

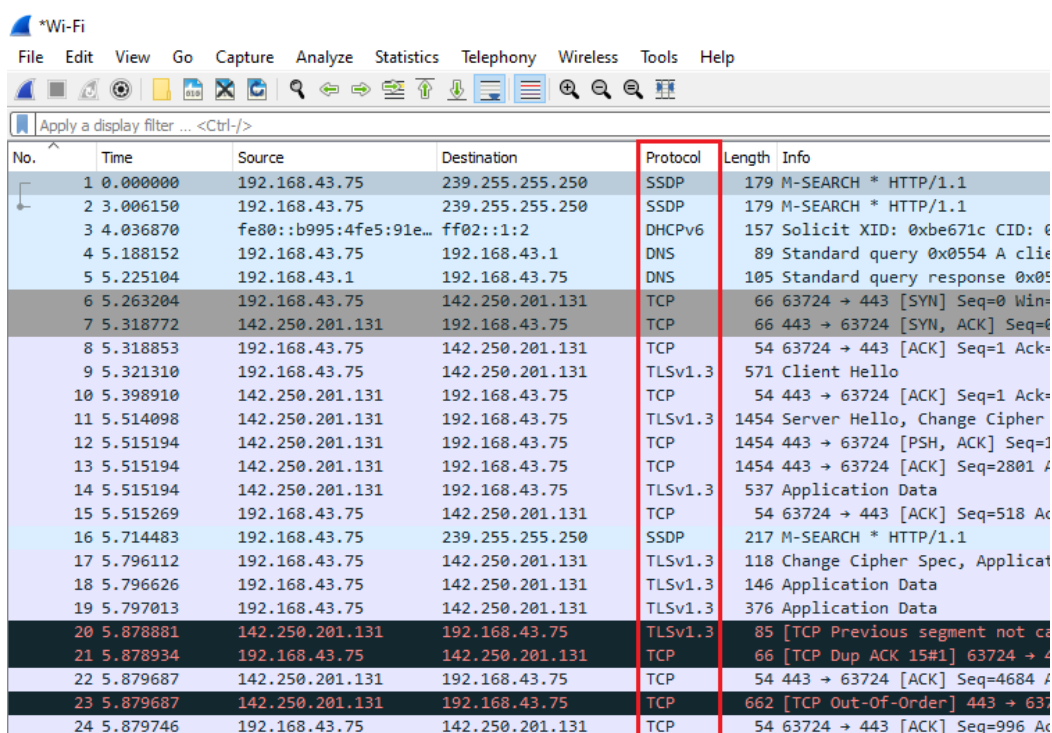
# گزارش دستور کار دوم آزمایشگاه درس شبکه‌های کامپیوتری

نگار موقتیان، ۹۸۳۱۰۶۲

۱. پروتکل‌های مشاهده شده در بسته‌های شنود شده

لیستی از پروتکل‌های مشاهده شده شامل موارد زیر است:

TLSv1.3 – TLSv1.2 – TCP – SSLv2 – SSDP – QUIC – MDNS – HTTP – DNS – DHCPv6 – ARP



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.43.75	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
2	3.006150	192.168.43.75	239.255.255.250	SSDP	179	M-SEARCH * HTTP/1.1
3	4.036870	fe80::b995:4fe5:91e...	ff02::1:2	DHCPv6	157	Solicit XID: 0xbe671c CID: 0
4	5.188152	192.168.43.75	192.168.43.1	DNS	89	Standard query 0x0554 A cli
5	5.225104	192.168.43.1	192.168.43.75	DNS	105	Standard query response 0x05
6	5.263204	192.168.43.75	142.250.201.131	TCP	66	63724 → 443 [SYN] Seq=0 Win=
7	5.318772	142.250.201.131	192.168.43.75	TCP	66	443 → 63724 [SYN, ACK] Seq=6
8	5.318853	192.168.43.75	142.250.201.131	TCP	54	63724 → 443 [ACK] Seq=1 Ack=
9	5.321310	192.168.43.75	142.250.201.131	TLSv1.3	571	Client Hello
10	5.398910	142.250.201.131	192.168.43.75	TCP	54	443 → 63724 [ACK] Seq=1 Ack=
11	5.514098	142.250.201.131	192.168.43.75	TLSv1.3	1454	Server Hello, Change Cipher
12	5.515194	142.250.201.131	192.168.43.75	TCP	1454	443 → 63724 [PSH, ACK] Seq=1
13	5.515194	142.250.201.131	192.168.43.75	TCP	1454	443 → 63724 [ACK] Seq=2801 A
14	5.515194	142.250.201.131	192.168.43.75	TLSv1.3	537	Application Data
15	5.515269	192.168.43.75	142.250.201.131	TCP	54	63724 → 443 [ACK] Seq=518 Ac
16	5.714483	192.168.43.75	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
17	5.796112	192.168.43.75	142.250.201.131	TLSv1.3	118	Change Cipher Spec, Applicat
18	5.796626	192.168.43.75	142.250.201.131	TLSv1.3	146	Application Data
19	5.797013	192.168.43.75	142.250.201.131	TLSv1.3	376	Application Data
20	5.878881	142.250.201.131	192.168.43.75	TLSv1.3	85	[TCP Previous segment not ca
21	5.878934	192.168.43.75	142.250.201.131	TCP	66	[TCP Dup ACK 15#1] 63724 → 4
22	5.879687	142.250.201.131	192.168.43.75	TCP	54	443 → 63724 [ACK] Seq=4684 A
23	5.879687	142.250.201.131	192.168.43.75	TCP	662	[TCP Out-Of-Order] 443 → 637
24	5.879746	192.168.43.75	142.250.201.131	TCP	54	63724 → 443 [ACK] Seq=996 Ac

## ۲. پروتکل‌های استفاده شده در لایه‌های مختلف یک بسته

شکل زیر اطلاعات مربوط به بسته انتخاب شده را نشان می‌دهد.

37	6.246658	192.168.43.1	192.168.43.75	DNS	118 Standard query response 0xbee7 A mail.google.com CNAME googlemail
36	6.232287	192.168.43.75	192.168.43.1	DNS	79 Standard query 0x3ace A accounts.google.com
35	6.229797	192.168.43.75	192.168.43.1	DNS	78 Standard query 0xbdf8 A www.googleapis.com
34	6.216761	192.168.43.75	192.168.43.1	DNS	75 Standard query 0xbee7 A mail.google.com

>	Frame 35: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{57753F9E-3774-4574-8665-51271CAA7952}, id 0
>	Ethernet II, Src: IntelCor_d1:bb:2c (00:1e:64:d1:bb:2c), Dst: SamsungE_3b:f1:ea (80:ce:b9:3b:f1:ea)
>	Internet Protocol Version 4, Src: 192.168.43.75, Dst: 192.168.43.1
>	User Datagram Protocol, Src Port: 60320, Dst Port: 53
>	Domain Name System (query)

با توجه به این اطلاعات پروتکل‌های استفاده شده در لایه‌های مختلف مانند زیر است:

۱. Physical Layer: Wi-Fi – wire interface و Ethernet II: Link Layer

۲. Network Layer: IPv4

۳. Transport Layer: UDP

۴. Application Layer: DNS

همچنین با توجه به داده‌هایی که در پایین صفحه نشان داده می‌شود می‌توان گفت بیت‌های داده به ترتیب لایه‌ها قرار گرفته‌اند. برای مثال در بسته فوق قرارگیری بیت‌های مربوط به هر لایه به ترتیب زیر می‌باشد (از لایه اول تا چهارم)

0000	80 ce b9 3b f1 ea 00 1e 64 d1 bb 2c 08 00 45 00	...;... d...E
0010	00 40 db 55 00 00 80 11 87 ba c0 a8 2b 4b c0 a8	@U...+K
0020	2b 01 eb a0 00 35 00 2c 07 b8 bd f8 01 00 00 01	+...5, .....
0030	00 00 00 00 00 00 03 77 77 77 0a 67 6f 6f 67 6c	...w ww-googl
0040	65 61 70 69 73 03 63 6f 6d 00 00 01 00 01	eapis.co m....

