

Network Sniffer Lab

Open Source Security Tools for Information Technology Professionals

School of Professional Studies (SPS)

The City University of New York (CUNY)

Aron Trauring
Adjunct Professor
CEO, Zoteca

Class 9 November 21th, 2005

Useful `tcpdump` options

- `tcpdump <options> <selector expression>`
- `-a` — attempts to convert addresses to names
- `-c` — stop `tcpdump` after count number of packets
- `-F` — take packets from file (post-analysis)
- `-e` — print link-level header (MAC address on Ethernet)
- `-i <interface>`
- `-n` — don't convert addresses to names
- `-t` — no timestamp

- `-v` `-vv` `-vvv` — level of verbosity
- `-w <filename>` — write to file

tcpdump Selectors

- host <hostname>
- net <IP Network>
- port <TCP port>
- Traffic Direction: src dst
- Protocol: tcp udp icmp
- Boolean: and or not

`tcpdump` Output for TCP Protocol

```
12:25:44.578546 csam.login > rtsg.1023: S 947648:947648  
ack 768513 win 4096 <mss 1024>
```

- Timestamp, broken down into fractions of a second — on a busy network many packets per second
- TCP sequence number
- Source IP address with port
- > (a greater than sign) — shows direction
- Destination address with port
- Flags — some combination of S (SYN), F (FIN), P (PUSH), R (RST), W (ECN CWR) or E (ECN-Echo), or a single `.` (no flags)

- Data-seqno describes the portion of sequence space covered by the data in this packet
- Ack is sequence number of the next data expected the other direction on this connection.
- Window is the number of bytes of receive buffer space available the other direction on this connection.
- Urg indicates there is 'urgent' data in the packet.
- Options are tcp options enclosed in angle brackets

`tcpdump` Output for UDP DNS Request

```
h2opolo.1538 > helios.domain: 3+ A? ucbvax.berkeley.edu  
(37)
```

- 3 — query id
- + — recursion
- A? — query type
- (37) — data length

tcpdump Output for UDP DNS Reply

```
helios.domain > h2opolo.1538:  3 3/3/7 A 128.32.137.3  
(273)
```

- 3 — query id
- 3/3/7 — 3 answer records, 3 name server records and 7 additional records
- A 128.32.137.3 — first record record is type A (answer) and IP address
- (273) — data length

```
helios.domain > h2opolo.1537:  2 NXDomain* 0/1/0 (97)
```

- 2 — query id

- `NXDomain` — response code of non-existent domain (`*` indicates that the authoritative answer bit was set)
- `0/1/0` — no answers, one name server and no authority records

`tcpdump` Output for ARP/RARP

- use `-n` option

```
arp who-has 128.3.254.6 tell 128.3.254.68
```

```
arp reply 128.3.254.6 is-at 02:07:01:00:01:c4
```

- 128.3.254.6 sent an arp packet asking for the Ethernet address of internet host csam 128.3.254.68
- 128.3.254.68 replies with its Ethernet MAC address

Terminating `tcpdump`

- Hit Ctrl-C prints a summary of all the traffic it saw
- Packets received by filter — count of packets processed by the `tcpdump` filter
- Packets dropped by kernel — number of packets that were dropped due to a lack of resources on your system

View All Traffic to and from a Particular Host

- To monitor only traffic to and from a specific host, filter everything else out with the `host` expression.
- can track network usage from/to a particular host

```
tcpdump -n host 192.168.1.1
```

```
tcpdump -n src host 192.168.1.1
```

```
tcpdump -n dst host google.com
```

Watch Only Traffic Coming in or out on a Certain Port

- To track usage of a certain application
- Use `tcpdump` to trap all traffic for a particular TCP/UDP port.
- Can see who is trying to access
- Useful when someone is trying an attack, e.g. SSH password attack

```
tcpdump -n port 22
```

View All Traffic to and from a Particular Host but Eliminate Some Kinds of Traffic

- Want to monitor a single host but want to filter out specific traffic
- e.g. if you were ssh'd into that host, unfiltered `tcpdump` output would show your own connection traffic
- Add `port` expression with a Boolean operator `not`
- Note that if using multiple expressions must also use Boolean operator `and`

```
tcpdump -n host 192.168.1.1 and not port 22
```

Find a Rogue Workstation

- If you are having network problems and suspect a rogue computer is swamping your network
- e.g. a bad network card or a trojanized PC causing a denial of service attack
- First try running it wide open to see what is generating the most traffic.
- Use the `-a` and `-e` options to generate names and MAC addresses

```
tcpdump -ae
```

- If this causes the output to scroll off the screen too fast, use the `-c 1000` option to only count 1,000 packets and then stop.

Monitor a Specific Workstation

- To log the traffic from a specific workstation to analyze later use `-w` switch to write to a file.
- `tcpdump -w logfile host 192.168.1.1`
- `logfile` is the file it will log to.
- May also use the `-c` or `-C` options to limit your output file size.

