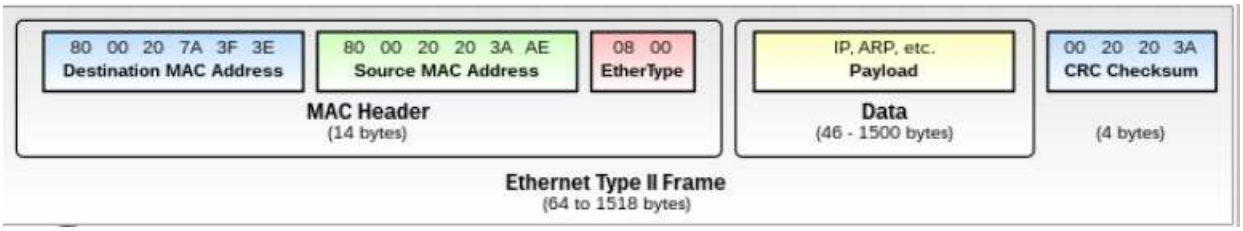
CN Wireshark homework – محمد اصولیان 99521073

Q1)

پاسخ گزینه B است.

سطر A در واقع اطلاعاتی است که wireshark در مورد فریم ارسال شده جمع آوری کرده و در واقعیت چنین سطری در frame وجود ندارد.

سطر B همان هدر فریم است. شکل زیر اطلاعات موجود ر هدر فریم را نشان میدهد:



در سطر B هم اطلاعات هدر فریم مانند source & destination mac address را مشاهده میکنیم پس این سطر همان هدر فریم است.

سطر c هدر پروتکل IP و در آن فیلد های هدر IP مثل ... ,ver, header length, total length, type of service را مشاهده میکنیم.

سطر D هم اطلاعات مربوط به پروتکل TCP است که مربوط به لایه 4 و بالاتر از IP است و به همین علت درون data بسته IP قرار میگیرد.

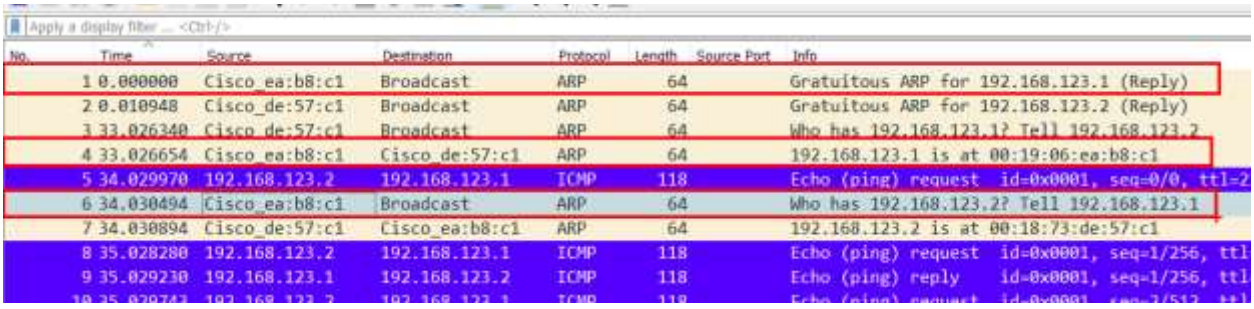
در نهایت سطر E داده های اصلی پکت و محتویات پروتکل HTTP است.

Q2)

a)

در صورت سوال گفته شده mac address مبدا و مقصد پکتی از cisco_ea:b8:c1 و پاسخ مربوط به آن را پیدا کنیم.

در لیست پکت ها سه پکت از طرف cisco_ea:b8:c1 وجود دارد:



پکت ردیف 1 یک بسته Gratuitous ARP است و وقتی انجام میشود که یک دستگاه جدید به شبکه متصل شود و برای این است که سایر دستگاه ها را از MAC خود با خبر کند. Gratuitous ARP ها reply ای ندارند پس این بسته، بسته مورد نظر سوال نیست.

پکت ردیف 4 خود یک ARP reply به درخواست موجود در ردیف 3 است پس این هم پکت مورد نظر صورت سوال نیست.

پکت ردیف 6 یک درخواست از طرف cisco_ea:b8:c1 است تا ببنید mac address مربوط به ip=192.168.123.2 چیست. پاسخ این درخواست هم در ردیف 7 آمده است. برای مشاهده mac address مبدا و مقصد این پکت ها، آن ها را در wireshark باز میکنیم.

