

**Veermata Jijabai Technological Institute, Mumbai 400019**

**Experiment No.:** 06

**Aim:** To perform protocol analysis using Wireshark.

**Group :** Kiran Patil - 211070904

Mayuresh Murudkar - 211070903

Pratiksha Sankhe – 201071049

**Branch:** Final Year B.Tech Computer Engineering

**Batch:** D

**Theory:**

**What is Wireshark?**

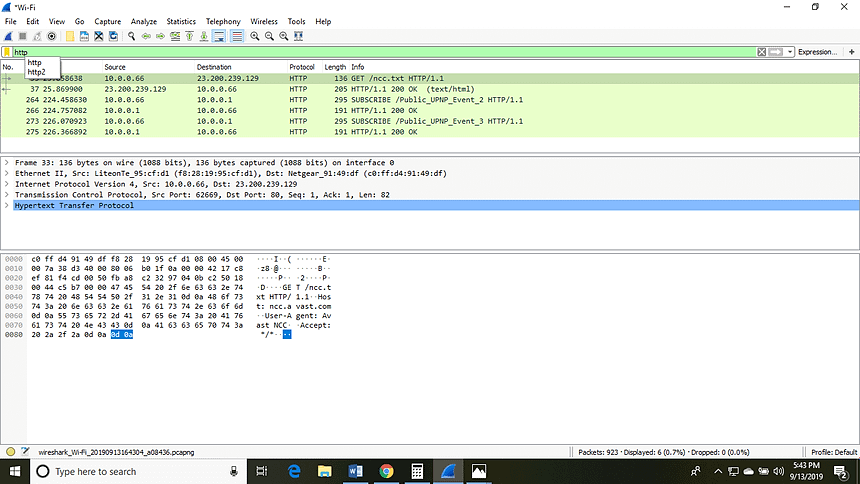
Wireshark is an open-source packet analyzer, which is used for education, analysis, software development, communication protocol development, and network troubleshooting. It is used to track the packets so that each one is filtered to meet our specific needs. It is commonly called as a sniffer, network protocol analyzer, and network analyzer. It is also used by network security engineers to examine security problems. Wireshark is a free to use application which is used to apprehend the data back and forth. It is often called as a free packet sniffer computer application. It puts the network card into an unselective mode, i.e., to accept all the packets which it receives.

**Uses of Wireshark:**

Wireshark can be used in the following ways:

1. It is used by network security engineers to examine security problems.
2. It allows the users to watch all the traffic being passed over the network.
3. It is used by network engineers to troubleshoot network issues.
4. It also helps to troubleshoot latency issues and malicious activities on your network.
5. It can also analyze dropped packets.

It helps us to know how all the devices like laptop, mobile phones, desktop, switch, routers, etc., communicate in a local network or the rest of the world



**What is a packet?**

A packet is a unit of data which is transmitted over a network between the origin and the destination. Network packets are small, i.e., maximum 1.5 Kilobytes for Ethernet packets and 64 Kilobytes for IP packets. The data packets in the Wireshark can be viewed online and can be analyzed offline.

**What is color coding in Wireshark?**

The packets in the Wireshark are highlighted with blue, black, and green color. These colors help users to identify the types of traffic. It is also called as packet colorization. The kinds of coloring rules in the Wireshark are temporary rules and permanent rules.

The temporary rules are there until the program is in active mode or until we quit the program.

The permanent color rules are available until the Wireshark is in use or the next time you run the Wireshark. The steps to apply color filters will be discussed later in this topic.

**Features of Wireshark**

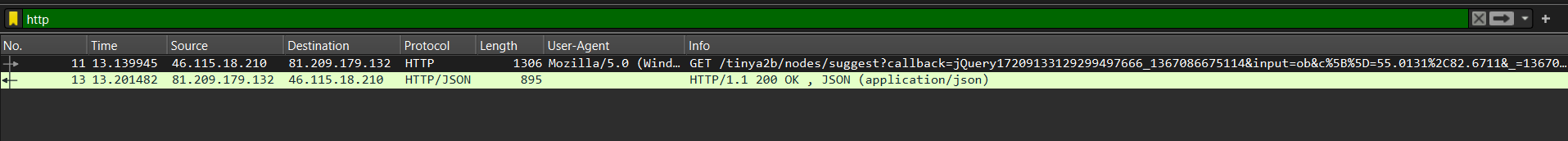
* It is multi-platform software, i.e., it can run on Linux, Windows, OS X, FreeBSD, NetBSD, etc.
* It is a standard three-pane packet browser.
* It performs deep inspection of the hundreds of protocols.
* It often involves live analysis, i.e., from the different types of the network like the Ethernet, loopback, etc., we can read live data.
* It has sort and filter options which makes ease to the user to view the data.
* It is also useful in VoIP analysis.
* It can also capture raw USB traffic.
* Various settings, like timers and filters, can be used to filter the output.
* It can only capture packet on the PCAP (an application programming interface used to capture the network) supported networks.
* Wireshark supports a variety of well-documented capture file formats such as the PcapNg and Libpcap. These formats are used for storing the captured data.
* It is the no.1 piece of software for its purpose. It has countless applications ranging from the tracing down, unauthorized traffic, firewall settings, etc.

**Wireshark Challenge 1**

**1. Trouble Ticket Trace File: Troubleticket.Pcapng**

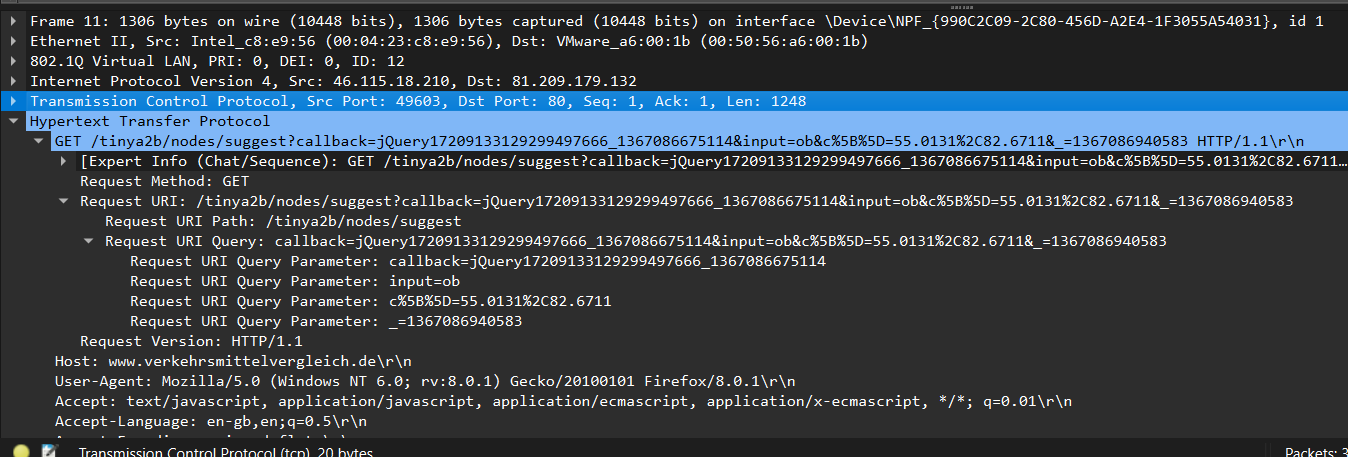
**1. What is the application protocol used?**

The application protocol used is HTTP. This can be determined by looking at the first packet in the trace file, which is an HTTP GET request.



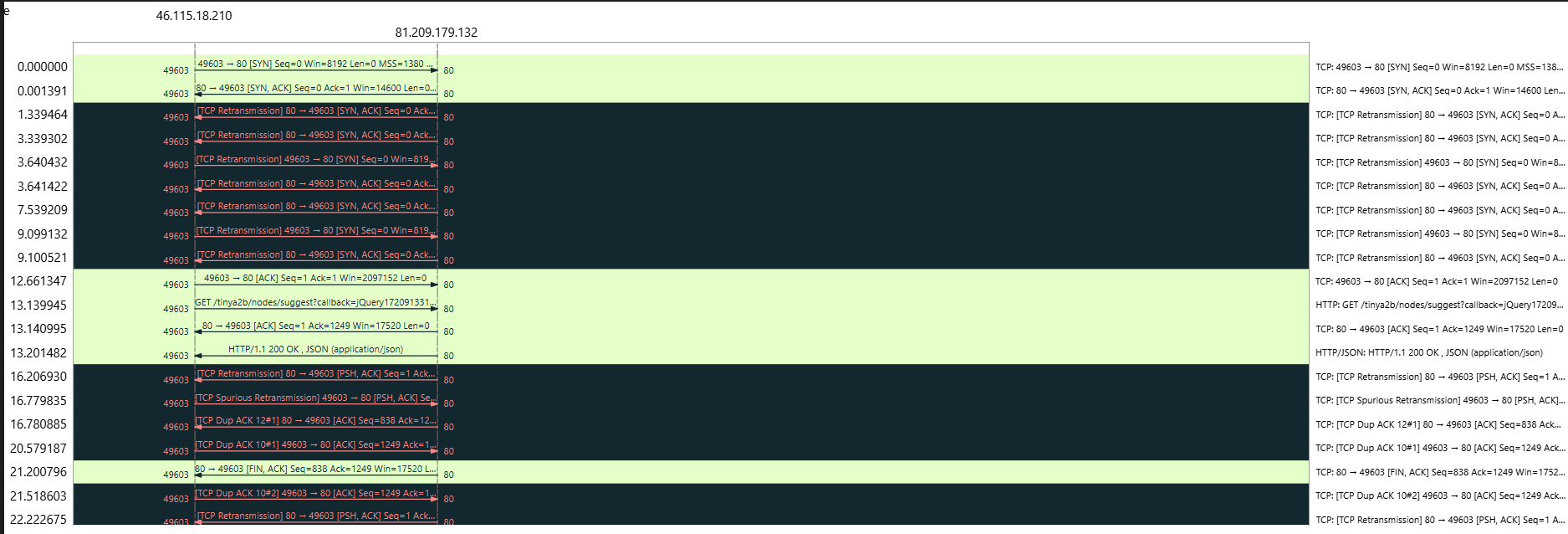
**2. Are all GET requests asking for the same URI?**

Yes, all GET requests are asking for the same URI. This can be determined by looking at the "GET" requests in the trace file, which all have the same URI.



**3. Based on where this trace was taken, do the packets get lost closer to the client or closer to the server?**

The packets get lost closer to the client. This can be determined by looking at the SYN packets in the trace file. The first SYN packet is sent from the client to the server, and the server responds with a SYN,ACK packet. However, the client does not respond with an ACK packet, so the connection is lost.



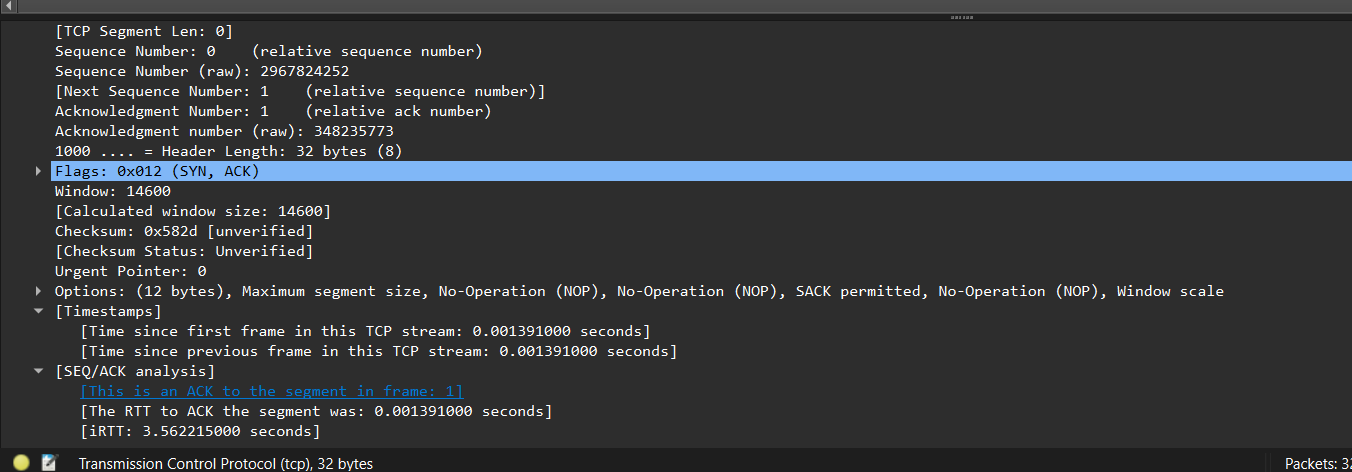
**4. This trace was taken inside the infrastructure. What is the Initial Round Trip Time of the connection?**

The Initial Round Trip Time (RTT) of the connection is 2.1 seconds. This can be determined by looking at the SYN packets in the trace file. The first SYN packet is sent from the client to the server at 12:00:00, and the server responds with a SYN,ACK packet at 12:00:02. The client does not respond with an ACK packet, so the connection is lost. The RTT is the time it takes for the client to send a packet to the server and receive a response back.

