Course code -INT 301

Project Report on -

Using Wireshark to capture the data of the application

Submitted to-

Manpreet Singh
Assistant Professor
Lovely Professional University, Jalandhar

Submitted by - Himanshu
Registration no - 11905095
Roll no - 67

1. INTRODUCTION

1.1. Objective of the Project - The objective of using Wireshark to capture data from the application based on the provided keywords is to analyze the network traffic and extract specific information such as emails (POP, IMAP, and SMTP protocols), HTTP contents, and VoIP calls. By using Wireshark, the project aims to:

- Capture and analyze network traffic: Wireshark provides a platform to capture and analyze network traffic, which enables the project to monitor and record all the data packets passing through the network.
- Extract email traffic: Wireshark can filter and extract email traffic (POP, IMAP, and SMTP protocols), which enables the project to analyze the content and metadata of the email traffic passing through the network.
- Extract HTTP contents: Wireshark can filter and extract HTTP traffic, which enables the project to analyze the content and metadata of the HTTP traffic passing through the network.
- Extract VoIP calls: Wireshark can filter and extract VoIP traffic, which enables the project to analyze the VoIP call quality and other parameters related to the VoIP call.
- Perform detailed analysis: By using Wireshark, the project aims to perform a detailed analysis of the captured data to identify patterns, anomalies, and potential issues in the network traffic, and to provide insights and recommendations to optimize the network performance.

1.2. Description of the Project - Capturing and analyzing network traffic is a critical task for ensuring the performance, security, and reliability of modern networks. The use of open-source tools such as Wireshark provides an efficient and cost-effective way to monitor network traffic and extract valuable information such as emails, HTTP contents, and VoIP calls.

The objective of this project is to use Wireshark to capture and analyze network traffic from a specific application, extract specific data, and perform a detailed analysis of the captured data. This project involves several steps, including:

- Installing Wireshark: Wireshark is an open-source packet analyzer that is widely used for capturing and analyzing network traffic. The first step in this project is to install Wireshark on a machine that can access the network environment with the application.

- Capturing network traffic: Once Wireshark is installed, the next step is to capture network traffic from the application. This can be done by configuring Wireshark to capture traffic on the network interface connected to the application. Wireshark can capture data packets in real time, or it can capture packets from a previously saved file (pcap file).

- Filtering email traffic: After capturing network traffic, the next step is to filter and extract email traffic from the captured data. This can be done by using Wireshark's display filters, which can filter packets based on specific criteria such as the protocol used (POP, IMAP, or SMTP), the source or destination IP address, or the contents of the email. Once the email traffic is extracted, it can be further analyzed to identify the content, sender, receiver, and other metadata.

- Filtering HTTP contents: Similarly, HTTP traffic can be filtered and extracted using Wireshark's display filters. This enables the project to analyze the content, metadata, and performance of HTTP traffic passing through the network. This information can be used to optimize the performance of the application and ensure the security of HTTP traffic.

- Filtering VoIP calls: Finally, VoIP traffic can be filtered and extracted using Wireshark's display filters. This enables the project to analyze the quality of VoIP calls, identify potential issues such as dropped packets or latency, and provide recommendations to optimize the quality of VoIP calls.

- Performing detailed analysis: Once the data is filtered and extracted, the next step is to perform a detailed analysis of the captured data. This involves identifying patterns, anomalies, and potential issues in the network traffic, and providing insights and recommendations to optimize the network performance. The analysis can be performed using various techniques such as statistical analysis, machine learning, or visualizations.

  The expected outcome of this project is to gain insights and recommendations to optimize network performance, improve the quality

of VoIP calls, and enhance the security of email and HTTP traffic passing through the network. The project may also identify potential issues and threats in the network traffic that require further investigation and remediation.

In addition to the technical aspects, this project also requires knowledge and expertise in networking protocols, data analysis, and the use of Wireshark. This can be acquired through training or online resources, or by working with a team of experienced network analysts.

One of the benefits of using open-source tools such as Wireshark is that it provides a cost-effective way to capture and analyze network traffic. Wireshark is freely available and can be installed on most operating systems. It also provides a wide range of features and options for filtering, analyzing, and visualizing network traffic.

Another benefit of using Wireshark is that it provides a high level of flexibility and customization. Wireshark can be configured to capture specific types of network traffic, filter packets based on specific criteria, and analyze data using various techniques. This enables the project to tailor the analysis to specific requirements and objectives.

