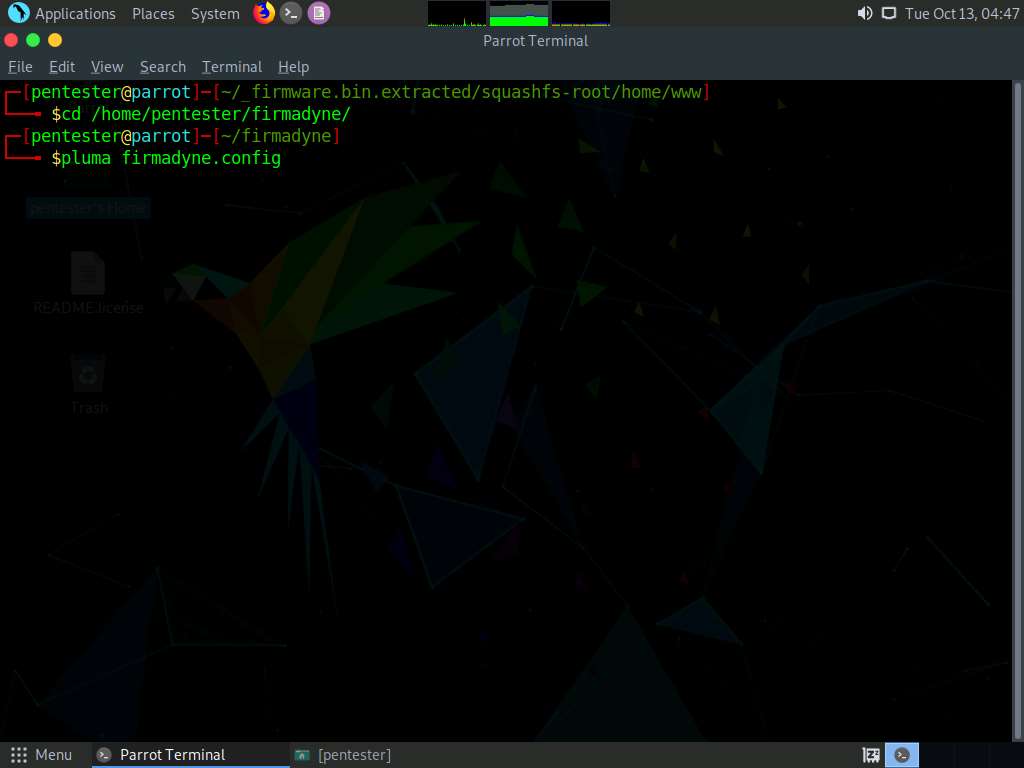
1. From here, you can explore it to see the contents. A good place to start is the home directory since it usually contains information about the installed applications. Type **cd home** and press **Enter**. Type **ls** and press **Enter** to review the results. Since we have a **www** folder, this can be of interest. Type **cd www** and **Enter**, and then type **ls** and press **Enter** to view the file contents.

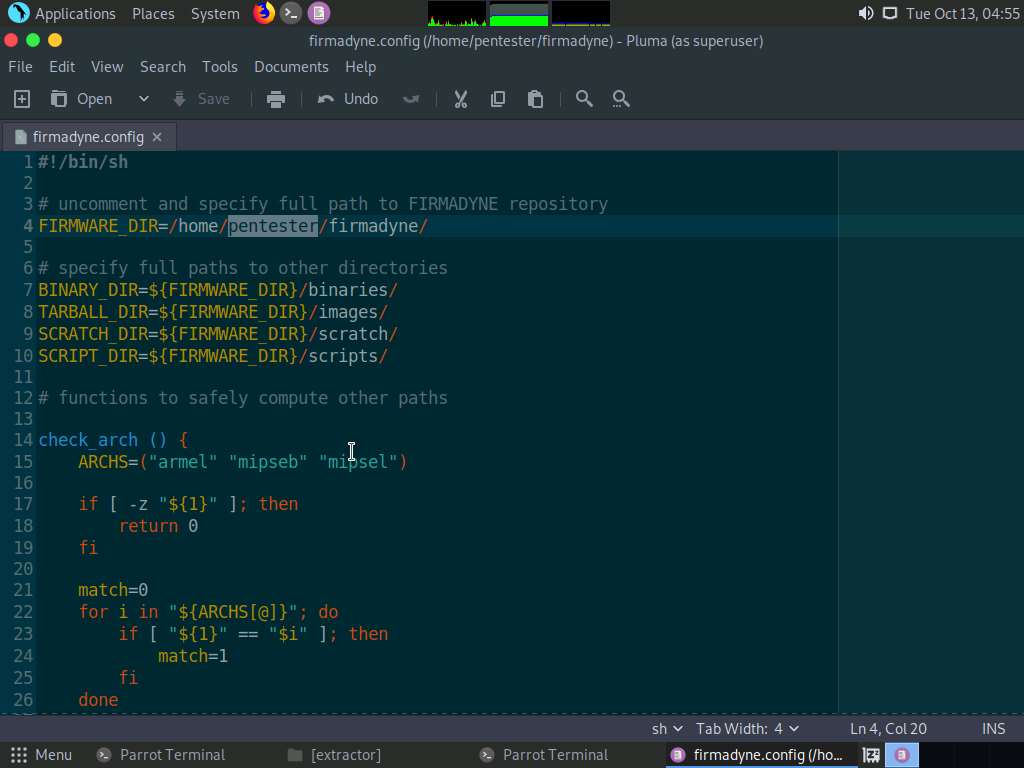


1. We have a lot of code here to examine. In most cases, this code is not written securely. The file we want to look at is the **boardDataWW.php**. Open this in the editor of your choice.
2. As you review the code, it is apparent that there is nothing in the way of input validation and boundary checking. This is a clear indication that this application has injection vulnerabilities. Minimize or close the text editor window. There are many files here for you to explore, but we will leave that to you. Next, we will work on the challenges of mounting and booting the firmware with an emulator.

