CESS Hands On Network Traffic Capture and Analysis

European Union

ASIACONNECT
DISTRIBUTED AND CLOUD-BASED
NETWORK DEFENSE SYSTEM
FOR NRENs

Asi@Connect

TEIN
COOPERATION CENTER

FYI

- Download Putty (just the putty.exe)
- Server IP Address: 103.94.135.209
- User IDs
  - Username: user1 Pass: dcnds@01
  - Username: user2 Pass: dcnds@02
  - …
  - Username: user30 Pass: dcnds@30

What is network traffic analysis?

- A process of recording, reviewing, analyzing network traffic and making decisions based on the results

- Meta data contains important information
  - the sender
  - the receiver
  - the protocol
  - the time
  - length of messages

Why analyze network traffic?

- To answer the hard questions:
  - When did it happen?
  - How did it happen?
  - Who was behind it?
  - What was exposed or stolen?
- To monitor download/upload speeds, throughput, content, etc.
- To identify any malicious or suspicious packets.

Popular Tools

- **Tcpdump**
- Wireshark
- **Tshark**
- Windump

Tcpdump

- tcpdump is a unix tool.
- Allows user to intercept and display TCP/IP and other packets being transmitted or received over a network.
- Used to gather data from network, decipher the bits, and display the output in a human readable format.
- tcpdump uses the libpcap library to capture packets.
- Can be used to intercepting and displaying the communications of another user or computer

Installing tcpdump

- To check if we have installed tcpdump on our system:
  - dpkg -l | grep tcpdump [for Ubuntu]
  - rpm –q tcpdump [for CentOS]
- Output
  - tcpdump 4.9.2-0ubuntu0.16.04.1
  - tcpdump-4.9.2-3.el7.x86_64
- To install tcpdump:
  - apt-get install tcpdump [for Ubuntu]
  - yum -y install tcpdump [for CentOS]

Tcpdump Syntax

- Syntax:
  - tcpdump [options] [filter expression]
  - *Running tcpdump requires root-privilege.
    - sudo tcpdump
  - Continue capturing packets until it is interrupted. (ctrl + c)
  - packets captured: no. of packets that tcpdump has received and processed
  - packets received by filter: counts only packets that were matched by the filter expression *
  - packets dropped by kernel: due to lack of buffer space in OS

What does a line convey?

```
01:46:28.808262 IP dcnds.iict.buet.ac.bd.ssh >
vip0x00f.map2.ssl.hwmdd_cdn.portal.net.2481:
seq2215593012:2215593859(847) ack 2268385237 win
2048
```

- Different output formats for different packet types

Reading the tcpdump log

- seq (sequence number) are used by the destination host to reassemble TCP traffic that arrives.
  - Sequence number changes from absolute to relative value after the first two messages, giving ISNs, have been exchanged.
  - seq 2215593012 (Absolute)
  - seq 1:1025 (1024) (Relative to ISN): 1st through 1025th (not including 1025th) bytes have been sent
- ack (acknowledgement number) the sequence no. of the next data expected the other direction of this connection (ISN+1).

Reading the tcpdump log (cont.)

- win (window) receiving buffer size available in the other direction of the, used for flow control
- mss (maximum segment size)informs the destination host that the physical network of source host will not receive more than 1024 bytes of TCP payload.
  - If 20 bytes of IP header and 24 bytes of TCP header (including 4 bytes of mss option) are included, the IP datagram may be 1068 bytes.
- TCP Timestamp option puts the timestamp of the sender. Since it is of 10 bytes, so 2 bytes of NOP are used.
- sackOK (selective acknowledgement) indicates that it can be used for this session.

Filters

- We are often not interested in all packets flowing through the network
- Use filters to capture only packets of interest to us

Traffic Filtering in Tcpdump

- Filtering Interface:
    - sudo tcpdump -i any
- To list interfaces:
    - tcpdump -D
    - sudo tcpdump -i ens192
- Filtering Hosts :
  - Match any traffic involving any IP as destination or source
    - sudo tcpdump -i ens192 host 103.94.135.209
  - As soure only
    - sudo tcpdump -i ens192 src host 103.94.135.209
  - As destination only
    - sudo tcpdump -i ens192 dst host www.buet.ac.bd

Traffic Filtering in Tcpdump (cont.)

- Network filtering :
  - sudo tcpdump -i ens192 net 103.94
  - sudo tcpdump -i ens192 src net 103.94.0.0/16
  - sudo tcpdump -i ens192 dst net 103.94

- Protocol filtering :
  - sudo tcpdump -i any arp
  - sudo tcpdump -i ens192 ip

- Similarly we can use tcp, udp, icmp, etc.