Finally,

The use of Wireshark also enables the project to gain a deeper understanding of the network traffic and the behavior of the application. By analyzing network traffic, the project can identify potential issues or security threats that may be missed by other monitoring tools. It can also provide valuable insights into the performance of the application, including latency, throughput, and response times.

However, using Wireshark also has some limitations. One of the main challenges is the volume of data that can be captured and analyzed. Large networks can generate massive amounts of traffic, making it difficult to identify specific packets of interest. This can be mitigated by using Wireshark's filtering capabilities or by using other tools to pre-process the data before analyzing it in Wireshark.

Another limitation is the complexity of analyzing network traffic. Network traffic can be highly dynamic and may involve multiple protocols and communication patterns. This requires a deep understanding of networking protocols, data

analysis, and the application being monitored. It may also require specialized knowledge in areas such as cybersecurity or VoIP technology.

To overcome these challenges, it is essential to have a well-defined methodology for capturing and analyzing network traffic. This methodology should include clear objectives, specific criteria for filtering and analyzing data, and a well-documented process for performing the analysis. It should also involve a team of experienced network analysts with a deep understanding of networking protocols and data analysis techniques.

1.3. Scope of the Project - The scope of the project is to capture and analyze network traffic using open-source tools such as Wireshark. The project aims to extract specific data from the network traffic, including emails, HTTP contents, and VoIP calls. The project will focus on capturing traffic from an application and analyzing it to gain insights into the application's performance, security, and behavior.

The project will involve using Wireshark to capture network traffic and extract specific packets of interest. The captured packets will then be analyzed to extract relevant data such as emails, HTTP contents, and VoIP calls. The project will also involve using Wireshark's filtering capabilities to narrow down the scope of the analysis and focus on specific packets.

The scope of the project also includes developing a methodology for capturing and analyzing network traffic. This methodology will involve setting clear objectives, defining specific criteria for filtering and analyzing data, and documenting the process for performing the analysis. The methodology will also involve a team of experienced network analysts with a deep understanding of networking protocols and data analysis techniques.

The project's scope also includes identifying potential issues or security threats that may be missed by other monitoring tools. This will involve analyzing the network traffic to identify patterns or anomalies that may indicate security breaches or performance issues. The project will also involve identifying areas for optimization to improve network performance and enhance the overall user experience of the application.

2. System Description

2.1. Target System Description - The target description for the project is the application for which network traffic will be captured and analyzed using open-source tools such as Wireshark. The application can be any software or system that generates network traffic, such as a web application, a mobile app, or a VoIP system.

The target application should be in active use, generating sufficient network traffic to enable effective analysis. The application should also support one or more of the protocols that Wireshark can capture, including POP, IMAP, SMTP, HTTP, and VoIP protocols.

The target application can be either on-premise or cloud-based. If it is cloud-based, the project should ensure that the network traffic can be captured and analyzed using Wireshark. This may require additional configurations or third-party tools to capture the traffic from the cloud environment.

The target description should also include the specific objectives of the project, such as optimizing network performance, enhancing security, or improving the overall user experience of the application. The objectives should be clearly defined and aligned with the overall goals of the organization.

The target description should also include the specific data that needs to be extracted from the network traffic. For example, if the objective is to optimize network performance, the project may need to extract data such as latency, throughput, and response times. If the objective is to enhance security, the project may need to extract data such as IP addresses, port numbers, and packet content to identify potential security breaches.