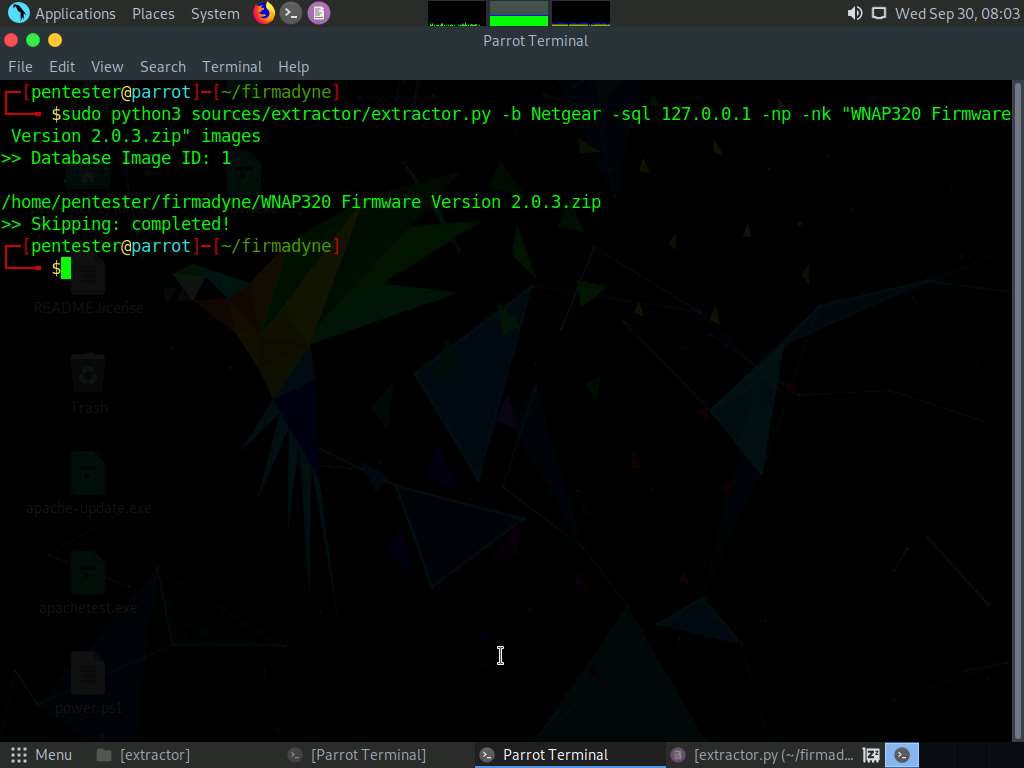


1. The one configuration we want to make is point the **firmadyne** directory to the correct address. Type **cd firmadyne** and press **Enter** and open the config file **firmadyne.config** in the text editor of your choice and set **FIRMWARE\_DIR** to point to the location of the tool, and **uncomment** the line. Save and close the text editor.

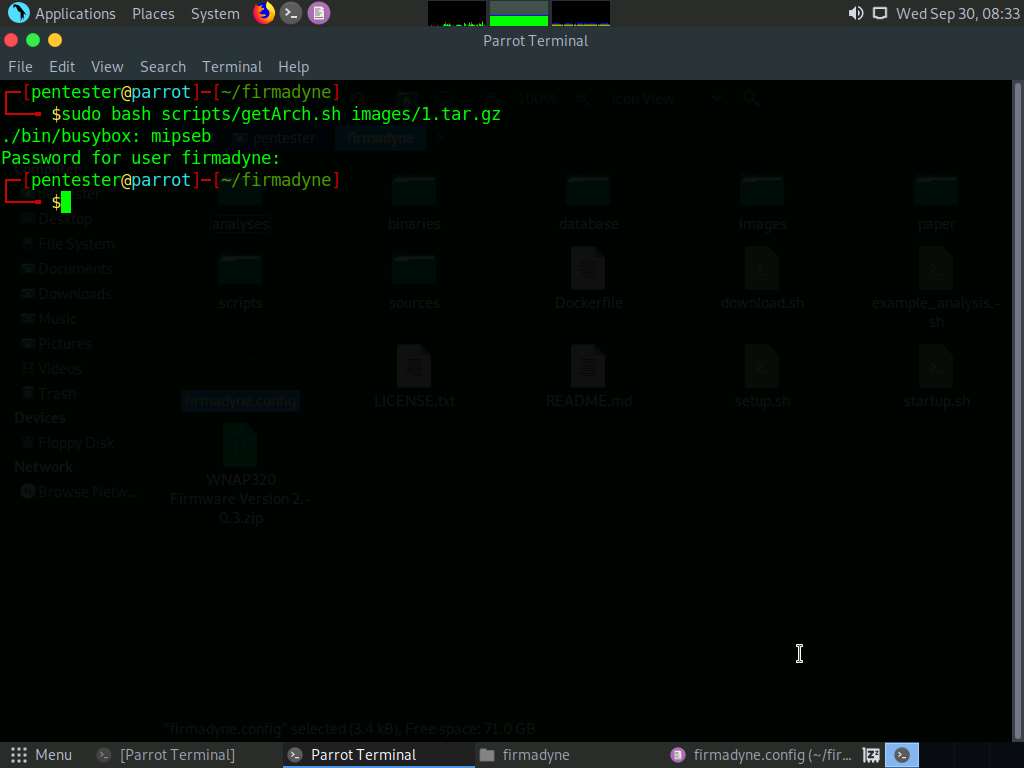




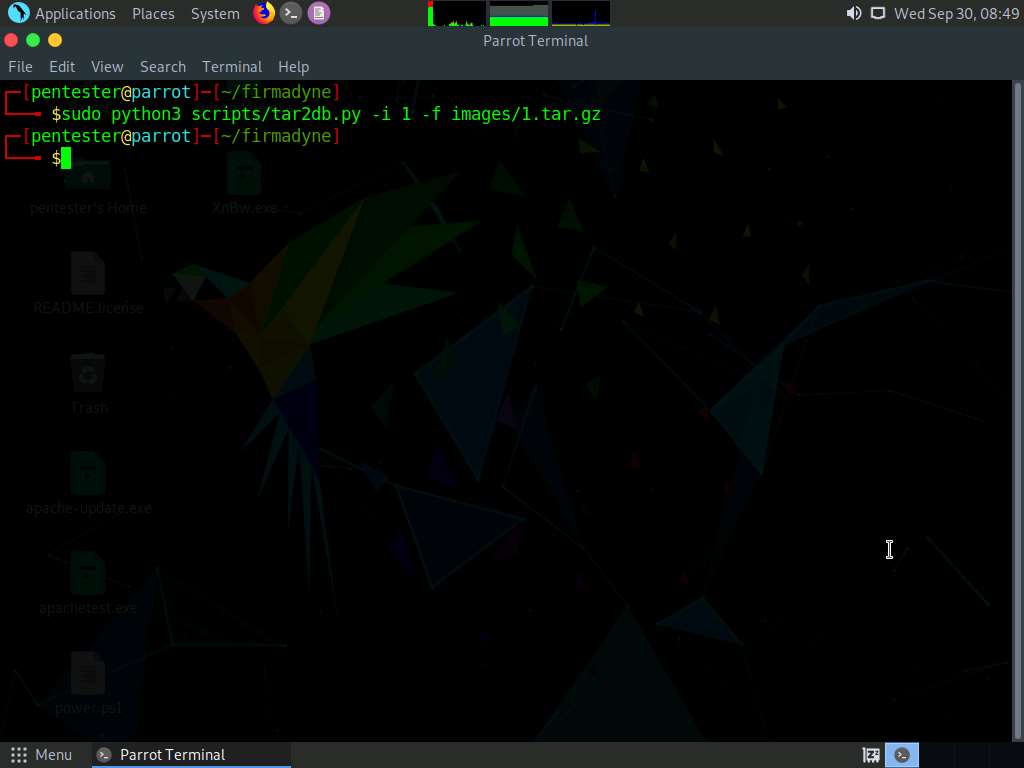
1. We will work with the compressed firmware version straight from the download. This is the same file as **firmware.bin**. Ensure that you are in the firmadyne directory.
2. Use the extractor to recover only the filesystem. Ensure that the following are set: no kernel (-nk), no parallel operation (-np), populating the image table in the SQL server at 127.0.0.1 (-sql) with the Netgear brand (-b), and storing the tarball in images. Type **sudo python3 sources/extractor/extractor.py -b Netgear -sql 127.0.0.1 -np -nk "WNAP320 Firmware Version 2.0.3.zip" images** and press **Enter**.