0000	80 ce b9 3b f1 ea 00 1e 64 d1 bb 2c 08 00 45 00	...;... d...E
0010	00 40 db 55 00 00 80 11 87 ba c0 a8 2b 4b c0 a8	@U...+K
0020	2b 01 eb a0 00 35 00 2c 07 b8 bd f8 01 00 00 01	+...5, .....
0030	00 00 00 00 00 00 03 77 77 77 0a 67 6f 6f 67 6c	...w ww-googl
0040	65 61 70 69 73 03 63 6f 6d 00 00 01 00 01	eapis.co m....

0000	80 ce b9 3b f1 ea 00 1e 64 d1 bb 2c 08 00 45 00	...;... d...E
0010	00 40 db 55 00 00 80 11 87 ba c0 a8 2b 4b c0 a8	@U...+K
0020	2b 01 eb a0 00 35 00 2c 07 b8 bd f8 01 00 00 01	+...5, .....
0030	00 00 00 00 00 00 03 77 77 77 0a 67 6f 6f 67 6c	...w ww-googl
0040	65 61 70 69 73 03 63 6f 6d 00 00 01 00 01	eapis.co m....

0000	80 ce b9 3b f1 ea 00 1e 64 d1 bb 2c 08 00 45 00	...;... d...E
0010	00 40 db 55 00 00 80 11 87 ba c0 a8 2b 4b c0 a8	@U...+K
0020	2b 01 eb a0 00 35 00 2c 07 b8 bd f8 01 00 00 01	+...5, .....
0030	00 00 00 00 00 00 03 77 77 77 0a 67 6f 6f 67 6c	...w ww-googl
0040	65 61 70 69 73 03 63 6f 6d 00 00 01 00 01	eapis.co m....

با توجه به اطلاعات این قسمت طول فریم این بسته ۷۸ بایت است.

```
[Time since reference or first frame: 6.2297s]
Frame Number: 35
Frame Length: 78 bytes (624 bits)
Capture Length: 78 bytes (624 bits)
[Frame is marked: False]
```

به علاوه طول بسته لایه Transport طبق اطلاعات زیر ۴۴ بایت است.

```
Source Port: 60320
Destination Port: 53
Length: 44
Checksum: 0x07b8 [unverified]
[Checksum Status: Unverified]
```

### ۳. بسته‌هایی که بدون پروتکل‌های لایه‌های Network، Transport و Application هستند

بله، مطابق شکل زیر بسته‌هایی که از پروتکل ARP (که پروتکلی برای یافتن آدرس لایه پیوند و ارتباطش با آدرس لایه شبکه است) استفاده کرده‌اند چنین ویژگی‌ای دارند.

17793	192.168.43.75	SamsungE_3b:f1:ea	IntelCor_d1:bb:2c	ARP	42	192.168.43.75 is at 00:1e:64:d1:bb:2c
17092	140.743215	IntelCor_d1:bb:2c	SamsungE_3b:f1:ea	ARP	42	192.168.43.75 is at 00:1e:64:d1:bb:2c
17091	140.743157	SamsungE_3b:f1:ea	IntelCor_d1:bb:2c	ARP	42	Who has 192.168.43.75? Tell 192.168.43.75
15456	119.537771	IntelCor_d1:bb:2c	SamsungE_3b:f1:ea	ARP	42	192.168.43.75 is at 00:1e:64:d1:bb:2c

> Frame 17092: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF\_{57753F9E-3774-4000-8000-000000000000} [Ethernet II]  
> Ethernet II, Src: IntelCor\_d1:bb:2c (00:1e:64:d1:bb:2c), Dst: SamsungE\_3b:f1:ea (80:ce:b9:3b:f1:ea)  
> Address Resolution Protocol (reply)

#### ۴. مقدار Checksum در پروتکل IP

با توجه به این قسمت مقدار Checksum برابر است با  $(87BA)_{16} = 34746$ .

```
Time to Live: 128
Protocol: UDP (17)
Header Checksum: 0x87ba [validation disabled]
[Header checksum status: Unverified]
Source Address: 192.168.43.75
```

#### ۵. شماره پورت و مقدار Checksum در پروتکل های TCP و UDP

با توجه به این قسمت شماره پورت مبدا ۶۰۳۲۰ و شماره پورت مقصد ۵۳ می باشد. به طور کلی شماره پورت راهی برای مشخص کردن پردازه ای است که زمانی که بسته به سرور می رسد باید به آن تحویل داده شود. شماره IP سیستم انتهایی را مشخص کرده و شماره پورت مشخص می کند بسته مربوط به کدام پردازه (یا application) روی این سیستم است.

