# Objective 3 Thaw Frost Tower’s Entrance

In this objective we will connect a Wi-Fi dongle to our Linux terminal, and then use a command line tool called curl to send commands to the tower’s thermostat through its website. Before we do that, we need to talk to Greasy Gopherguts and solve his terminal to get hints.

A screenshot of a computer

Description automatically generated with medium confidence

## Terminal Grepping for Gold

Greasy can be found in the North Pole area, but way to the right in front of the Frost Tower.

A picture containing text, toy

Description automatically generated Text

Description automatically generated When you talk to him, he puts this handy hint in your badge.

|  |  |
| --- | --- |
| Table  Description automatically generated with medium confidence | <https://ryanstutorials.net/linuxtutorial/cheatsheetgrep.php> |

Here’s the terminal. Answer the questions by typing quizme at the prompt.

Text

Description automatically generated

## Assignment

The ability to extract useful information from large text files is important. A great tool for that is grep, using regular expressions. The Grepping for Gold terminal will help you get started.

When the terminal asks you, “How many?” you can pipe your grep results into word count (wc) like this:  
grep ThingsIWant FileName | wc -l ( the -l, lower case L, counts the number of lines)

Note that you can get hints from the terminal at no cost.

Text

Description automatically generated

### Step 1 question

The first three questions are straight forward, so we’ll do them all at once.

Graphical user interface, text, application, chat or text message

Description automatically generated

### Step 1 answer

The answers built in to quizme are good.

Text

Description automatically generated

A picture containing text

Description automatically generated

### Step 2 question

The next question asks to to find three ports: How many hosts have a web port open? (Let's just use TCP ports 80, 443, and 8080). You need to search these with the equivalent of an OR. If you search them individually you will get overcounts, for example when both 80 and 443 are open. This link is helpful for this question. <https://www.thegeekstuff.com/2011/10/grep-or-and-not-operators/>

### Step 2 answer

The answer from quizme is very good.

A picture containing text

Description automatically generated

### Step 3 question

This one is a bit tricky: How many hosts with status Up have no (detected) open TCP ports? Any host that is up will have a line with “Status: Up”. The line with “Ports:” only appears when there are open ports. You already know how many hosts are up; find out how many have ports open and do the math!

### Step 4 answer

Their answer is a bit complicated but works if you must get the answer in one line of BASH. I chose to grep 'Ports:’ | wc -l and subtract the results (25652) from the results in from the “Up” query we already did (26054) to get 402.

A screenshot of a computer

Description automatically generated with medium confidence

### Step 5 question

This one is also tricky: What's the greatest number of TCP ports any one host has open? Their solution uses a regex quantifier, {}. The regex (stuff.\*){7} will match if stuff.\* repeats 7 times. You can do that for the ports and increase the number until you don’t get any more matches and that will tell you the greatest number.

Another way is to notice that the “Ignored State:” in the log tells you how many ports were closed or filtered. By default, nmap scans the 1000 most common ports, so open ports will be very close to 1000 – Ignored.

Host: 34.76.0.3 () Status: Up  
Host: 34.76.0.3 () Ports: 22/open/tcp//ssh///, 110/open/tcp//pop3///, 443/open/tcp//https///, 5900/open/tcp//vnc///, 8080/open/tcp//http-proxy/// Ignored State: filtered (995)

### Step 6 answer

Their answer is here. You don’t need to do it all in one line. Just do the part before the “&&” and keep increasing the number in {}. You should get 5 hits with 12 and 0 with 13, so 12 is the answer.A screenshot of a computer

Description automatically generated with medium confidence

The query below shows another way to solve this problem. It first selects all the lines that end with “Ignored State:.\*”. The “.\*” says match any character (the ‘.’) any number of times (the ‘\*’). The -o tells grep to only take the portion that matches. The second grep grabs only the numbers from the previous match. Finally sort the results and count the unique results.

grep -o "Ignored State:.\*$" bigscan.gnmap | grep -o "[0-9]\*" | sort | uniq -c

The smallest number of closed ports is 988 so the largest number of open ports must be 12.

Graphical user interface, text

Description automatically generated

## Hints after solving the Grepping for Gold terminal.

When you talk to Greasy after solving his terminal, he gives you hints which will be useful in Objective 3, Thaw Frost Tower’s Entrance.

|  |  |
| --- | --- |
| Text  Description automatically generated | Table  Description automatically generated  <https://linux.die.net/man/8/iwlist> <https://linux.die.net/man/8/iwconfig> |
| A green screen with white text  Description automatically generated with medium confidence  <https://linux.die.net/man/1/curl> |
| A green screen with white text  Description automatically generated with medium confidence  <https://www.educative.io/edpresso/how-to-perform-a-post-request-using-curl> |

## Objective 3: Thaw Frost Tower’s Entrance

For this objective you must connect your Wifi dongle (in Items in your badge) to the Internet connected thermostat that is visible just inside the window of Frost Tower.

|  |  |
| --- | --- |
| A screenshot of a computer  Description automatically generated with medium confidence | Text  Description automatically generated |
| Text, whiteboard  Description automatically generated |

Greasy’s hint about the Linux Wi-Fi commands will help. You need to use iwlist to find the SSID of the thermometer’s AP, and iwconfig to connect to it. Once you’ve done that, Greasy’s hints about cURL will help you turn up to heat to melt the door.

# Assignment

This assignment will give you more practice at the Linux command line, first in connecting to the thermostat, and second in using cURL to interact with the thermostat’s web site. Using cURL to examine web sites through the command line is a handy tool for your ITSec toolbag.

### Step 1 question

Find the SSID (technically it is the ESSID) of the Frost Tower thermostat using the Wi-Fi dongle and the iwlist command. Remember that this is wireless, so your avatar will need to be near the thermostat to get a connection.

Text

Description automatically generated

### Step 1 answer

If you look at Greasy’s link on iwlist, you will see this.

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

The command to enter is iwlist scan or iwlist scanning.

Note that it gives us the interface ID, wlan0, as well which is helpful.

Text

Description automatically generated

### Step 2 question

Now that you know the ESSID and the interface ID, wlan0, use iwconfig to connect to the thermostat.

### Step 2 answer

The help for iwconfig shows is the proper format, although it should be  
iwconfig [interface] essid

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

We have a successful connection.

### Step 3 question

Connect to the thermostat with curl, using the URL shown in the connection. All you need do is put curl in front of the URL.

### Step 3 answer

Connect to the thermostat web site with curl http://nidus-setup:8080/

Text

Description automatically generated

### Step 4 question

The thermostat says it must be registered! Oh no! There is hope though. All we want to do is change the temperature, which may be allowed due to North Pole Health and Safety regulations. Turn the temperature up enough so that the entrance thaws. If you read the apidoc, you may even find that it gives you the exact command to use!

### Step 4 answer

It was nice of them to give us the command. Changing -40 to 40 ought to do it.

Text

Description automatically generated

curl -XPOST -H 'Content-Type: application/json' --data-binary '{"temperature": 40}' http://nidus-setup:8080/api/cooler

Text

Description automatically generated

|  |  |
| --- | --- |
| Text  Description automatically generated | The doors are open! |