Look for Suspicious Network Traffic

- If you are worried about what is happening on your network after hours, leave `tcpdump` running to flag traffic you might deem questionable.
- Run it with the `gateway <IP Address>` flag set, where IP address is that of your own Internet gateway (assuming you have access to that gateway)
- Assuming your network was in the IP Range of 192.168.0.0 through 192.168.0.254, this would flag any traffic coming or going from your Internet gateway.
- If you have an internal server and don't want to log that traffic since that would be valid traffic, add the statement: `and host != <address of server>`
- The exclamation point also acts as the Boolean `not` operator.

```
tcpdump -w logfile gateway 192.168.0.1 and host !=  
192.168.0.2
```

- If you are looking for users using a particular application, e.g. streaming video or audio program, specify that using port number.

```
tcpdump -w logfile gateway 192.168.0.1 and host !=  
192.168.1.2 and dst port 1000
```

Look for abnormal traffic passing by or destined to your system

- To display start and end packets (SYN and FIN packets) of each TCP conversation involving a non-local host:

```
tcpdump 'tcp[13] & 3 != 0 and not src and dst net <localnet>'
```

- `<localnet>` is your local network
- Put in quotes so shell won't interpret punctuation characters
- To print traffic that does not have a local host as its source or destination:

```
tcpdump ip and not net <localnet>
```

- If your network has only one gateway between you and one other network, this traffic should *never* traverse your local network.

- To display IP broadcast or multicast packets that were not sent via Ethernet broadcast or multicast

```
tcpdump 'ether[0] & 1 = 0 and ip[16] >= 224'
```

- 'To display all ICMP packets that are not echo requests/replies (i.e., not ping packets)
- `tcpdump 'icmp[0] != 8 and icmp[0] != 0'`

Ethereal

- Like tcpdump with a friendly graphical interface
- Offers many more analytical and statistical options.
- Output is much easier to read and understand than the raw packet captures of Tcpdump.
- Can interpret over 300 different network protocols, which covers just about every network type ever invented.
- More physical network formats are supported.
- Output can be saved as plain text or in Postscript format.
- A rich display filter mode. This includes the ability to highlight certain packets in color.

- There is a filter creation GUI to walk you through the process of creating filters easily.
- The ability to follow a TCP stream and view the content in ASCII.
- The ability to work with dedicated hardware
- The ability to save sessions in multiple formats.
- A command-line terminal mode. (so can run on servers without GUI - can dump file for analysis in GUI)
- Can also use serve as a general network analysis tool.

Using Ethereal

- Choose Capture --> Start
- Choose Prepare
- Can do in real time or capture for later analysis

Capture Options

- Interface — Picks the interface to capture from the pull-down menu.
- Ethereal automatically senses all the available interfaces and lists them.
- You can also choose to capture from all interfaces at once, just like Tcpdump
- Limit each packet to x bytes — Sets a maximum size for the packets captured.
- Use this if you fear some of the packets may be very large and you don't want to overload your machine.
- Capture packets in promiscuous mode — on by default.

- Filter — creates a filter using `tcpdump`-style expressions.
- Can name the filter so can be used in future sessions
- Capture file(s) — if want to read from a file rather than capture live data.
- Display options — these are disabled by default, but enable them if you want to watch the packets scroll by in real time.
- Not recommended for busy network or slow machine is slow,
- Capture limits — automatic stop
- After x number of packets or kilobytes of data have been captured,
- After x number of seconds have elapsed.

- Name resolution — specify whether you want Ethereal to resolve names at various levels of the network model.
- Enabling all of these, especially DNS, can slow down your capture significantly.

Packet List Window

- The top third of the screen is where the packet stream is displayed in order of receipt,
- Can sort this in just about any way by clicking on the headings.
- Packet number — Assigned by Ethereal
- Time — The time the packet was received, set from the elapsed time from the start of the capture session.
- Alternately, this can be configured to show the clock time, the clock time and date, or even the time between packets (this is helpful for network performance analysis).
- Source address — Where the packet came from. This is an IP address on IP networks.

- Destination address —Where the packet is going to, also usually an IP address.
- Protocol — The level 4 protocol that the packet is using
- Info — Some summary information about the packet, usually a type field.

Packet Detail Window

- Goes into more detail on each packet that is highlighted.
- Arranged in an order that basically conforms to the OSI model, so the first item listed is detail on the data link layer, and so on.
- The little pluses can be expanded to show even more information on each level. It is amazing how much detail you can see on each packet.

Packet Content Window

- Actual packet contents, in both hexadecimal and translated into ASCII where possible.
- Binary files will still look like garbage, as will encrypted traffic, but anything in clear text will appear.
- Highlights the power (and danger) of having a sniffer on your network.

Preferences

- Allows you to change layout
- Allows you to modify and add headings
- Allows you to adjust ports and such for protocols

Display Filters

- Choose `Expression`
- Choose protocol and filter type
- Choose apply

Ethereal Session Statistics Window

- Displays protocol statistic
- You can stop your session at any time by clicking Stop
- If you set a limit in the options, it will automatically stop when it reaches it.

Analyze

- Follow TCP stream — shows connection traffic “conversation”

Statistics

tethereal

- `tethereal -w <filename>`
- `ethereal -r <filename>`

ntop

- `ntop -u root`
- first time asks for password
- run in browser <http://localhost:3000/>