پکت درخواست ARP در ردیف 6:

```
> Frame 6: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
v Ethernet II, Src: Cisco_ea:b8:c1 (00:19:06:ea:b8:c1), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
    > Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    > Source: Cisco_ea:b8:c1 (00:19:06:ea:b8:c1)
      Type: 802.1Q Virtual LAN (0x8100)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 123
v Address Resolution Protocol (request)
      Hardware type: Ethernet (1)
      Protocol type: IPv4 (0x0800)
      Hardware size: 6
      Protocol size: 4
      Opcode: request (1)
      Sender MAC address: Cisco_ea:b8:c1 (00:19:06:ea:b8:c1)
      Sender IP address: 192.168.123.1
      Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
      Target IP address: 192.168.123.2
```

میبینیم که source mac address = 00:19:06:ea:b8:c1 و destination mac address = ff:ff:ff:ff:ff:ff = broadcast است.

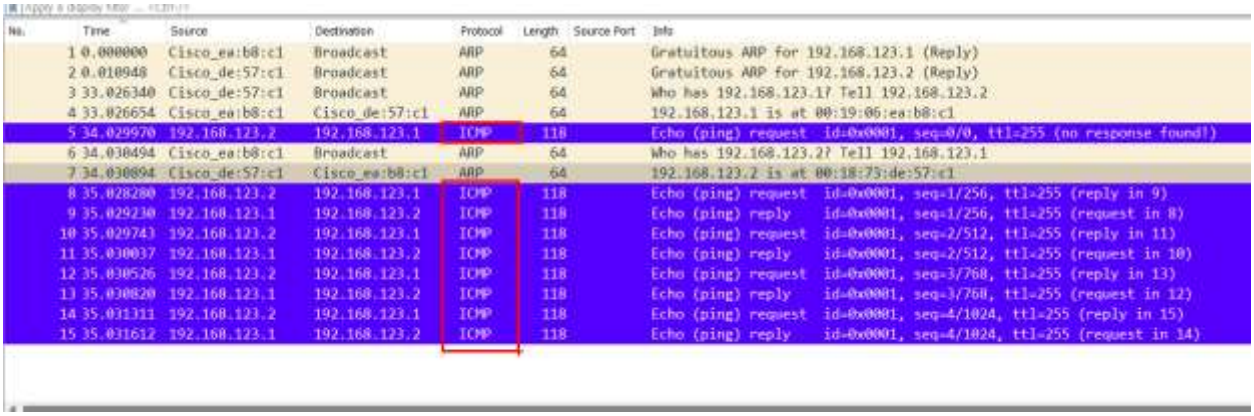
پکت ریپلای ARP در ردیف 7:

```
> Frame 7: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
v Ethernet II, Src: Cisco_de:57:c1 (00:18:73:de:57:c1), Dst: Cisco_ea:b8:c1 (00:19:06:ea:b8:c1)
    > Destination: Cisco_ea:b8:c1 (00:19:06:ea:b8:c1)
    > Source: Cisco_de:57:c1 (00:18:73:de:57:c1)
      Type: 802.1Q Virtual LAN (0x8100)
> 802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 123
v Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: Cisco_de:57:c1 (00:18:73:de:57:c1)
    Sender IP address: 192.168.123.2
    Target MAC address: Cisco_ea:b8:c1 (00:19:06:ea:b8:c1)
    Target IP address: 192.168.123.1
```

میبینیم که source mac address = 00:18:73:de:57:c1 و destination mac address = 00:19:06:ea:b8:c1 میباشد.

b)