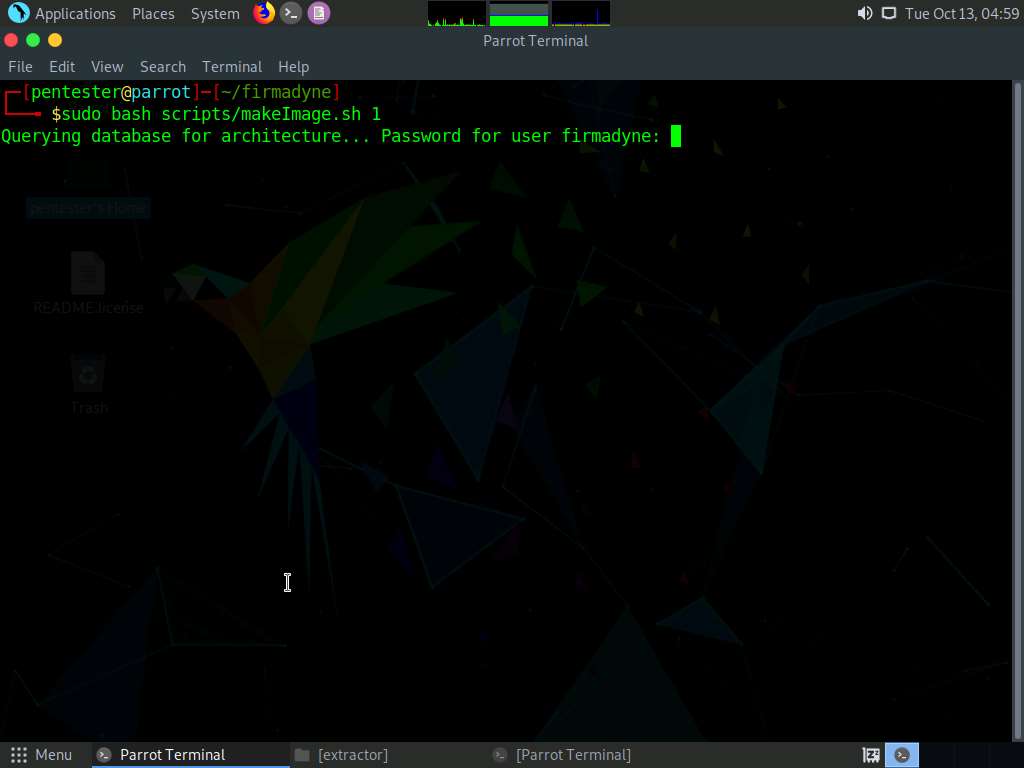
1. If it works correctly, you will have a tarball in the image directory. Next, enter **sudo bash scripts/getArch.sh images/1.tar.gz**. Enter the password of firmadyne which is **toor** when prompted.

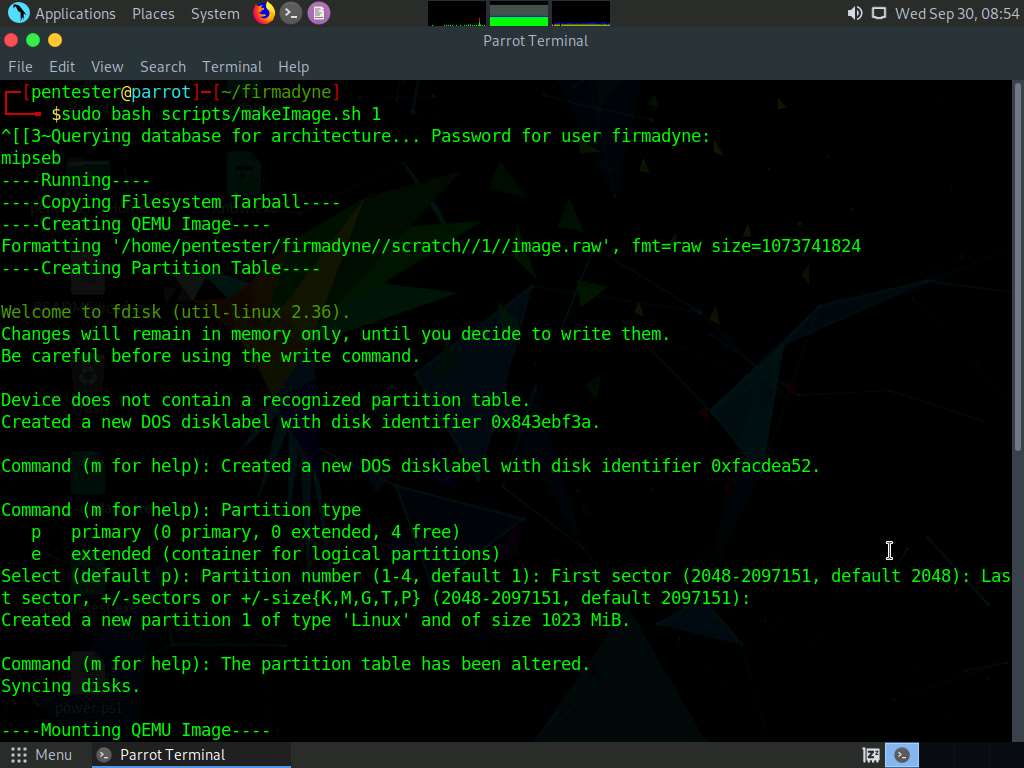


1. This command is used to identify the architecture of firmware **1** and store the result in the image table of the database.
2. Now we want to load the contents of the filesystem for firmware **1** into the database and populate the object and object\_to\_image tables. Enter **sudo python3 scripts/tar2db.py -i 1 -f images/1.tar.gz**.

