Introduction to Web Science

Assignment 1

TANGO

Mariya Chkalova
mchkalova@uni-koblenz.de

Arsenii Smyrnov
smyrnov@uni-koblenz.de

Simon Schauß
sschauss@uni-koblenz.de

1 Ethernet Frame (5 Points)

Ethernet Frame is of the given structure:

Preamble	Destination MAC address	Source MAC address	Type/Length	User Data	Frame Check Sequence (FCS)
8	6	6	2	46 - 1500	4

Figure 1: Ethernet Frame Structure

```
00 27 10 21 fa 48 00 13     10 e8 dd 52 08 06 00 01
08 00 06 04 00 01 00 13     10 e8 dd 52 c0 a8 02 01
00 00 00 00 00 00 c0 a8     02 67
```

Find:

1. Source MAC Address
2. Destination MAC Address
3. What protocol is inside the data payload?
4. Please mention what the last 2 fields hold in the above frame.

Answer:

1. Source MAC Address: `00:13:10:e8:dd:52`
2. Destination MAC Address: `00:27:10:21:fa:48`
3. Protocol: Address Resolution Protocol
4. The penultimate field is the targets MAC Address and the last field is the targets IP Address.

2 Cable Issue (5 Points)

Let us consider we have two cables of 20 meters each. One of them is in a 100MBps network while the other is in a 10MBps network. If you had to transfer data through each of them, how much time it would take for the first

bit to arrive in each setting? (For your calculation you can assume that the speed of light takes the same value as in the videos.) Please provide formulas and calculatoins along with your results.
Answer:
Let c be the speed of light, l the length of the cable and t the time it takes for the first bit to travel the length l. As the length of the cables are equal and the networks bandwidth doesn't change the propagation delay, the calculation for both networks are the same. Given the speed of light $c = 3 \cdot 10^8 \frac{m}{s}$ and the formula for the propagation delay $t = \frac{l}{c}$, the propagation delay is $t = \frac{20}{3 \cdot 10^8} s \approx 67ns$

3 Basic Network Tools (10 Points)

Listed below are some of the commands which you need to "google" to understand what they stand for:

1. *ipconfig / ifconfig*
2. *ping*
3. *traceroute*
4. *arp*
5. *dig*

Consider a situation in which you need to check if `www.wikipedia.org` is reachable or not. Using the knowledge you gained above to find the following information:

1. The *% packet loss* if at all it happened after sending 100 packets.
2. *Size* of the packet sent to *Wikipedia* server
3. *IP address* of your machine and the *Wikipedia* server
4. *Query Time* for DNS query of the above url.
5. Number of *Hops* in between your machine and the server
6. MAC address of the device that is acting as your network gateway.

Do this once in the university and once in your home/dormitory network. With your answers, you must paste the screen shots to validate your find.
Answer:
1. The % packet loss if at all it happened after sending 100 packets.
Home: 0%
University: 0%

Listing 1: ping home

```
ping -c 100 -i 0.2  www.wikipedia.de
...
100 packets transmitted, 100 received, 0%
   packet loss, time 19883ms
rtt min/avg/max/mdev =
   18.037/21.074/29.851/1.646 ms
```

Listing 2: ping university

```
100 packets transmitted, 100 received, 0%
   packet loss, time 19880ms
rtt min/avg/max/mdev =
   9.323/10.315/19.560/1.547 ms
```

2. Size of the packet sent to Wikipedia server.
Home: 64 bytes
University: 64 bytes

Listing 3: man ping

```
          -s packetsize
                 Specifies the number of
                    data bytes to be sent.
                     The default is 56,
                    which translates into
                    64 ICMP data
                 bytes when combined with
                    the 8 bytes of ICMP
                    header data.
```

3. IP address of your machine and the Wikipedia server
Home: 192.168.2.115, 91.198.174.192
University: 141.26.186.205, 91.198.174.192

