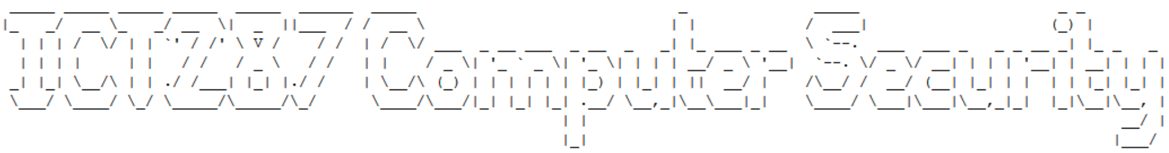


Tutorial One: Kali Linux, Basic Information Gathering

The purpose of this tutorial is to get Kali Linux up and running and familiarise yourself with the lab environment, Kali Linux and some of the important tools on Kali Linux. This platform will be used for almost all of the subsequent labs so it's important to do this first.

It is assumed that you have a basic working knowledge of Linux from the unit's prerequisite, so we don't cover this here. If you lack the basic, you should work through a basic Linux tutorial first (see link on LMS).

If you are logging in from our computer lab on the Perth campus, there is a pre-installed Kali Linux image in VMware which your tutor will show you how to access. If you are setting up your own workspace, then you will need to obtain a copy of **Kali**.

It is available from here https://www.kali.org/downloads/

This is available as a Live bootable CD image so you just burn the CD (or USB) and its ready to go! However, we recommend you install this in a VM (e.g. VirtualBox or VMware) or if you have a spare machine you can also install it directly on your hard drive. You can find pre-build VM images via the above download page too and it is recommended to use these.

Since 2020 Kali has changed to a non-root user policy and you log in as normal user. In a standard Kali install the default credentials are:

Login: kali
Password: kali

If these are not the credentials of the Kali installed in the lab, ask your instructor.

The tool **sudo** should be used to access tools, ports, or services that need administrative privileges, see https://www.kali.org/docs/general-use/sudo/

Note that in Kali 2019 and older versions you need to login as "root" with password "toor" and the attacker VM provided for lab 11 is still based on Kali 2019.

Checking the network configuration

The first thing we need to do is check that we have a functioning network connection. The network configuration file is in /etc/network/interface if you need to check it.

To get status on the network adapters:

```
#ifconfig -a
```

If you have an eth0 network adapter, and it has an IP address then it will probably work.

Test it out by pinging a known site:

```
#ping www.google.com
```

If this doesn't work, check the settings in the **/etc/network/interfaces**

For DHCP

```
auto eth0
allow-hotplug eth0 # detect link
iface eth0 inet dhcp #  using DHCP method
```

For static IP

```
auto eth0 #interface name
allow-hotplug eth0 #link detection
iface eth0 inet static # define IPV4 the ip using static
method
address 192.168.0.252 # IP address
netmask 255.255.254.0 # subnet mask
gateway 192.168.0.253 # gateway (router)
```

To restart the network service after changing settings:

```
#/etc/init.d/networking restart
or
#service networking restart
```

Getting Help with Commands

In the labs you will often have to figure out commands or command line options yourself as the labs are not designed as copy&paste labs (copy&paste labs are not very useful for learning how to solve problems).

Many Linux tools support the -h switch that gives you an overview of how to use the tool. Try:

```
#ifconfig -h
```

Now often we need more details. Then we can use the **man** command to look at the manual page for a tool. Try:

```
#man ifconfig
```

You exit a man page by pressing q. You can search a man page by pressing / followed by a search string and ENTER. Jump to the next occurrence of the string by pressing n. Use this technique to search for "netmask" in the ifconfig man page.

Finally, you can also use Google or other search engines to search for the man pages. Google "man ifconfig".

Now let's move onto a few tools for information gathering.

Viewing Network Connections

The netstat command allows you to view connections and ports in use. For example to see programs listening on TCP ports, use the command:

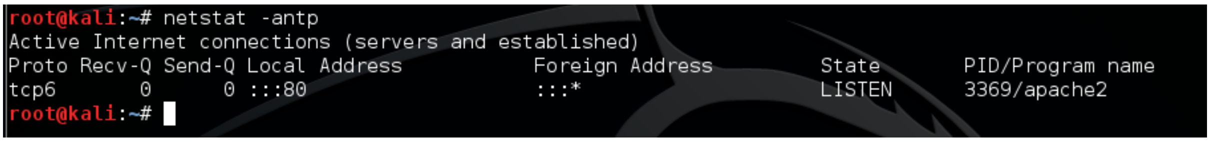
```
#netstat -antp
```

As with all commands, use the man (short for manual page) command to find out more information about the command line options. Pressing the / key allows to search in man pages.

To learn more about netstat

```
#man netstat
```

We can see that on this machine, we have a web server running on port 80:



```
root@kali:~# netstat -antp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp6       0      0 :::80                   :::*                    LISTEN      3369/apache2
root@kali:~# 
```

If your machine doesn't have Apache running you can always start it with the command: service apache2 start.

netcat – the swiss army knife of TCP/IP connections

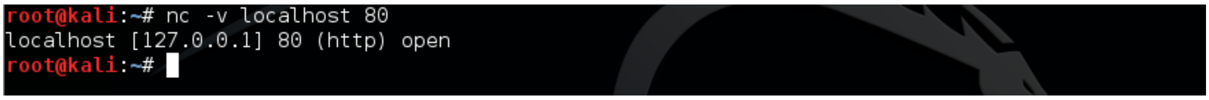
netcat is a computer networking service for reading from and writing to network connections using TCP or UDP. To see the various options for netcat:

```
#nc –h
```

Checking if a port is listening

Let's say we want to check if the web server we saw in netstat is really listening on port 80

```
#nc –v localhost 80
```



```
root@kali:~# nc -v localhost 80
localhost [127.0.0.1] 80 (http) open
root@kali:~# 
```

We can also listen on a port for a connection

```
#nc –lvp 1234
```

Task 1: Create a listener in one terminal window using the above command, and then connect to it from another terminal window.

Task 2: When you set up your nc listener, tell it to execute /bin/bash to give a shell when a connection is received. The command is

```
 #nc –lvp 1234 –e /bin/bash
```

Task 3: So far, we have used localhost for our nc sessions. Next try connecting to another machine. Use nc to create a chat session between yourself and one of your classmates. The same commands as above can be re-used.

You can close down nc processes with Ctrl-C

Information Gathering

Netcraft

Sometimes the information that web servers and hosting companies give out publically can tell you a lot about a site. Netcraft aggregates some of this data and also keeps stats on the availability of various websites. Go to www.netcraft.com and query a domain name (try Murdoch.edu.au for a start). What do you discover?

Whois Lookups

Domain registrars keep records of the domains they host. These are kept in a public database that can be queried with a whois tool. From the command line try

```
#whois murdoch.edu.au
```

What do you find? Who is the registrant of this domain?

DNS Recon

DNS servers also tell us a lot. A tool like nslookup will allow us to query DNS

```
#nslookup murdoch.edu.au
```

If that's not very interesting we can try some additional options. For example, we may be interested in the mail exchangers for this domain

```
#nslookup –query=mx murdoch.edu.au
```

Task 4: There are numerous commands that can find out similar information but with slightly different functionality. Look up the host and dig commands and find out how they differ.

Packet Sniffing

The Wireshark program allows inspecting packets sent over a network interface.

Task 5: Run Wireshark and start capturing packets. Use the chat program from Task 3 to send some messages. Inspect the traffic with Wireshark. Can you see the chat messages you sent?