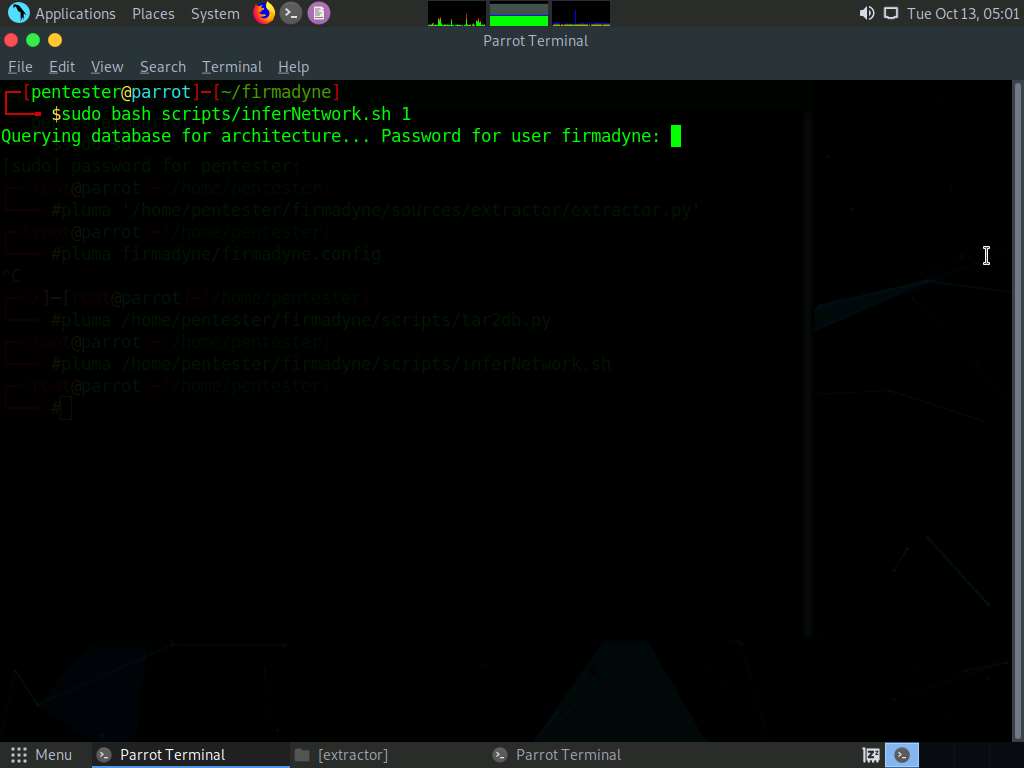


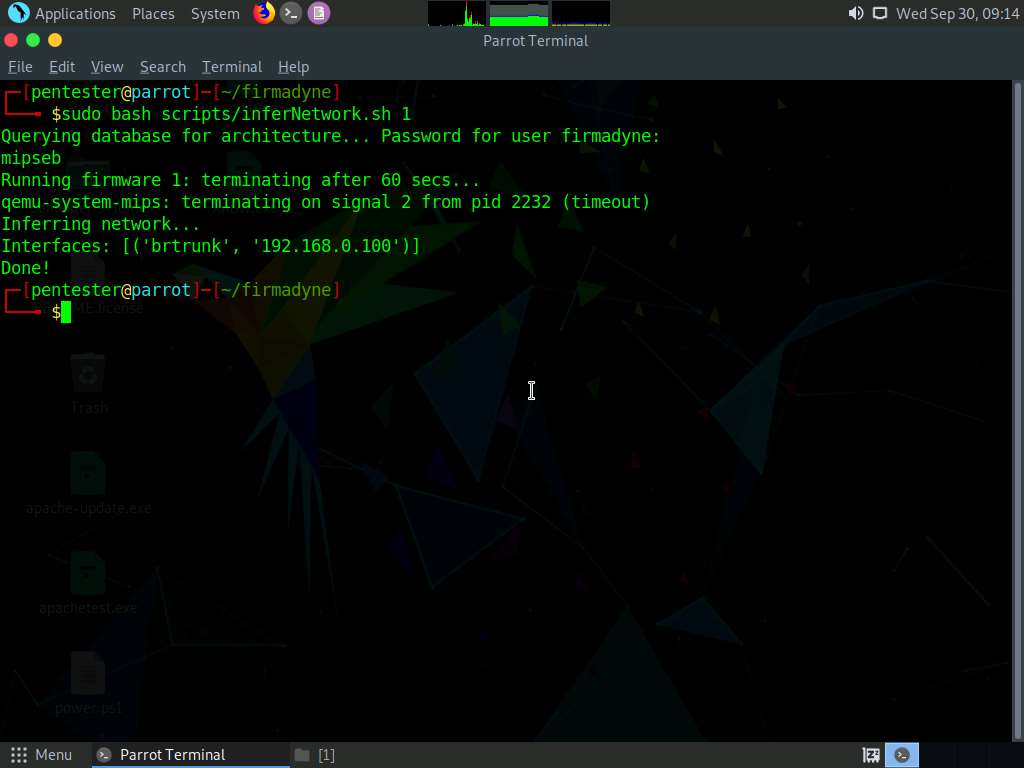
1. Next, we need to create the QEMU disk image for firmware **1**. Enter **sudo bash scripts/makeImage.sh 1**. Enter the password of firmadyne which is **toor** when prompted.



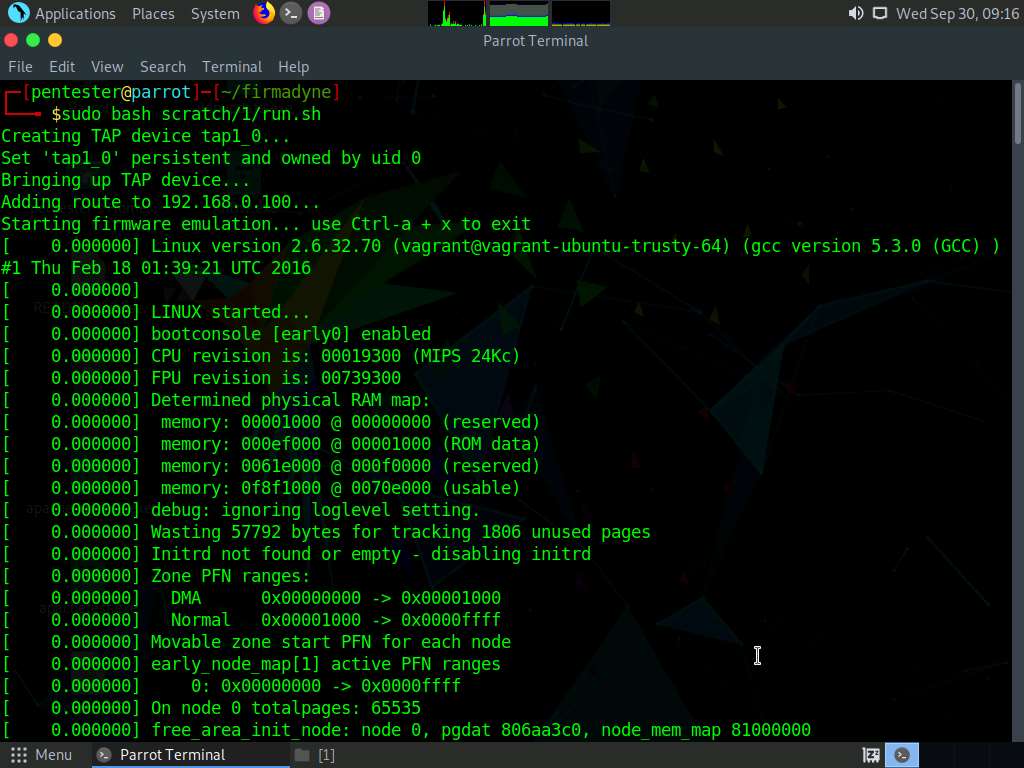


1. Now you are ready to setup networking. It should then be available on the network like any other node. Enter **sudo bash scripts/inferNetwork.sh 1**. Enter the password of firmadyne i.e., **toor** when prompted.