Traffic Filtering in Tcpdump (cont.)

- Filtering ports :
  - Match any traffic involving port 22 as source or destination
    - sudo tcpdump -i ens192 port 22
  - Source
    - sudo tcpdump -i ens192 src port 443
  - Destination
    - sudo tcpdump -i ens192 dst port 53
    - sudo tcpdump -i ens192 portrange 53-80

Traffic Filtering in Tcpdump (cont.)

- Capture only TCP packets with https requests
  ```
  sudo tcpdump -i any tcp and port 443
  ```
- Capture only TCP packets http requests:
  ```
  sudo tcpdump -i any tcp && http
  ```
- Capture only UDP packets with DNS replies or http requests:
  ```
  sudo tcpdump -i any port 53 or http
  sudo tcpdump -i any port 53 || http
  ```
- Capture packets and not ARP packets:
  ```
  sudo tcpdump -i ens192 not arp
  sudo tcpdump -i ens192 ! arp
  ```

Write & Read a Captured file

- To write the packets to a file:
  - sudo tcpdump -i ens192 port 443 -w write.pcap
  - sudo tcpdump -i ens192 port 443 -w write.pcapng
- To read from the packets:
  - sudo tcpdump -r write.pcap
- To view the details of the captured file:
  - capinfos http.pcap

Commonly Used Filtering Options

- -c : exit after receiving count packets
- -e : to print the link level header (eg. MAC addresses)
- –n : do not to resolve the IP address into names
- -nn : do not convert the protocol and port number into names
- -s : snaplen (snapshot length)

  Default packet size is 262144 bytes or 65535 bytes
- -# : to print an optional packet number at the beginning of the line

Commonly Used Filtering Options

- -t : don’t print a timestamp on each dump line
- -ttt : to print a delta (between current and previous line on eac dump line.
- -tttt : print a timestamp as hours, minutes, seconds and fractions of a second since midnight.
- -v : slightly more verbose output
- -vv : even more verbose output
- -A : to print each packet in ASCII.
- -X : to print the data of each packet in hex and ASCII.

Converting .pcap file to .csv

-
  ```
  tshark -r tcp.pcap -T fields -E separator=, -E header=y -e frame.number -e frame.time_relative -e ip.src -e ip.dst -e ip.proto -e frame.len -e frame.cap_len -e ip.hdr_len -e tcp.hdr_len -e tcp.flags > tcp.csv
  ```

Traffic Analysis

- To list the hosts (ip address and name) from a captured file
  - tshark -r dcnds.pcap -q -z hosts
- To show the warning packets and related information of the captured file:
  - tshark -r dcnds.pcap -q -z expert,warn
- To show http statistics
  - tshark -r http.pcap -q -z http,stat
  - tshark -r http.pcap -q -z http,tree

Traffic Analysis (cont.)

- To lists the hosts with counts
  - `tshark -r http.pcap -q -z ip_hosts,tree`
  - `tshark -r http.pcap -q -z ip_srcdst,tree`
- To show protocol hierarchy statistics
  - `tshark -r http.pcap -q -z io,phs`
- To see who is doing what:
  - `tshark -r http.pcap -q -z conv,ip`

