**PACKET SNIFFING USING SCAPY**

**Shanya, Shreya Ghosal and Hriday Banerjee**

**Abstract -**

*The issue of packet sniffing in a switched environment is the main emphasis of this work, while a brief examination of the impact in a non-switched context is included. ARP (Address Resolution Protocol) spoofing, which enables an attacker to eavesdrop on network traffic in a switched environment, and other methods are described in detail.* *The process of intercepting each packet as it travels over the network is called packet sniffing. It is a method where an user sniffs data that belongs to other network users. Both switched networks and non-switched networks can benefit from it. Packet sniffers may be used maliciously or as a tool for administration. Depending on the users' intentions.* *The various techniques for packet sniffing on Layer 2 switched networks are covered in this study. The various sniffing techniques will each be thoroughly discussed. The aim of the study is to demonstrate sniffing on switched networks and to comprehend how it may be avoided.*

**Keywords**- Packet Sniffing, Network Monitoring, Sniffing Tools, Wireshark, Linux, Scapy.

• Hriday Banerjee • Dept. of Computer Science, University of Engineering & Management, Jaipur, India • hriday.banerjee@uem.edu.in

**I. Introduction-**

People are becoming increasingly aware of the significance of network security as computer network technologies and internet technology evolve more swiftly. The primary problem in computing is network security since there are more and more different sorts of assaults every day. One of the most important challenges is safeguarding network and computer security. The malicious nodes lead to network issues.

Authorization, which is often done with a username and password, is where network security begins. A network administrator's rules and procedures for preventing and monitoring illegal access, system modification, abuse, and denial of a computer network and network-accessible resources make up network security. The malware is relatively easy to find with using antivirus software or an intrusion detection system (IDS). Due to the ease with which intellectual property may be obtained over the internet, network security is becoming increasingly crucial.[1]

Attempts by malicious attackers to obtain access without authorization, steal data, or harm computers, computer networks, or other computing systems are known as cyber attacks. Any site may be used to conduct a cyberattack. A data breach might even be caused by a cyberattack. The data may then be modified, sold, held for ransom, or stolen by attackers. Data backup, penetration testing, bounty training, and fixing security flaws are a few prevention strategies.

One of the most common attack is packet sniffing. The process of watching and capturing data traffic on a computer network is known as packet sniffing. Data is transmitted between computers over the internet or between any networks in the form of discrete units known as packets, which are then forwarded to their destination and reconstituted into a full message. An application called a packet sniffer runs on a network-attached device and passively captures all data link layer frames that are travelling via the network adapter of the device. It is often referred to as an Ethernet Sniffer, Network Analyzer, or Protocol Analyzer. [2]

A user can review and analyse web traffic that has been recorded using a variety of technologies. TCPDUMP, WIRESHARK, TCPFLOW, and Ethercap are the most well-known tools. This might make it difficult for network monitoring or users to recreate the visually focused web page. Therefore, we must create a software programme to facilitate users' and monitors' viewing, analysing, and decoding of their acquired network data. This can be done using scapy.[3]

With the help of the Python interpreter Scapy, we can produce, spoof, or decode packets over a network, as well as capture and examine them. We can also use it to introduce new packets into the network. It is capable of handling and manipulating wireless communication packets and supports a large variety of network protocols. Scapy can carry out the tasks carried out by several network tools, including nmap, hping, arpscan, and tshark (the command line of wireshark).

A packet sniffer can be used to gather information that can aid an attacker in breaking into a network, including passwords, IP addresses, protocols being used on the network, and other details. Wiretapping, intrusion detection, network administration, and hacking are the main uses of packet sniffing.

