WIRESHARK

Arş. Gör. Melek ŞENTÜRK

HOW DOES DATA TRANSMİSSİON WORK İN TCP?

TCP (Transmission Control Protocol) is the most common data transmission protocol on the Internet.

Its main feature is that it provides communication without data loss and in the correct order.

After the data is sent, an ACK (acknowledgement) message is waited from the receiver. This checks whether the data has been successfully delivered.

After the data is sent, an ACK (acknowledgement) message is waited from the receiver. This checks whether the data has been successfully delivered.

Retransmission: If the package does not arrive, it will be sent again.

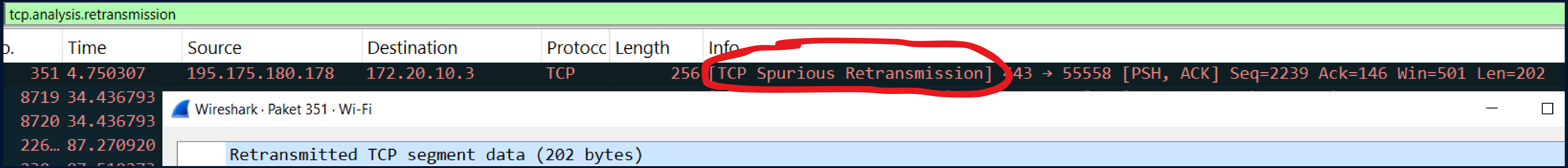
Fast Retransmission: If the same ACK message is received 3 times, TCP quickly resends.

Spurious Retransmission: In fact, the packet arrived but TCP did not notice it and sent it again.

TCP ON WEAK CONNECTİONS

Purpose: To see packet loss and resend events.

- The network connection needs to be weakened by using a mobile internet connection (Hotspot) or a low-speed VPN connection.
- The slow network provides an environment for observing TCP behaviors such as packet loss and retransmission in Wireshark.



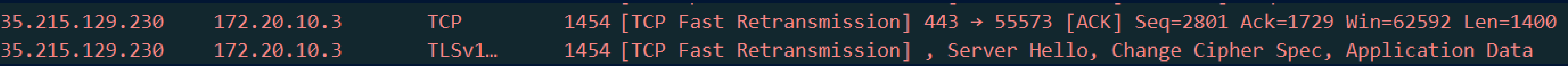
```
tcp.analysis.retransmission

o.    Time          Source            Destination    Protocc Length  Info
  351 4.750307     195.175.180.178   172.20.10.3    TCP         256 [TCP Spurious Retransmission] 443 → 55558 [PSH, ACK] Seq=2239 Ack=146 Win=501 Len=202
 8719 34.436793
 8720 34.436793
 226… 87.270920
```

```
Retransmitted TCP segment data (202 bytes)
```

- Spurious Retransmission is when the sender incorrectly senses the packet and resends it, even though the original packet has already reached the receiver. This error is usually caused by network latency, out of sequence, or a delay/loss of the ACK packet.

TCP ON WEAK CONNECTİONS



```
35.215.129.230    172.20.10.3    TCP       1454 [TCP Fast Retransmission] 443 → 55573 [ACK] Seq=2801 Ack=1729 Win=62592 Len=1400
35.215.129.230    172.20.10.3    TLSv1…    1454 [TCP Fast Retransmission] , Server Hello, Change Cipher Spec, Application Data
```

- TCP detects that a packet is lost when it receives three repeat ACKs (dupe ACKs) without waiting for a timeout and quickly resends it.

- In this line, a 1400 byte data packet is being sent again, probably because the previous transmission did not receive an ACK from the receiver or received 3 duplicate ACKs.

- The server (35.215.129.230) quickly resends the 1400 byte data it previously sent to the client (172.20.10.3) because the ACK response did not arrive in time. This data includes important TLS messages such as “Server Hello”, which are necessary to establish a secure connection.

