

tcpdump

List the interfaces

```
sudo tcpdump -D
```

Capture All traffic

```
tcpdump -i eth0  
tcpdump -i wlan0
```

To a File

```
tcpdump -i eth0 -w capture.pcap  
tcpdump -i any -w capture.pcap -nn 'ip and port 80'  
  
# Set Timeout  
timeout 6m tcpdump -i eth0 -w capture.pcap
```

Read a file (.pcap)

- `-nn` : Disable port and protocol name lookup.
- `-r` : Read capture data from the named file.
- `-v` : Display detailed packet data.
- `-X` : Display the hexadecimal and ASCII output format packet data. Security analysts can analyze hexadecimal and ASCII output to detect patterns or anomalies during malware analysis or forensic analysis.

```
tcpdump -r capture.pcap  
tcpdump -r capture.pcap -nn -v 'ip and (port 80 or port 443)'  
tcpdump -nn -r capture.pcap -X
```

Filter

```
# Filter by Source IP  
tcpdump src 192.168.0.1
```

```
# Filter by Destination IP
```

```
tcpdump dst 192.168.0.1
```

```
# Filter by Port
```

```
tcpdump port 80
```

```
# Filter by Protocol
```

```
tcpdump icmp
```

```
# Protocol and Port
```

```
tcpdump tcp port 443
```

```
# Source and Destination
```

```
tcpdump src 192.168.0.1 and dst 192.168.0.2
```

```
tcpdump -i any -w capture.pcap -n 'ip and port 80'
```

Display in ASCII

```
# Display in ASCII
```

```
tcpdump -A
```

```
# Display in Hexadecimal
```

```
tcpdump -X
```

Specific Number of Packets

```
tcpdump -c 100
```

Display

```
# Capture and Display IPv6 Traffic
```

```
tcpdump -6
```

```
# Capture and Display Traffic in Timestamp Format
```

```
tcpdump -tttt
```

SSH Connections

```
# -l: real-time
# -e: including ethernet headers
tcpdump -i eth0 'tcp port 22' -l -e
```

HTTP Request and Response

```
tcpdump -i eth0 -s 0 -A -n 'tcp dst port 80'
```

IP Range and Protocol

```
tcpdump -i eth0 'net 192.168.0.0/24 and (tcp port 22 or icmp)'
```

DNS Traffic

```
tcpdump -i eth0 'udp port 53' -nnvvv
```

FTP Traffic

```
tcpdump -i eth0 -s 0 'tcp port 21'
```

?? DDos ??????????

```
interface=ens1
dumpdir=/home/user/automatic-tcp-dump/
while /bin/true; do
    pkt_old=`grep $interface: /proc/net/dev | cut -d : -f2 | awk '{ print $2 }'`
    sleep 1
    pkt_new=`grep $interface: /proc/net/dev | cut -d : -f2 | awk '{ print $2 }'`
    pkt=$(( $pkt_new - $pkt_old ))
    echo -ne "\r$pkt packets/s\033[0K"
    if [ $pkt -gt 30000 ]; then
        echo -e "\n`date` Under Attack. Capturing Packets..."
        sudo tcpdump -n -i $interface -s0 -c 20000 -w $dumpdir/dump.`date +%Y%m%d-%H%M%S`.pcap
        echo "`date` Packets Captured."
        sleep 300 && pkill -HUP -f /usr/sbin/tcpdump
    else
        sleep 1
    fi
done
```



Tcpdump Command Examples

- ✓ **tcpdump** listen on the first non-loopback interface detected
- ✓ **tcpdump -i eth0** capture packets on eth0 and display their content
- ✓ **tcpdump -i eth0 -w my.pcap** save packets received on eth0 to my.pcap
- ✓ **tcpdump -i any** capture packets from all available interfaces
- ✓ **tcpdump arp|tcp|udp|icmp** capture only a specific protocol
- ✓ **tcpdump src 10.0.0.1** capture traffic from 10.0.0.1
- ✓ **tcpdump port 80** capture traffic with ether src/dst port 80
- ✓ **tcpdump dst net 10.1.1.0/24** capture traffic for specific subnet
- ✓ **tcpdump tcp and src 10.0.0.1 and port 80** combine multiple filters
- ✓ **tcpdump tcp dst portrange 22-1023** capture packets with port range
- ✓ **tcpdump -vvv** show protocol-specific info with full verbosity
- ✓ **tcpdump -tt** use UNIX timestamp as packet timestamp format
- ✓ **tcpdump not port 22** capture all traffic except ssh traffic
- ✓ **tcpdump -c 1000** capture the first 1000 packets only
- ✓ **tcpdump -n** do not convert IP addresses/ports to names
- ✓ **tcpdump -e** display layer-2 info such as MAC addresses
- ✓ **tcpdump -X** show payload content in hex/ASCII format
- ✓ **tcpdump ip6** capture IPv6 packets only
- ✓ **tcpdump 'tcp port 80 or udp port 67'** use complex filters
- ✓ **tcpdump greater 200** capture packets whose length > 200
- ✓ **tcpdump ether dst ff:ff:ff:ff:ff:ff** capture layer-2 broadcast packets
- ✓ **tcpdump 'tcp[tcpflags] == tcp-syn'** capture TCP SYN packets
- ✓ **tcpdump 'tcp[tcpflags] & (tcp-syn|tcp-fin) != 0'** match TCP SYN or FIN
- ✓ **tcpdump -e vlan 10** capture traffic with VLAN tag 10
- ✓ **tcpdump 'icmp[0] = 8'** capture ICMP echo request packets (ping)
- ✓ **tcpdump outbound** capture only outbound traffic



Created by
Dan Nanni
study-notes.org