Listing 4: ifconfig home

```
ifconfig
...
bond0: flags=5187<UP,BROADCAST,RUNNING,MASTER,
   MULTICAST>  mtu 1500
inet 192.168.2.115  netmask 255.255.255.0
   broadcast 192.168.2.255
inet6 fd21:22dd:f528:1:d6b5:5652:241e:f450
   prefixlen 64  scopeid 0x0<global>
inet6 fd21:22dd:f528:1:f2de:f1ff:fe03:c9c9
   prefixlen 64  scopeid 0x0<global>
inet6 2003:c5:5bd7:2653:d8fd:5b7d:730d:9337
   prefixlen 64  scopeid 0x0<global>
inet6 2003:c5:5bd7:2653:f2de:f1ff:fe03:c9c9
   prefixlen 64  scopeid 0x0<global>
inet6 fe80::f2de:f1ff:fe03:c9c9  prefixlen 64
   scopeid 0x20<link>
ether f0:de:f1:03:c9:c9  txqueuelen 1000  (
   Ethernet)
RX packets 7563  bytes 6410345 (6.1 MiB)
RX errors 0  dropped 0  overruns 0  frame 0
TX packets 5621  bytes 1106251 (1.0 MiB)
TX errors 0  dropped 0 overruns 0  carrier 0
   collisions 0
```

Listing 5: arp www.wikipedia.org home

```
arp www.wikipedia.org
...
www.wikipedia.org (91.198.174.192) -- no entry
```

Listing 6: ipconfig university

```
ifconfig
...
bond0: flags=5187<UP,BROADCAST,RUNNING,MASTER,
   MULTICAST>  mtu 1500
inet 141.26.186.205  netmask 255.255.240.0
   broadcast 141.26.191.255
```

```
        inet6 fe80::f2de:f1ff:fe03:c9c9  prefixlen 64
           scopeid 0x20<link>
        ether f0:de:f1:03:c9:c9  txqueuelen 1000  (
           Ethernet)
        RX packets 16144  bytes 2051137 (1.9 MiB)
        RX errors 0  dropped 391  overruns 0  frame 0
        TX packets 148  bytes 18728 (18.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0
           collisions 0
```

Listing 7: arp www.wikipedia.org university

```
        arp www.wikipedia.org
        ...
        www.wikipedia.org (91.198.174.192) -- no entry
```

4. Query Time for DNS query of the above url.
Home: 3msec
University: 82msec

Listing 8: dig home

```
        dig www.wikipedia.org
        ...
        ; <<>> DiG 9.11.0 <<>> www.wikipedia.org
        ;; global options: +cmd
        ;; Got answer:
        ;; ->>HEADER<<- opcode: QUERY, status: NOERROR,
            id: 7395
        ;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1,
           AUTHORITY: 0, ADDITIONAL: 0

        ;; QUESTION SECTION:
        ;wikipedia.org.                 IN      A

        ;; ANSWER SECTION:
        www.wikipedia.org.              38      IN
                A       91.198.174.192

        ;; Query time: 3 msec
```

```
;; SERVER: 192.168.2.1#53(192.168.2.1)
;; WHEN: Wed Nov 02 06:22:14 UTC 2016
;; MSG SIZE  rcvd: 47
```

```
dig www.wikipedia.org
...
; <<>> DiG 9.11.0 <<>> www.wikipedia.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR,
    id: 478
;; flags: qr rd ra; QUERY: 1, ANSWER: 1,
   AUTHORITY: 6, ADDITIONAL: 13

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.wikipedia.org.              IN      A

;; ANSWER SECTION:
www.wikipedia.org.      456     IN      A
          91.198.174.192

;; AUTHORITY SECTION:
org.                    165879  IN      NS
        c0.org.afilias-nst.info.
org.                    165879  IN      NS
        d0.org.afilias-nst.org.
org.                    165879  IN      NS
        a0.org.afilias-nst.info.
org.                    165879  IN      NS
        a2.org.afilias-nst.info.
org.                    165879  IN      NS
        b2.org.afilias-nst.org.
org.                    165879  IN      NS
        b0.org.afilias-nst.org.

;; ADDITIONAL SECTION:
```

```
a0.org.afilias-nst.info. 165879 IN      A
        199.19.56.1
a0.org.afilias-nst.info. 165879 IN      AAAA
        2001:500:e::1
a2.org.afilias-nst.info. 165879 IN      A
        199.249.112.1
a2.org.afilias-nst.info. 165879 IN      AAAA
        2001:500:40::1
b0.org.afilias-nst.org. 165879  IN      A
        199.19.54.1
b0.org.afilias-nst.org. 165879  IN      AAAA
        2001:500:c::1
b2.org.afilias-nst.org. 165879  IN      A
        199.249.120.1
b2.org.afilias-nst.org. 165879  IN      AAAA
        2001:500:48::1
c0.org.afilias-nst.info. 165879 IN      A
        199.19.53.1
c0.org.afilias-nst.info. 165879 IN      AAAA
        2001:500:b::1
d0.org.afilias-nst.org. 165879  IN      A
        199.19.57.1
d0.org.afilias-nst.org. 165879  IN      AAAA
        2001:500:f::1

;; Query time: 81 msec
;; SERVER: 141.26.64.60#53(141.26.64.60)
;; WHEN: Wed Nov 02 07:51:43 UTC 2016
;; MSG SIZE  rcvd: 464
```

