Using tcpdump

The tcpdump utility is used to read either packets captured live from a network or packets that have been saved to a file. Although there is a Windows version called windump, tcpdump only works on Unix-like operating systems.

- Use this command to capture traffic on all interfaces:

```
# tcpdump -i any
```

- Here is the command to capture traffic on a particular interface:

```
# tcpdump -i eth0
```

- And to filter traffic by IP, whether it's the source or the destination, use this command:

```
# tcpdump host 192.168.5.5
```

Using the File Transfer Protocol

File Transfer Protocol (FTP) is a subset of TCP/IP and that FTP is used for the transfer of files. In recent years, FTP has become a truly cross-platform protocol for transferring files. Almost every client and server platform has implemented FTP. Windows is no exception. Its TCP/IP stack comes with a command-line ftp utility.

To start the ftp utility, enter ftp at a command prompt. The result is an ftp command prompt:

```
C:\Users\clarusway>ftp
ftp>
```

From this prompt, you can open a connection to an FTP server and upload and download files as well as change the way FTP operates. To display a list of all the commands you can use at the ftp command prompt, type help or ? and press Enter. To get help on a specific command, type help, a space, and then the name of the command. Here is some output from the help command:

```
C:\Users\clarusway>ftp
ftp> ?
Commands may be abbreviated.  Commands are:

!               delete          literal         prompt          send
?               debug           ls              put             status
append          dir             mdelete         pwd             trace
ascii           disconnect      mdir            quit            type
bell            get             mget            quote           user
binary          glob            mkdir           recv            verbose
bye             hash            mls             remotehelp
cd              help            mput            rename
close           lcd             open            rmdir
```

💡**Tip:**

- Third-party applications are available that provide a GUI interface for FTP, which is easier to use than a command line.

Starting FTP and Logging In to an FTP Server

Of the two FTP file operations (download and upload), the ability to download files is definitely the more crucial for you to have down as a network technician or sysadmin.

The first steps in starting an FTP download session are to determine the address of the FTP site and start the ftp utility. The FTP site typically has the same name as the website except that the first three characters are ftp instead of www. For example, Microsoft's website is www.microsoft.com. Its FTP site, on the other hand, is ftp.microsoft.com.

First, start the ftp utility as demonstrated in the preceding section, and then follow these steps:

1. At the ftp command prompt, type open, a space, and the name of the FTP server, like this:

```
C:\Users\clarusway> ftp
ftp> open ftp.claruswaytrainer.com
Connected to ftp.claruswaytrainer.com.
220---------- Welcome to Pure-FTPd [TLS] ----------
220-You are user number 1 of 100 allowed.
220-Local time is now 11:45. Server port: 21.
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 15 minutes of inactivity.
User (ftp.claruswaytrainer.com:(none)): enter
230 Anonymous user logged in
ftp>
```

As shown here, if the FTP server is available and running, you'll receive a response welcoming you to the server and asking you for a username. Right now, we used *Anonymous* as the username (enabled by default on the FTP server), which means that anyone can log in to it.

You can also start an FTP session by typing ftp, a space, and the address of the FTP server. This allows you to start the ftp utility and open a connection in one step. Here's an example:

```
C:\Users\clarusway> ftp ftp.claruswaytrainer.com
```

2. Enter a valid username, and press Enter.
3. Enter your password, and press Enter.

If you enter the wrong username and/or password, the server will tell you so by displaying the following and leaving you at the ftp command prompt:

```
 530 Login Incorrect
 Login failed.
```

This means you've got to try again and must start the login process over. If you're successful, the FTP server will welcome you and drop you back at the ftp command prompt.

Downloading Files

After you log in to the FTP server, you'll navigate to the directory that contains the files you want. The FTP command-line interface is similar to the DOS command-line interface.

The below table lists and describes the common navigation commands for FTP. After you navigate to the directory and find the file you want to download, it's time to set the parameters for the type of file. Files come in two types:

- ASCII, which contains text
- Binary, which is all other files

If you set ftp to the wrong type, the file you download will contain gibberish. So if you're in doubt, set ftp to download files as binary files. Check out the below table.

Command	Description
ls	Short for list. Displays a directory listing. Very similar to the DIR command in MS-DOS.
cd	Short for change directory. Works almost identically to the MS-DOS CD command. Use it to change to a different directory and navigate the server's directory structure.
pwd	Short for print working directory. Displays the current directory on the server. Useful if you forget where you are when changing to several locations on the server.
lcd	Short for local change directory. Displays and changes the current directory on the local machine. Useful when you are downloading a file and aren't in the directory where you want to put the file.