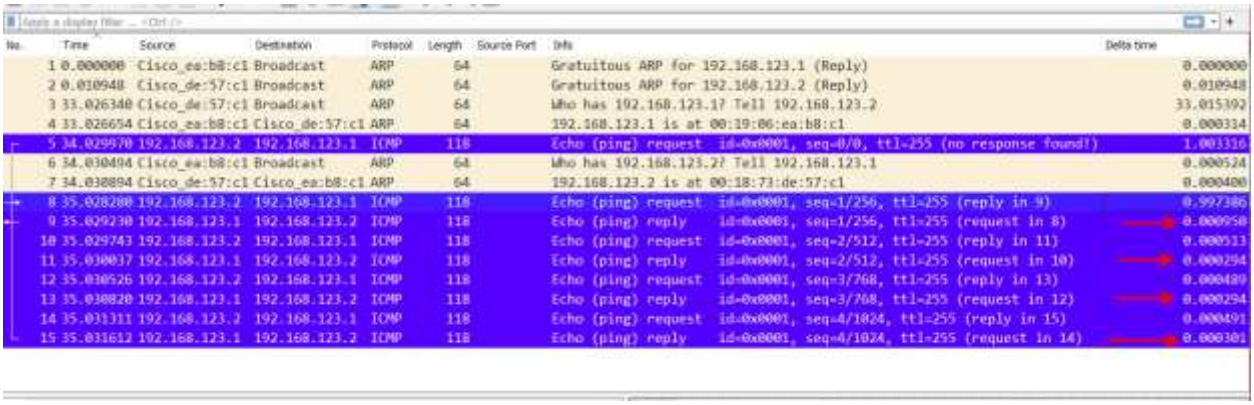
با توجه به نوع پروتکل های مشخص شده، 9 بسته icmp داریم:



No.	Time	Source	Destination	Protocol	Length	Source Port	Info
1	0.000000	Cisco_ea:b8:c1	Broadcast	ARP	64		Gratuitous ARP for 192.168.123.1 (Reply)
2	0.010948	Cisco_de:57:c1	Broadcast	ARP	64		Gratuitous ARP for 192.168.123.2 (Reply)
3	33.026340	Cisco_de:57:c1	Broadcast	ARP	64		Who has 192.168.123.1? Tell 192.168.123.2
4	33.026654	Cisco_ea:b8:c1	Cisco_de:57:c1	ARP	64		192.168.123.1 is at 00:19:06:ea:b8:c1
5	34.029970	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=0/0, ttl=255 (no response found!)
6	34.030494	Cisco_ea:b8:c1	Broadcast	ARP	64		Who has 192.168.123.2? Tell 192.168.123.1
7	34.030894	Cisco_de:57:c1	Cisco_ea:b8:c1	ARP	64		192.168.123.2 is at 00:18:73:de:57:c1
8	35.028280	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=1/256, ttl=255 (reply in 9)
9	35.029230	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=1/256, ttl=255 (request in 8)
10	35.029743	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=2/512, ttl=255 (reply in 11)
11	35.030037	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=2/512, ttl=255 (request in 10)
12	35.030526	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=3/768, ttl=255 (reply in 13)
13	35.030820	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=3/768, ttl=255 (request in 12)
14	35.031311	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=4/1024, ttl=255 (reply in 15)
15	35.031612	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=4/1024, ttl=255 (request in 14)

Q3)

طبق تعریف، round trip time زمانی است که طول میکشد تا پکت ICMP ارسال شود، به علاوه زمانی که طول میکشد تا پاسخ ICMP دریافت شود. برای محاسبه round trip time، کافیست تفاوت زمانی پکت های ICMP reply با پکت های ICMP request را پیدا کنیم. برای این کار هم در wireshark کافیست یک ستون جدید time delta اضافه کنیم که خود تفاوت زمانی را محاسبه میکند.