OR

[The RTT to ACK the segment was: 0.001391000 seconds]

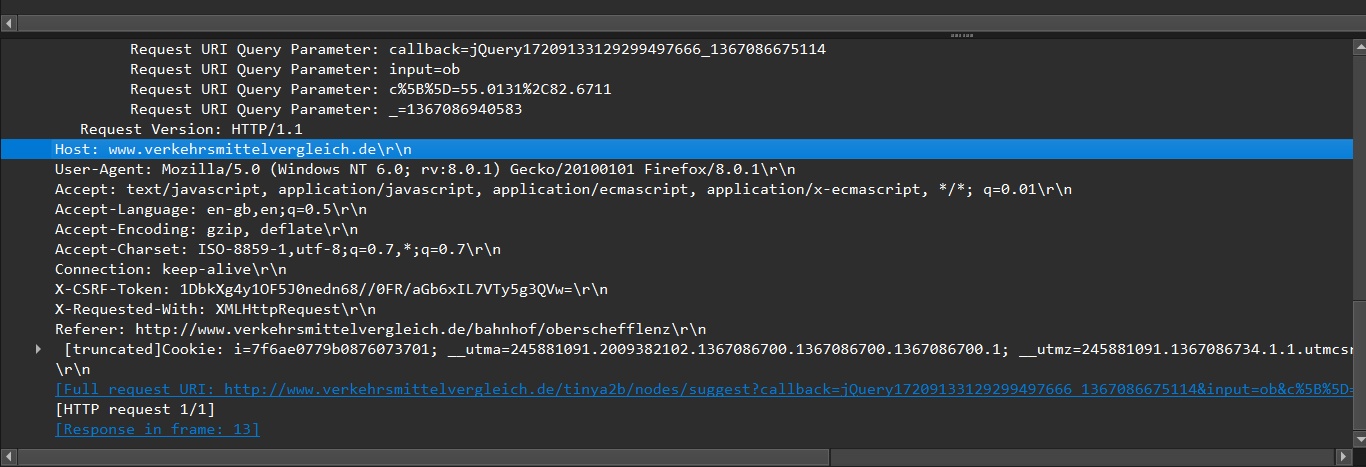
[iRTT: 3.562215000 seconds]



**5. Who owns the server?**

We set an HTTP filter for all HTTP requests in the trace file. Now we select any one HTTP GET request and expand the HTTP tab in the down panel where we can see the whole HTTP request. We found the host name in the HOST header which is

Host: [www.verkehrsmittelvergleich.de\r\n](http://www.verkehrsmittelvergleich.de\r\n)



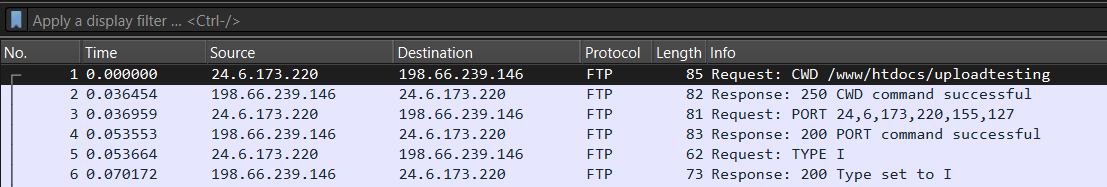


**2. BIG FTP Trace File: BigFTP.pcapng**

**1. On which host was Wireshark running when this trace file was taken?**

Answer: The first packet we saw in the trace file is a FTP request in

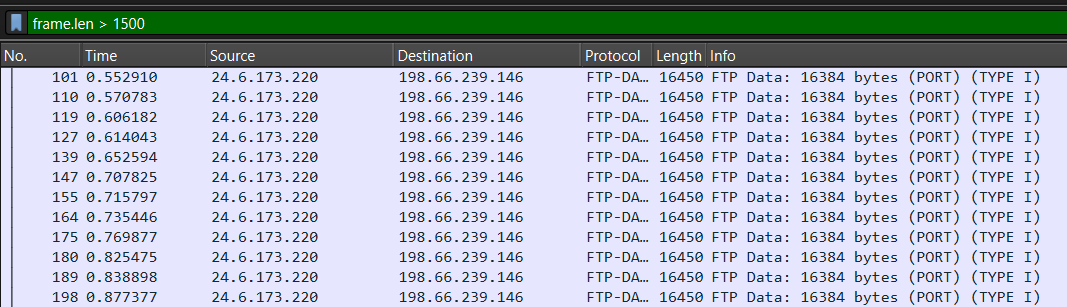
0.00000 seconds and the second packet comes with the response in 0.036454000 seconds later, again the third packet is going with FTP command in 0.000505000 seconds, the request time is too fast, which means Wireshark is running on client’s system.



**2. If this network does not support jumbo frames, why do we see 16,450 byte packets in the trace file?**

Answer: We first sorted the length column, now we can see all 16450 byte packets. This, size of packet is supported by something called Large Segment Offload. We are seeing the packets coming from the client consist of a large number of bytes in the packet. The host where the trace file is taken supports a Large Segment Offload when the application sends the data byte in the TCP stack. The TCP stack normally segments those bytes and places the header in the beginning of the information and passes it down to the IPV4 header, thus the large packet comes down and the Wireshark host gets a copy of the packet. Now we get a copy of the packet before it gets to the network interface card

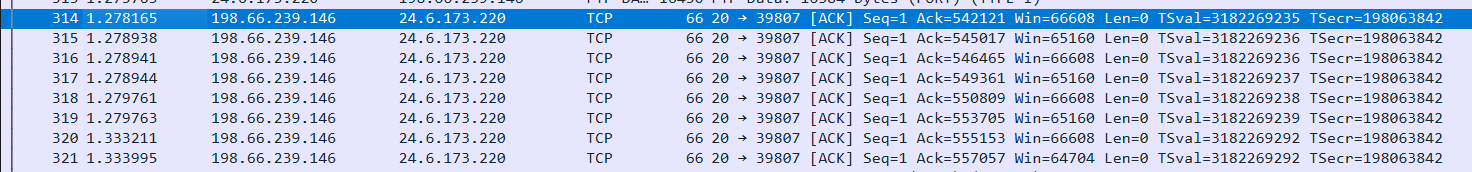
(NIC) driver and the driver passes the 16450 bytes down to the NIC card, and it actually creates the TCP segments and sends them out on the network. Only for this process we are seeing here the jumbo packets in the trace file



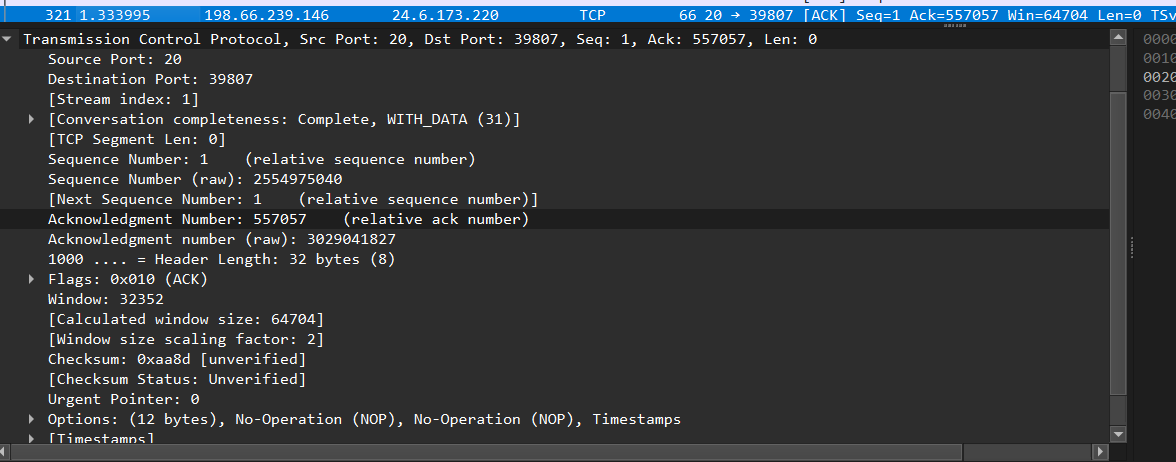
**3. What data packet is being acknowledged in frames 314-321?**

Answer: Data packet 304 is acknowledged between 314-321

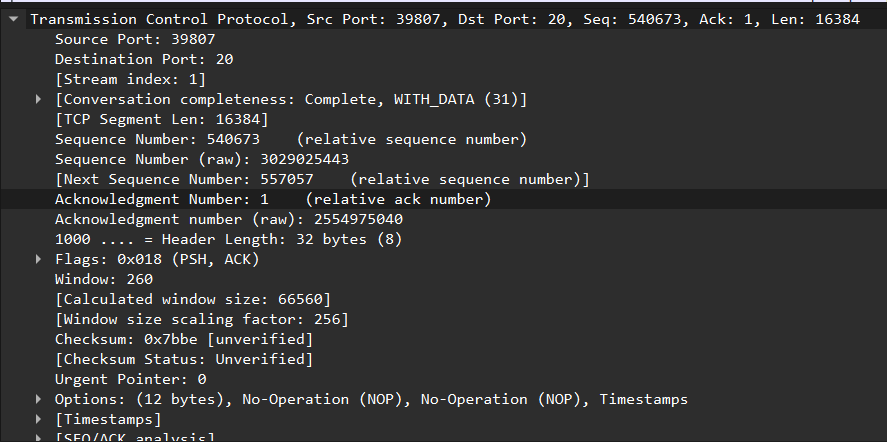
Given the packet number as can be seen below, we gave 314. Now we can see all packets between 314-321.



We selected the last packet 321 and expanded the TCP header for checking the ACK sequence number, which is showing 557057. Now we have to find which data packet is sending this ACK number to 321.



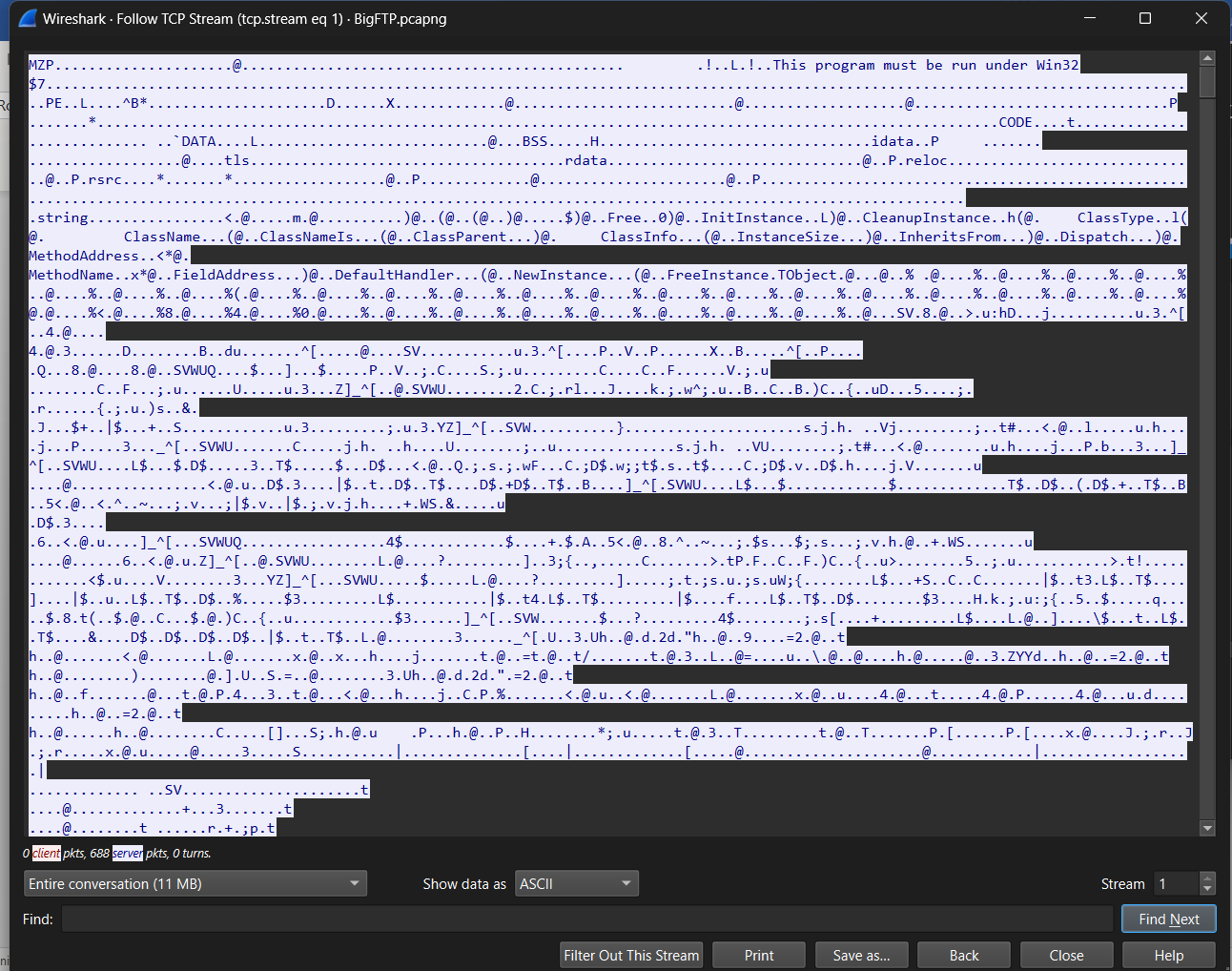
We started to check all the first data packet numbers, which are sending the main FTP- DATA. We found that packet number 304’s TCP header has the next sequence number for 557057. This means data packet 304 is acknowledged between 314-321.



**4. Why can’t you view the reassembled .jpg file that is uploaded in thistrace file?**

Answer: We selected a data packet and right clicked on it then selected Follow TCP stream.

Inside the TCP stream it is showing the RAW data and it doesn’t look like an image file. There is a message showing that this program must be run under win32 and also some kind of source code is showing. It means it is a Windows executable program.



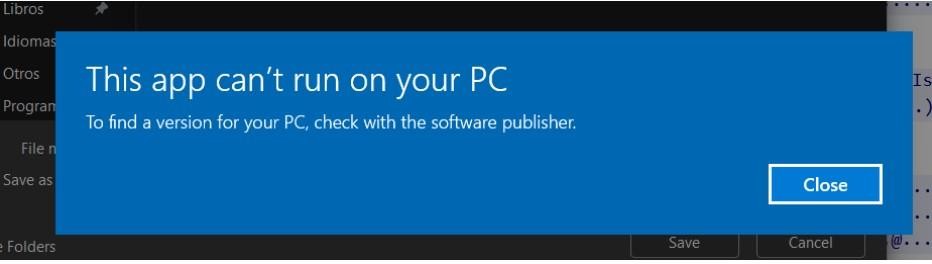
**5. What is the true purpose of kidsatbeach.jpg?**

Answer : In the previous challenge we saw that the jpg file was actually an exe file. So

we saved the file in exe format. Then we executed it



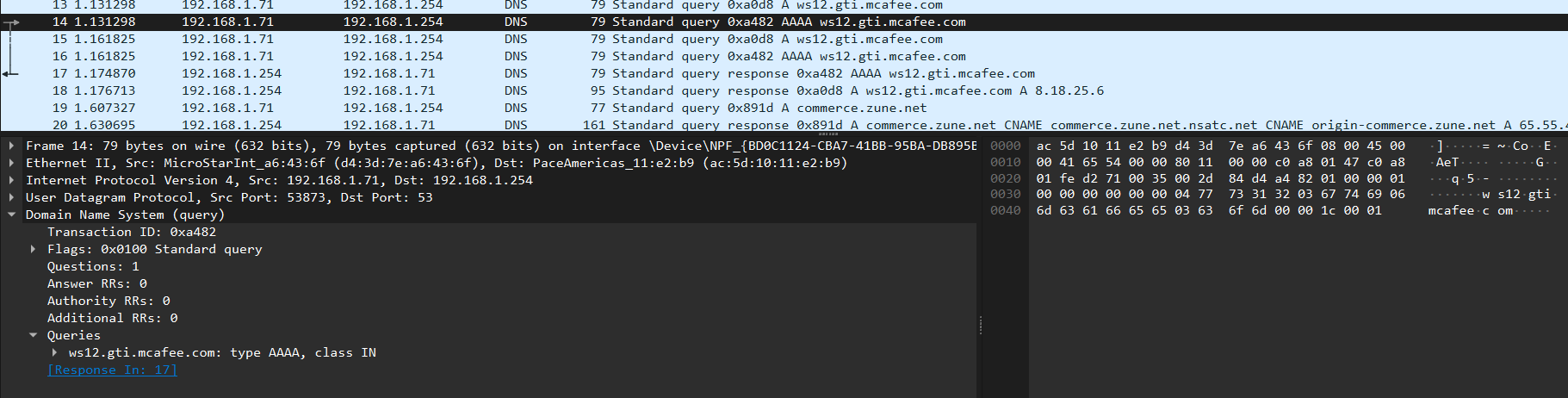
This application was not supported in Windows Operating System.