  - TLS ensures that communication over the internet is secure. (Encryption, authentication, etc.)
  - TCP is a transport protocol that ensures that data arrives in order and without loss.
  - TLS data is transported over TCP. In other words:

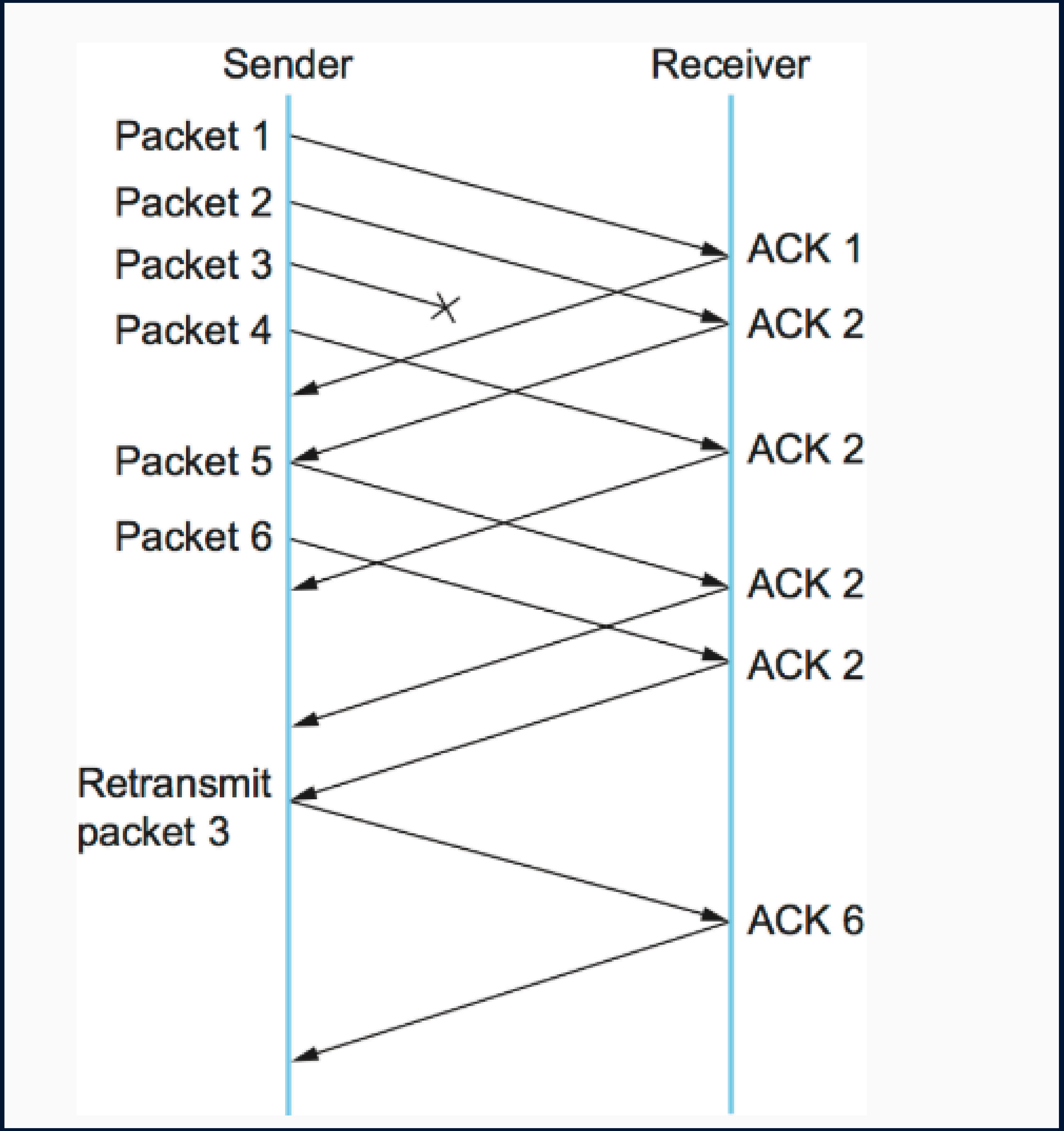
TLS determines what to send (such as encrypting and authenticating the user) → TCP delivers it to the recipient properly.

3 DUPLİCATE ACK

- The TCP protocol uses ACK (acknowledgement) messages to detect data loss in transmission.

- If a receiver does not receive the packet it was expecting, it sends the same ACK number to each subsequent packet. This means "I'm still waiting for that packet."