همچنین با توجه به شکل زیر مقدار Checksum برابر است با  $(07B8)_{16} = 1976$ .

```
▼ User Datagram Protocol, Src Port: 60320, Dst Port: 53
  Source Port: 60320
  Destination Port: 53
  Length: 44
  Checksum: 0x07b8 [unverified]
  [Checksum Status: Unverified]
```

## ۶. پروتکل لایه Transport، آدرس IP مقصد و اطلاعات مربوط سرآیند لایه دوم

با توجه به شکل زیر می توان گفت که این بسته برای انتقال در لایه Transport از پروتکل UDP استفاده می کند. همچنین آدرس IP مقصد آن برابر است با 192.168.43.1.

No.	Time	Source	Destination	Protocol
1	0.000000	192.168.43.75	192.168.43.1	DNS
2	0.005572	192.168.43.1	192.168.43.75	DNS
3	4.724645	192.168.43.75	192.168.43.1	DNS
4	4.730907	192.168.43.1	192.168.43.75	DNS
5	4.733454	192.168.43.75	192.168.43.1	DNS
6	4.738639	192.168.43.1	192.168.43.75	DNS

```
> Frame 5: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on  
> Ethernet II, Src: IntelCor_d1:bb:2c (00:1e:64:d1:bb:2c), Dst: HuaweiT  
> Internet Protocol Version 4, Src: 192.168.43.75, Dst: 192.168.43.1  
> User Datagram Protocol, Src Port: 52575, Dst Port: 53  
> Domain Name System (query)
```

همچنین با توجه به اطلاعات لایه دوم آدرس مبدا آن برابر با 192.168.43.75 و آدرس مقصد آن برابر با 192.168.43.1 می باشد.

```
Protocol: UDP (17)  
Header Checksum: 0x5554 [validation disabled]  
[Header checksum status: Unverified]  
Source Address: 192.168.43.75  
Destination Address: 192.168.43.1  
> User Datagram Protocol, Src Port: 52575, Dst Port: 53
```

## ۷. آدرس مشترک میان قسمت قبل و خروجی دستور ipconfig

با توجه به خروجی دستور ipconfig آدرس IP مربوط به سیستمی که از آن استفاده می کنیم برابر است با 192.168.43.75 که همان آدرس مبدا در قسمت قبل آزمایش است. بنابراین می توان گفت بسته فوق از طرف سیستم ما ارسال شده است.

```
Link-local IPv6 Address . . . . . : fe80::b995:4fe5:91e4:efda%10(Preferred)  
IPv4 Address. . . . . : 192.168.43.75(Preferred)  
Subnet Mask . . . . . : 255.255.255.0
```

## ۸. Type استفاده شده در پروتکل DNS در اجرای دستور Ping

با توجه به اطلاعات زیر در این بخش از Type A استفاده شده است.

```
▼ Queries
  > google.com: type A, class IN
    [Response In: 2]
```

پاسخ این query یک hostname و آدرس IP مربوط به آن را ذخیره می کند. بنابراین می توان گفت از این درخواست برای یافتن IP سرور گوگل با توجه به hostname آن استفاده شده است.

## ۹. Type استفاده شده در پروتکل DNS در اجرای دستور nslookup

با توجه به اطلاعات زیر در این بخش از Type PTR استفاده شده است.

```
▼ Queries
  > 1.1.1.1.in-addr.arpa: type PTR, class IN
  > Answers
```

پاسخ این query برعکس کاری است که DNS در حالت عادی انجام می دهد. بنابراین می توان گفت از این درخواست برای یافتن hostname مربوط به IP ای با شماره 1.1.1.1 استفاده شده است.

## ۱۰. دیگر Type های استفاده شده در پروتکل DNS

از دیگر type هایی که در پروتکل DNS استفاده می شوند می تواند به موارد زیر اشاره کرد:

۱. NS: مشخص می کند کدام سرور حاوی رکوردهای اصلی DNS مربوط به یک دامنه است و برای یافتن

IP دامنه مورد نظر باید به کجا مراجعه کنیم.