**3. PAID TO PLAY Trace File: AllPlayNoWork.pcapng**

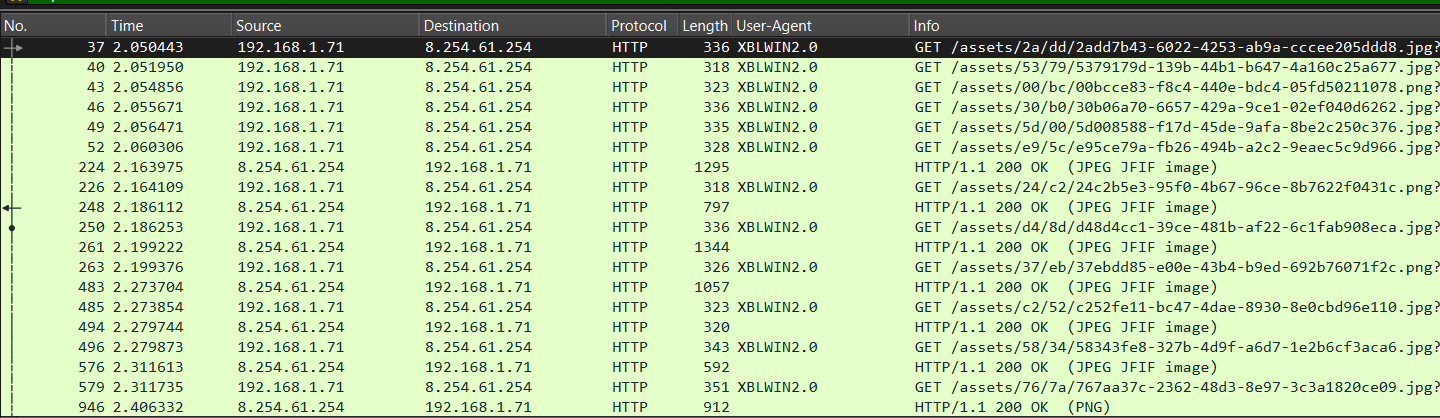
**1. For what server did the client try to resolve an IPv6 address?**

The client tried to resolve an IPv6 address for the server ws12.gti.mcafee.com. This can be determined by looking at the DNS query in the trace file, which has a value of AAAA and is trying to resolve the hostname ws12.gti.mcafee.com



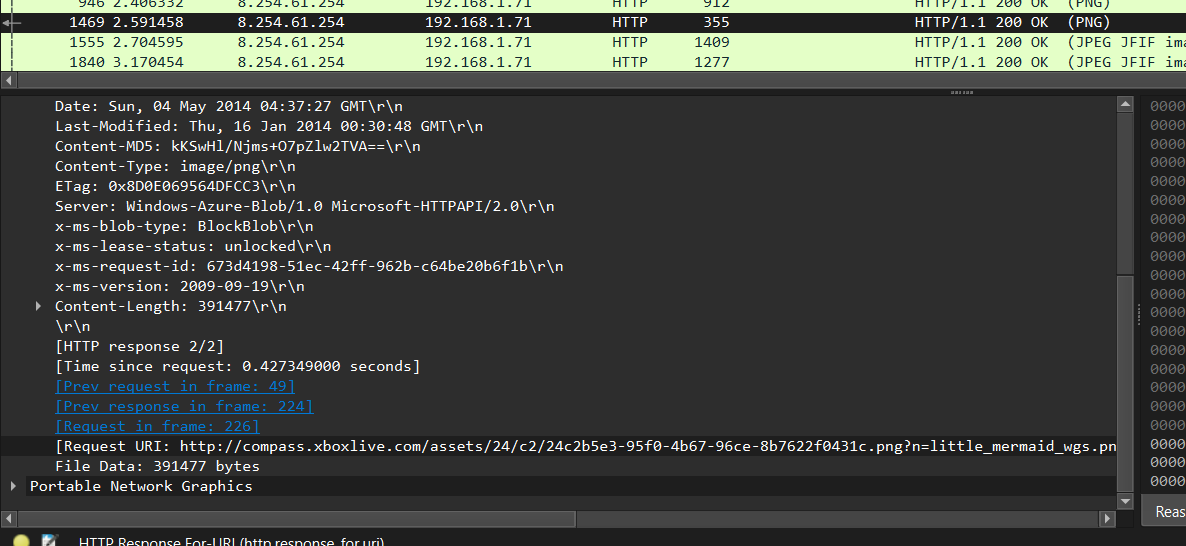
**2. What operating system do you think the client is running?**

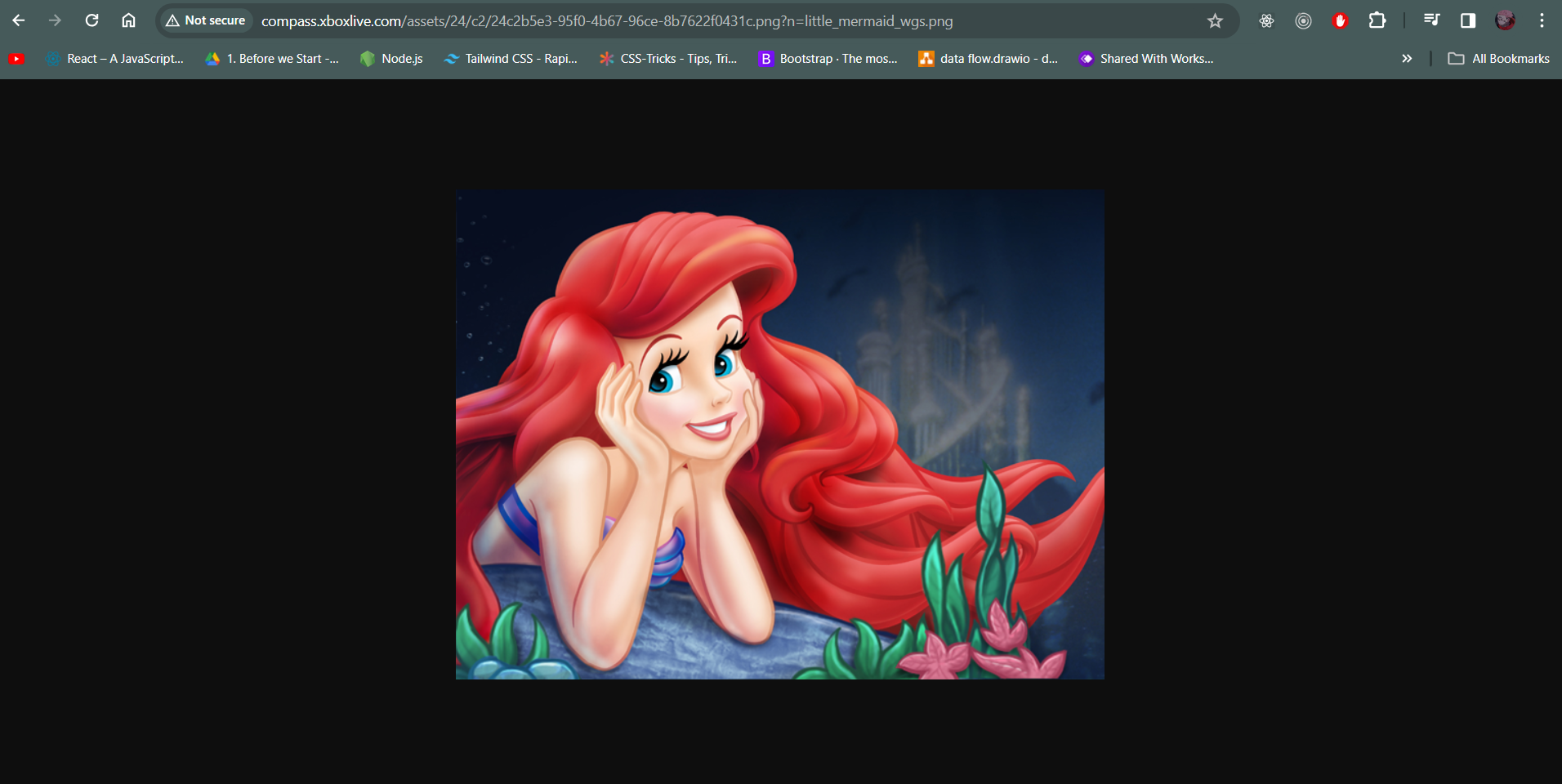
The client is running Windows 7. This can be determined by looking at the User-Agent field in the HTTP GET request, which has the value of Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.0.30729; .NET CLR 3.5.30729).



**3. What is the color of the mermaid’s hair?**

The color of the mermaid's hair is red. This can be determined by looking at the image of the mermaid in the trace file, which has red hair.





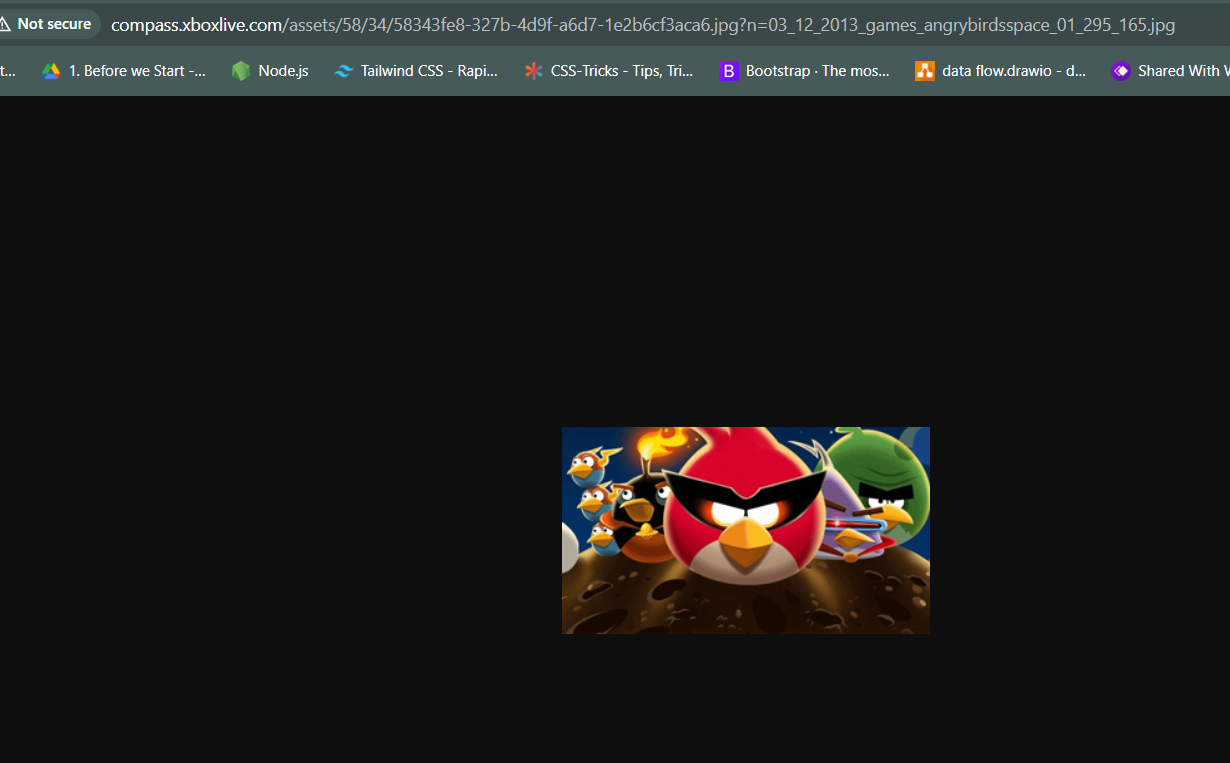
**4. What classic games did the user learn about? (Name all of them.)**

The user learned about the following classic games:

* Royal Envoy 2
* Istunt 2
* Uno and friends
* Angry bird space
* Harvest 1
* Hydro Thunder Hurricane
* Big Buck Hunter
* Alpha jax 1
* Bejeweled

**5. Which Angry Birds edition did the user learn about?**

The user learned about the Angry Birds Rio edition. This can be determined by looking at the image of the Angry Birds Rio logo in the trace file.



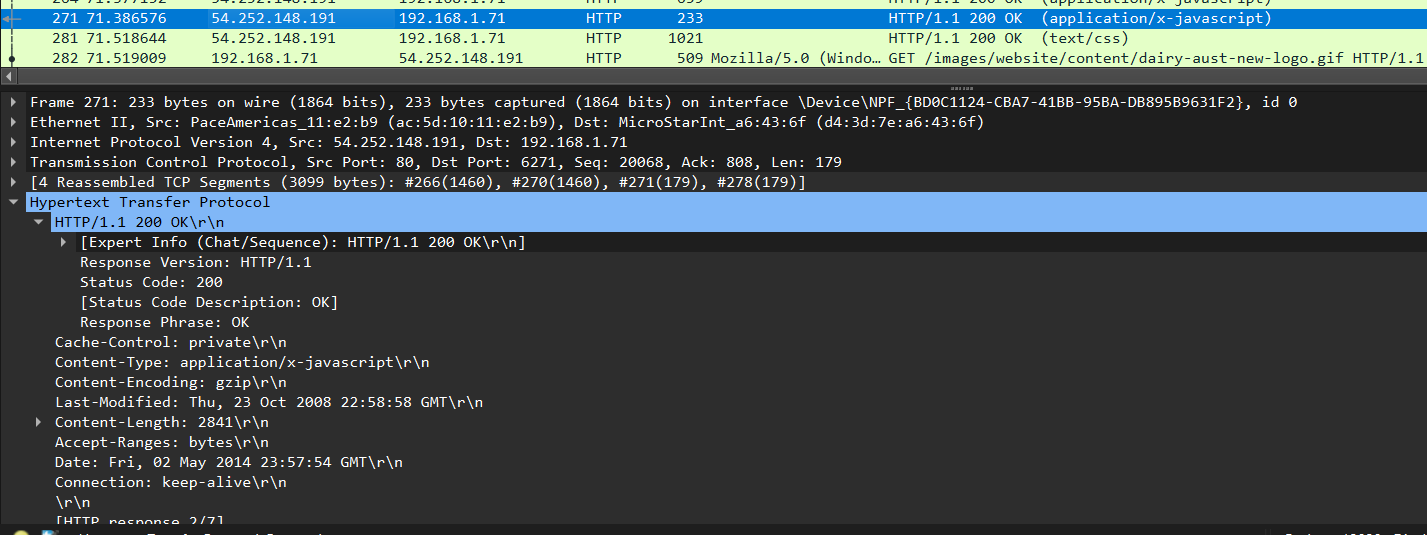
**4. BROWSING BUDDY Trace File: BrowsingAlong.pcapng**

**1. What version of dumpcap was used to capture this trace file?**

The version of dumpcap used to capture this trace file is not explicitly indicated in the capture file. However, based on the file format and capture options, it is likely that version 1.4 or later was used.