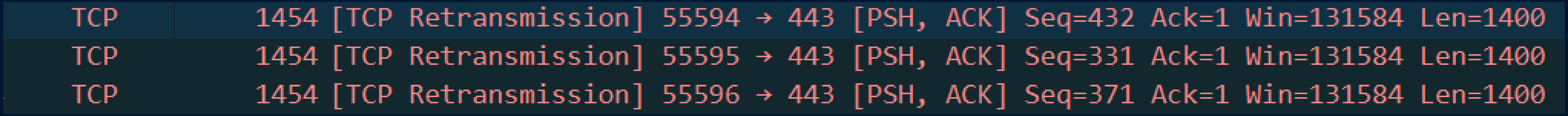
- This is called a "Duplicate ACK".



- 1–2 duplicate ACK → Thinks there might be a small delay in the network, waits.
- 3 duplicate ACK → This might be a data loss, says:
- Activates the "Fast Retransmit" mechanism.

Immediately resends the packet it thinks is lost.

TCP ON WEAK CONNECTİONS



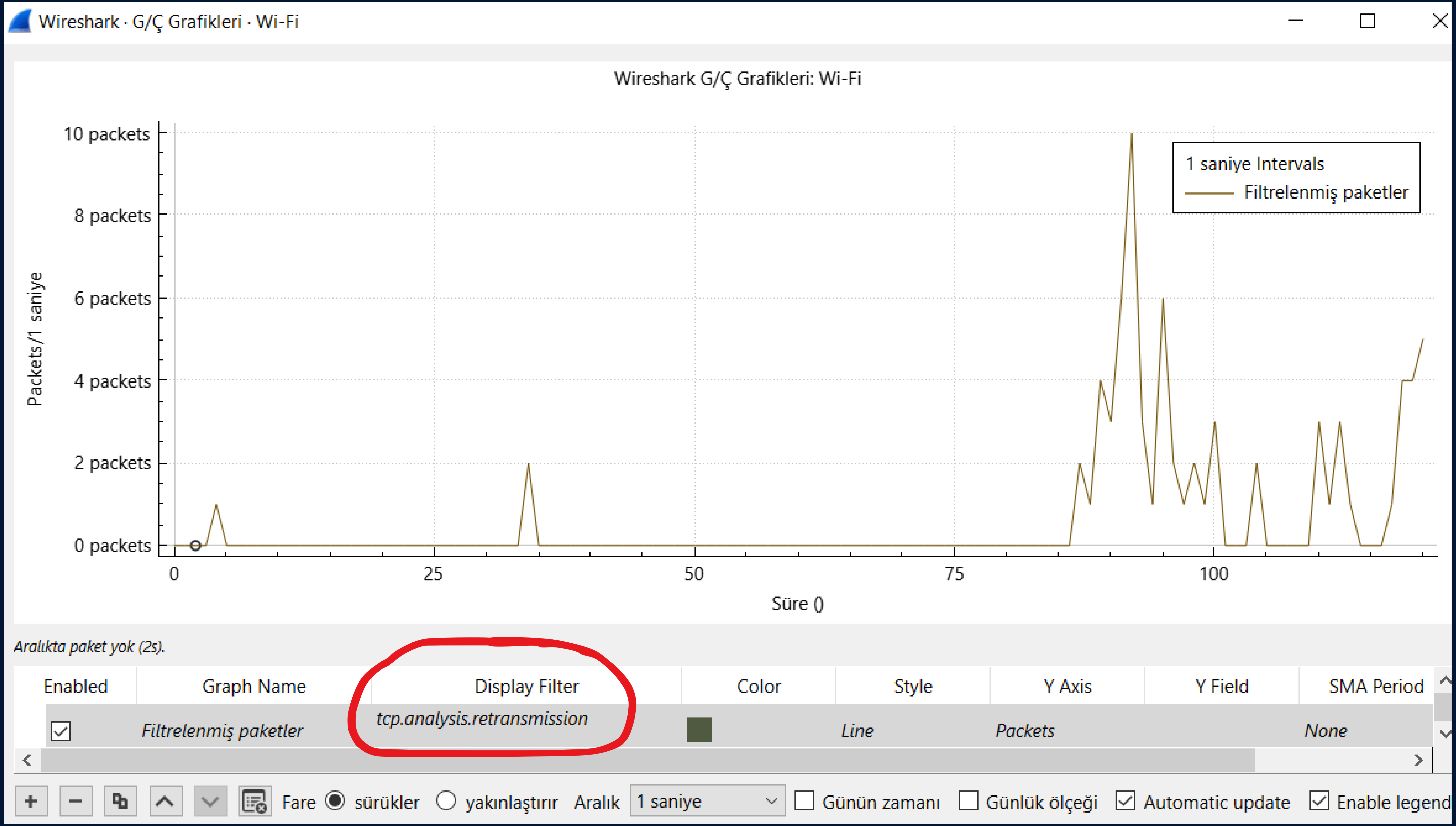
```
TCP      1454 [TCP Retransmission] 55594 → 443 [PSH, ACK] Seq=432 Ack=1 Win=131584 Len=1400
TCP      1454 [TCP Retransmission] 55595 → 443 [PSH, ACK] Seq=331 Ack=1 Win=131584 Len=1400
TCP      1454 [TCP Retransmission] 55596 → 443 [PSH, ACK] Seq=371 Ack=1 Win=131584 Len=1400
```

