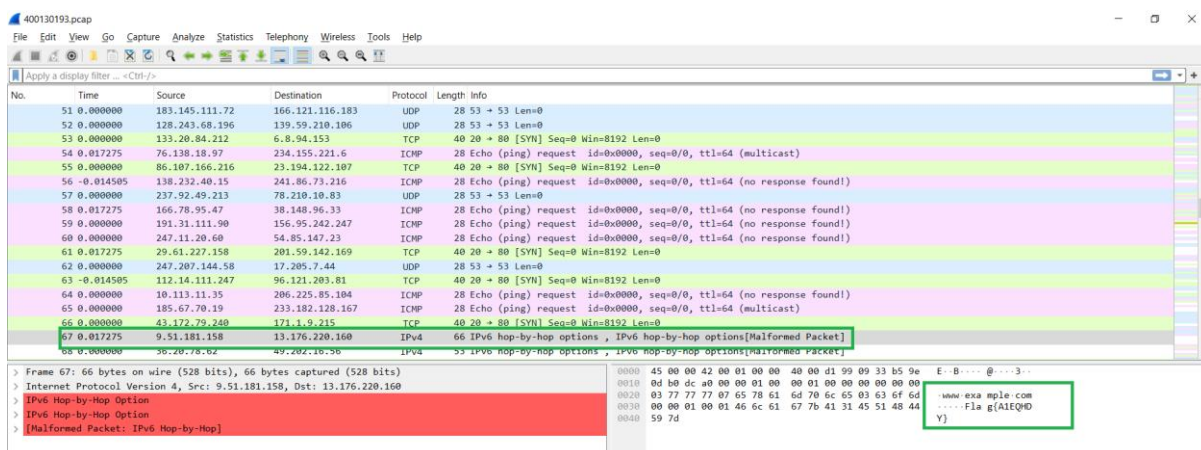


# به نام خدا

عنوان	پروژه چهارم شبکه های کامپیوتری
نام و نام خانوادگی	آرش دانش نژاد
شماره دانشجویی	۴۰۰۱۳۰۱۹۳

## 1. Flag را بیابید. در چه پکتی بود؟ پروسه یافتن آن را شرح دهید.

Flag را در پکت شماره 67 پیدا کردم که در پروتکل ipv4 بود. ابتدا با استفاده از فیلتر ها پروتکل های TCP را سرچ کردم چون شانس بیشتری برای وجود flag داشتند که در این پروتکل ها وجود نداشت و سپس به دنبال فلگ در UDP گشتم که در انجا هم فلگی نبود و با جستجو پروتکل های ipv4 فلگ را پیدا کردم.



## ۲. فایل ترافیک ارائه شده را با استفاده از فیلتر های مختلف تحلیل کنید و تا حد امکان، جزئیات مختلف در موردش را بیان نمایید.

## نمونه هایی از تحلیل پکت ها :

در اینجا پکت هایی که پورت ۵۳ را دارند فیلتر کردیم که source پورت و destination پورت این عدد هست.

UDP.port == 53

به عنوان مثال checksum بررسی میکنیم.

در زمینه تأیید داده ها و خطایابی، checksum مقداری است که از یک مجموعه داده محاسبه می شود تا بررسی شود آیا در طول ذخیره سازی یا انتقال داده ها خطا یا خرابی رخ داده است یا خیر. checksum یک نوع افزونگی است که به داده ها اضافه می شود؛ این امکان را می دهد تا بررسی کنید که داده ها تغییر نکرده یا خراب نشده اند.

checksum 0xBED9 یک مقدار هگزادسیمال (پایه ۱۶) است که نشان دهنده checksum یک مجموعه داده خاص است. با این حال، اینکه به عنوان "[تأیید نشده]" مشخص شده است به این معنی است که checksum تأیید یا تایید نشده است. این می تواند به دلایل مختلفی رخ دهد:

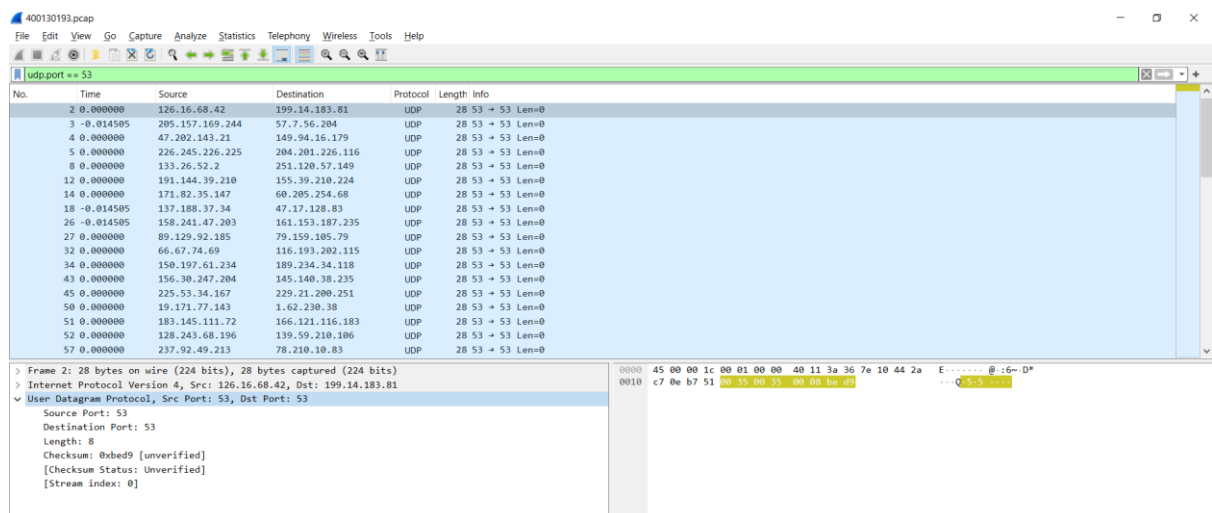
عدم محاسبه مجدد: ممکن است checksum دوباره محاسبه نشده باشد تا با مقدار اصلی مقایسه شود، بنابراین صحت آن ناشناخته است.

خرابی داده: ممکن است در طول انتقال یا ذخیره سازی داده ها خرابی رخ داده باشد که باعث تفاوت بین checksum و داده واقعی می شود.

الگوریتم یا پارامترهای نادرست: اگر checksum با پارامترهای متفاوت یا الگوریتم دیگری نسبت به آنچه در ابتدا استفاده شده است دوباره محاسبه شده باشد، ممکن است تأیید شکست بخورد.

خطای انسانی یا نرم افزاری: یک خطا در فرآیند محاسبه یا تأیید checksum می تواند منجر به عدم تأیید آن شود.

در هر صورت، توصیه می شود checksum را دوباره از مجموعه داده با استفاده از الگوریتم مناسب محاسبه کنید و با xBED9 مقایسه کنید تا صحت داده ها را تأیید کنید. اگر آنها مطابقت داشتند، احتمالاً داده ها سالم هستند؛ در غیر این صورت، ممکن است مشکلی در داده ها وجود داشته باشد.



No.	Time	Source	Destination	Protocol	Length	Info
2	0.000000	126.16.68.42	199.14.183.81	UDP	28	53 → 53 Len=0
3	0.014505	205.157.169.244	57.7.56.204	UDP	28	53 → 53 Len=0
4	0.000000	47.202.143.21	149.94.16.179	UDP	28	53 → 53 Len=0
5	0.000000	226.245.226.225	204.201.226.116	UDP	28	53 → 53 Len=0
8	0.000000	133.26.52.2	251.120.57.149	UDP	28	53 → 53 Len=0
12	0.000000	191.144.39.210	155.39.210.224	UDP	28	53 → 53 Len=0
14	0.000000	171.82.35.147	60.205.254.68	UDP	28	53 → 53 Len=0
18	0.014505	137.188.37.34	47.17.128.83	UDP	28	53 → 53 Len=0
26	0.014505	158.241.47.203	161.153.187.235	UDP	28	53 → 53 Len=0
27	0.000000	89.129.92.185	79.159.105.79	UDP	28	53 → 53 Len=0
32	0.000000	66.67.74.69	116.193.202.115	UDP	28	53 → 53 Len=0
34	0.000000	150.197.61.234	189.234.34.118	UDP	28	53 → 53 Len=0
43	0.000000	156.30.247.204	145.140.38.235	UDP	28	53 → 53 Len=0
45	0.000000	225.53.34.167	229.21.200.251	UDP	28	53 → 53 Len=0
50	0.000000	19.171.77.143	1.62.230.38	UDP	28	53 → 53 Len=0
51	0.000000	183.145.111.72	166.121.116.183	UDP	28	53 → 53 Len=0
52	0.000000	128.245.68.196	139.59.210.106	UDP	28	53 → 53 Len=0
57	0.000000	237.92.49.213	78.210.10.83	UDP	28	53 → 53 Len=0