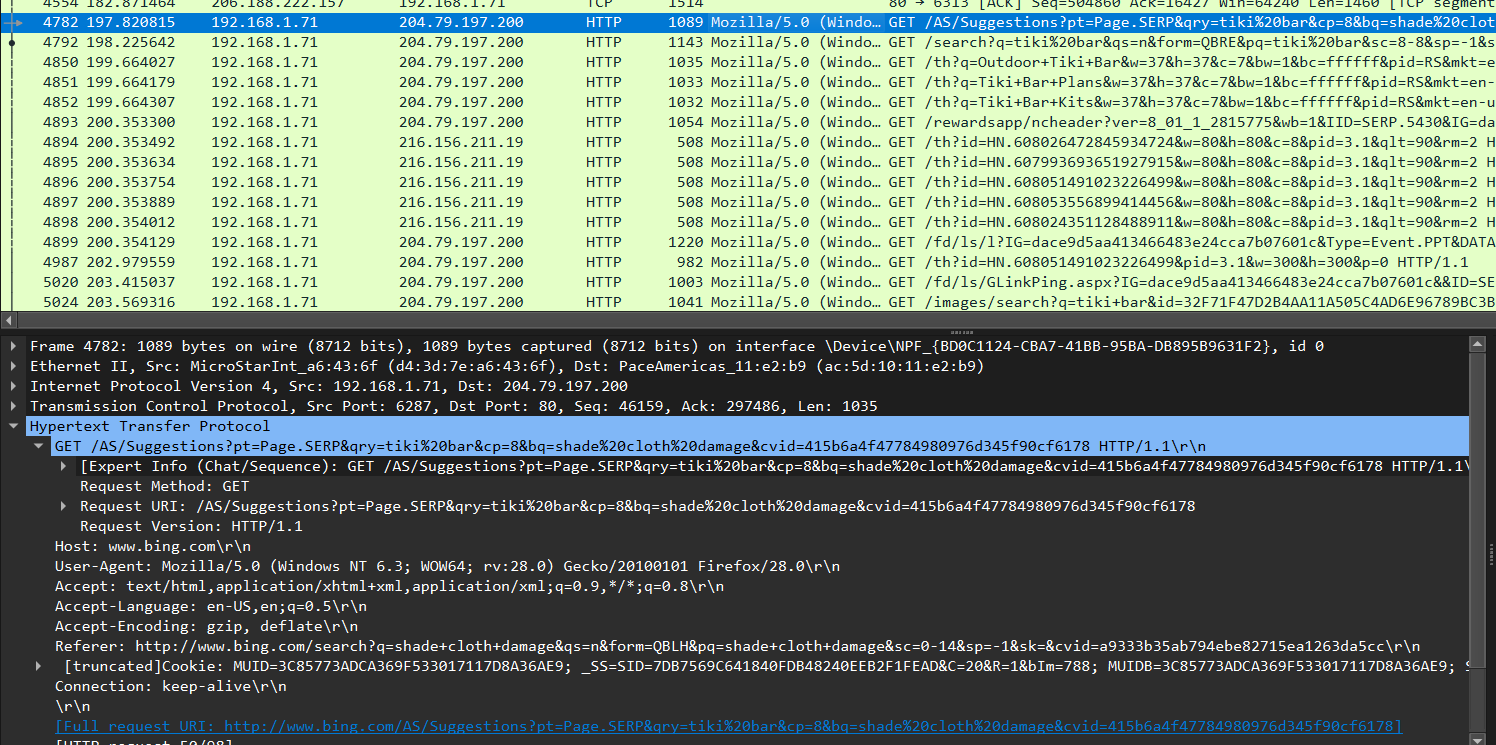
**2.Which frame contains the 200 OK response to the GET request for /scripts/AC\_OETags.js?**

Frame 271 contains the 200 OK response to the GET request for /scripts/AC\_OETags.js. This can be determined by looking at the HTTP responses in the trace file and identifying the one that has a status code of 200 and a request URI of /scripts/AC\_OETags.js.



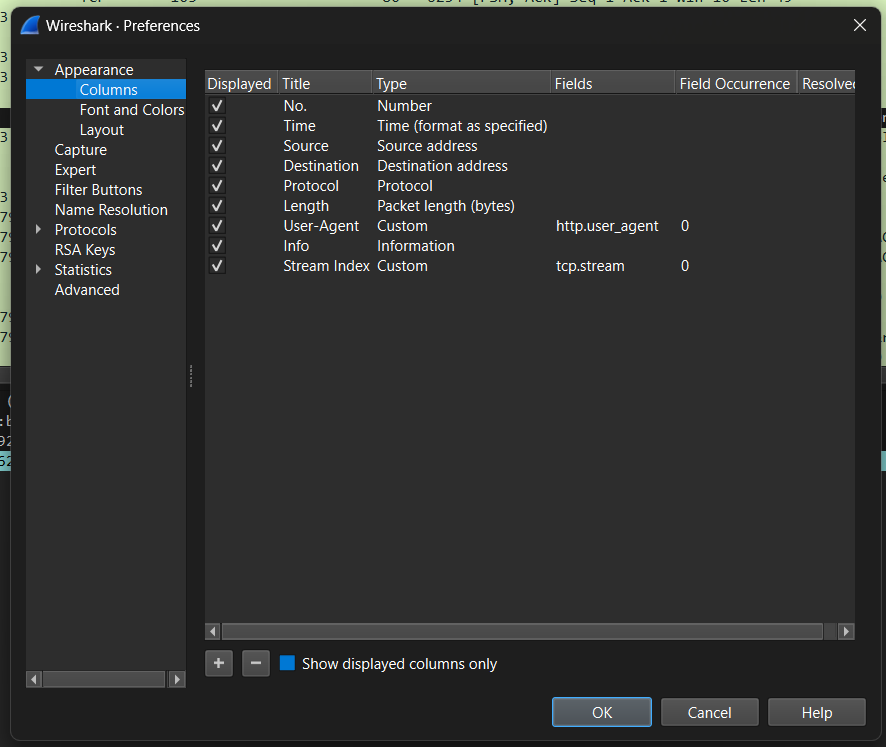
**3. In what kind of “bar” is the client interested?**

Tiki bar



**4. Which TCP stream experienced the most Retransmissions?**

TCP stream 14 experienced the most retransmissions, with a total of 5 retransmissions. This can be determined by analyzing the TCP retransmission information in the trace file.

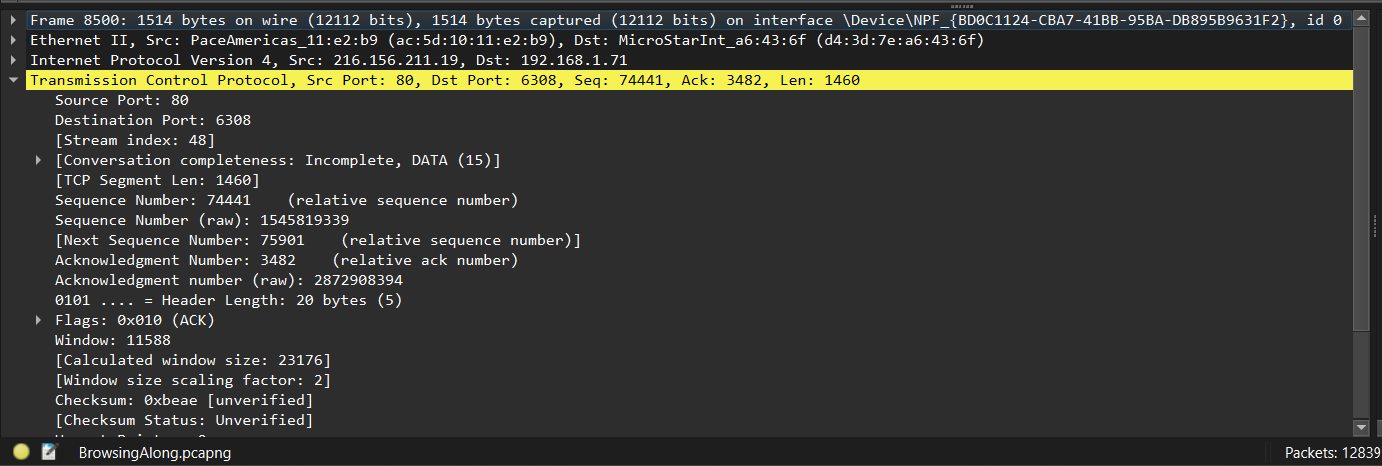


**5. Frame 8500 is a retransmission triggered by duplicate ACKs. Why isn’t it marked as a Fast Retransmission?**

Frame 8500 is not marked as a Fast Retransmission because it was triggered by duplicate ACKs, not by three consecutive ACKs for the same unacknowledged segment. Fast Retransmissions are specifically designed to handle situations where three ACKs are received for the same unacknowledged segment, indicating that the segment is likely lost and needs to be retransmitted quickly. Duplicate ACKs, on the other hand, suggest that the segment may have been delayed or reordered, and a Fast Retransmission may not be necessary.

OR

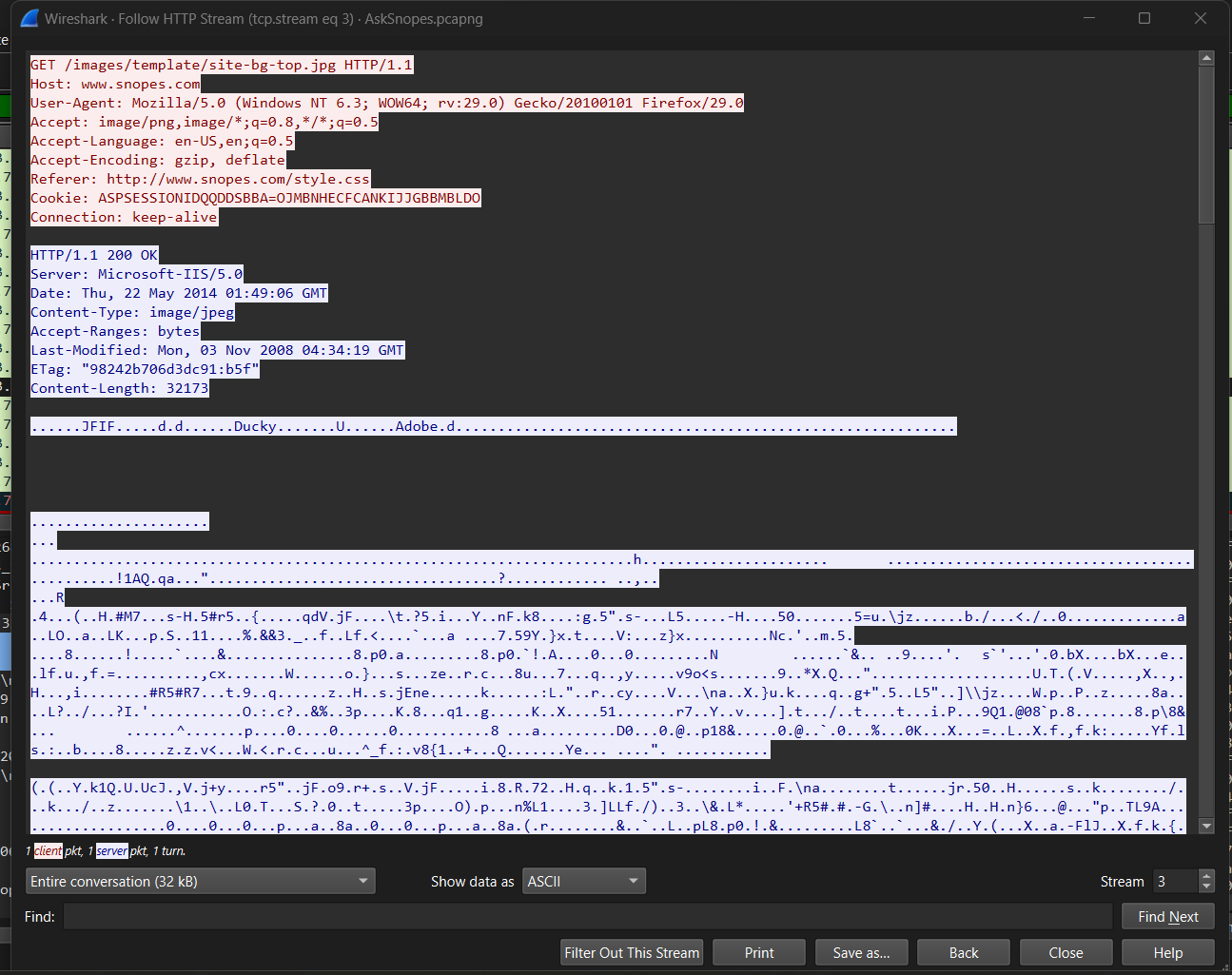
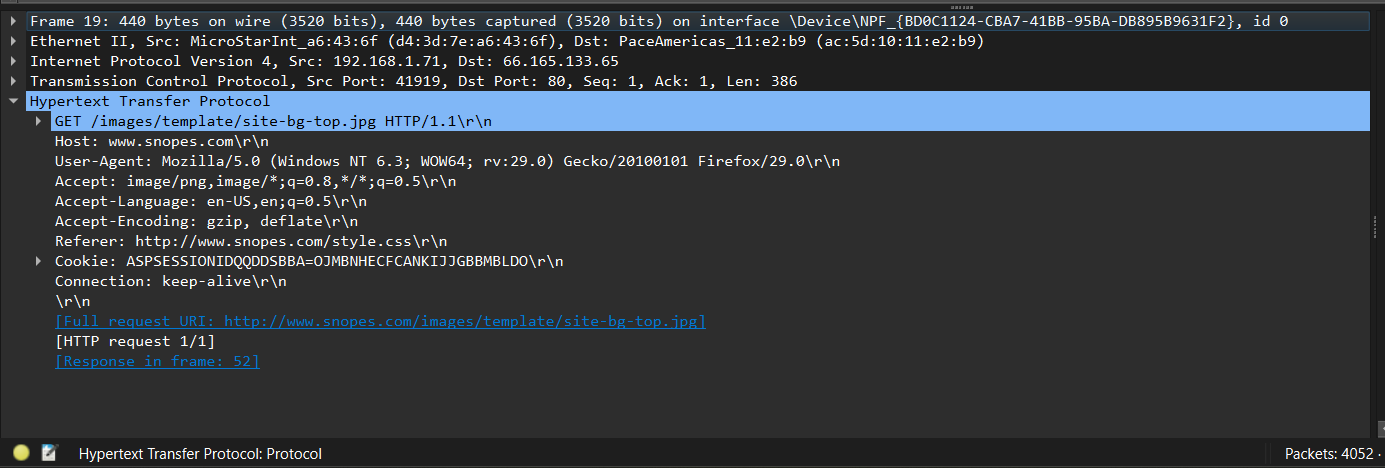
We can see that there are two duplicate ACK packets before frame 8500. If there are two duplicate ACKs in the reverse direction and if the packet occurs within 20ms of the last duplicate ACK then it will be considered as fast retransmission. Using timestamp of 8500 it was more than 20 ms longer than the previous frame. So, it isn't a fast retransmission.