- While connected to the network, sometimes data can be lost, delayed, or arrive late.
- In such cases, TCP says "I think the data was lost" and sends the same data again.
- This is normal behavior to avoid connection failure.

- Sequence Number: Indicates the byte from which the data in this packet starts in the stream.

- Window Size: Determines the maximum amount of data that the other party can send at the same time.
- Here it says: "You can send 131,584 bytes of data right now, don't send more than that, or my buffer will overflow."

- By filtering resent packets in I/O graphs, it is possible to determine the time intervals in which resends are intense.



Aralıkta paket yok (2s).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	*Filtrelenmiş paketler*	*tcp.analysis.retransmission*	■	*Line*	*Packets*		*None*

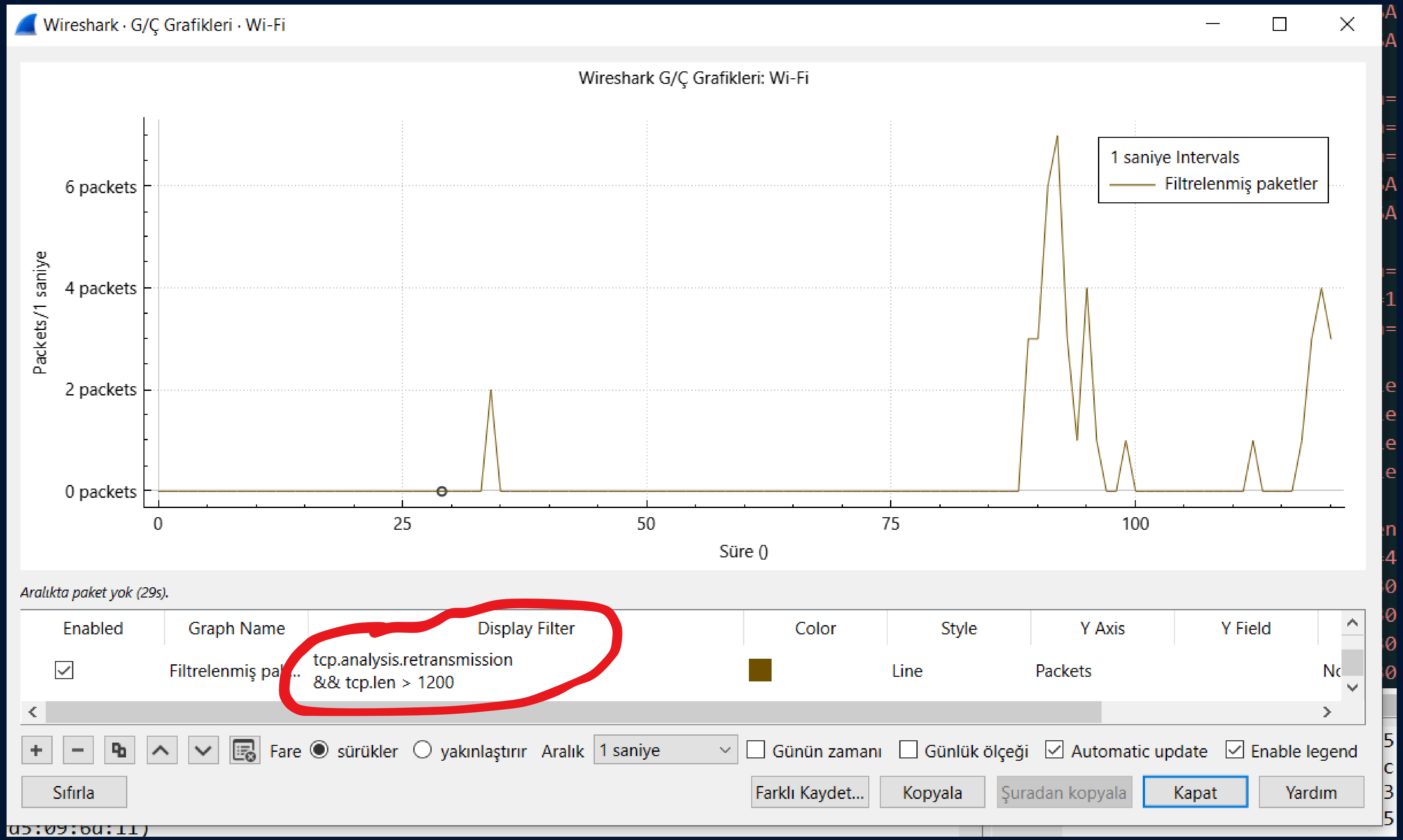
Fare ● sürükler ○ yakınlaştırır Aralık 1 saniye ☐ Günün zamanı ☐ Günlük ölçeği ☑ Automatic update ☑ Enable legend