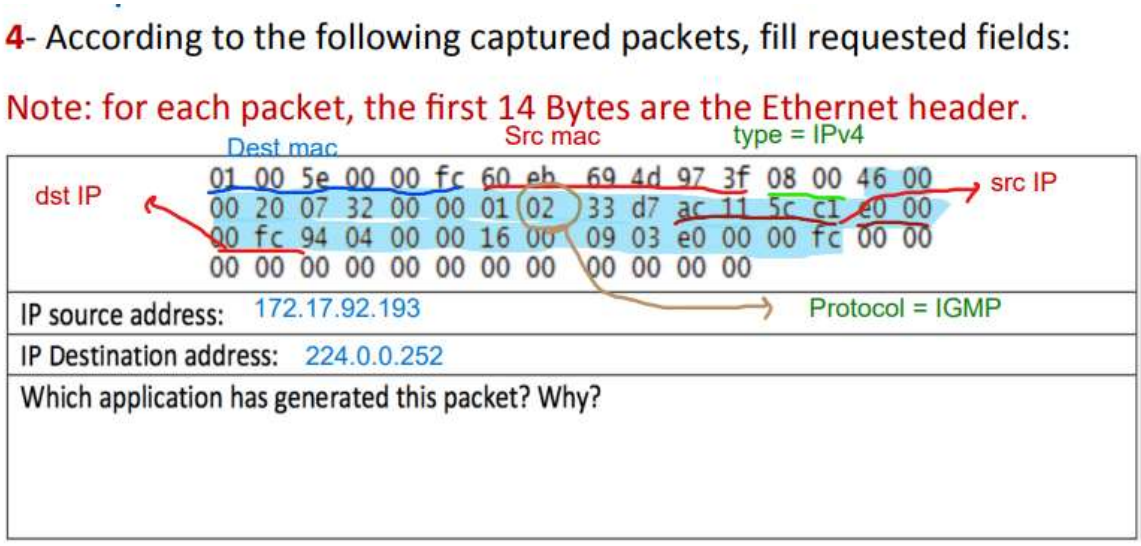
No.	Time	Source	Destination	Protocol	Length	Source Port	Info	Delta time
1	0.000000	Cisco_ea:b8:c1	Broadcast	ARP	64		Gratuitous ARP for 192.168.123.1 (Reply)	0.000000
2	0.010948	Cisco_de:57:c1	Broadcast	ARP	64		Gratuitous ARP for 192.168.123.2 (Reply)	0.010948
3	33.026348	Cisco_de:57:c1	Broadcast	ARP	64		Who has 192.168.123.1? Tell 192.168.123.2	33.015392
4	33.026654	Cisco_ea:b8:c1	Cisco_de:57:c1	ARP	64		192.168.123.1 is at 00:19:06:ea:b8:c1	0.000314
5	34.029970	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=0/0, ttl=255 (no response found!)	1.003316
6	34.030494	Cisco_ea:b8:c1	Broadcast	ARP	64		Who has 192.168.123.2? Tell 192.168.123.1	0.000524
7	34.030894	Cisco_de:57:c1	Cisco_ea:b8:c1	ARP	64		192.168.123.2 is at 00:18:73:de:57:c1	0.000400
8	35.028288	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=1/256, ttl=255 (reply in 9)	0.997386
9	35.029230	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=1/256, ttl=255 (request in 8)	0.000950
10	35.029743	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=2/512, ttl=255 (reply in 11)	0.000513
11	35.030037	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=2/512, ttl=255 (request in 10)	0.000294
12	35.030526	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=3/768, ttl=255 (reply in 13)	0.000489
13	35.030820	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=3/768, ttl=255 (request in 12)	0.000294
14	35.031311	192.168.123.2	192.168.123.1	ICMP	118		Echo (ping) request id=0x0001, seq=4/1024, ttl=255 (reply in 15)	0.000491
15	35.031612	192.168.123.1	192.168.123.2	ICMP	118		Echo (ping) reply id=0x0001, seq=4/1024, ttl=255 (request in 14)	0.000301

مشاهده میکنیم که min RTT = 0.294ms و max RTT = 0.950ms و avg RTT = 0.45975ms است.

Q4)

بسته نمایش داده شده یک Ethernet frame است. با کمک فرمت این بسته مشخص شد که محتوای این بسته، داده های یک پکت IP است. با کمک فرمت پکت IP اطلاعات خواسته شده را استخراج کردم. src ip و ds tip را در عکس مشخص کردم و سوال سوم را پایین تر توضیح دادم

a)