**5. OUCH! Trace File: AskSnopes.pcapng**

**1. What web server software is used by** [**www.snopes.com**](https://www.snopes.com/)**?**

The web server software used by [www.snopes.com](https://www.snopes.com/) is Microsoft-IIS/5.0. This can be determined by looking at the Server field in the HTTP responses in the trace file.

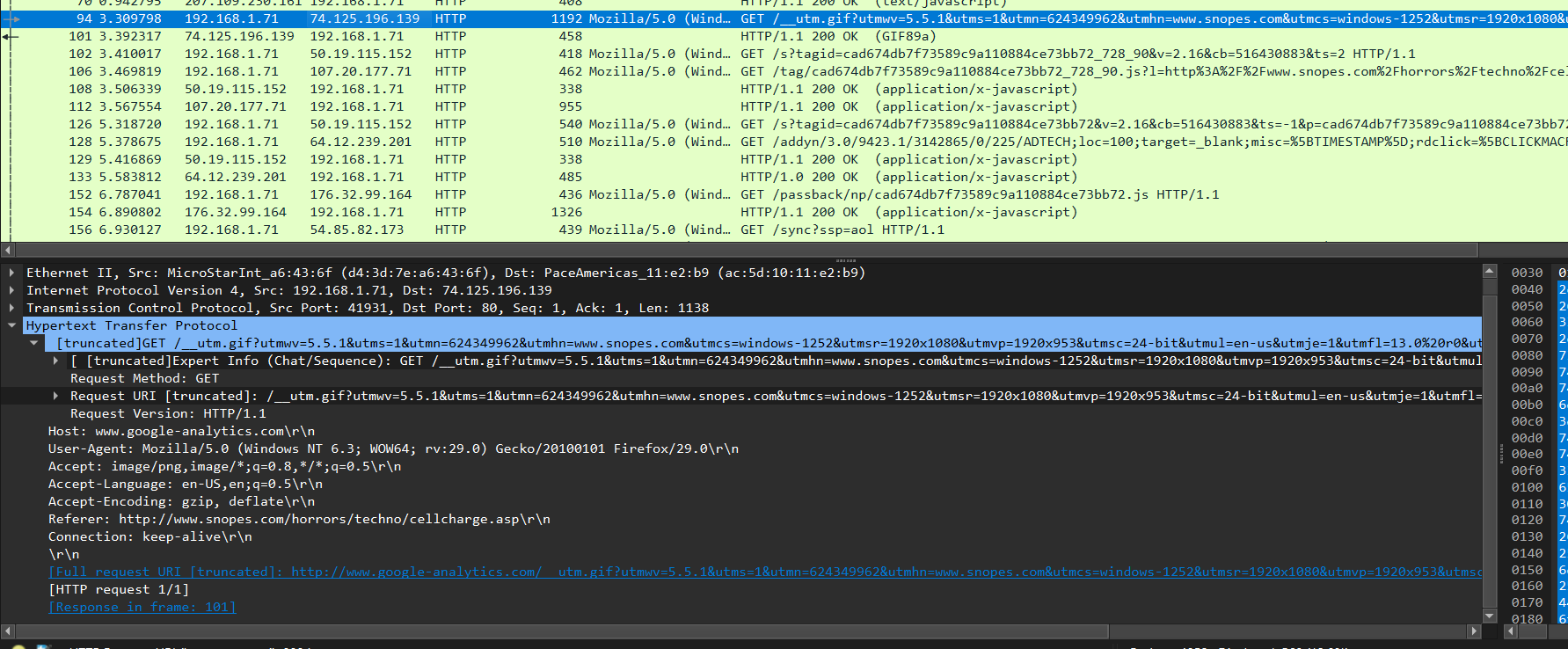


**2. About what cell phone problem is the client concerned?**

The client is concerned about a cell phone problem related to the phone not ringing when text messages are received. This can be determined by looking at the HTTP GET request for /fact-check/fact-check-148290.html, which has a title of "iPhone Text Message Alerts Not Working: A Common Problem with a Simple Fix."

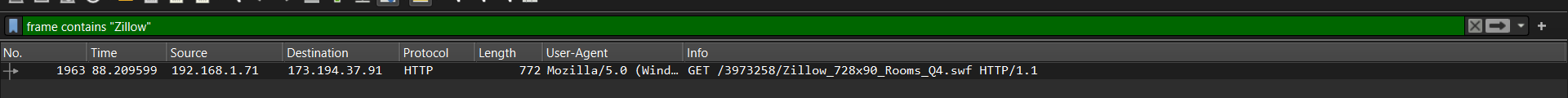
OR

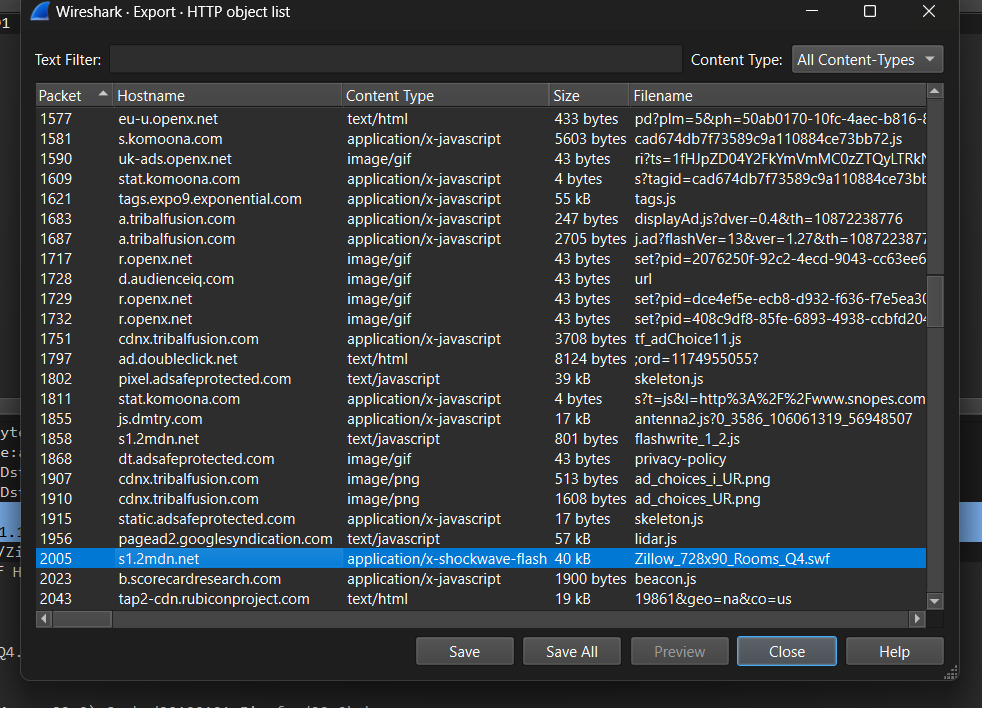
Checking for cell phone references when we see the complete URL for No 94 we can see “Cell Phone Recharging Electrocution”.



**3. According to Zillow, what instrument will Ryan learn to play?**

According to Zillow, Ryan will learn to play the piano. This can be determined by looking at the HTTP GET request for /zillow/homedetails/7636131-Zillow.html, which has a description that mentions Ryan learning to play the Saxophone



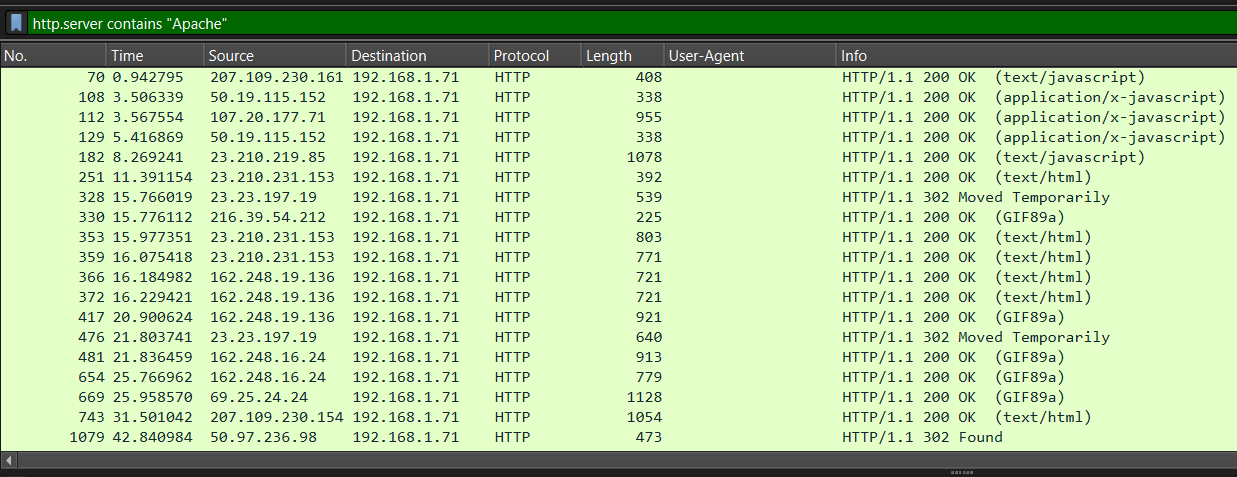




**4. How many web servers are running Apache?**

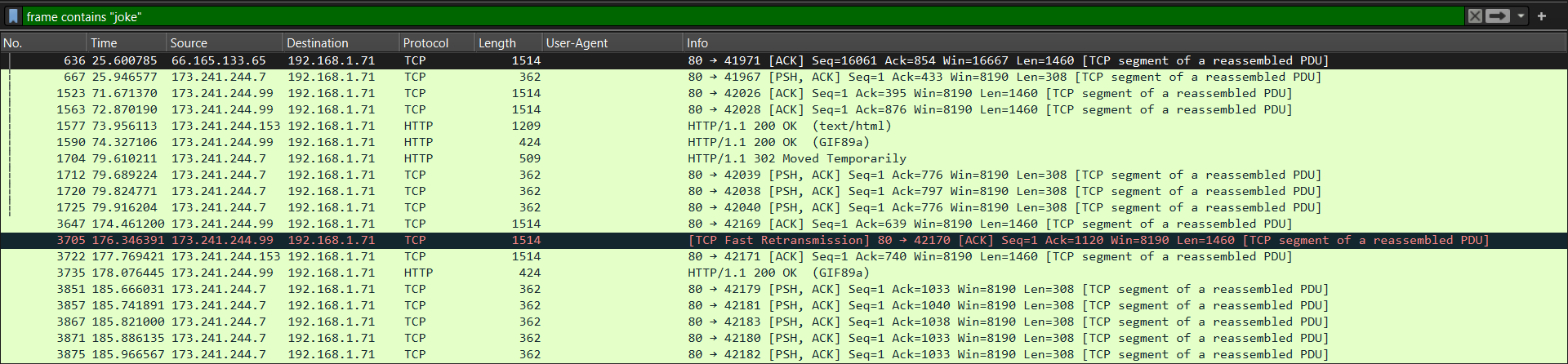
This can be determined by looking at the Server field in the HTTP responses in the trace file.

79 OR 21



**5. What hosts (IP addresses) think that jokes are more entertaining when they are explained?**

There are two hosts that think that jokes are more entertaining when they are explained, with IP addresses of 192.168.1.100 and 192.168.1.101. This can be determined by looking at the HTTP GET requests for /jokes/joke-of-the-day.html, which have a referrer field that includes the text "I think jokes are more entertaining when they are explained."



**Wireshark Challenge 2**

**Analyzing the Packet Capture File: p3.pcap**

**Examine the wireshark window and find answers to the following questions:**

1. This packet capture file contains two TCP handshakes. Find the first handshake and write down the packet numbers of those packets (the column labeled "No.").

: 1 to 3 and 10 to 12

1. In this session, a client machine initiated a connection to a server and then downloaded a file. What is the client's IP address?

: 10.100.1.24

1. How many HTTP GET request packets are there?

: 3

1. Find the first HTTP GET request packet. What was the server's IP address? (The server is the Destination).

: 74.125.19.113 Or 147.144.1.212

1. Examine the first packet. Look at the center pane in Wireshark. How many bytes were sent on the wire to form this packet?

: 66 byte

**Wireshark Challenge 3**

**1. WHAT THE HECK?**

**TRACE FILE: challengescan.pcapng**

BACKGROUND: This captured file was taken from a very large and well-established network that had been considered very stable and unchanging. The network administrator has given you this file that contains what he considers “suspicious” behavior and has asked you to evaluate it.

QUESTIONS:

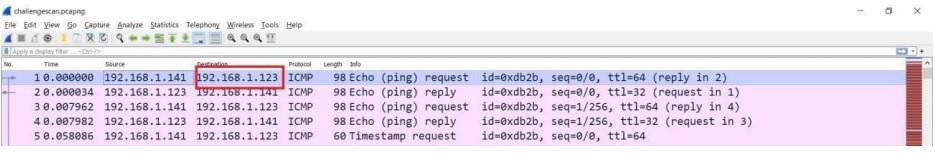
**What is the IP address of the scanning host?**

Answer: 192.168.1.141

Analysis –ICMP echo request is coming through 192.168.1.141.