- By filtering both large and resent packets in I/O graphs, resending analysis of large packets can be performed.

- Generally, the risk of retransmission is higher in larger packages. But this is not always the case, but in certain cases.

- Larger data → greater chance of error (e.g. in wireless network).

- If the network is of good quality and empty, large packets can pass through easily.



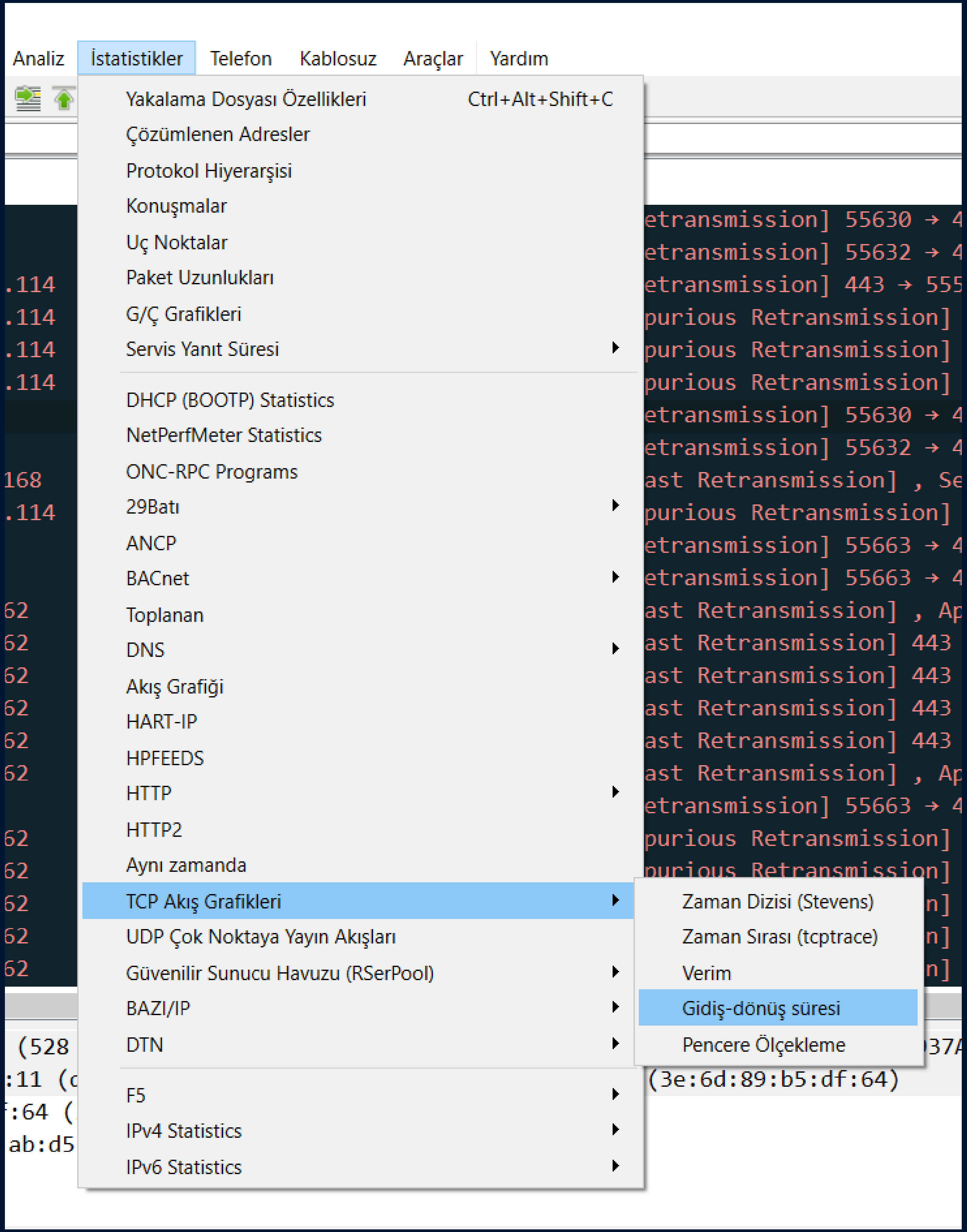
Aralıkta paket yok (29s).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field
☑	Filtrelenmiş pa...	tcp.analysis.retransmission && tcp.len > 1200	🟫	Line	Packets	Nc