۲. CNAME: مشخص می کند نام اصلی یک سرور چیست (گاهی برای سهولت در استفاده کاربران نام

دیگری به جای نام اصلی سرور به طور متداول استفاده می شود).

۳. MX: سرور ایمیل SMTP مربوط به دامنه را مشخص می کند.

## ۱۱. استفاده از Display Filter

با توجه به فیلتر توصیف شده در دستورکار خروجی برنامه مطابق شکل زیر می‌باشد.

ip.addr == 5.144.130.115						
No.	Time	Source	Destination	Protocol	Length	Info
344	21.889787	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=21/5376, ttl=1 (no response found!)
345	21.891310	192.168.1.1	192.168.1.34	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
346	21.893801	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=22/5632, ttl=1 (no response found!)
347	21.897147	192.168.1.1	192.168.1.34	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
348	21.900078	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=23/5888, ttl=1 (no response found!)
349	21.901326	192.168.1.1	192.168.1.34	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
411	31.048703	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=24/6144, ttl=2 (no response found!)
412	31.080386	172.31.0.100	192.168.1.34	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
413	31.084446	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=25/6400, ttl=2 (no response found!)
414	31.116440	172.31.0.100	192.168.1.34	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
415	31.120223	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=26/6656, ttl=2 (no response found!)
417	31.157591	172.31.0.100	192.168.1.34	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
456	46.900281	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=27/6912, ttl=3 (no response found!)
457	46.935634	81.91.128.9	192.168.1.34	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
458	46.939188	192.168.1.34	5.144.130.115	ICMP	106	Echo (ping) request id=0x0001, seq=28/7168, ttl=3 (no response found!)
459	46.970357	81.91.128.9	192.168.1.34	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)

همانطور که مشاهده می‌شود این فیلتر تمام بسته‌هایی را نشان می‌دهد که آدرس IP مبدا و یا مقصد آن برابر با IP وبسایت p30download باشد. همچنین تمام پروتکل‌های استفاده شده ICMP می‌باشند زیرا دستوراتی مانند ping و traceroute برای دریافت اطلاعات گره‌ها از پیغام‌های ICMP (Internet Control Message Protocol) استفاده می‌کنند.

## ۱۲. بررسی مقادیر IP Layer TTL و ICMP Type

با توجه به اطلاعات زیر در این بخش از Type 8 که مربوط به یک درخواست ping می‌باشد استفاده شده‌است.

```
Internet Control Message Protocol
Type: 8 (Echo (ping) request)
Code: 0
```

همچنین مقدار TTL در پروتکل IP برابر است با ۱.

```
Fragment Offset: 0
> Time to Live: 1
Protocol: ICMP (1)
```

### ۱۳. هدف از تغییر TTL در بسته‌های ارسال شده توسط دستور `tracert`

در صورتی که بسته‌ها را به ترتیب بررسی کنیم متوجه می‌شویم که مقدار TTL در هر چند بسته متوالی یکی افزایش می‌یابد. هدف دستور `tracert` بررسی و شناسایی هر یک از گره‌های موجود در مسیر تا گره مقصد است. همچنین می‌دانیم TTL طول عمر بسته را مشخص می‌کند. بنابراین با تغییر طول TTL می‌توانیم آخرین گره‌ای که بسته به آن می‌رسد را مشخص کرده و اطلاعات آن گره را استخراج کنیم و از این طریق تمام گره‌های موجود در طول مسیر به ازای TTL های مختلف را بررسی کنیم.

### ۱۴. استفاده از فیلتر `ip.proto`

طبق جدول موجود در لینک زیر شماره ۶ معادل با پروتکل TCP می‌باشد.

[https://en.wikipedia.org/wiki/List\\_of\\_IP\\_protocol\\_numbers](https://en.wikipedia.org/wiki/List_of_IP_protocol_numbers)

بنابراین این فیلتر بسته‌هایی که از پروتکل TCP استفاده کرده‌اند را نشان می‌دهد. البته این فیلتر با فیلتر TCP متفاوت است و تنها بسته‌هایی که پروتکل آن‌ها TCP است را نشان نمی‌دهد، بلکه بسته‌هایی با پروتکل‌های متفاوت (مانند ICMP و TLS) که از پروتکل TCP استفاده می‌کنند را هم نمایش می‌دهد.