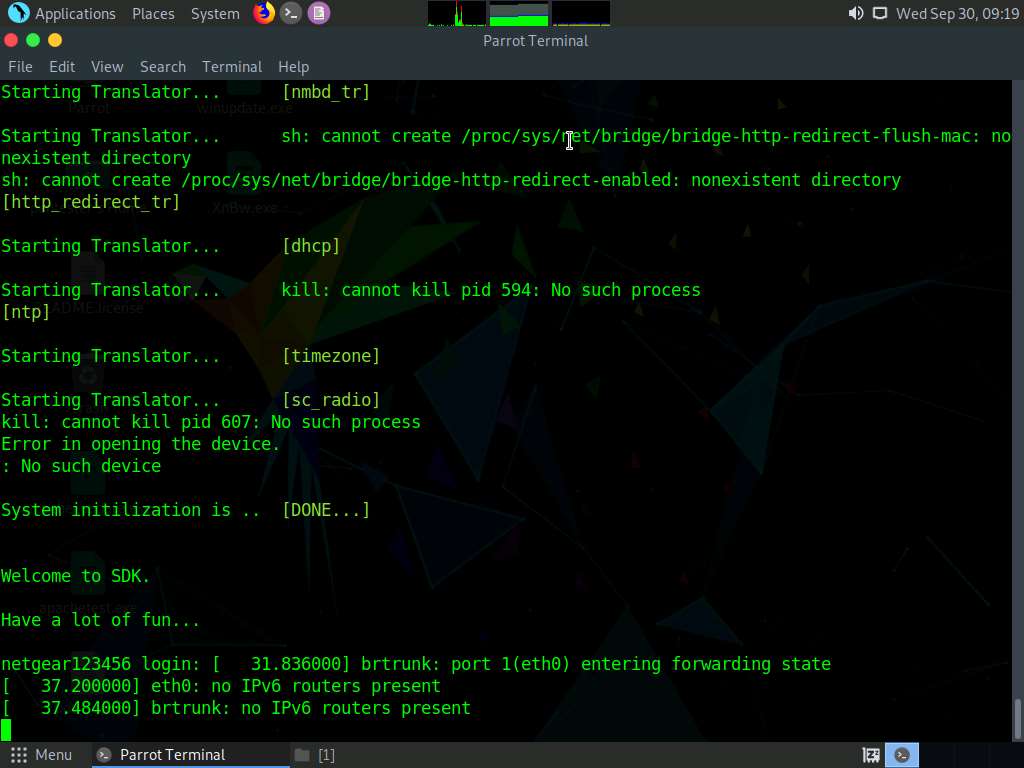




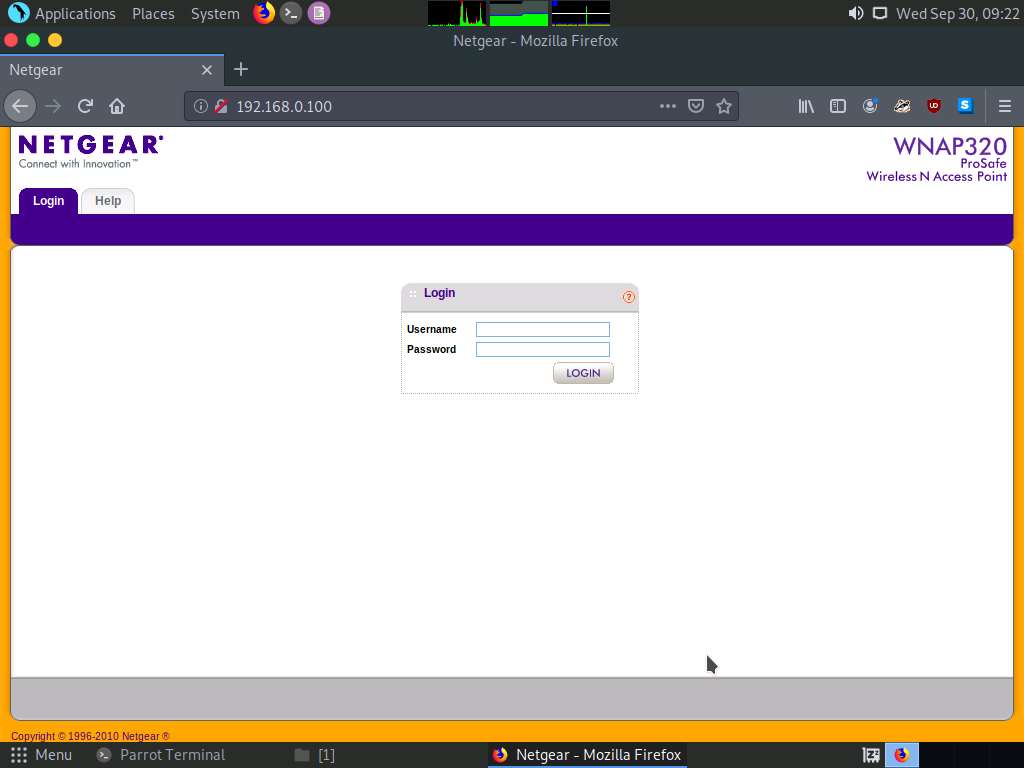
1. Now we are ready to emulate firmware **1** with the inferred network configuration. This will modify the configuration of the host system by creating a Test Access Point (TAP) device and adding a route. Enter **sudo bash scratch/1/run.sh**.



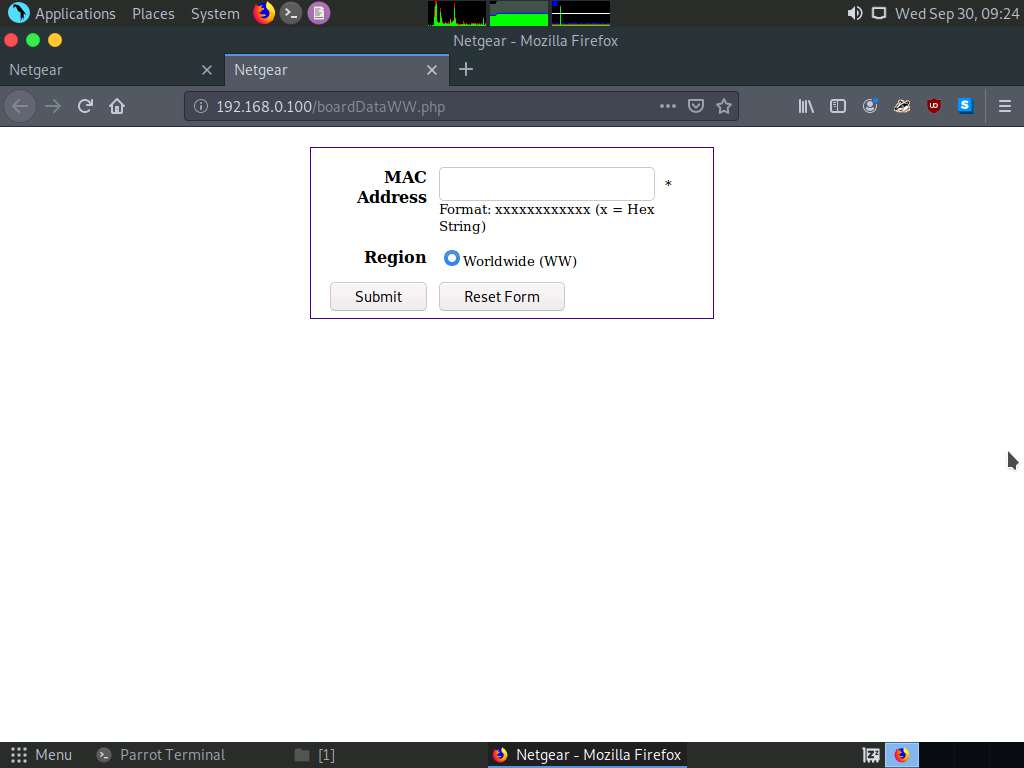
1. This will take some time and execute a chain of instructions as shown in the screenshot. Minimize the terminal window.