Capturing MAC address

- sudo tcpdump -i ens192 -nn -e -#c5 -tttt port 443

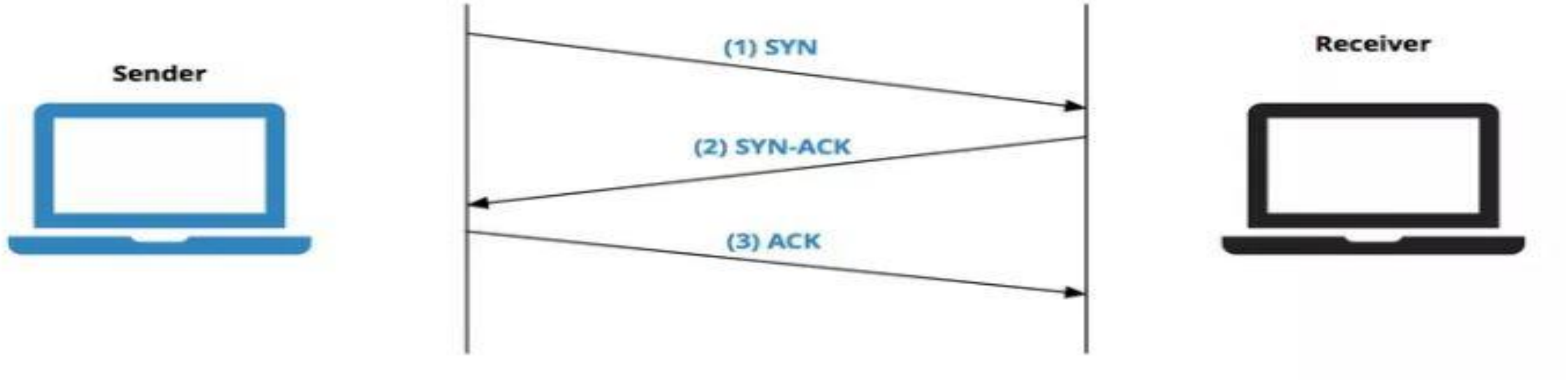
- Output:
  - 2019-02-03 02:51:25.793143 **00:0c:29:35:6a:d3** > **e4:8d:8c:1b:26:50**, ethertype IPv4 (0x0800), length 437: 172.16.8.49.42008 > 74.125.200.105.443: Flags [P.], seq 2343668656:2343669027, ack 370423994, win 2182, options [nop,nop,TS val 31475980 ecr 3405846089], length 371

Fixed number of packet

- Capture a fixed number of packets using “-c” flag
  - sudo tcpdump -i ens192 port 443 -#c1
  - sudo tcpdump -i ens192 port 80 -#c6