To set the file type to ASCII, type ascii at the ftp command prompt. ftp will respond by telling you that the file type has been set to A (ASCII):

```
ftp>ascii
Type set to A
```

To set the file type to binary, type binary at the ftp command prompt. ftp will respond by telling you that the file type has been set to I (binary):

```
ftp>binary
Type set to I
```

To download the file, just use the get command like this:

```
ftp>get test.exe
200 PORT command successful.
150 Opening BINARY mode data connection for 'test.exe'
(567018 bytes).
```

The file will start downloading to your hard drive. Unfortunately, with its default settings, the ftp utility doesn't give you any indication of the progress of the transfer. When the file has downloaded, the ftp utility will display the following message and return you to the ftp command prompt:

```
226 Transfer complete.
567018 bytes received in 116.27 seconds (4.88 Kbytes/sec)
```

Uploading Files

To upload a file to an FTP server, you've got to have rights on that specific server. These rights are assigned on a directory-by-directory basis. To upload a file, log in and then follow these steps:

1. At the ftp command prompt, type `lcd` to navigate to the directory on the local machine where the file resides.
2. Type `cd` to navigate to the destination directory.
3. Set the file type to `ASCII` or `binary`.
4. Use the `put` command to upload the file.

The syntax of the `put` command looks like this:

```
ftp> put local file destination file
```

Let's say you want to upload a file called `test.txt` on the local server but you want it to be called `my.txt` on the destination server. To accomplish that, use the following command:

```
ftp> put test.txt my.txt
```

You'll get the following response:

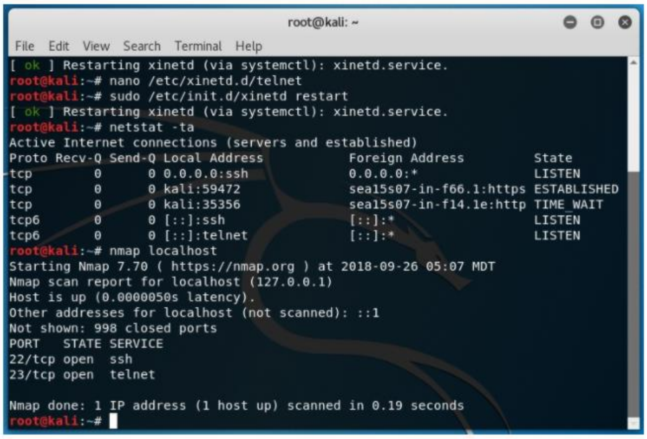
```
200 PORT command successful.
150 Opening BINARY mode data connection for my.txt
226 Transfer complete.
743622 bytes sent in 0.55 seconds (1352.04 Kbytes/sec)
```

💡**Tips:**

- You can upload multiple files using the **mput** command. Simply type **mput**, a space, and then a wildcard that specifies the files. For example, to upload all the text files in a directory, type **mput *.txt**. And in the same way, you can download multiple files with **mget** command.
- **get** and **put** commands require the perfect match of the file names intented to be downloaded and uploaded. On the other hand, **mget** and **mput** commands can run on partial match, a pattern with wildcards or asteriks.

Using the Telnet Utility

Part of the TCP/IP protocol suite, **Telnet** is a virtual terminal protocol utility that allows you to make connections to remote devices, gather information, and run programs. Telnet was originally developed to open terminal sessions from remote Unix workstations to Unix servers. Although it's still used for that purpose, we now use it as a troubleshooting tool as well. The below figure shows the basic Telnet interface as it's being used to start a terminal session on a remote Unix host.



```
[ ok ] Restarting xinetd (via systemctl): xinetd.service.
root@kali:~# nano /etc/xinetd.d/telnet
root@kali:~# sudo /etc/init.d/xinetd restart
[ ok ] Restarting xinetd (via systemctl): xinetd.service.
root@kali:~# netstat -ta
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:ssh             0.0.0.0:*               LISTEN
tcp        0      0 kali:59472              sea15s07-in-f66.1:https ESTABLISHED
tcp        0      0 kali:35356              sea15s07-in-f14.1e:http TIME_WAIT
tcp6       0      0 [::]:ssh                [::]:*                  LISTEN
tcp6       0      0 [::]:telnet             [::]:*                  LISTEN
root@kali:~# nmap localhost
Starting Nmap 7.70 ( https://nmap.org ) at 2018-09-26 05:07 MDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000050s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
23/tcp open  telnet

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
root@kali:~# 
```

The Telnet

In today's Windows environments, Telnet is a basic command-line tool for testing TCP connections. You can telnet to any TCP port to see if it's responding—something that's especially useful when checking *Simple Mail Transfer Protocol (SMTP)* and *HTTP* (web) ports.