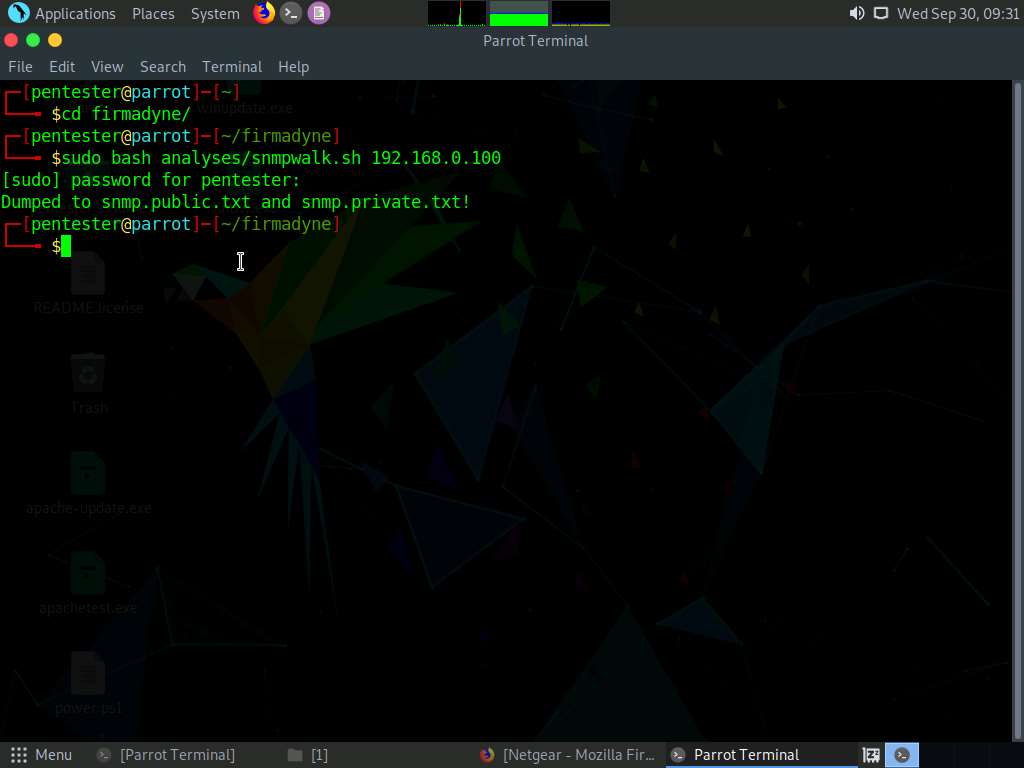
1. The system should be available over the network. Open a browser, and enter **http://192.168.0.100**. Netgear WNAP320 access point login page should appear as shown in the screenshot. This is the web interface from the running firmware that is being emulated.



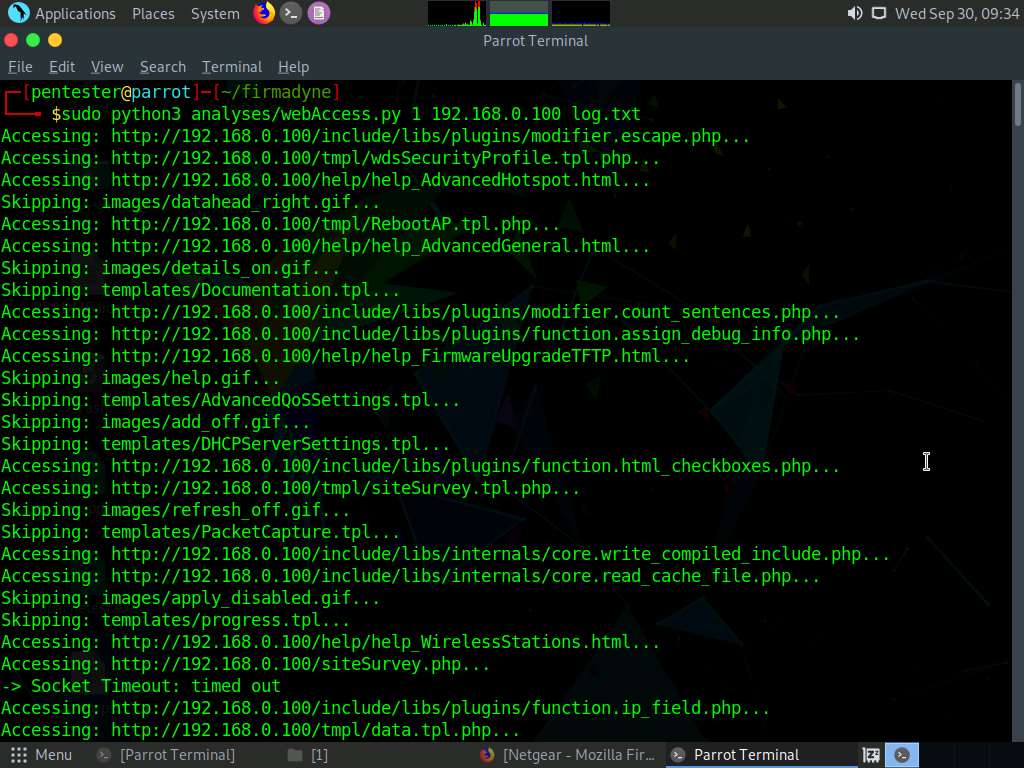
1. We can next look at any of the PHP code that we saw when we explored the file system. In the browser, enter http://192.168.0.100/boardDataWW.php. This will bring up another form as shown in the following screenshot.