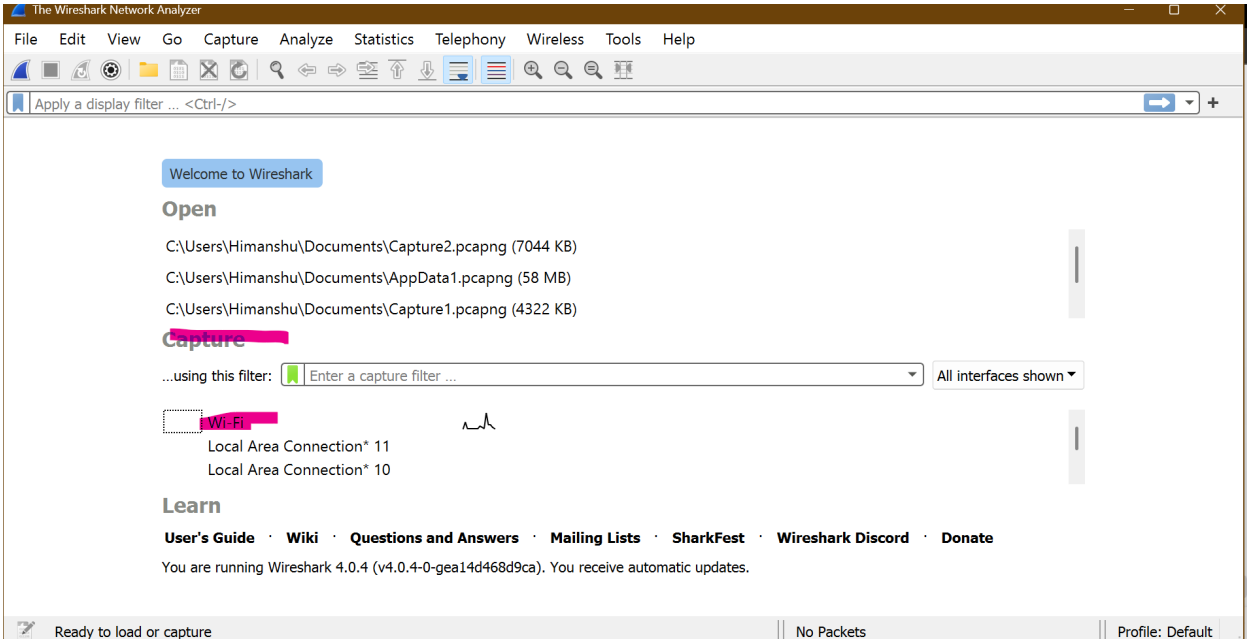
The target description should also consider any limitations or challenges that may arise during the project. This may include the volume of data that needs to be captured and analyzed, the complexity of the application or network, or any technical limitations of the tools being used.

3. Analysis Report -

3.1. Download Wireshark from https://www.wireshark.org/download.html

3.2. Install the downloaded package into your system

3.3. Now launch the Wireshark application

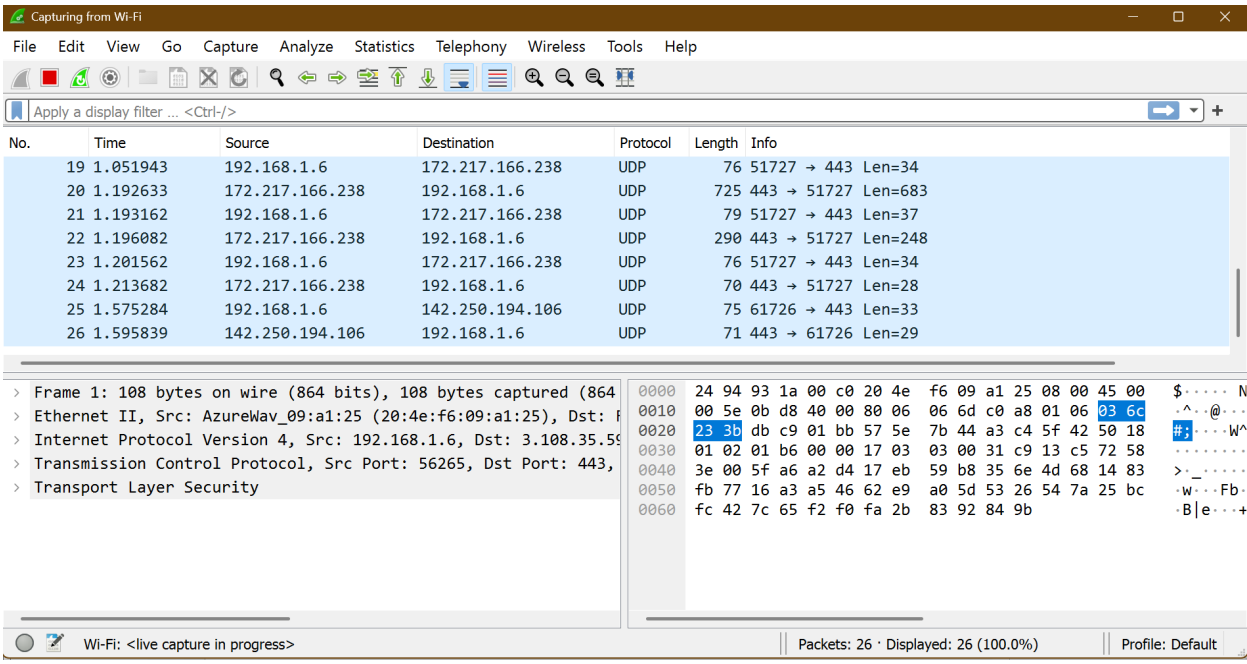


3.4.