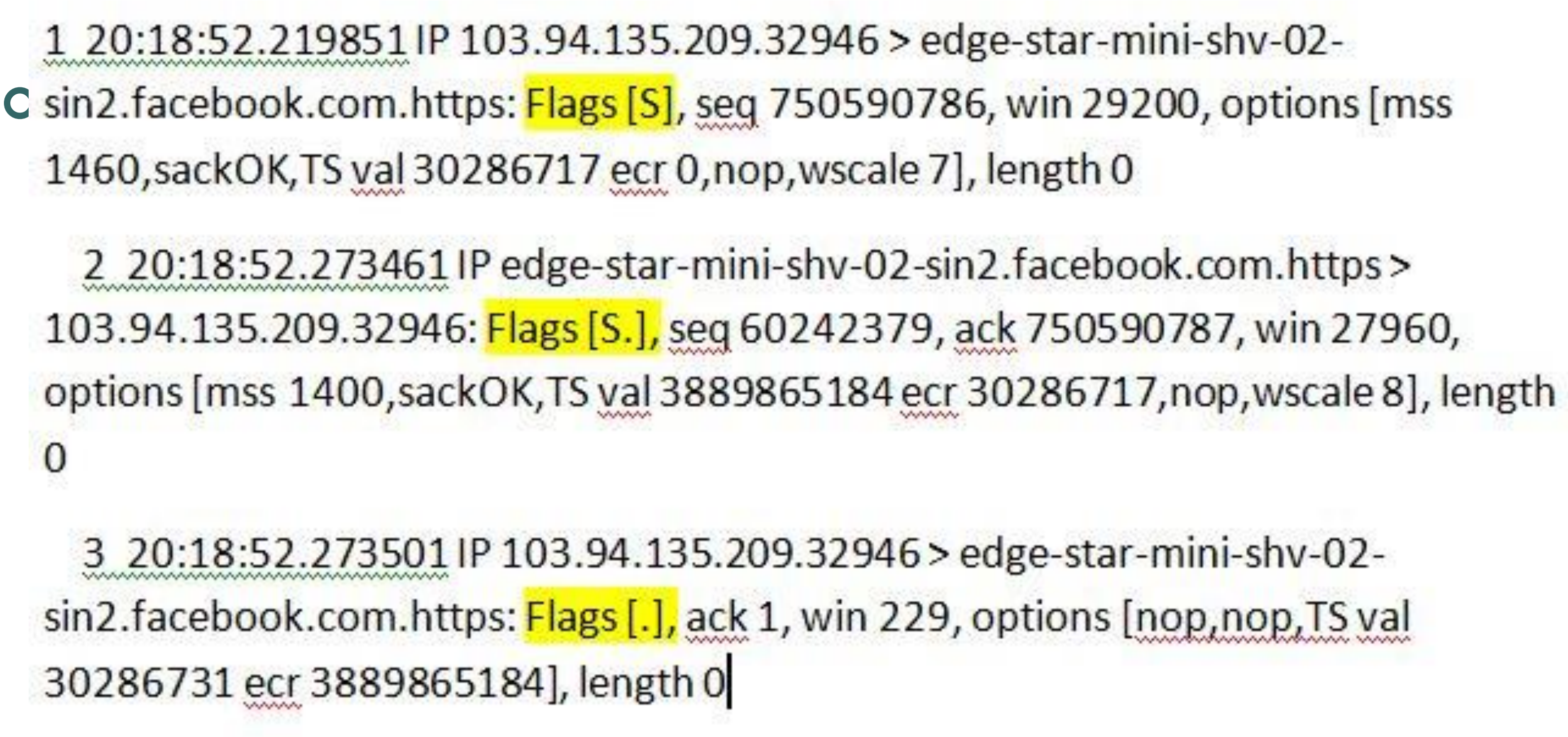
Connection stablish

- tcpdump –I interfaceName host www.facebook.com



- wget www.facebook.com

Handshake



```
1 20:18:52.219851 IP 103.94.135.209.32946 > edge-star-mini-shv-02-
sin2.facebook.com.https: Flags [S], seq 750590786, win 29200, options [mss
1460,sackOK,TS val 30286717 ecr 0,nop,wscale 7], length 0

   2 20:18:52.273461 IP edge-star-mini-shv-02-sin2.facebook.com.https >
103.94.135.209.32946: Flags [S.], seq 60242379, ack 750590787, win 27960,
options [mss 1400,sackOK,TS val 3889865184 ecr 30286717,nop,wscale 8], length
0

   3 20:18:52.273501 IP 103.94.135.209.32946 > edge-star-mini-shv-02-
sin2.facebook.com.https: Flags [.], ack 1, win 229, options [nop,nop,TS val
30286731 ecr 3889865184], length 0
```

Make a list of IP’s

- Step 1: read a pcap file
  - tcpdump -nnr dcnds.pcap

Make a list of IP’s

- Step 2: take a row from the packet
  - tcpdump -nnr dcnds.pcap |awk '{print $3}'

Make a list of IP’s

- Step 3: split and cut

  -
    ```
    tcpdump -nnr dcnds.pcap |awk '{print $3}'|cut -d. -f1,2,3,4
    ```

Make a list of IP’s

- Step 4: sorting

  -
    ```
    tcpdump -nnr dcnds.pcap |awk '{print $3}'|cut -d. -f1,2,3,4 | sort
    ```

Make a list of IP’s

- Step 5: uniq sort

  -
    ```
    tcpdump -nnr dcnds.pcap |awk '{print $3}'|cut -d. -f1,2,3,4 | sort | uniq -c
    ```

Make a list of IP’s

- Step 6: Sorting by frequency

  ```
  tcpdump -nnr dcnds.pcap |awk '{print $3}'|cut -d. -f1,2,3,4 | sort | uniq -c | sort -nr
  ```

- Output: we will get a list of IP used in a captured file

Number of unique IP used by source

```
tcpdump -nnr dcnds.pcap  tcp[tcpflags]==2 and  src host 103.94.135.209 | awk '{print $5}' | cut -d. -f1,2,3,4 | sort |uniq -c |wc -l
```

- $5 represents the destination IPs
- Output : count the number of unique IPs this host is trying to talk to
- So 103.94.135.209 has tried to connect to “x” unique destination IPs.
- Output : “X(number of unique destination IP)”

Ports Between src and dst

- For a particular source and destination IP pair, lets see how many ports are hit

  -
    ```
    tcpdump -nnr dcnds.pcap  tcp[tcpflags]==2 and src host 103.94.135.209 and dst 74.125.200.128 |  awk '{print $5}' | cut -d. -f5 | sort |uniq -c |wc -l
    ```
- Output: will show the number of ports used by the source and destination IPs

Count the frequency of the port used

```
tcpdump -nnr dcnds.pcap  tcp[tcpflags]==2 and src host 103.94.135.209 and dst 74.125.200.128 |  awk '{print $5}' | cut -d. -f5 | sort |uniq -c
```

- Output: How many times the ports used by the source and destination

Sniffing userName and Password

- Only ASCII value:
  - sudo tcpdump -i ens192 -A port 80
- (now run another window to run http websites. For example: “wget www.teletalk.com.bd”)

Sniffing User Name and Password

  - sudo tcpdump -i ens192 port http -l -A | egrep -i -B5 'username=|password='

- Browse in a http website and provide user name and password (in another terminal or using VNC)
  - w3m www.teletalk.com.bd

- Output: username and password is seen when we are in the same network interface