Source Port: 53	Destination Port: 53	Length: 8	Checksum: 0xbed9 [unverified]	[Checksum Status: Unverified]	[Stream Index: 0]
-----------------	----------------------	-----------	-------------------------------	-------------------------------	-------------------

0000	45	00	00	1c	00	01	00	00	40	11	3a	35	7e	10	44	2a	E.....@:~0*
0010	c7	0e	b7	51	00	03	00	03	00	03	b4	02					...Q:5:5:5:5

## نمونه بعدی :

یکی از پورت های ipv4 را بررسی می کنیم به عنوان مثال به دنبال ادرس 133.252.222.45 میگردیم با این فیلتر ان را پیدا می کنیم.

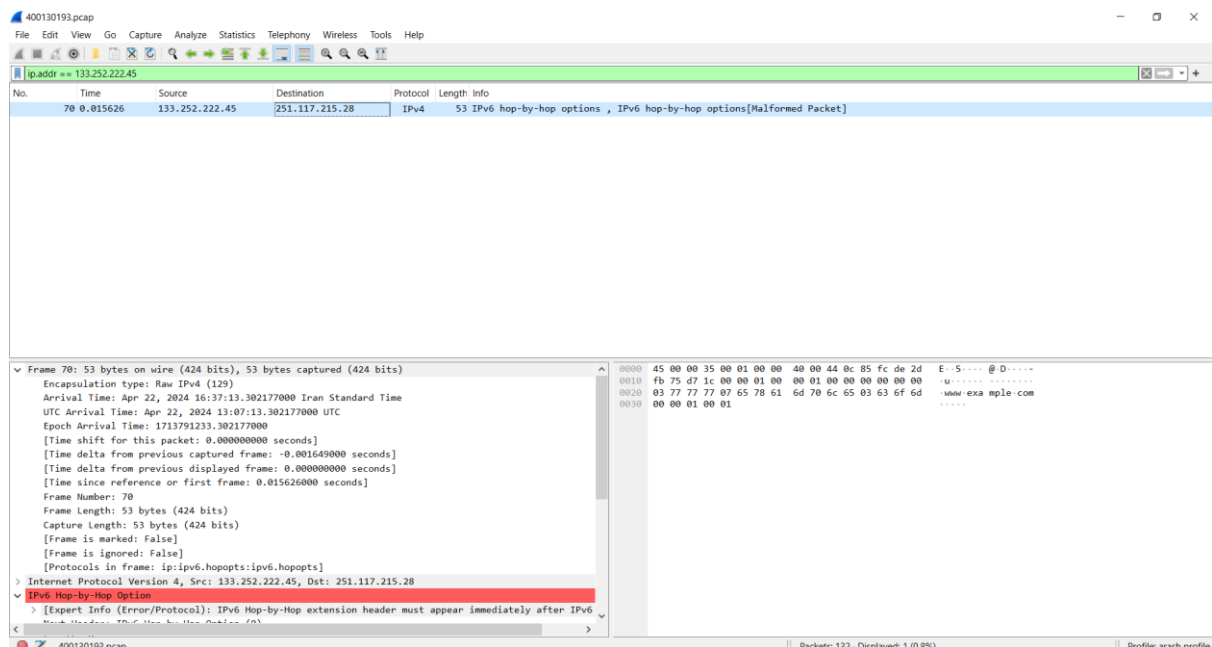
**ip.addr == 133.252.222.45**

آدرس های IP (IPv4 و IPv6): به ما این اجازه می دهد تا ترافیک شبکه را با توجه به آدرس های IP مورد بررسی قرار دهید. آدرس های IP نشان دهنده مبدأ و مقصد بسته ها در شبکه هستند.

در این پکت framenumber=70 هست، شماره فریم 70 به یکی از بسته های داده در ترافیک شبکه اشاره دارد. در Wireshark، هر بسته داده با یک شماره فریم مشخص می شود تا بتوانیم آن ها را به راحتی پیدا کنیم.

شماره فریم نشان دهنده ترتیب دریافت یا ارسال بسته ها در ترافیک شبکه است. به عبارت دیگر، اگر شماره فریم 70 باشد، این بسته 70امین بسته ای است که در ترافیک شبکه دریافت یا ارسال شده است.

همچنین در اینجا طول فریم ما 53 بایت است.