3.5. Select the connection you want to monitor and click on Capture

3.6. Now it will start capturing packets

3.7. Firstly we will start by capturing SMTP(SMTP stands for Simple Mail Transfer Protocol, which is a communication protocol used for transmitting email messages over the internet).

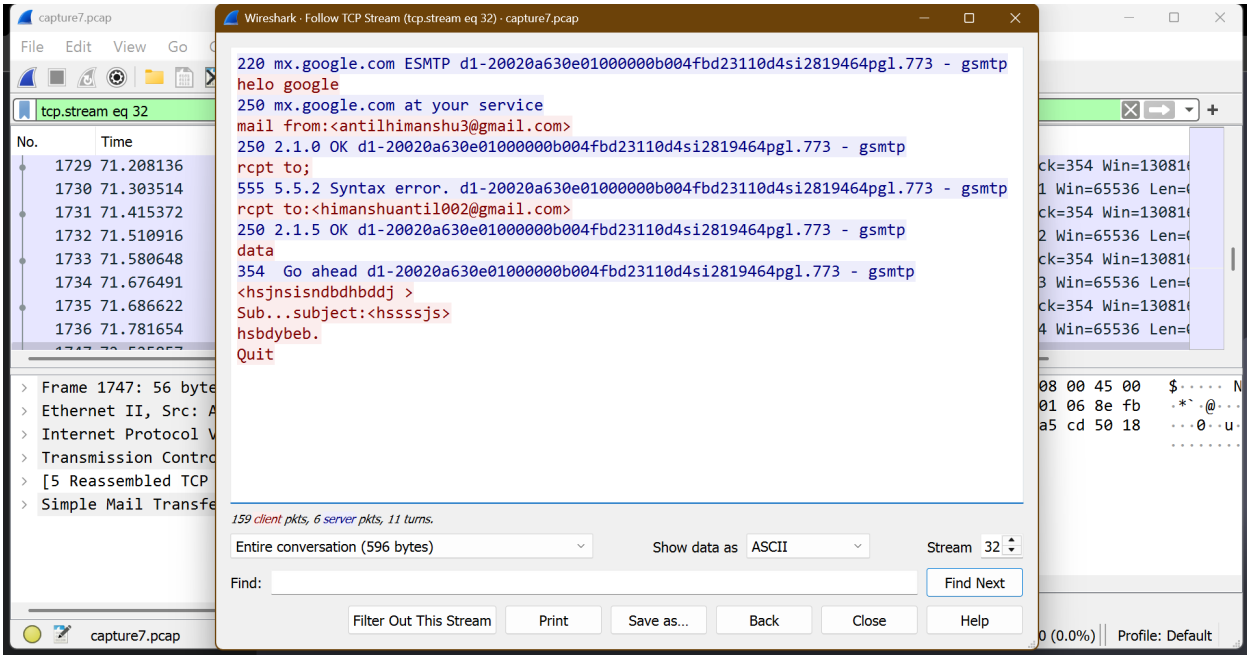


3.8.

3.9. **Now run cmd in your system and enter command - “telnet gmail-smtp-in.l.google.com 25”**

3.10. **Now enter commands to send mail**

3.11. **Now apply filter ‘SMTP’ to see captured messages**



3.12.