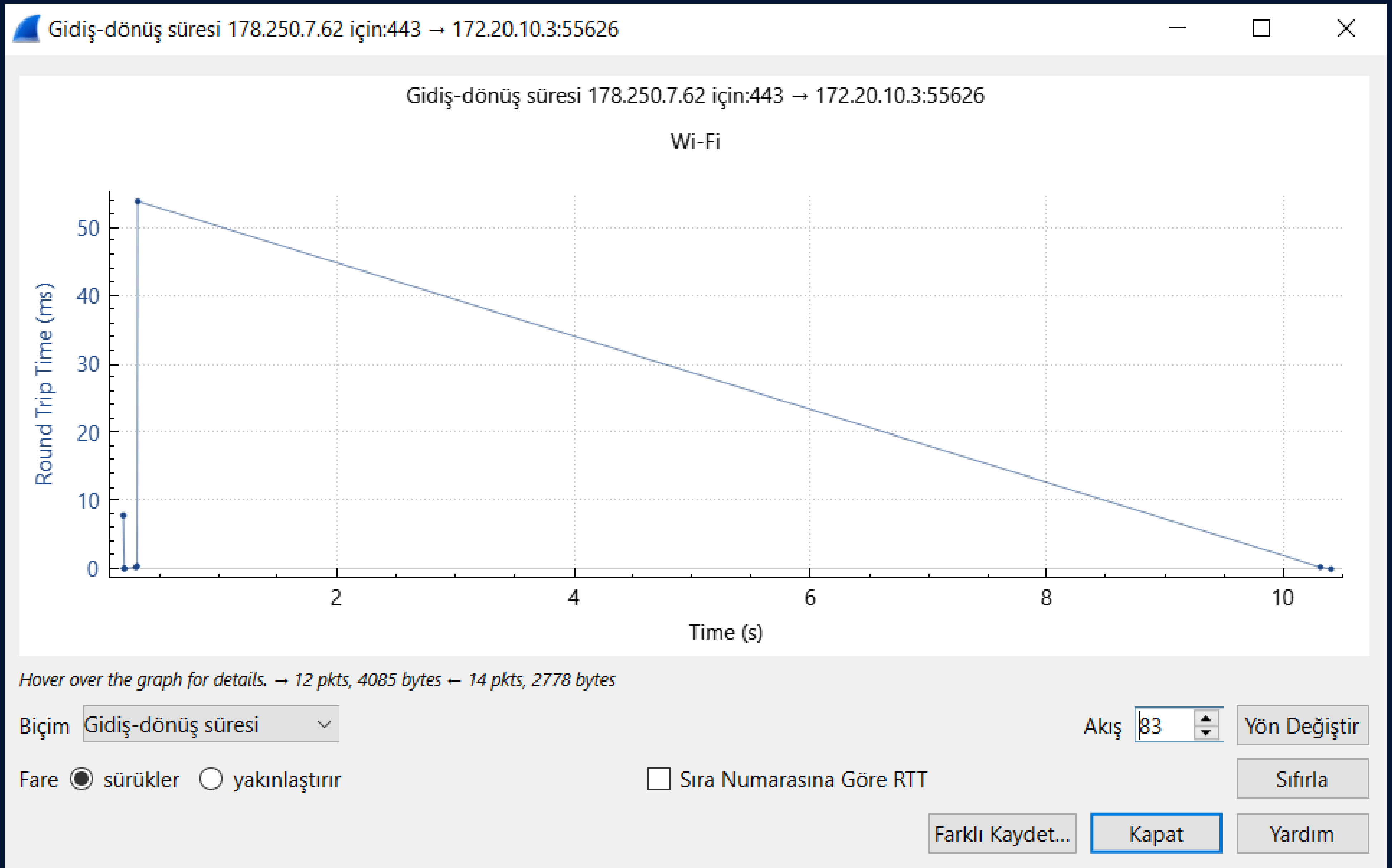
Fare ● sürükler ○ yakınlaştırır Aralık 1 saniye