**Sniffing packets in non-switched environment:**

Sniffing packets is simple to achieve in a non-switched network environment. This is due to the fact that network traffic is broadcast to everyone via a hub. Every port that is linked to a hub receives traffic that is routed via it. So, to examine a computer linked to a hub, just connect a packet sniffer to an open port on the hub, which will enable seeing all traffic to and from all computers connected to that hub. It is proposed that a computer's network card be switched into a specific "promiscuous" mode in order to listen in on network traffic. Once in this mode, any application can access any network traffic (regardless of its destination) that reaches the network card (such as a packet sniffing program).

**Sniffing packets in switched environment:**

The most frequent sort of network is a switched environment. Data may be efficiently transported using switches for broadcast, unicast, and multicast traffic. Switches, however, bring a whole new degree of complexity to the task of a packet analyzer. According to their destination MAC addresses, packets in a switched network environment are only routed to the port they are intended for. In a switched environment, devices only receive packets that are intended for them, preventing promiscuous devices from sniffing extra packets. Only broadcast traffic and traffic sent and received by that computer are visible when sniffer is connected to a switch port. On a switched network, there are three main techniques for intercepting data from a target device: port mirroring, hubbing out, and ARP cache poisoning.

**II. Literature Review-**

[4] The main purpose of this research paper is to provide a secure atmosphere for our fellow classmates and peers in this internet-driven society. Unlike a standard tool that produces a large amount of data and takes a long time to filter out the desired results, the primary goal of this research is to enhance the sniffer private certificates. The primary motivation behind this project is to offer a highly recommended solution that individuals can use to simply keep track of the duties involved in monitoring their home network. Many things are uncommon because we lack network monitoring equipment, like TCP dump and wireshark. Recent study has also shown that this project may be applied successfully in the field of education in addition to being valuable for network monitoring.

[5] The objective of the research is to demonstrate sniffing on switched networks and to comprehend how it may be avoided. Computers can connect with other local computers more effectively thanks to switched networks. The way that switches are utilised now as security features was never intended. They are intended to improve the utilisation of network capacity. By restricting traffic to the desired destination, this is achieved. Sniffing is therefore more difficult on switched networks. Modern networking protocol flaws have given rise to a number of sniffer techniques. Fortunately, there are various ways to avoid problems.

[6] The numerous packet sniffing techniques, as well as the many strategies AntiSniff employs to catch these sniffing programmes, are covered in this paper along with how sniffing may be done in a hub and switched network. Computers often merely listen to their own communications when they communicate via networks. Network cards can, however, go into promiscuous mode, allowing users to listen to every network data, whether or not it is intended for them. Passwords and usernames in plain text as well as other sensitive data can be captured by packet sniffers. As a result, packet sniffers constitute a significant issue for network security. Encrypting data transfers is a good idea since sniffing is feasible on both switched and non-switched networks.

[7] The purpose of this article is to inform network administrators and specialists of the benefits of implementing Wireshark, a free and open source packet sniffer, to monitor the network. Since all packet sniffing tools function similarly to Wireshark, it was chosen for this paper's experimental goals. There are other packet-sniffing tools available. Additionally, the article provides real-world examples of frequent LAN assaults and shows how Wireshark may be used to spot them. Additionally, this article is separated into parts that illustrate several actual assaults on local networks, including port monitoring, DHCP flooding, DNS spoofing, ARP spoofing, and DDoS attacks.

[8] The topic of packet sniffing and methods for enhancing network security are covered in this article. On a switched network, sniffing is possible with third-party software. The outcome of using a variety of these tools on a switched, isolated network is shown, and it amply supports the assertion that perhaps the threat they present is substantial and genuine. The last part discusses defences against network eavesdropping in switched and non-switched contexts. The main argument of this essay is that encryption is the only real protection against the threat of sniffing.

**III. Objective-**

This project's major goal is to show that sniffing over switched networks is feasible. We'll use Scapy to see how to create and transmit packets, as well as how to examine and capture packets using packet sniffing software like tcpdump and wireshark. With the help of packet sniffer we can figure out why computer A and computer B can't interact, for instance (the communication may be impossible due to a variety of factors, such as an issue with the system or the transmission media). And also, recover the user names and passwords of network users.In addition to scanning, fundamental tasks like network tracing and port identification may also be performed, as well as packet sniffing and traffic monitoring. The scapy module will enable us to create a programme that will steal user names and passwords from unprotected websites.