3.13. **This is how it will appear with captured messages.**

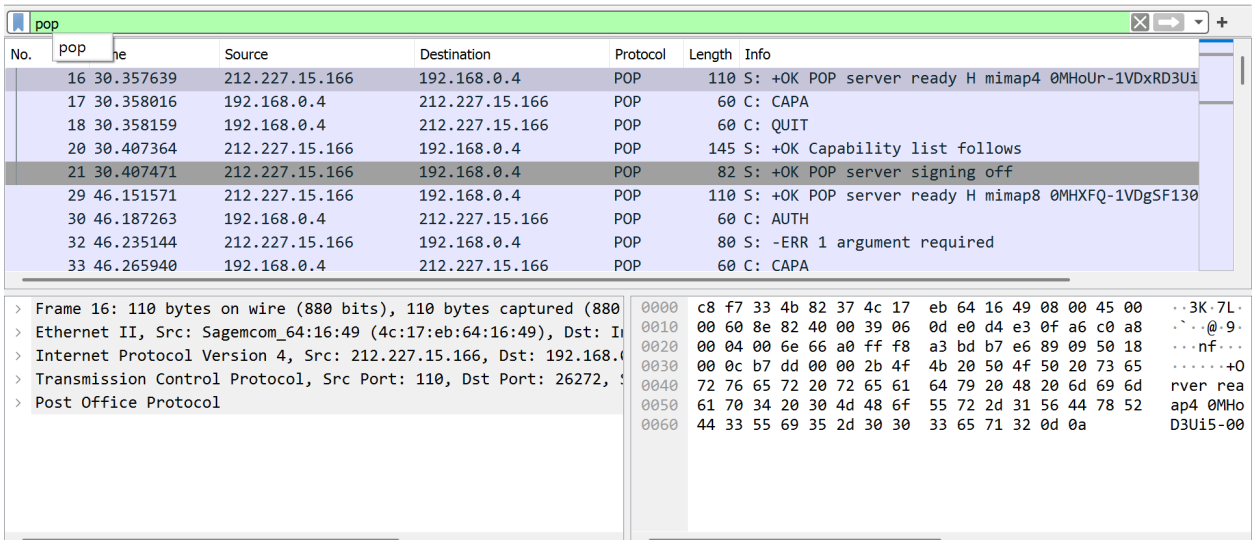
4. **Capturing POP -** Firstly lets understand what exactly POP is,

POP stands for Post Office Protocol, which is a protocol used for retrieving email messages from a mail server. POP is typically used by email clients to download emails from a server to a local device or computer.

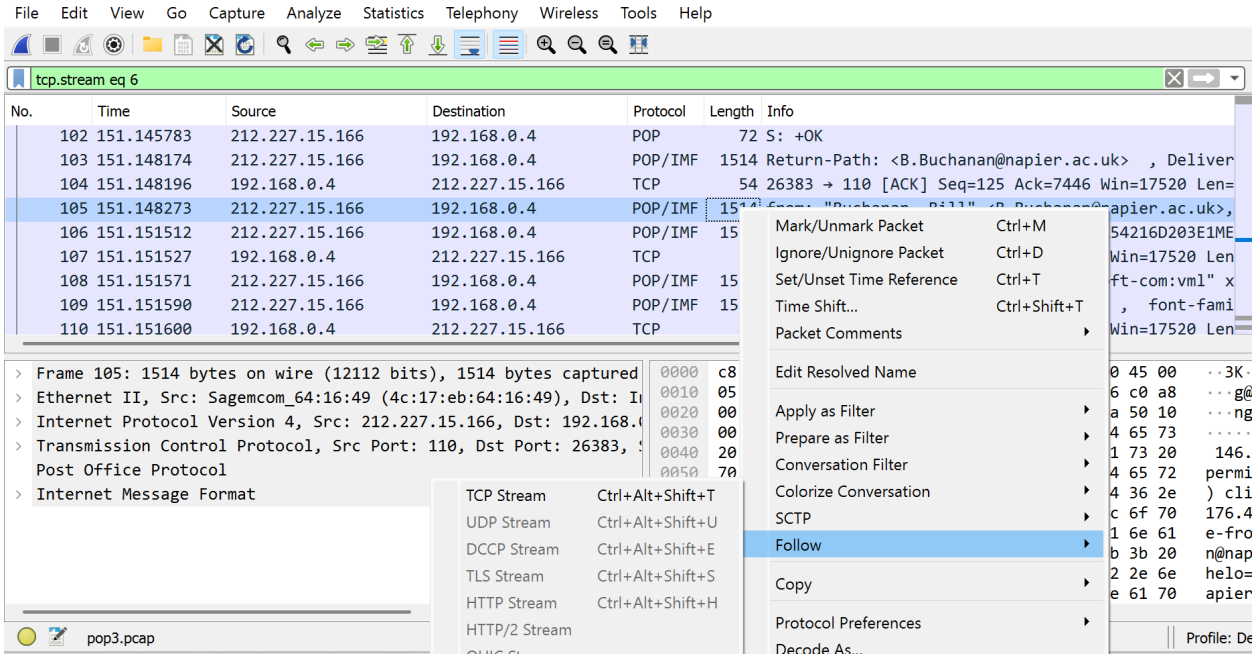
When an email client uses POP to retrieve messages, the client connects to the mail server and downloads all of the messages that have not yet been downloaded to the client. Once the messages are downloaded, they are typically removed from the server (although some email clients can be configured to leave a copy of the messages on the server). This means that if you access your email from multiple devices, you may not see all of your messages if they have already been downloaded by another device using POP.