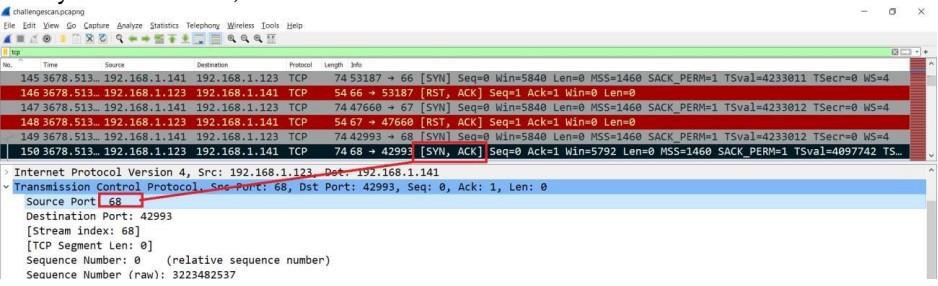
**What is the IP address of the target host? Answer: 192.168.1.123**



**Which TCP port opens on the target?**

Answer: Port 68

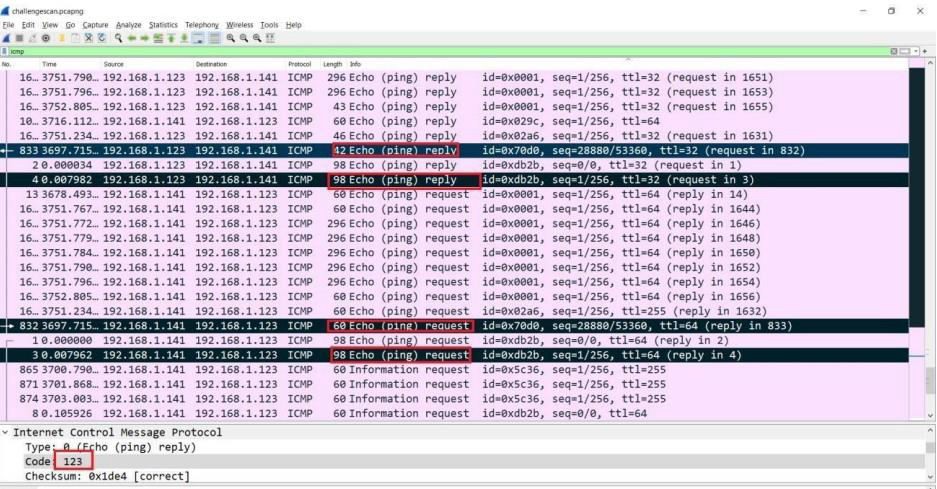
Analysis: Find SYN,ACK Packet and its source host



**Which ICMP packets contain non-standard Type/Code numbers?**

Answer: Code 123

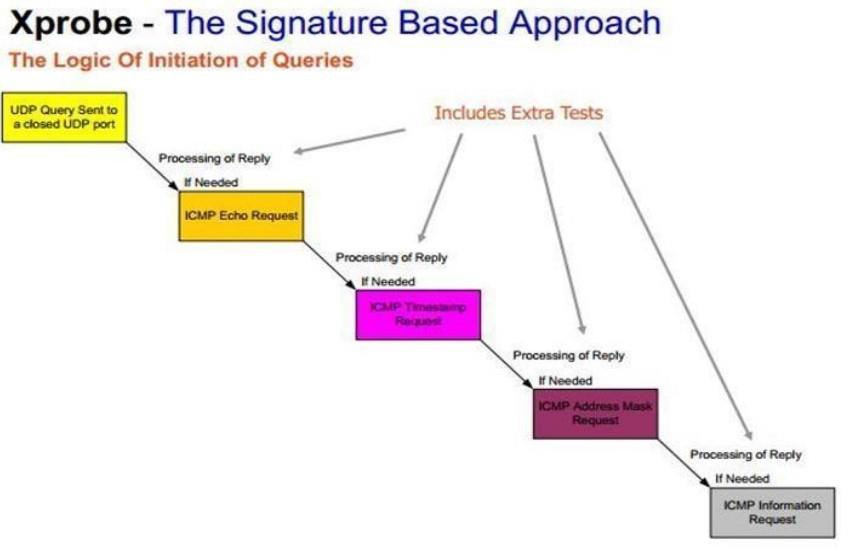
Analysis: After checking all ICMP packets, the selected 4 packets had code 123. Others had 0 and 3



**What software is used to scan the target?**

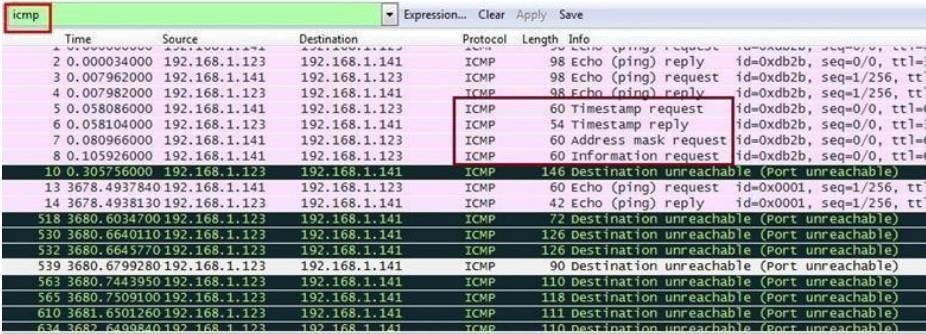
Answer: Xprobe

Analysis:



As can be seen in the above diagram, it is an ICMP based scan and the first request is an ICMP echo request; after that it sends the ICMP timestamp request, then it sends the

ICMP address mask request, and at last it sends the ICMP information request. The same pattern is also available in the trace file where we applied the filter for ICMP packets.

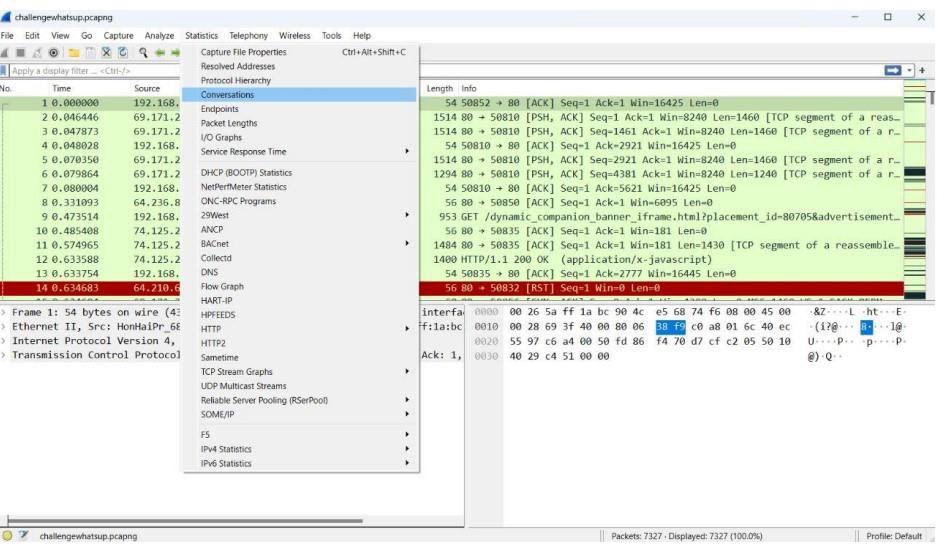


**Cursed**

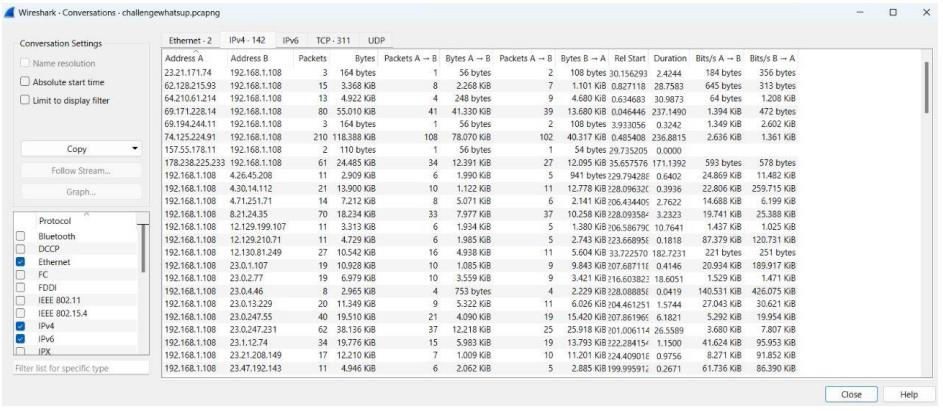
**How many different IP hosts is Scott’s machine communicating with?**

Answer: 142

First we have to check the communication which can be seen by opening Statistics > Conversations



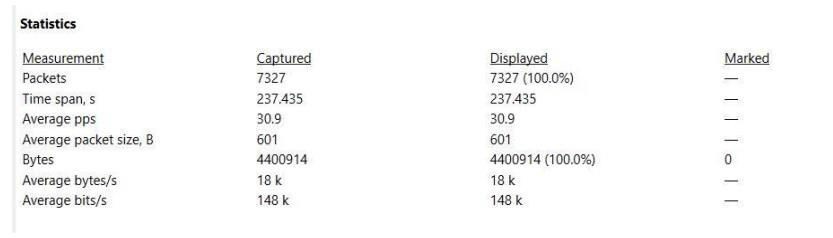
Then we can select the Ipv4 tab and see the count is 142 which means that there are 142 Ips communicating with Scott’s machine.



**What is the average packets per second rate seen in this trace file?**

Answer: 30.9

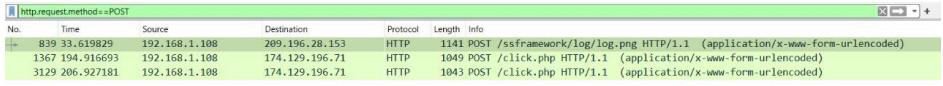
We can check out the packets rate per second from the summary options. Select > Statistics> Capture File Properties And as shown below, we can see the average packets rate per second



**How many HTTP POST requests did Scott’s machine send?**

Answer: 3

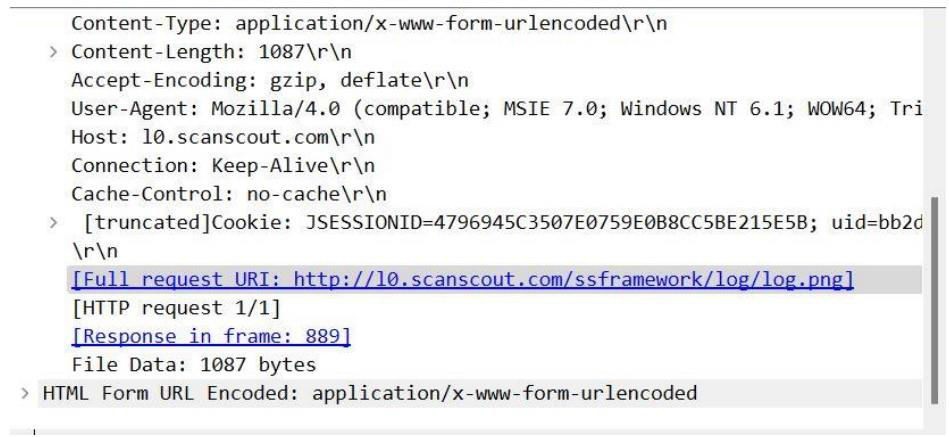
We applied a filter i.e. http.request.method == POST to filter only POST requests



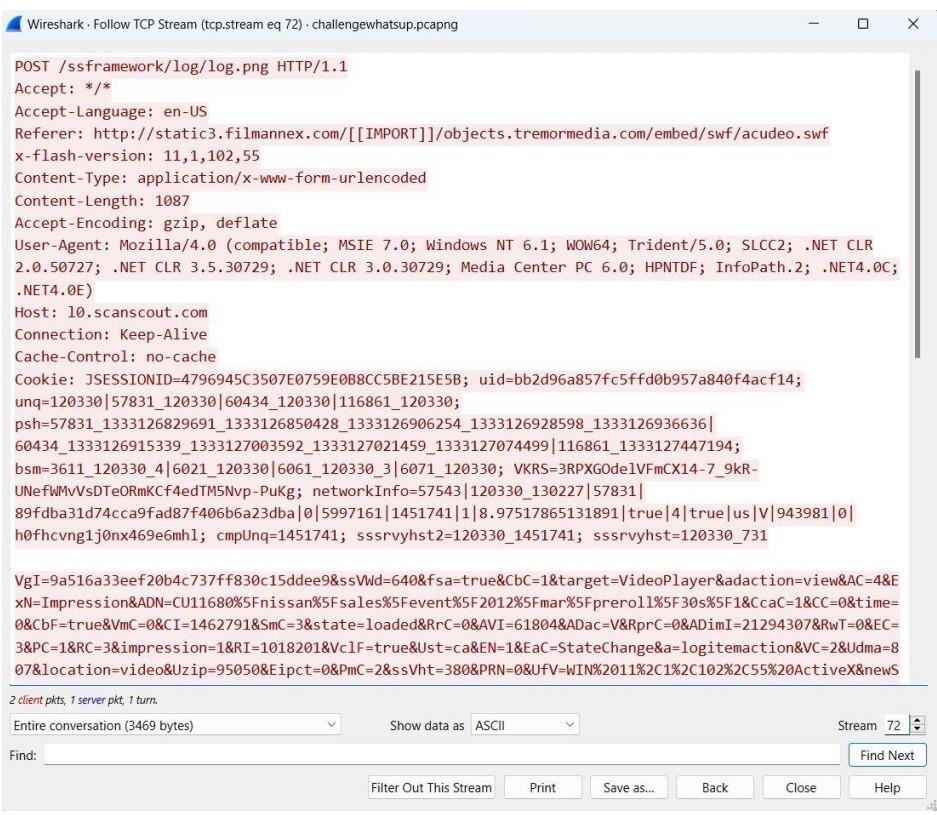
**What location information is contained in the POST toscanscout.com?**

Answer: San Francisco – Oakland – San Jose

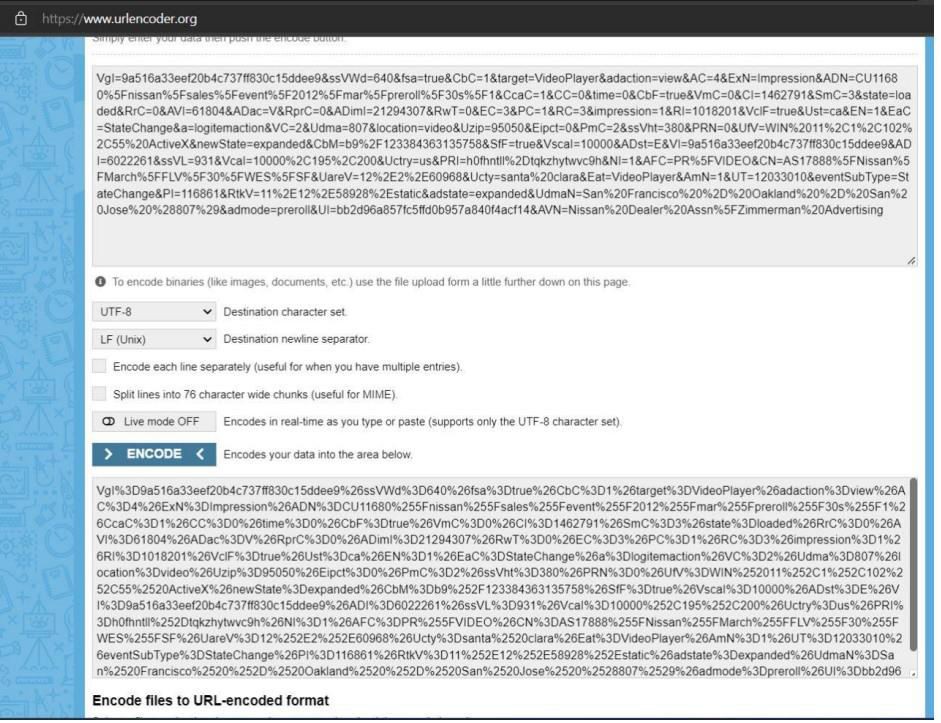
First find the POST request, and there are only three of them, and next we have to find which request header contains the scanscout.com. We checked out all three requests and found one of the request headers is for scanscout.com.