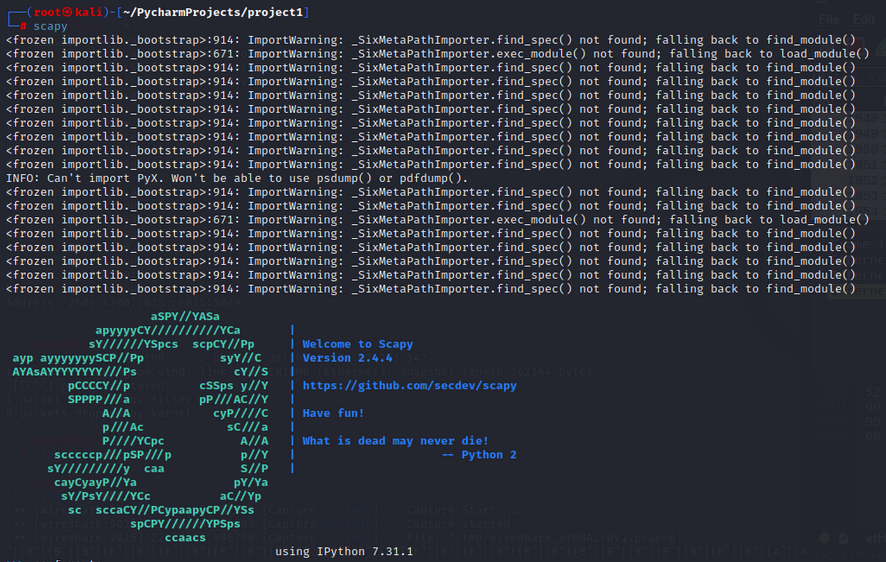
**IV. Working :-**

**Network Analysis Methods:**

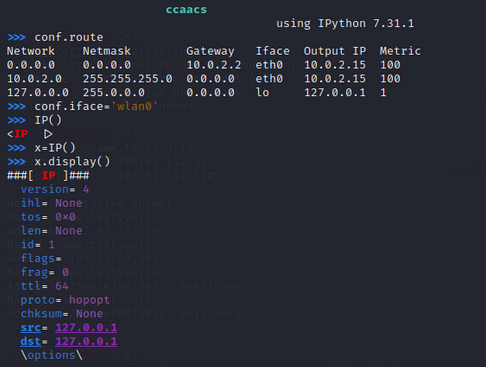
1. **Scapy:**

Scapy is a robust software for manipulating interactive packets. It can transmit packets over the wire, collect them, match requests and responses, forge or decode packets of many different protocols, and do a lot more. The majority of traditional activities, such as scanning, tracerouting, probing, unittests, assaults, or network discovery, may be handled with ease using Scapy. Hping, arpspoof, arp-sk, arping, and even some features of Nmap, tcpdump, and tshark can be replaced by it. Python's interpreter serves as the command line interface for Scapy. Sending packets and getting responses are Scapy's key tasks. We create a collection of packets, it transmits them, gets responses, matches requests with responses, and then produces a list of packet pairs (request, answer) and a list of packets that weren't matched. This has the significant benefit over tools like Nmap or hping in that the entire packet is returned as an answer rather than just a (open/closed/filtered) response. To create our own automated tools, we can use Scapy. Scapy can also be expanded without modifying its source code. Scapy may be readily integrated into our own tools. Importing what we require will suffice.

