Packet captures with tcpdump

Tcpdump is a popular network analyzer.
It's pre-installed on many Linux distributions and can be installed on most Unix-like operating systems, like macOS.
You can easily capture and monitor network traffic such as TCP, IP, ICMP, and many more.

Tcpdump is a command line tool.
This means that it does not have a graphical user interface.
Earlier in the program, you learned that the command line is a very powerful and efficient tool, and we'll practice using it together.
With tcpdump, you can apply options and flags to your commands to easily filter network traffic so that you can find exactly what you're looking for.
You can filter for a specific IP address, protocol, or port number.

Let's examine a simple tcpdump command used to capture packets.
Keep in mind that your computer's traffic may appear different when you use this command.

At first glance, this looks like a lot of information.
Let's examine it line by line.
The command we ran is:
sudo tcpdump -i any -v -c 1.
We're using sudo because the Linux account we're logged in on doesn't have the permission to run tcpdump.
Then, we specify tcpdump to start tcpdump and -i to specify which interface we want to sniff traffic on.
The -v stands for verbose, which displays detailed packet information.
The -c stands for count, which specifies how many packets tcpdump will capture.
Here we've specified one.
Now let's examine the output.

Tcpdump has told us that it's listening on any available network interfaces, and it's also given us additional information, like the capture size.

The first field is the packet's timestamp, which details the specific time of the packet travel.
It begins with hours, minutes, seconds, and fractions of a second.
Timestamps are especially helpful during an incident investigation when you want to determine timelines and correlate traffic.

Next, IP is listed as the Version field.
It's listed as IP, which means it's IPv4.
The verbose option has given us more details about the IP packet fields, such as protocol type and the length of the packet. Let's check it out.

The first field, ToS stands for Type of Service.
Recall that this tells us if certain packets should be treated with different care.
This is represented by a value in hexadecimal.

The TTL field is Time to Live, which tells us how long a packet can travel across a network before it gets dropped.
The next three fields are Identification, Offset, and Flags, which provide three fields with information relating to fragmentation.
These fields provide instructions on how to reassemble packets in the correct order.
For example the DF, beside flags stands for Don't Fragment.

Next, the proto is the Protocol field.
It specifies the protocol in use and also provides us with the value that corresponds to the protocol.
Here the protocol is tcp, which is represented by the number 6.
The last field, length, is the Total Length of the packet, including the IP header.

Next, we can observe the IP addresses that are communicating with each other.
The direction of the arrow indicates the direction of the traffic flow.
The last piece of the IP address indicates the port number or name.

Next, the cksum or checksum field corresponds to the Header Checksum, which stores a value that's used to determine if
any errors have occurred in the header.
Here, it's telling us it's correct with no errors.

The remaining fields are related to TCP.
For example, Flags indicate TCP flags.
The P is the push flag, and the period indicates it's an ACK flag.
This means that the packet is pushing out data.

This is just one of many commands you can use in tcpdump to capture network traffic.
Isn't it fascinating to observe all the information contained within these invisible packets?
Go ahead and try it out for yourself!

Revision #1
Created 11 September 2023 06:56:17 by naruzkurai
Updated 1 November 2023 01:10:46 by naruzkurai