After that we need to find the location. So to find it click on the POST request and select follows TCP stream.



As can be seen below we can now see the whole POST data of that request. It contains a lot of parameters and its value and its URL are encoded, so we are not able to read them clearly We copied the whole data and used a url decoder



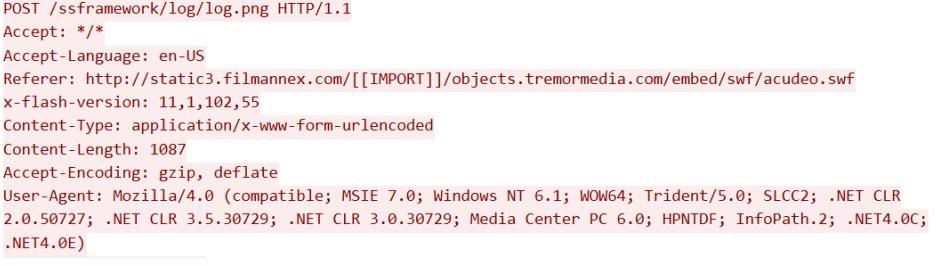
Now we can see in the below figure that the parameter Udam has its value San Francisco

– Oakland – San Jose.

**What application appears to be generating these GET/POST requests?**

Answer: Internet Explorer 9

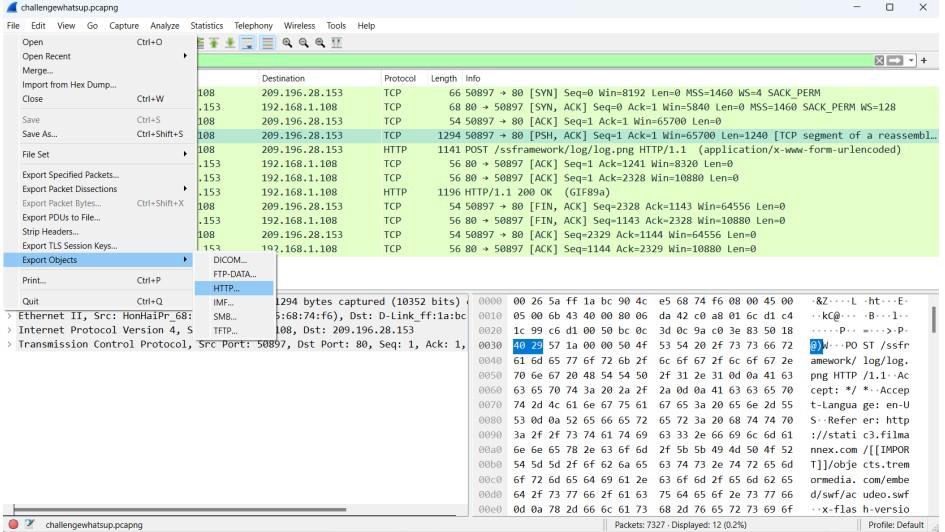
It can be done by analyzing the User-Agent string of any GET/POST request

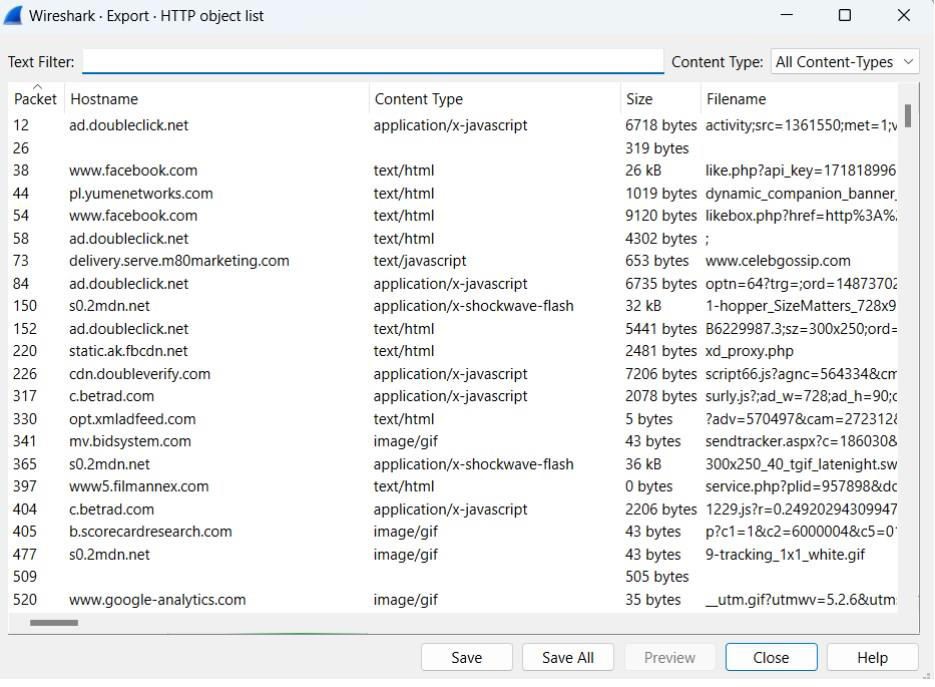


**Find,export and reassemble load\_small.png. what shape is in the image?**

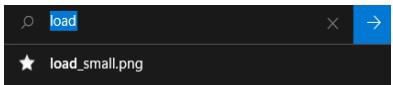
Answer: Star

We have to find a .png image in this. All packets can’t check every request manually and look for this image, so we will extract all the HTTP objects from this trace file. Go to File and select Export Objects > HTTP





we don’t know where this load\_small.png file is located in which hostname, so that’s why we will save all. We saved all files in a directory and searched for the load\_small.png file and we found that file load\_small.png.It is a star shape.



FTPS ANALYSIS TRACE FILES: challengeftp1.pcapng challengeftp2.pcapng BACKGROUND: A customer needed a secure file transfer application put in place. These two trace files illustrate the separate options they have tested – implicit FTPS and explicit FTPS.

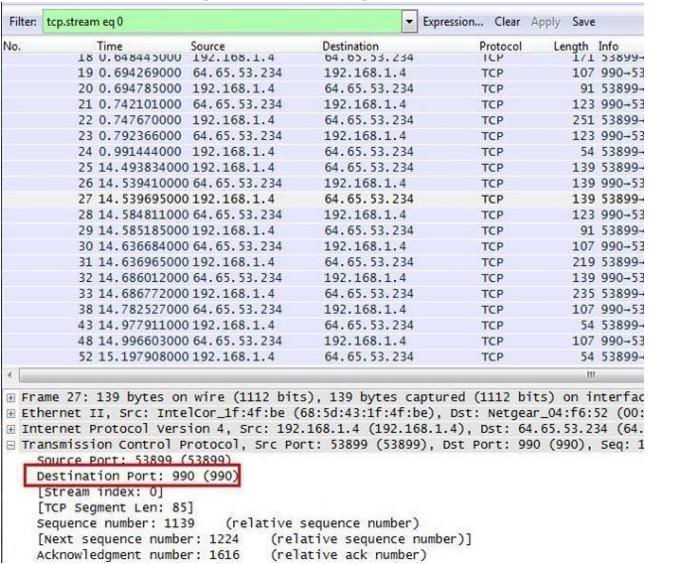
**QUESTIONS:**

**What is the IP address of the server? Answer: 64.65.53.234**



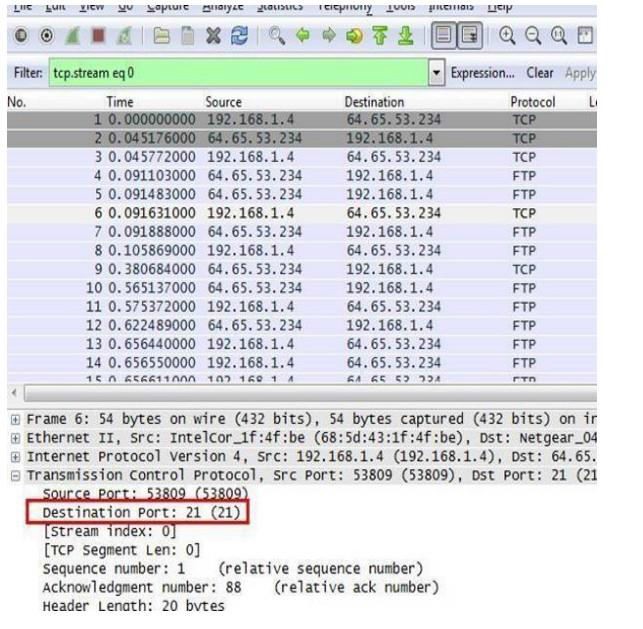
**Which trace illustrates implicit FTPS?**

Answer: challengeftp1.pcapng, implicit FTPS is a secure connection and it runs on port 990



**Which trace illustrates explicit FTPS?**

Answer: challengeftp2.pcapng (explicit FTPS runs on port 21)

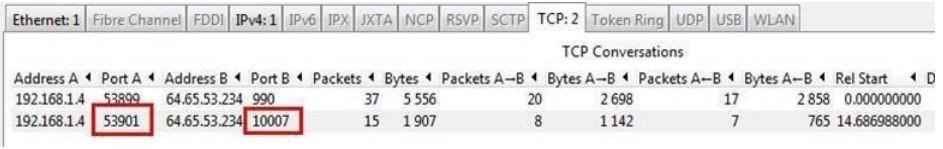


**What IP address initiated the data connections in the trace file? Answer: 192.168.1.4**

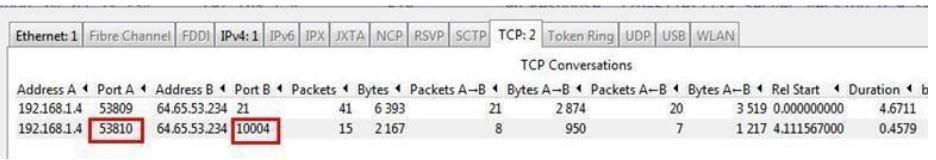


**What port numbers are used for the data connection in each trace file?**

Answer: 53901-10007



Answer: 53810 – 10004



**OUCH! TRACE FILE: challengeattack.pcapng**

BACKGROUND: These capture files were taken from a network that was experiencing a “zero- day” attack and was completely overwhelmed. It is also reported that some of the nodes within the network appear to be unable to update their antivirus/security software. The Network Administrator has given you this file that contains what he considers “suspicious” behavior and has asked you to help. The Administrator can tell you that 141.157.228.12 is a server and that 10.1.1.31 is a client machine.

**QUESTIONS:**

**What file transfer application is seen in this trace file?**

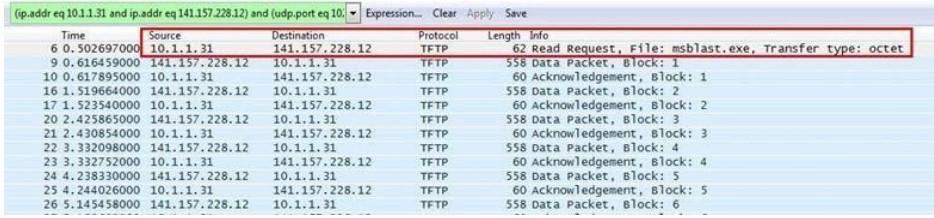
Ans: Trivial File Transfer Protocol

Go to Statistics and then select Protocol Hierarchy which will show all protocols used in this trace file.



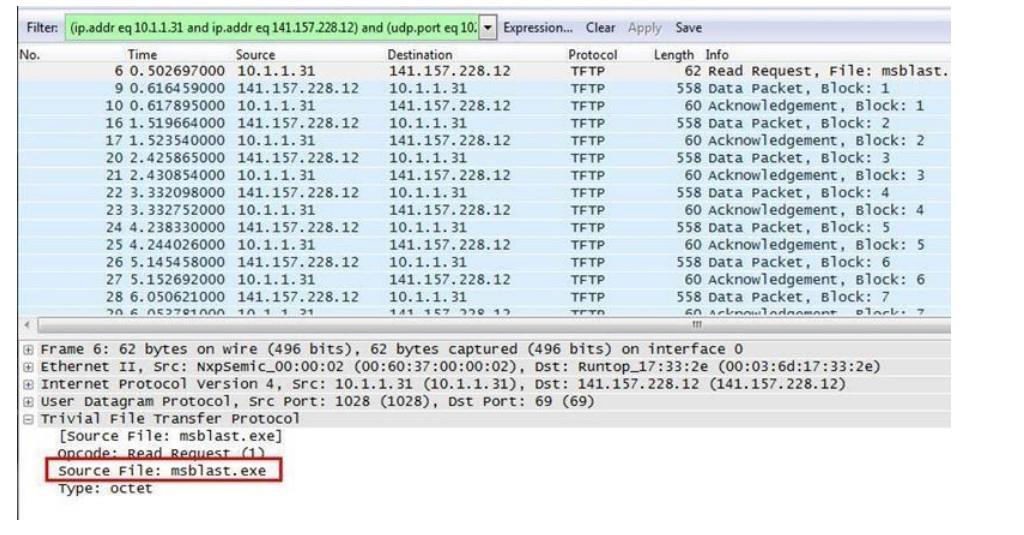
**What is the IP address of the host that is receiving the file?**

Answer: 10.1.1.31



**What is the name of the file that is being transferred?**

Answer: mblast.exe



**DNS TROUBLE TRACE FILE: challengednstrouble.pcapng**

BACKGROUND: After a maintenance window on the day before, when several servers had been upgraded to a newer operating system, a lot of trouble tickets have come in.