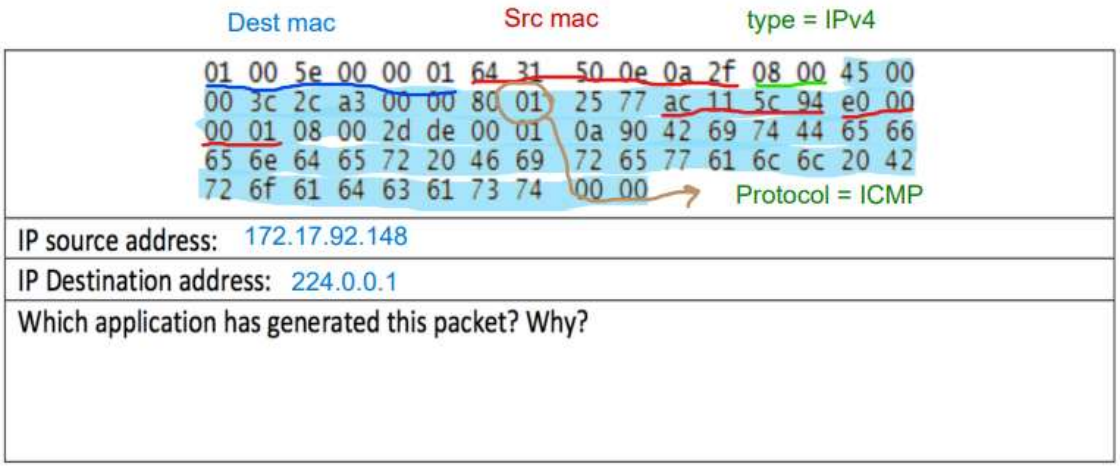


برای این که بدانیم این بسته توسط چه برنامه ای تولید شده است، باید متوجه data درون پکت IP مربوط به چه پروتکلی است. برای این کار فیلد protocol در هدر IP را چک میکنم که مقدار آن 02 است و این مقدار بنابر RFC مربوط به پروتکل IGMP است.

IGMP یا Internet Group Management Protocol یک پروتکل است که به چندین دستگاه اجازه میدهد که یک IP مشترک را داشته باشند تا بتوانند داده یکسانی دریافت کنند. این پروتکل در Multicasting کاربرد دارد و در مواردی مانند stream کردن ویدیو استفاده میشود.

(b

IP مبدا و مقصد در شکل مشخص شده. و سوال سوم پایین تر توضیح داده شده.



فیلد Protocol در بسته IP موجود در این شکل، مقدار 01 دارد که با توجه به RFC این عدد، عدد پروتکل ICMP است.

ICMP یا Internet Control Message Protocol یک پروتکل برای بررسی برقراری ارتباط بین دو نود در شبکه استفاده میشود. این پروتکل در دستور ping در ترمینال دستگاه‌ها استفاده می‌شود.

