Network Lab2

Ex. 1: File: /ex1-tcpdump...

Ex. 2: File: */ wireshark_ex2*

Ex. 3: Files: / wireshark_host_ex3; /wireshark_src_ex3

Ex 4: Files: / wire_ICMP_ex4; /wire_tcp_ex4; / wire_tcp_port_ex4

Ex. 5: Files: / arp_ex5

Questions about recorded data in this Exercise:

1. What is the typical destination MAC address of an ARP Request packet?

The typical MAC-ADDRESS is ff:ff:ff:ff:ff:ff. (Broadcast Adress).

2. What are the different values of the Type field in the Ethernet headers that you captured?

The Type field gives information about the protocol in the payload.

User Datagram Protocol

Internet Control Message Protocol

Address Resolution Protocol (request)

Address Resolution Protocol (relpy)

3. Describe the process which ARP uses to acquire a MAC address for a given IP address.

When ARP needs to resolve a given IP address to Ethernet address, it broadcasts an ARP request packet. The ARP request packet contains the source MAC address and the source IP address and the destination IP address. Each host in the local network receives this packet. The host with the specified destination IP address sends an ARP reply packet to the originating host with its MAC address.

Ex. 6: / mac_adress_ex6

```
10.10.10.14 dev eth0 lladdr 00:18:8b:23:e8:27 REACHABLE
10.10.10.13 dev eth0 lladdr 00:21:9b:77:11:eb REACHABLE
10.10.10.12 dev eth0 lladdr 00:18:8b:1a:be:17 STALE
```

Ex7: File: / failedarp_ex7

1. Use the saved data to determine the interval between each ARP Request. Describe the method used by ARP to determine the time between retransmissions of an unsuccessful ARP Request. Include the relevant parts of your save data in the lab report.

For each request the ARP tries to retransmission 2 times after a failed request with a time interval of 1 second. ARP determines the time between retransmissions by looking up the set time in */proc/sys/net/ipv4/neigh/interface/retrans_time_ms* . The default is set to 1000 ms.

2.

```
No.   Time       Source              Destination          Protocol Info

 1 0.000000   00:18:8b:1a:be:34    ff:ff:ff:ff:ff:ff    ARP     Who has 10.10.10.10?  Tell 10.10.10.11
 2 0.999454   00:18:8b:1a:be:34    ff:ff:ff:ff:ff:ff    ARP     Who has 10.10.10.10?  Tell 10.10.10.11
 3 1.999456   00:18:8b:1a:be:34    ff:ff:ff:ff:ff:ff    ARP     Who has 10.10.10.10?  Tell 10.10.10.11
```

Ethernet II, Src: Dell_1a:be:34 (00:18:8b:1a:be:34), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request)
Frame 1 (42 bytes on wire, 42 bytes captured)

2. Why are ARP packets not encapsulated like IP packets?

Because ARP-packages never leave the Local Area Network (LAN).

Ex8:

1. File: / ip_stattistics_ex8; / iproute_ex8
2. What are the network interfaces of PC1 and what are their respective maximum transmission unit (MTU) values?

```
lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN

eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000

eth1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state
DOWN qlen 1000

pan0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN
```

Ex9:

File: / ntsat_ex9

	Transmitted/requests/out	Received/in
IP datagram	0	0
ICMP	0	0
UDP	0	0
TCP	0	0

Ex10: File: / ipaddr_1_ex10; / ipaddr_2_ex10

1. Explain the output of `ip addr`.

The output of command "ip addr" or "ip a" is a summary of the momentary configuration of all network interfaces of the PC. It for example includes the IP-address, the mtu value, the gateway, and whether or not an interface is connected or not.

Ex11: File: / ip_neigh_ex11; / wire_arp_tcp_ex11

1.Explain: the host determined in step 4

PC4, because it is the last known PC with the used IP-address

2.and why the telnet session was estabished at all and did not just return an error message.

The ARP request returned a legal destination for the telnet-connection, so a session could be established

Ex12: Files: /netmasks_wiresh_ex12; /ping_pc1-pc2_ex12; /ping_pc1-pc3_ex12 ; /ping_pc1-pc3_ex12; /ping_pc2-pc3_ex12; /ping_pc2-pc4_ex12; /ping_pc4-pc1_ex12

PC1 to PC3: ICMP message could be send
PC1 to PC2: ICMP message could be send
PC1 to PC4: Destination Host Unreachable (different subnet)

PC2 to PC4: ICMP message could be send
PC2 to PC3: ICMP message could be send because ARP stored destination address already

PC4 to PC1: Destination Host Unreachable (different subnet)

Ex13:

File:/etchosts_ex13

1.Explain why the presented method is impractical for networks with a large number of hosts

Everytime an ip address changes, you need to change the respective line in the file. Doing that manually, you can get lost quite easily. Also, it is problematic that if you don't have a synchonization server (like puppet), the file has to be correct and up to date on every single computer, which may be a problem. Using a DNS server, the list has to be present only once and every PC in the net can ask the DNS server.

2.What will be the result of the host name resolution process if there are more thatn one IP addresses associated with the same host name in `/etc/hosts`?

If two or more IP adresses are associated with the same host name, the first address in the file with the name wanted will be taken. Therefore it is kind of impossible to get the other adresses with this host name.

Ex14: File: /ftp_ex14

1. identify the port numbers used by ftp client and ftp server

Port client: 37414 / Port Server: 21

2.find out the login name and password of the ftp user.

User: ubuntu / Password: lab

Ex15: File: /telnet_ex15

Does telnet have the same security flaws as FTP? Where in your saved data can you find the credentials?

The same flaws exists. If the option “Follow TCP stream” is used, the username and password can be read in plain text as well.

Ex16: File: /telnet_ex16

3 Packets are sent with the Telnet-protocol and 1 with the TCP protocol for lower-layer communication. 2 Telnet-packets because PC2 gives a feedback for each character even if it does not result in any action.