Network Tools

The purpose of this lesson is to introduce some of the basic command-line tools used for networking and how to use these both as troubleshooting tools and how to use them in the context of cybersecurity. Helpful commands for both Windows and Linux will be highlighted, although only examples for the Linux commands will be shown.

COMMAND LINE TOOLS

IP & Ipconfig

IP, or *ipconfig* on Windows, is a command-line tool that shows the current network configuration of the device that you are on. This includes information such as the current private IP address of the device, the gateway address, and the DNS server. This tool is often used when a system is having connectivity issues and is a good place to start to diagnose those issues.

Some common examples of *ip* commands include:

- *ip a* – Shows the IP addresses on the device
- *ip r list* – Displays the current routing table on the device
- *ip link set dev [Device Name] [up|down]* – This sets the network interface to either up (enabled) or down (disabled)

```
student@SBTLab3:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:d6:04:31 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.55/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
       valid_lft 84869sec preferred_lft 84869sec
    inet6 fe80::20c:29ff:fed6:431/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
student@SBTLab3:~$
```

Using the ip command to see the local network configuration

Traceroute & Tracert

Traceroute, or tracert on Windows, is a command-line tool that allows you to see the path that network packets take when going from one host to another. This tool is often used to troubleshoot routing issues between two systems.

Some common examples of *traceroute* commands include:

- *traceroute [url]* – Runs the basic traceroute to see the path it takes to get to a specified address
- *traceroute [url] -p [port number]* – Allows the trace to be run with a specific port

```
student@SBTLab3: $ traceroute go
```

Running traceroute on Google.com

Dig & Nslookup

Dig, or Nslookup on Windows (and Linux), is a command-line tool that is used to query DNS servers for information about a specific domain. This tool can often be helpful when you need to quickly search for the IP address of a malicious URL or if you need to find out what mail server a domain routes their emails through.

Some common examples of *dig* commands include:

- *dig [domain name]* – Queries the DNS server for the A record for the specified domain
- *dig [domain name] MX* – Queries the DNS server for mail (MX) records for the specified domain
- *dig [domain name] ANY +nocomments +noauthority +noadditional +nostats* – Queries the DNS server for all DNS records for the specified domain and removes the extra information provided by dig

Using dig to query the MX records for securityblue.team

Netstat

Netstat is a Linux and Windows-based command-line tool that monitors the TCP and UDP connections on your host system. This tool can often be used for application troubleshooting or if a computer is suspected of containing malware, to see if a system has open connections to remote servers, which could be a sign of it being controlled by a C2 Server

Some common examples of *netstat* commands include:

- *netstat -a* – Displays all of the current connections and listening ports on the system
- *netstat -a -b* – Displays all of the current connections and listening ports on the system, as well as their corresponding executable
- *netstat -s -p tcp -f* – Displays the statistics for all connections using TCP and then displays them in an FQDN format

Using netstat to see active connections on the host machine

NMAP

Nmap, or Network Mapper, is most often the tool of choice for performing Network Discovery. It is capable of revealing ports, discovering devices on a network, revealing running services, identifying operating systems, and many other functions. All of these capabilities make Nmap an effective, easy-to-use, and versatile tool. Nmap is an open-source tool that comes pre-installed on the Kali operating system, among various other Linux distributions. However, there are also Windows, Mac OS, and other operating system versions on their website.

The Syntax for Nmap commands are fairly simple: *nmap [Scan Type] [Options] [target specification]*

There are multiple options that you can use with the Nmap tool:

OPTIONS	COMMAND	OPERATION
-v (Verbosity level)	***nmap -v <target>***	Provides more detailed text output on the scan result.
-O (OS Detection)	***nmap -O <target>***	Provides information about a host's operating system.
-F (Fast Scan)	***nmap -F <target>***	Allows for fast scanning through the most commonly used ports.
-sS (TCP SYN Scan) **(half Scan)**	***nmap -sS <target>***	This is the most popular type of scan that Nmap has. It is a fast and stealthy scan that checks a host's TCP ports. (Is also known as a half-scan because it does not complete the 3-way handshake of the TCP protocol).
-sT (TCP Connect Scan)	***nmap -sT <target>***	It's the Nmap's default TCP scanner, unlike -sS this one does complete the 3-way TCP handshake, thus getting all open TCP ports but losing the stealth of the scan.
-sU (UDP Scan)	***nmap -sU <target>***	This scanner allows you to check the UDP ports of a host.
-sA (TCP ACK Scan)	***nmap -sA <target>***	It's a different scanner from the previous ones, this one allows to identify if the rules of a Firewall work correctly or not. In other words, it allows to identify Firewalls.
-sV (Version detection)	***nmap -sV <target>***	It allows us to obtain the current version of the service running on each port we find that responds.

(All these options can be written together in the form: ***nmap -v -sT -sU -sV <target>***)

If you would like more information on different commands, you can use these Nmap cheatsheets from SANS and StationX.

Running a TCP SYN Scan with verbosity on scanme.nmap.org

Nmap also has scripting capabilities through the Nmap Scripting Engine (NSE) and this allows for quick and easy scripting to allow tasks like network discovery, version detection, vulnerability detection, backdoor detection, and vulnerability exploitation to be more efficient. However, the NSE is out of scope for this lesson and BTL1, if you would like to learn more about NSE, you can go to its official chapter on the Nmap website.

PORT SCANNING EXAMPLE

To demonstrate how ports and services function, we have conducted a port scan against "scanme.nmap.org" – **it's important to mention that you should not scan any website or IP address unless you have expressed permission to do so**. The founders behind Nmap have stated you are legally allowed to scan "scanme.nmap.org" so feel free to practice against this site.

For this example, we used the following Nmap command to conduct a TCP Connect scan. on scanme.nmap.org:
nmap -v -sT -sV scanme.nmap.org

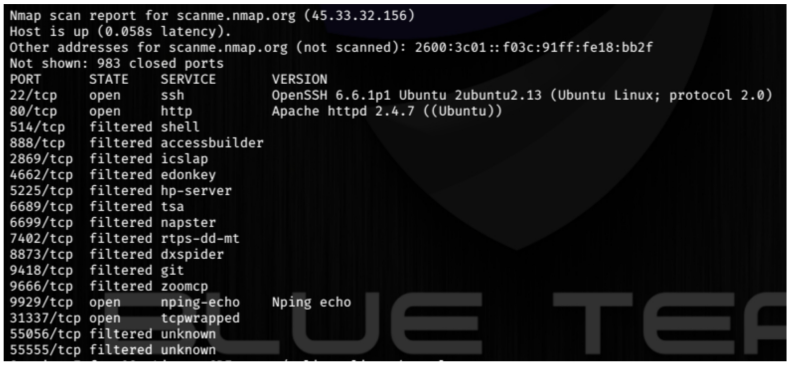
In the Network Fundamentals lesson we covered the TCP three-way handshake, in this case, we are sending an SYN packet to ports on the webserver (step 1), the server is sending us back SYN-ACK packets (step 2), then we send an ACK packet (step 3) allowing us to connect and perform "banner grabbing" – the process of collecting information about running services from a system.

In the below screenshot you can see the output from Nmap, with a PORT column, STATE column, and SERVICE

column.

- **PORT –** The port number that is open and has a service running.
- **STATE –** Whether Nmap was able to connect to the port (open) or not able to connect (filtered).
- **SERVICE –** The service (program) that is running on that specific port number.

We can see that Port 80 TCP is open, running the HTTP service (hypertext transfer protocol). This is why we're able to view a webpage if we go to www.scanme.nmap.org. As we have conducted a service version scan using Nmap, we can also see that this specific web server is running Apache. Port 22 TCP is also open, which is the standard port for secure shell (SSH) that allows remote sessions on the system.



Previous Topic

Back to Lesson

Next Topic