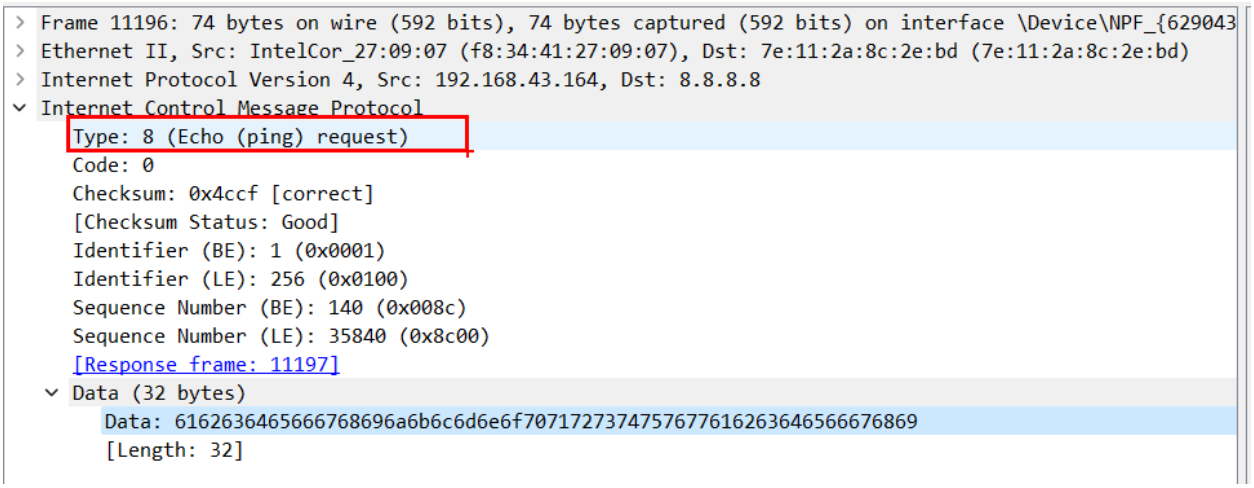
(Q5

A) چرا بسته ICMP سورس پورت ندارد؟

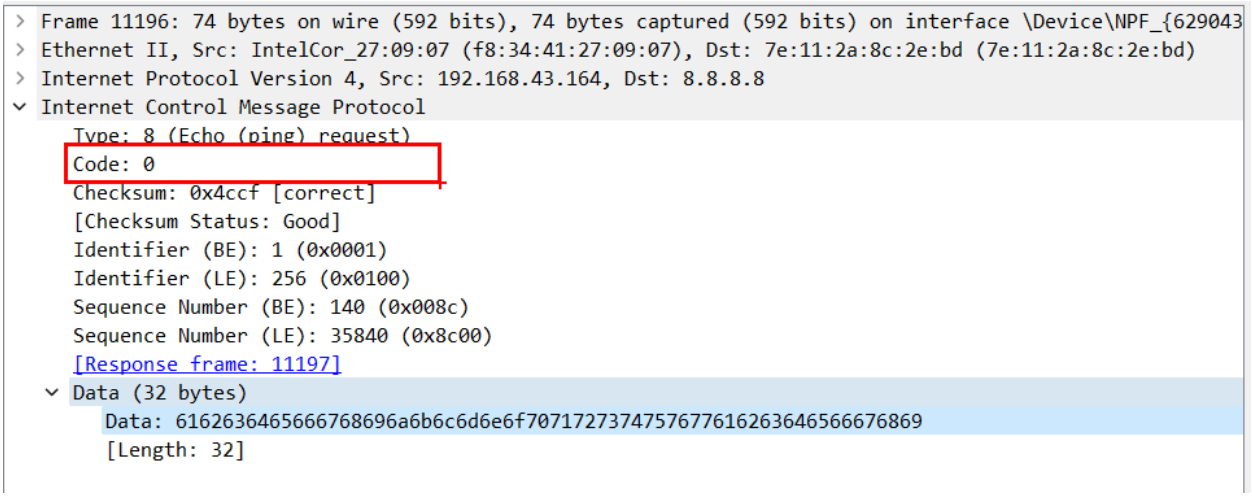
زیرا پروتکل ICMP در لایه 3 کار میکند. در واقع هدف بسته ICMP برقراری ارتباط بین دو نود تا لایه 3 است و وارد لایه 4 نمیشود که برای تحویل به یک application نیاز به پورتی داشته باشد.

(B

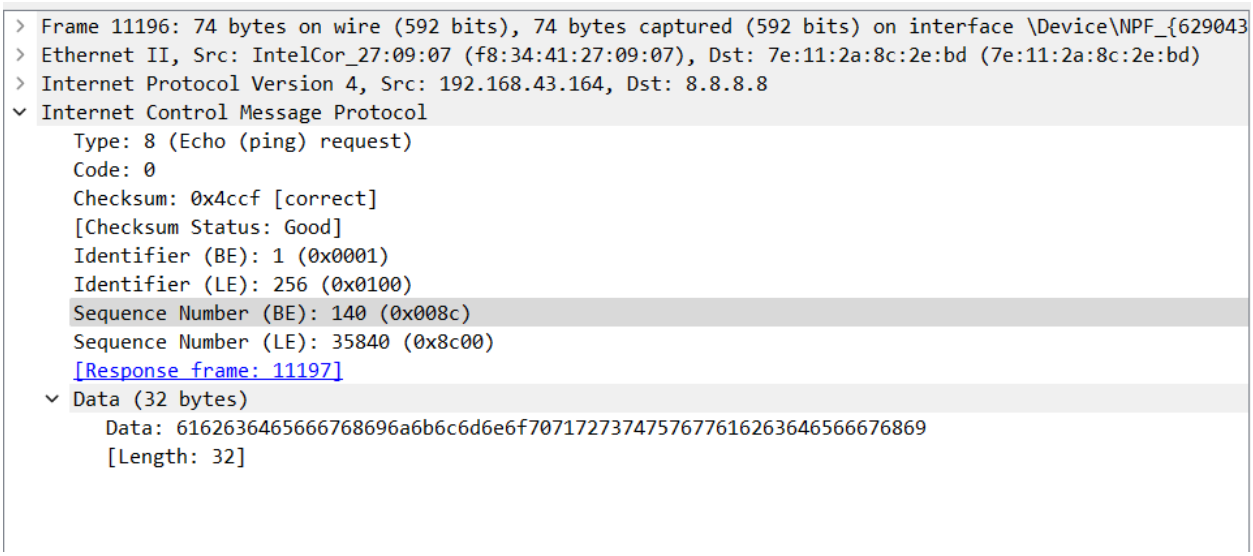
icmp type مشخص میکند که نوع پکت icmp ارسال شده چیست. مثلا میتواند مقادیر 0(Echo reply), 8(Echo request), 5(Redirect), 3(Destination Unreachable), 11(Time Exceeded) و ... را داشته باشد. در اینجا مقدار 8=Echo Request دارد:



icmp code جزعیات بیشتری را در مورد پکت ICMP مشخص میکند. مثلا در صورتی که icmp type برابر 8(Destination Unreachabl) باشد، مقادیر مختلف code مشخص میکند که مقصد به چه صورت غیر قابل دسترس بوده. اگر code = 0 یعنی شبکه مقصد غیر قابل دسترس بوده و اگر code = 1 یعنی هاست مقصد غیر قابل دسترس بوده. در اینجا code مقدار 0 را دارد.