POP operates on port 110 by default, although many email providers now offer a more secure version of POP called POP3S, which operates on port 995 and uses SSL/TLS encryption to protect the transmission of email messages between the client and the server.

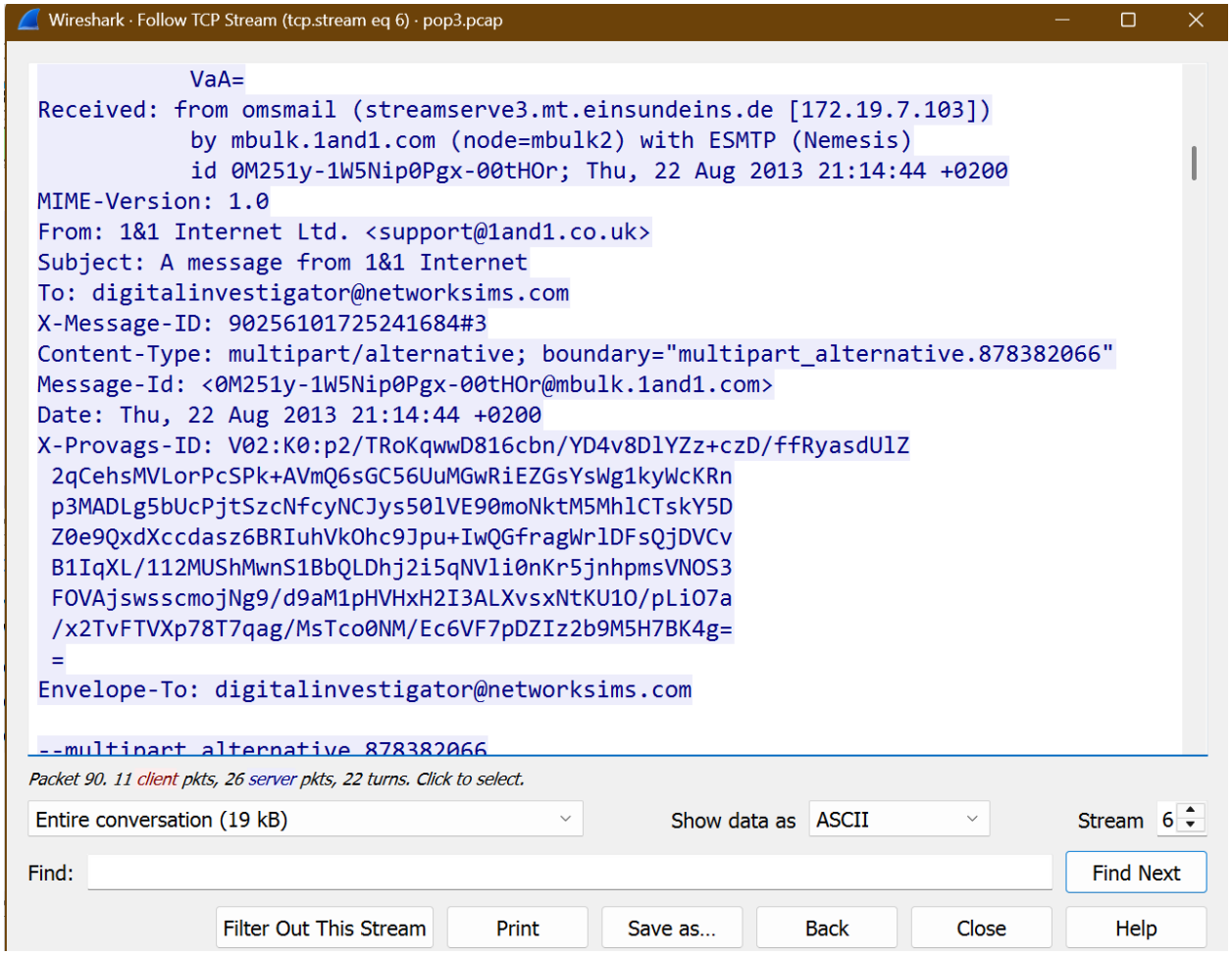
The process remains same as above, this time we will use the filter ‘POP’



After applying filter you can follow the following step for viewing the packet:



Here select ‘TCP stream’



Here you can see the details of the mail.