## نمونه بعدی :

### این پکت یک پکت با پروتکل TCP است.

TCP یا Transmission Control Protocol یکی از مهم‌ترین پروتکل‌ها یا استانداردها برای فعال‌سازی ارتباط بین دستگاه‌ها در یک شبکه خاص است. این پروتکل الگوریتم‌هایی دارد که خطاهای پیچیده‌ای که در ارتباط بسته‌ها به وجود می‌آید (مانند بسته‌های خراب، بسته‌های نامعتبر، تکراری و غیره) را حل می‌کند. از آنجا که با پروتکل اینترنت (IP) استفاده می‌شود، بسیاری اوقات به عنوان TCP/IP نیز اشاره می‌شود.

در این پکت ادرس مقصد و سورس به این صورت هست:

Source address : 53.78.110.139

Destination address : 166.234. 110 .139

در پکت طول فریم ۴۰ بایت است.

The image shows a Wireshark packet capture analysis. The top pane displays a list of captured packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The bottom pane shows the detailed view of the selected packet (No. 127), including its encapsulation type, arrival time, and the raw data bytes.

No.	Time	Source	Destination	Protocol	Length	Info
115	0.000000	53.166.42.27	50.288.173.153	UDP	28	53 → 53 Len=0
116	0.016733	142.159.209.109	3.180.65.242	IPv4	53	IPv6 hop-by-hop options , IPv6 hop-by-hop options[Malformed Packet]
117	0.000000	225.102.73.2	79.239.29.5	UDP	28	53 → 53 Len=0
118	0.000000	52.80.202.5	21.249.123.69	ICMP	28	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (no response found!)
119	0.000000	95.170.147.84	239.6.32.66	UDP	28	53 → 53 Len=0
120	0.017275	254.99.72.129	11.0.153.202	UDP	28	53 → 53 Len=0
121	0.000000	71.108.210.192	77.110.55.242	ICMP	28	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (no response found!)
122	0.000000	106.121.57.212	117.98.93.227	IPv4	53	IPv6 hop-by-hop options , IPv6 hop-by-hop options[Malformed Packet]
123	0.000000	18.76.234.78	177.79.150.169	IPv4	53	IPv6 hop-by-hop options , IPv6 hop-by-hop options[Malformed Packet]
124	0.000000	135.163.122.13	228.173.13.125	IPv4	53	IPv6 hop-by-hop options , IPv6 hop-by-hop options[Malformed Packet]
125	0.000000	125.250.165.48	58.254.113.219	IPv4	53	IPv6 hop-by-hop options , IPv6 hop-by-hop options[Malformed Packet]
126	0.000000	105.237.115.193	73.112.59.223	ICMP	28	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (no response found!)
127	0.000000	53.78.110.139	166.234.139.244	TCP	40	20 → 80 [SYN] Seq=0 Win=8192 Len=0
128	0.000000	211.229.77.197	74.107.141.226	ICMP	28	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (no response found!)
129	0.000000	202.203.195.118	13.50.21.150	IPv4	53	IPv6 hop-by-hop options , IPv6 hop-by-hop options[Malformed Packet]
130	0.000000	137.242.40.183	84.205.42.203	ICMP	28	Echo (ping) request id=0x0000, seq=0/0, ttl=64 (no response found!)
131	-0.014505	104.30.116.138	202.185.135.226	TCP	40	20 → 80 [SYN] Seq=0 Win=8192 Len=0
132	0.000000	139.57.0.0	197.222.191.55	UDP	28	53 → 53 Len=0

Frame 127: 40 bytes on wire (320 bits), 40 bytes captured (320 bits) on interface 0  
Encapsulation type: Raw IPv4 (129)  
Arrival Time: Apr 22, 2024 16:37:13.286551000 Iran Standard Time  
UTC Arrival Time: Apr 22, 2024 13:07:13.286551000 UTC  
Epoch Arrival Time: 1713791233.286551000  
[Time shift for this packet: 0.000000000 seconds]  
[Time delta from previous captured frame: 0.000000000 seconds]  
[Time delta from previous displayed frame: 0.000000000 seconds]  
[Time since reference or first frame: 0.000000000 seconds]  
Frame Number: 127  
Frame Length: 40 bytes (320 bits)  
Capture Length: 40 bytes (320 bits)  
[Frame is marked: False]  
[Frame is ignored: False]  
[Protocols in frame: ip:tcp]  
[Coloring Rule Name: HTTP]  
[Coloring Rule String: http || tcp.port == 80 || http2]  
> Internet Protocol Version 4, Src: 53.78.110.139, Dst: 166.234.139.244  
> Transmission Control Protocol, Src Port: 20, Dst Port: 80, Seq: 0, Len: 0