How to Enable Telnet in Windows

Because most people have the Windows 10 operating system running on their PCs these days, it's good to know that, by default, these operating systems install without Telnet available. But there's a way around that one—if you really must have a Telnet client enabled in these operating systems, here's how to do it:

1. Open Control Panel.
2. Select Programs And Features.
3. In the left column, select Turn Windows Features On Or Off
4. Select the Telnet checkbox (and any other obscure services you may want enabled), and wait while Windows installs for a while and then reboots.

Now you can go to Start and then type telnet in the Start search box to get a Telnet window to open for you. You can also open a DOS prompt and just type telnet from there. Here are the options that Windows provides with Telnet:

```
Microsoft Telnet> ?
Commands may be abbreviated. Supported commands are:
c - close close current connection
d - display display operating parameters
o - open hostname [port] connect to hostname (default port 23).
q - quit exit telnet
set - set set options (type 'set ?' for a list)
sen - send send strings to server
st - status print status information
u - unset unset options (type 'unset ?' for a list)
?/h - help print help information
```

Don't Use Telnet, Use Secure Shell

Telnet is totally insecure because it sends all data in crystal-clear text—including your name and password. If Microsoft doesn't even enable it on its latest OSs, then you know it really must be insecure.

So if you shouldn't use Telnet, what should you use instead? **Secure Shell (SSH)** is your answer. It provides the same options as Telnet, plus a lot more; but most importantly, it doesn't send any data in cleartext. The thing is, your servers, routers, and other devices need to be enabled with SSH, and it's not configured by default on most devices.

Some configuration is usually necessary if you want things to work as they really should, and yes, sometimes it's a little painful to get everything running smoothly, but it's all worth it in the long run.

Secure Shell (ssh)

The **secure shell** or `ssh` is a collection of tools using a secure protocol for communications with remote computers.

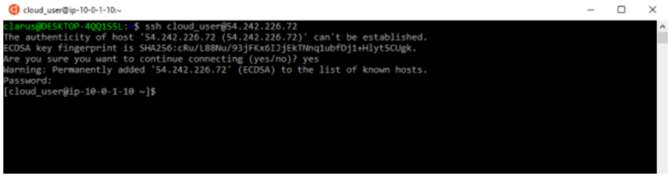
It is a protocol used to securely connect to a remote server/system. `ssh` is secure in the sense that it transfers the data in encrypted form between the host and the client. It transfers inputs from the client to the host and relays back the output. `ssh` runs at TCP/IP port 22.

```
ssh user_name@host(IP/Domain-name)
```

`ssh` command instructs the system to establish an encrypted secure connection with the host machine.

- `user_name` represents the account that is being accessed on the host.
- `host` refers to the machine which can be a computer or a router that is being accessed. It can be an IP address (e.g. 54.164.151.235) or domain name(e.g. www.clarusway.com).

Example:



```
clarus@DESKTOP-4QQ1S5L:~$ ssh cloud_user@54.242.226.72
The authenticity of host '54.242.226.72 (54.242.226.72)' can't be established.
ECDSA key fingerprint is SHA256:cRu/L88Nu/93jFKx6IJjEkTNnq1ubfDj1+Hlyt5CUgk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '54.242.226.72' (ECDSA) to the list of known hosts.
Password:
[cloud_user@ip-10-0-1-10 ~]$
```

> Q: How do you use **ssh** to connect to a **remote** server in Linux.
> A: Most servers in the world are run on Linux servers. They're dependable, affordable and highly configurable. However, servers aren't always accessed, nor accessible, directly. Hence they require remote access. The most frequently used, and secure, method of accessing servers remotely is via SSH, otherwise known as Secure Shell.
> For connect to remote server, first we should open to linux terminal and then type
> ssh user@www.remote_server_name.com
> or
> ssh user@82.178.72.19
>
> — - Interview Q&A