□ Günün zamanı □ Günlük ölçeği ☑ Automatic update ☑ Enable legend

- To analyze the RTT values of packets, Round Trip values can be seen graphically according to TCP flows in the Statistics section.



```
.114    etransmission] 55630 → 4
.114    etransmission] 55632 → 4
.114    etransmission] 443 → 555
.114    purious Retransmission]
        purious Retransmission]
        purious Retransmission]
        etransmission] 55630 → 4
        etransmission] 55632 → 4
168     ast Retransmission] , Se
.114    purious Retransmission]
        etransmission] 55663 → 4
        etransmission] 55663 → 4
62      ast Retransmission] , Ap
62      ast Retransmission] 443
62      ast Retransmission] 443
62      ast Retransmission] 443
62      ast Retransmission] 443
62      ast Retransmission] , Ap
        etransmission] 55663 → 4
62      purious Retransmission]
62      purious Retransmission]
62      n]
62      n]
62      n]
 (528                           37A
:11 (c  (3e:6d:89:b5:df:64)
:64 (
ab:d5
```


Gidiş-dönüş süresi 178.250.7.62 için:443 → 172.20.10.3:55626
Gidiş-dönüş süresi 178.250.7.62 için:443 → 172.20.10.3:55626
Wi-Fi
Round Trip Time (ms)
0
10
20
30
40
50
2
4
6
8
10
Time (s)
Yön Değiştir
Sıfırla
Farklı Kaydet...
Kapat
Yardım