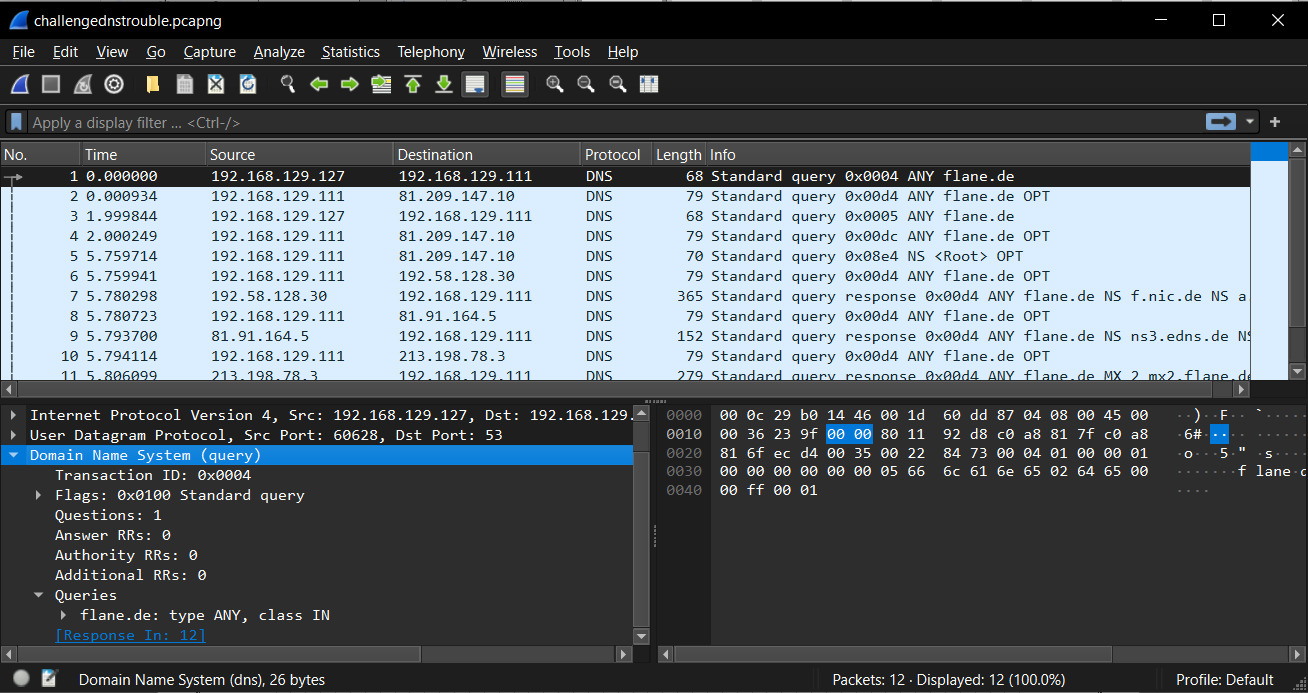
Users complain that connecting to web sites and other services takes a long time now, especially when connecting for the first time. A quick check on all relevant switches, routers and servers reveals no bottlenecks in CPU, memory or disk I/O, so of course the tickets are handed over to the network guys – it must be the network, right? Finally, one of the network engineers comes to you and asks you to help him with analyzing a trace he took. He suspects that there is something wrong with the DNS name resolution, but

even after filtering away most of the other stuff he can’t put his finger on it. Can you take a look at his trace to find out what happened and if this is a network problem at all?

**QUESTIONS:**

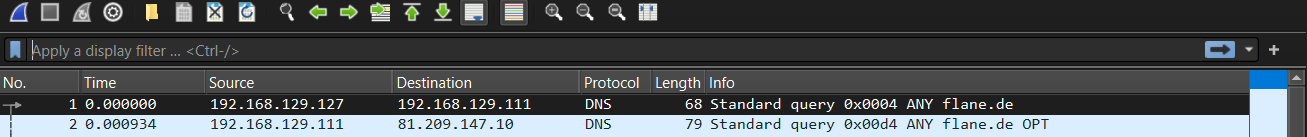
**What FQDN is the client attempting to resolve?**

Answer: flane.de



**To what IP address is the first recursive DNS query sent?**

Ans: 192.168.129.111.



**To what IP address is the second recursive DNS query sent?**

Answer: 81.209.147.10

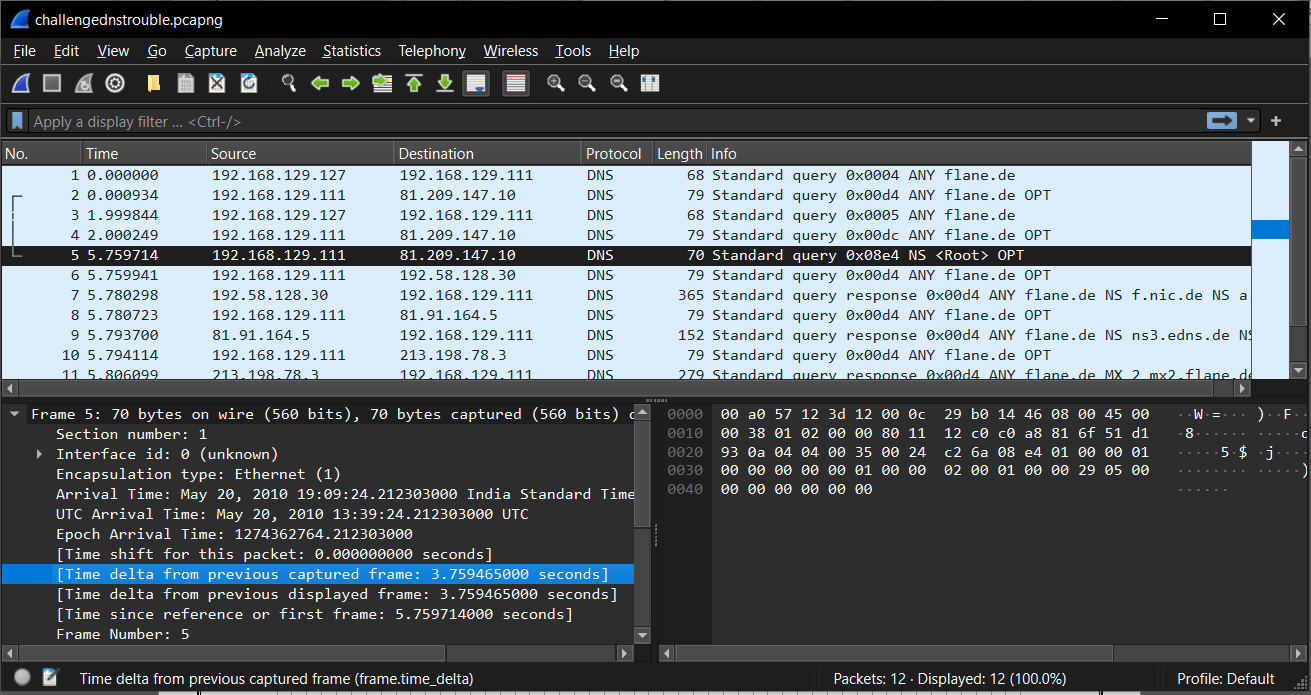


**The trace file includes authoritative DNS servers responsible for whattop level country code domain?**

Answer: .de is the country code top-level domain (ccTLD) for Federal Republic of Germany

**What is the IP address of the host that is responsible for the longdelay in resolving the host name?**

Answer: 81.209.147.10

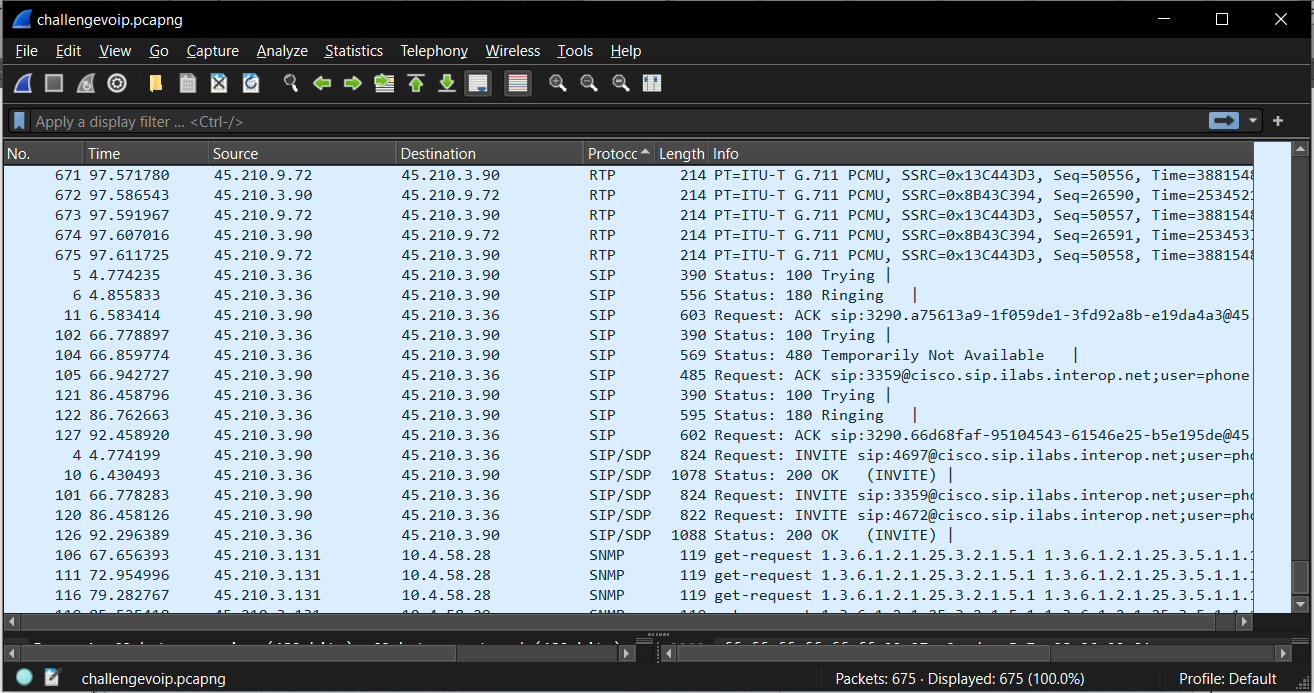


VOIP RECONSTRUCTION TRACE FILES: challengevoip.pcapng BACKGROUND: This captured file was collected from a recently installed VoIP network that is experiencing performance issues, and you have been asked to evaluate it and recommend corrective action.

**QUESTIONS:**

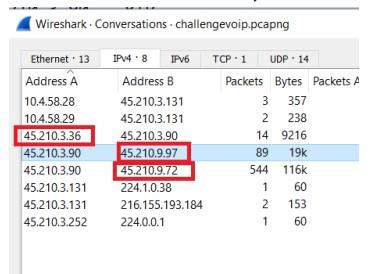
**What three UDP-based protocols are used for the VoIP call and call setup?**

Answer: SIP, SDP, RTP, SNMP



**With what three IP addresses is 45.210.3.90 communicating?**

Answer: 45.210.3.36, 45.210.3.97, 45.210.3.72



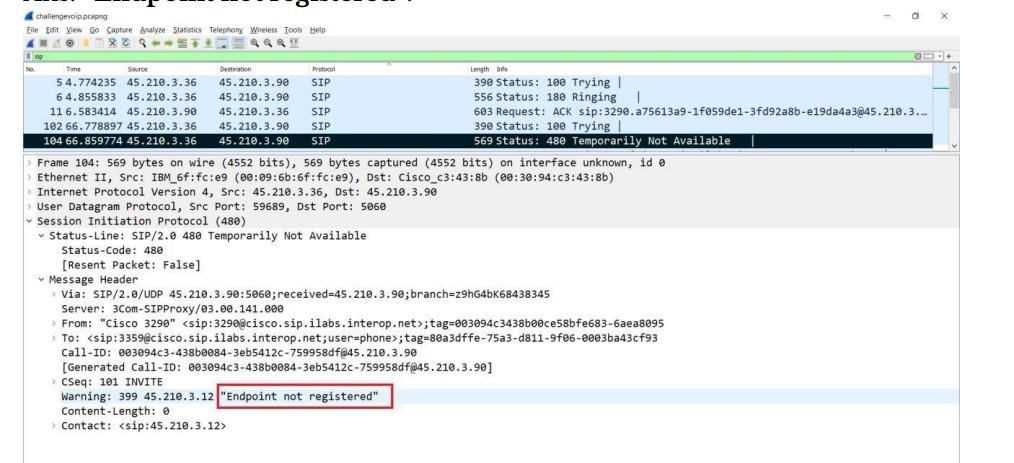
**What SIP error code is seen in this trace file?**

Answer: Error code 480



**What is the stated cause of this SIP error?**

Ans: “Endpoint not registered”.



**BOY SCOUT TRACE FILES: challengeboyscout.pcapng**

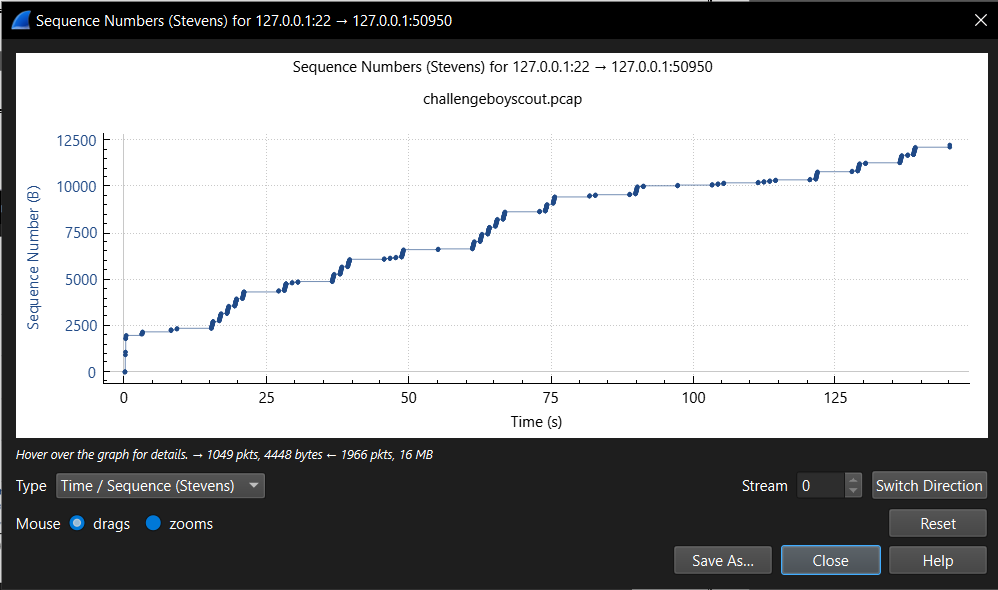
BACKGROUND: Information leaks from all sorts of place

**QUESTIONS:**

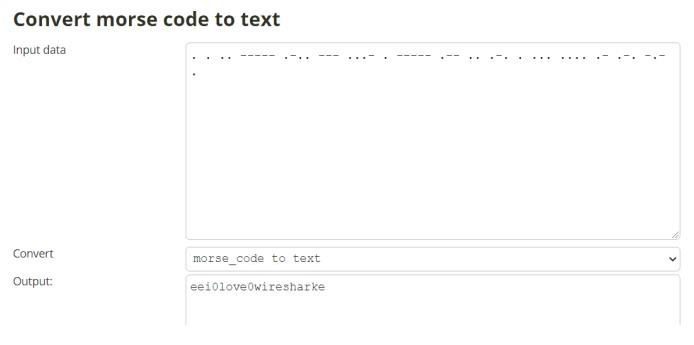
**1. What is the secret message?**

Answer: eei0love0wiresharke

Analysis:Select Statistics > TCP StreamGraph > TCP Sequence Graph (Stevens).



For decoding the Morse code, we used an online Morse code decoder.



**Conclusion**:

Thus, from this experiment we used Wireshark to analyze the protocol used and frames of each protocol, applied filters to find out the information being send, downloaded files

that are sent as HTTP Objects to check its content, decoded secret messages and other challenges