فیلد های دیگر بسته icmp در این بسته:



Checksum: یک error correction code برای اطمینان از صحت داده های بسته icmp است.

Identifier: این فیلد برای تمایز بین پکت های icmp استفاده میشود. مثلا زمانی که چندین دستور ping همزمان در حال اجرا هستند.

sequence number: این فیلد هم برای تمایز بین بسته های imcp است. فرق این فیلد با قبلی این است که حتی اگر یک دستور ping در حال اجرا باشد، در هر بسته نسبت به بسته قبلی این مقدار تغییر میکند.

- در وایر شارک این دو فیلد را هم به صورت Big Endian هم به صورت Little Endian نشان داده تا خوانایی راحت تر باشد.

data: این فیلد هم همان داده های اصلی این پکت icmp هستند.

تعداد بایت های هر فیلد را میتوانیم در wireshark بشماریم:

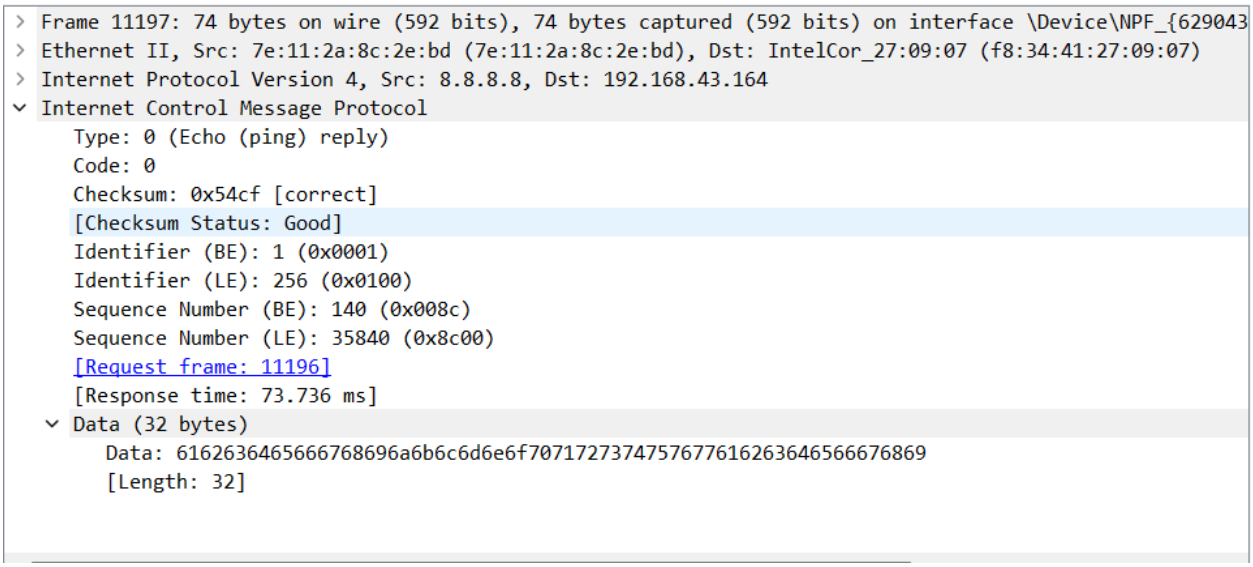
checksum: 2 بایت

Identifier: 2 بایت

Sequence Number: 2 بایت

C)

خود نرم افزار wireshark فریم پاسخ این بسته را هم در پایین بسته مشخص کرده با کلیک کردن روی آن به بسته پاسخ میرویم.



مقدار type، 0 به معنای echo reply است و مقدار code، 0 است.

سایر فیلد های این بسته:

checksum: که متفاوت از checksum بسته قبلی است.

Identifier: برابر این مقدار در بسته request است.

Sequence Number: برابر با این مقدار در بسته request است.

Data: برابر این مقدار در بسته Request است.

اندازه هر فیلد:

checksum: 2 بایت

Identifier: 2 بایت

Sequence Number: 2 بایت