Hover over the graph for details. → 12 pkts, 4085 bytes ← 14 pkts, 2778 bytes

Biçim Gidiş-dönüş süresi

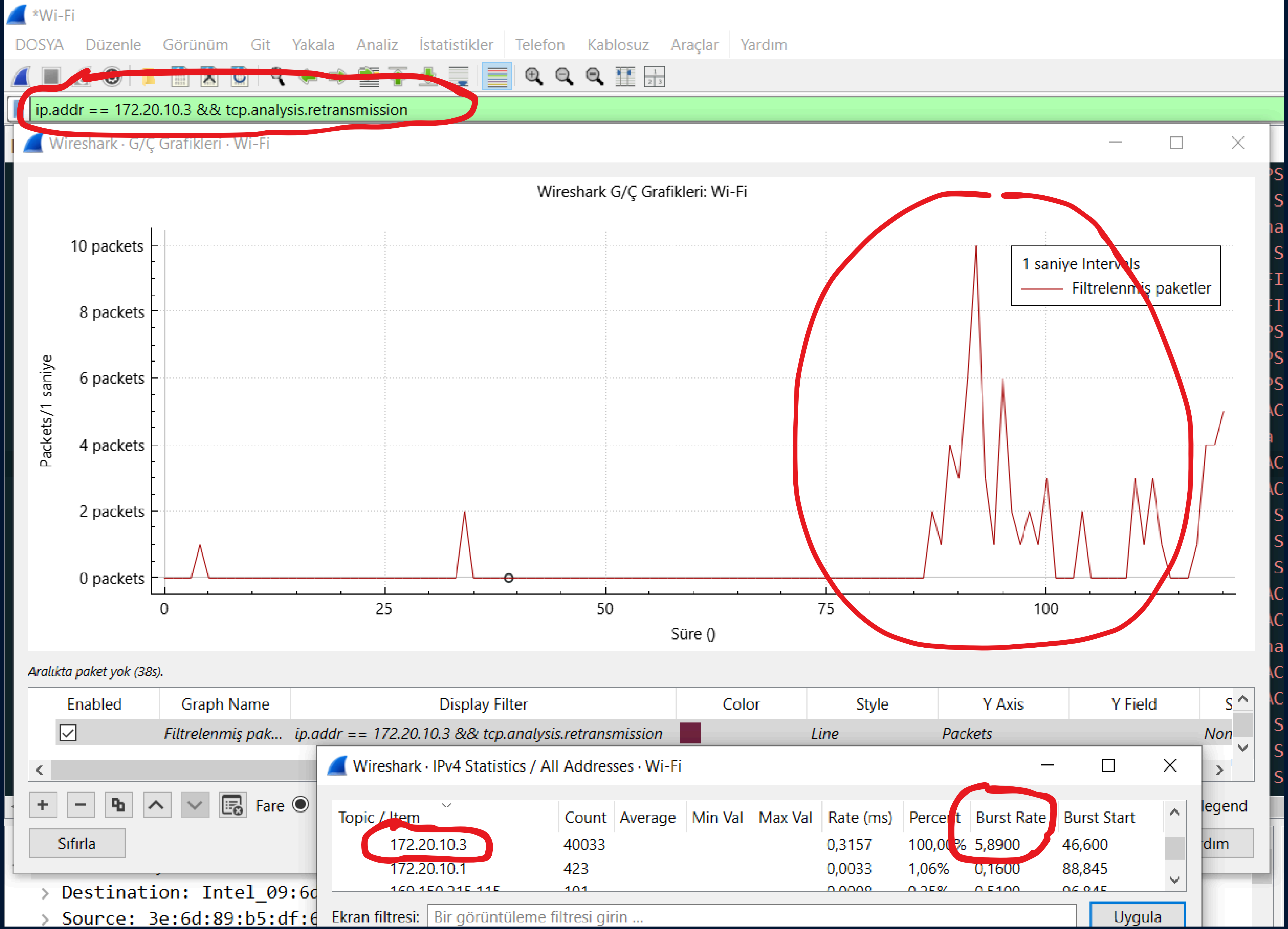
Akış 83

Fare ● sürükler ○ yakınlaştırır

□ Sıra Numarasına Göre RTT

- 83. When we look at the RTT values of the packets in the flow, we see that the RTT value increases at some point. This indicates that there are delays in the network. The lower the RTT, the lower the delays in the network.

- Burst Rate oranı fazla olan IP adresi ile yeniden gönderim değerleri analiz edilebilir. En fazla retransmission olan zaman aralığında belirli IP nin yer aldığını görüyoruz.



```
ip.addr == 172.20.10.3 && tcp.analysis.retransmission
```

Aralıkta paket yok (38s).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	S
☑	*Filtrelenmiş pak...*	*ip.addr == 172.20.10.3 && tcp.analysis.retransmission*	■	*Line*	*Packets*		*Non*

Topic / Item	Count	Average	Min Val	Max Val	Rate (ms)	Percent	Burst Rate	Burst Start
172.20.10.3	40033				0,3157	100,00%	5,8900	46,600
172.20.10.1	423				0,0033	1,06%	0,1600	88,845

Ekran filtresi: Bir görüntüleme filtresi girin ...