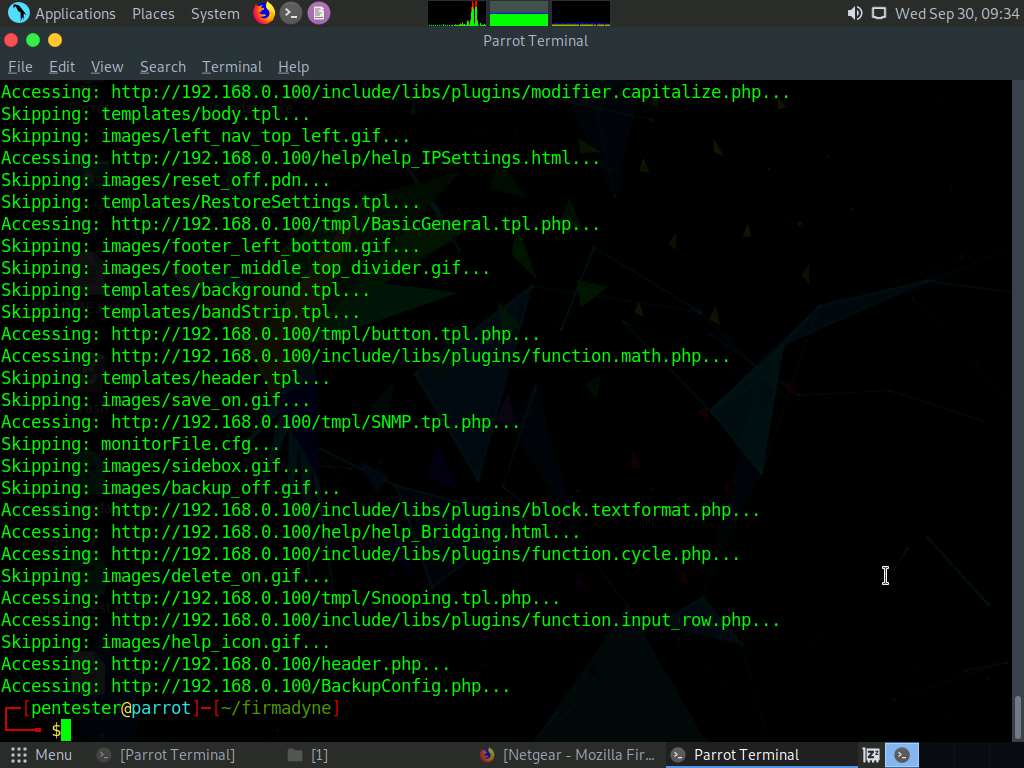


1. Since we had the advantage of reviewing the code, we know that we do not have any input validation, which means that there is a possibility of command injection. We will leave that for you to explore. Hint: You can intercept the query using Burpsuite or any other proxy, and then attempt to enter commands such as cat /etc/passwd.
2. Within the Firmadyne tool, we can leverage additional scripts. Open a new terminal and change the directory to **firmadyne**. Now, enter **sudo bash analyses/snmpwalk.sh 192.168.0.100**. As the command states, it will enumerate data using SNMP. If you are asked to enter the password, type **toor** and press **Enter**.