5. Number of Hops in between your machine and the server
Home: didn't complete after 100+ hops
University: 11

Listing 9: traceroute university

```
traceroute -I www.wikipedia.org
...
traceroute to www.wikipedia.org
```

```
    (91.198.174.192), 30 hops max, 60 byte
    packets
1   wlanrouter.uni-koblenz.de (141.26.176.1)
    3.303 ms  6.096 ms  6.104 ms
2   g-uni-ko-1.rlp-net.net (217.198.241.129)
    6.123 ms  6.124 ms  8.683 ms
3   g-hbf-ko-1.rlp-net.net (217.198.240.69)
    6.102 ms  8.684 ms  8.684 ms
4   g-hbf-mz-2.rlp-net.net (217.198.240.21)
    8.684 ms  8.684 ms  11.187 ms
5   g-interxion-1.rlp-net.net (217.198.240.13)
    11.206 ms  14.308 ms  14.319 ms
6   r1fra3.core.init7.net (80.81.192.67)  14.327
     ms  5.593 ms  4.234 ms
7   r1ams1.core.init7.net (77.109.128.154)
    16.680 ms  16.691 ms  16.697 ms
8   r1ams2.core.init7.net (77.109.128.146)
    18.935 ms  18.934 ms  19.091 ms
9   gw-wikimedia.init7.net (77.109.134.114)
    18.920 ms  18.915 ms  18.909 ms
10   ae1-403.cr2-esams.wikimedia.org
    (91.198.174.254)  18.904 ms  18.914 ms
    18.888 ms
11   text-lb.esams.wikimedia.org
    (91.198.174.192)  18.898 ms  21.779 ms
    21.767 ms
```

6. MAC address of the device that is acting as your network gateway.
Home: d4:21:22:dd:f5:28
University: 14:18:77:45:b1:bd

Listing 10: arp home

```
arp -n
...
Address                  HWtype  HWaddress
                Flags Mask            Iface
192.168.2.1              ether   d4:21:22:dd:f5
   :28    C                      bond0
```

Listing 11: arp university

```
arp -n
...
Address                  HWtype  HWaddress           Flags Mask            Iface
141.26.176.1             ether   14:18:77:45:b1:bd   C                     bond
```

4 Simple Python Programming

Write a simple python program that does the following:

1. Generate a random number sequence of 10 values between 0 to 90.
2. Perform `sine` and `cosine` operation on numbers generated.
3. Store the values in two different arrays named SIN & COSIN respectively.
4. Plot the values of SIN & COSIN in two different colors.
5. The plot should have labeled axes and legend.

Answer:

```
# Maps the values of an empty array of length 10 to a
    random number between 0 and 90.
import random

randlist = list(map(lambda x: random.randint(0,90), [
    None] * 10))

# Maps the previous generated random numbers to their
    sine and cosine values.
import math

sin = list(map(math.sin, randlist))
cos = list(map(math.cos, randlist))
```

```
# First plot the values of the sin and cos list, set
    the line type to dots and define the labels. Then
    the legend for the labels is created. After that the
     viewport is defined. Finally the plot is rendered.
import matplotlib.pyplot as plt

plt.plot(sin, "o", label="sine")
plt.plot(cos, "o", label="cosine")

plt.legend(loc=1, borderaxespad=0, numpoints=1)
plt.xlim([-1, 14])
plt.ylim([-5/4, 5/4])
plt.show()
```