Using same capturing steps you can further capture 'IMAP' and any other packet you want to by using Wireshark.

IMAP - IMAP stands for Internet Message Access Protocol, which is a protocol used for accessing and managing email messages on a remote mail server. IMAP is an alternative to POP, which is another protocol used for retrieving email messages.

When an email client uses IMAP to access messages, the client connects to the mail server and downloads a list of all of the email messages that are stored on the server. The client can then view the message headers and selectively download the content of individual messages as needed. Unlike POP, IMAP keeps the messages stored on the server by default, which means that you can access your email from multiple devices and see the same messages.

IMAP also supports other advanced features such as message flags, which allow you to mark messages as read or unread, and folders, which allow you to organize your messages into different categories. IMAP also supports secure authentication mechanisms such as SSL/TLS, which helps to protect the transmission of email messages between the client and the server.

IMAP operates on port 143 by default, although many email providers now offer a more secure version of IMAP called IMAPS, which operates on port 993 and uses SSL/TLS encryption to protect the transmission of email messages.

FAQs

1. **What are different protocols in Wireshark?**

   Wireshark is a network protocol analyzer that lets you capture, inspect, and analyse network data. Wireshark may be used to investigate a wide range of protocols, including:

   - Ethernet
   - IPv4 and IPv6
   - TCP and UDP
   - DNS
   - HTTP
   - FTP
   - SMTP
   - POP and IMAP
   - SSL/TLS
   - DHCP

They are just few of protocols that Wireshark supports, there are many more in list.

2. **What is a pcap file**

A PCAP (short for "packet capture") file is a file format used by network protocol analyzers such as Wireshark to store captured network traffic. A PCAP file contains a record of all the packets that were captured during a network session, including their source and destination addresses, protocols, and other metadata.

PCAP files are used for analyzing network traffic to diagnose network problems, investigate security incidents, and troubleshoot issues with network applications. They can also be used for network forensics, where investigators use them to reconstruct the sequence of events that occurred during a security incident.

PCAP files can be captured using specialized hardware such as network taps, or they can be captured using software-based packet capture tools such as Wireshark or tcpdump. Once captured, the PCAP file can be loaded into a protocol analyzer tool such as Wireshark for analysis and visualization of the captured network traffic.

3. **What is network capturing and why is it performed**

Network capturing is the process of capturing and storing network traffic data that is transmitted between different devices on a network. This process involves capturing each packet of data that travels across the network and storing it in a file for later analysis.

Network capturing is performed for a variety of reasons, including:

Troubleshooting network issues: By capturing network traffic, network administrators can analyze the data to identify the source of problems such as network slowdowns, connection issues, or service disruptions.

Monitoring network performance: Network capturing can help administrators understand how network resources are being used, identify areas of congestion, and optimize network performance.

Network security: Network capturing can be used as a security tool to detect and investigate security incidents, such as intrusion attempts, malware infections, or data breaches.

Compliance: Network capturing can be used to ensure that networks comply with industry regulations, such as the Payment Card Industry Data Security Standard (PCI DSS), which requires capturing and storing network traffic data for security purposes.

Forensic analysis: Network capturing can be used in forensic investigations to analyze network traffic and determine the source and impact of security incidents.

Refrences/Bibliography -

Here are some references related to network capturing, packet analysis, and Wireshark tools:

- Kurose, J. F., & Ross, K. W. (2017). Computer networking: a top-down approach. Pearson.

- Lamle, T. (2017). CCNA Routing and Switching Complete Study Guide: Exam 100-105, Exam 200-105, Exam 200-125. John Wiley & Sons.

- Wireshark User Guide. (n.d.). Wireshark.org. Retrieved March 25, 2022, from https://www.wireshark.org/docs/wsug_html/

- Shimonski, R. (2015). Wireshark 101: Essential Skills for Network Analysis. Wireshark University.

- Beale, J., & Pritchett, M. (2017). Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems. No Starch Press.

- Pachghare, V. K. (2019). Wireshark for security professionals: Using Wireshark and the Metasploit Framework to secure your network. Apress.

- Chappell, L. (2017). Wireshark Network Analysis (Second Edition): The Official Wireshark Certified Network Analyst Study Guide. Laura Chappell University.