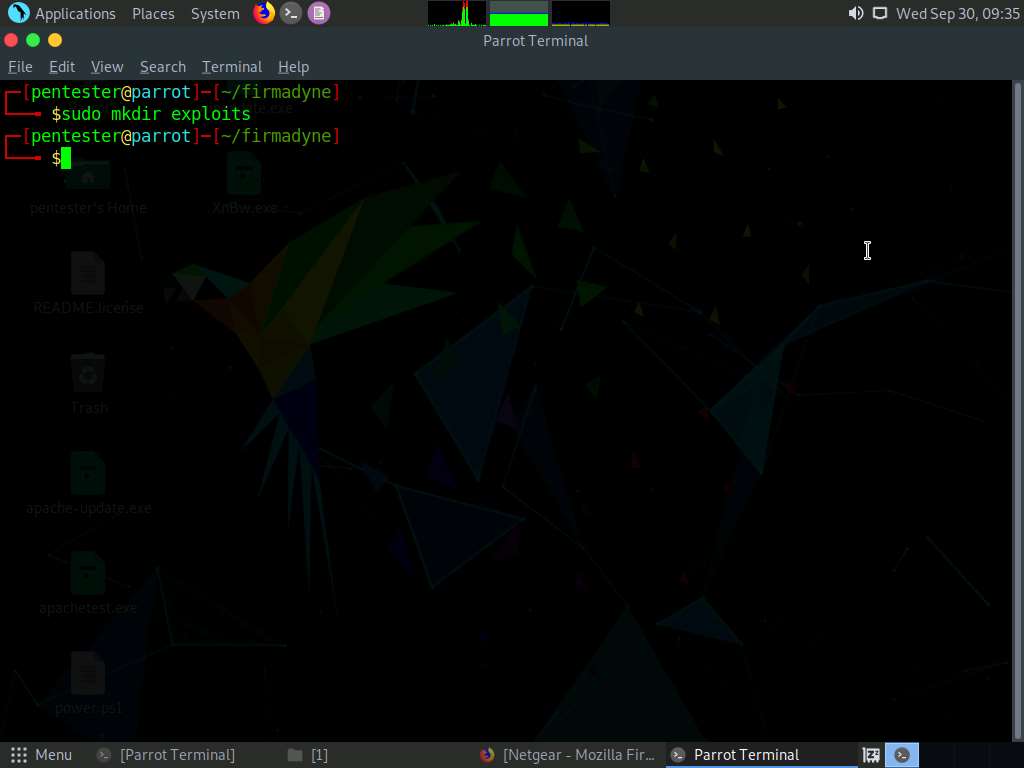


1. We can next enter **sudo python3 analyses/webAccess.py 1 192.168.0.100 log.txt**.

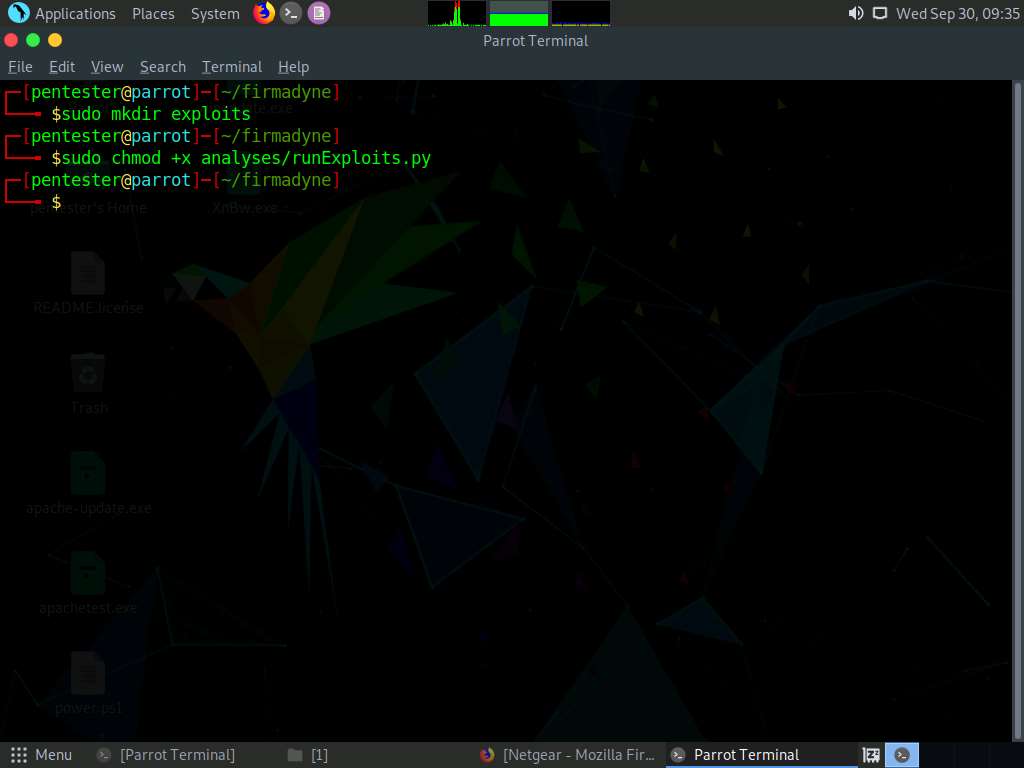




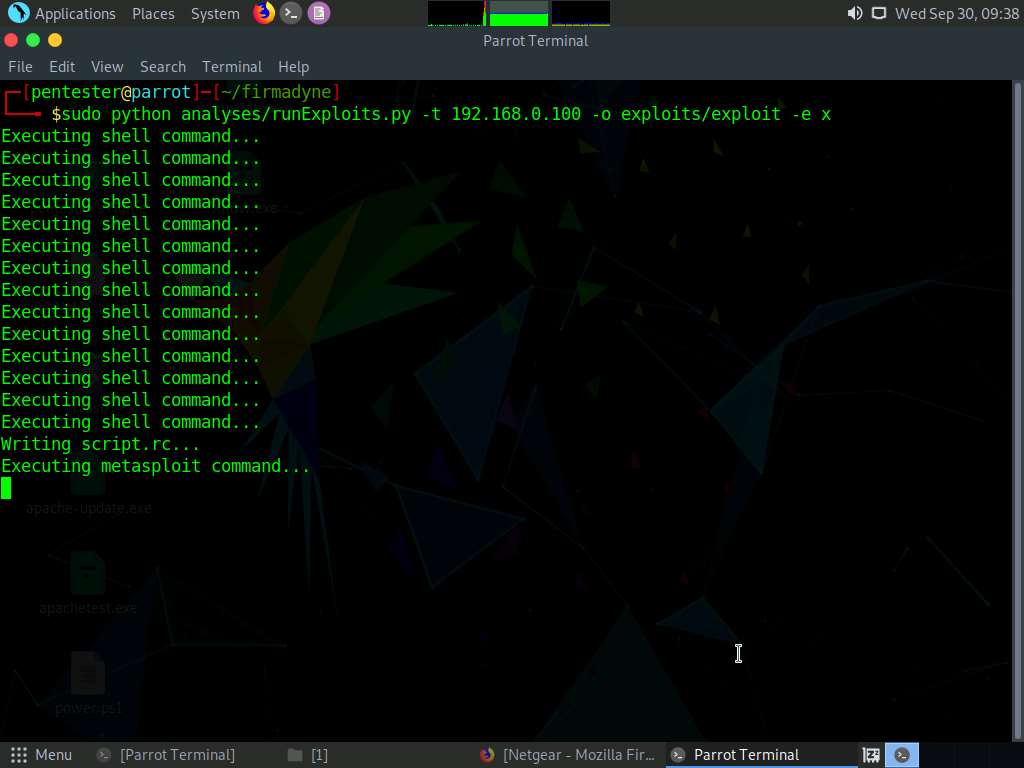
1. There is also a Metasploit interface that will look for vulnerabilities and corresponding exploits. Enter **sudo mkdir exploits**.



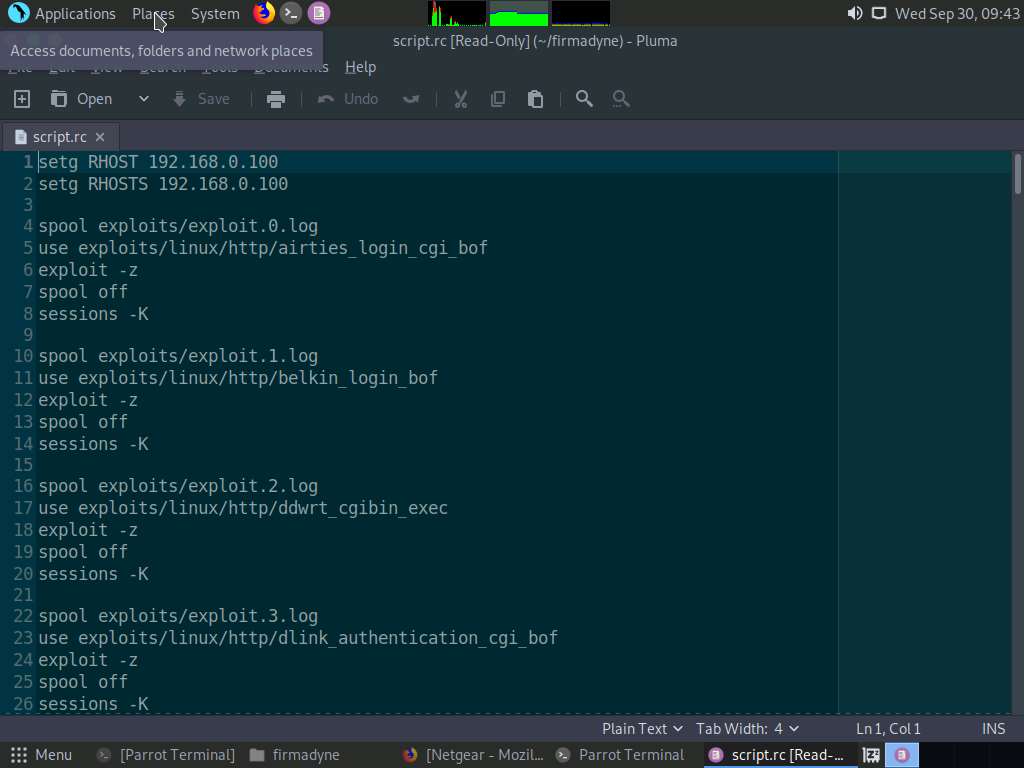
1. Change the permissions on the file with **sudo chmod +x analyses/runExploits.py**.



1. Enter **sudo python analyses/runExploits.py -t 192.168.0.100 -o exploits/exploit -e x**. An example of the output of the command is shown in the following screenshot:



1. Once the script finishes, open the file (**script.rc**) in any text editor. The file can be found in **/home/pentester/firmadyne** directory.



1. As you can see in the above screenshot, the tool is only trying a list of exploits. Therefore, it is not a powerful and effective script. Therefore, it should be used as a reference. Similar to other findings, your ability to research will determine success. The exercise objectives have been achieved.