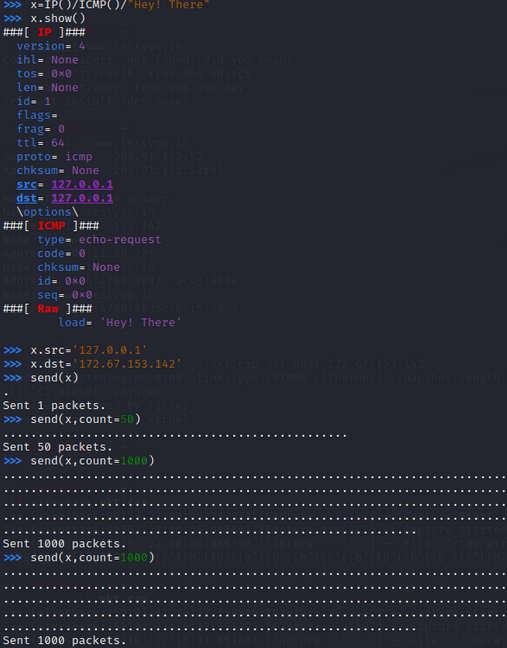
**Starting Scapy:**



**To view Default setting:**

****

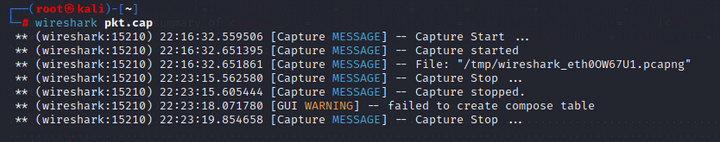
**Generating and sending packets:**

****

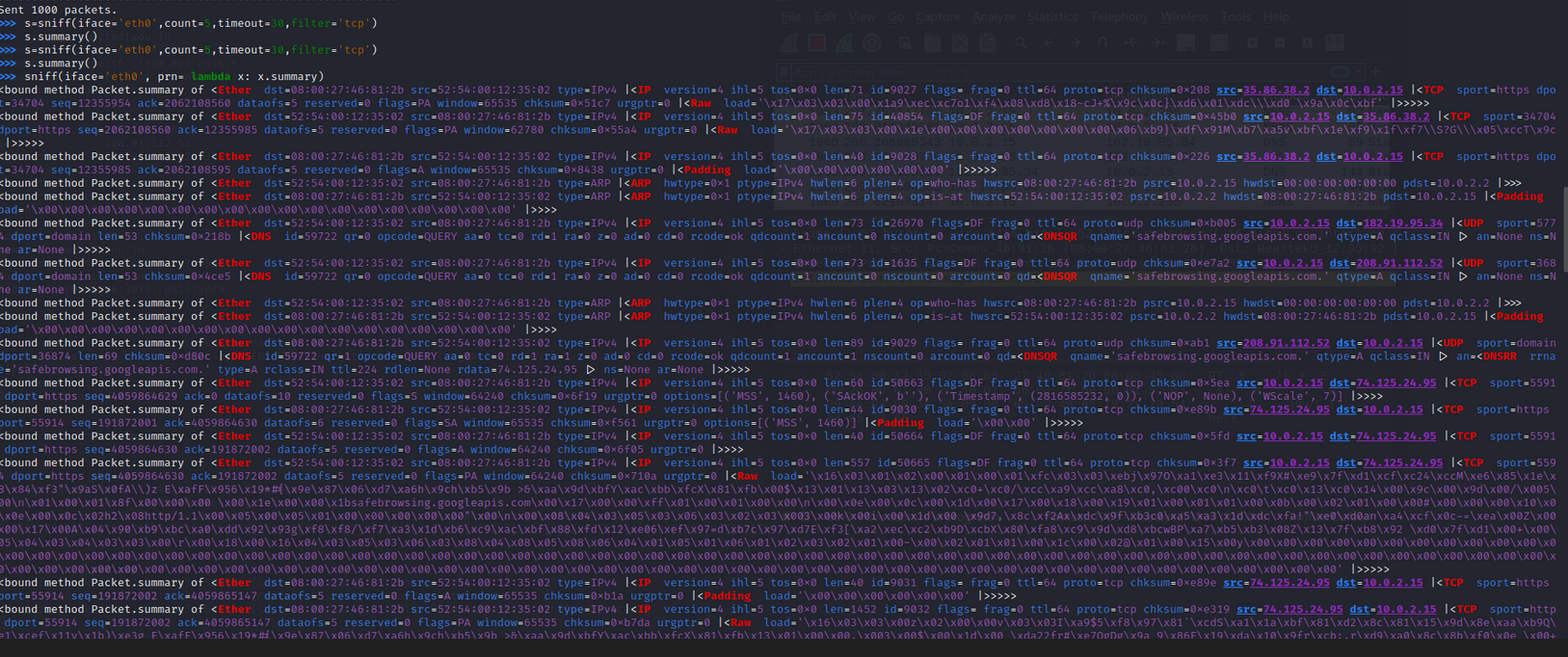
**Capturing Packet Using Tcpdump:**

****

**Viewing Data Packets in Wireshark:**

****

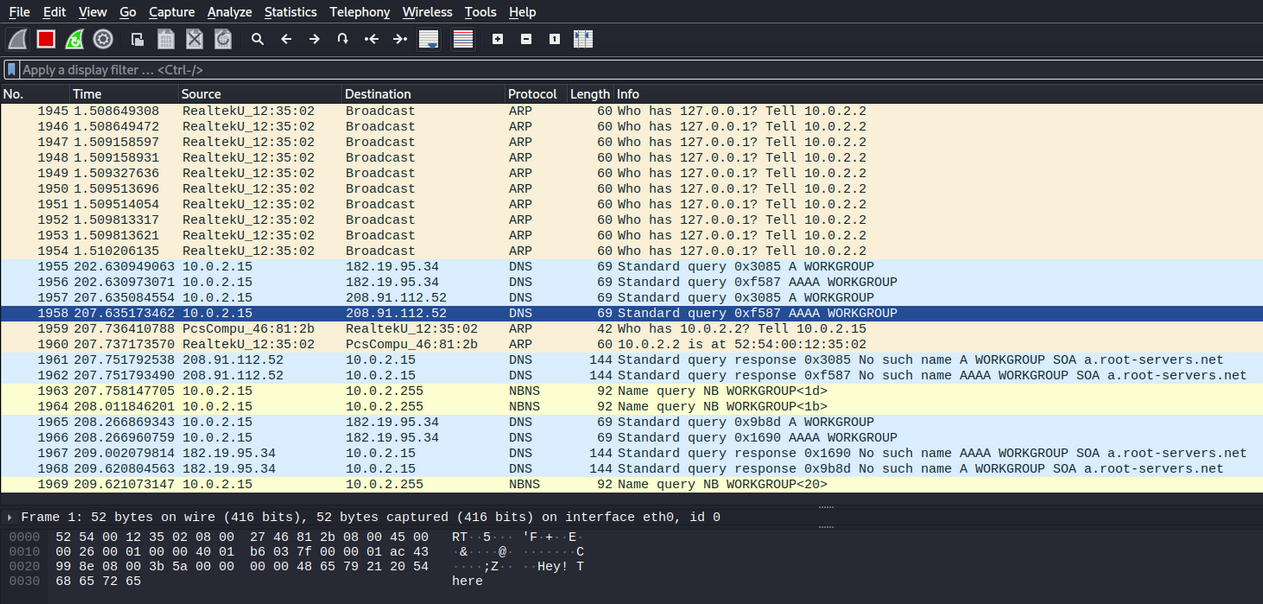
**Sniffing Packets:**



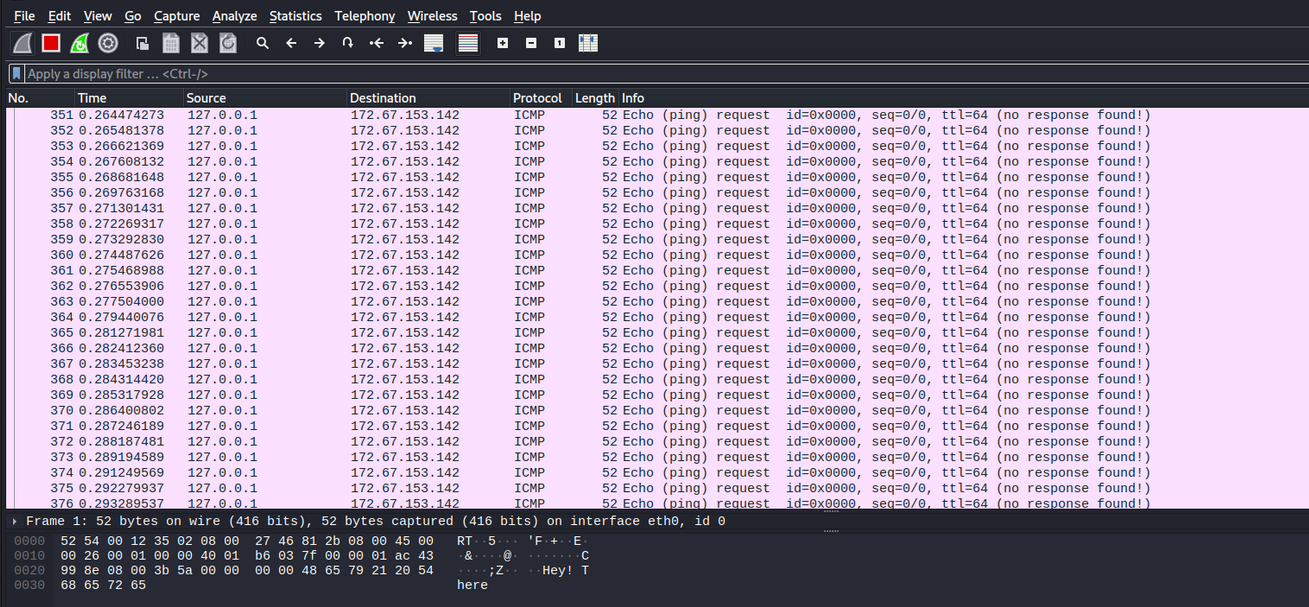
1. **Wireshark:**

Wireshark is an open source packet analyzer that is free to use. It is employed in the creation of software and communications protocols, network troubleshooting, analysis, and teaching. Simply said, Wireshark grabs data from a LAN line, from a live network connection, or from a file of previously collected packets. Wireshark is an open source programme for analysing network traffic. Various networks, including Ethernet, IEEE 802.11, PPP, and loopback, may read live data. For the traffic that was recorded, a tcpdump file may be made. Although Wireshark enables users to save files with a variety of extensions, we will utilise the.pcap extension for our needs. The most prevalent format is this one. Two filtering languages are available in Wireshark, one for capturing packets and the other for displaying them. Display filters let the administrator focus on the packets that interest them while obscuring the ones that aren't of interest right now.  Packets may be chosen based on the protocol, the existence of a field, its value, a comparison of fields, etc. The expression tab or the queries that can be typed in the field may be chosen to offer much more complex definitions and a list of all the protocols from a broad variety of protocols in Wireshark. [9]

**Viewing Packets of Different Protocols:**



Viewing Packets of Same Protocol:



1. **Tcpdump:**

It is a standard packet analyzer that operates from the command line. It enables the user to view and intercept TCP/IP and other packets that are sent across a network to which the machine is connected. Tcpdump is free software that is distributed under the BSD licence. The majority of Unix-like operating systems support tcpdump: Tcpdump utilises the libpcap library in those systems to collect packets. WinPcap, the Windows version of libpcap, is used by the port of tcpdump for Windows known as Win Dump.

**D. Ettercap:**

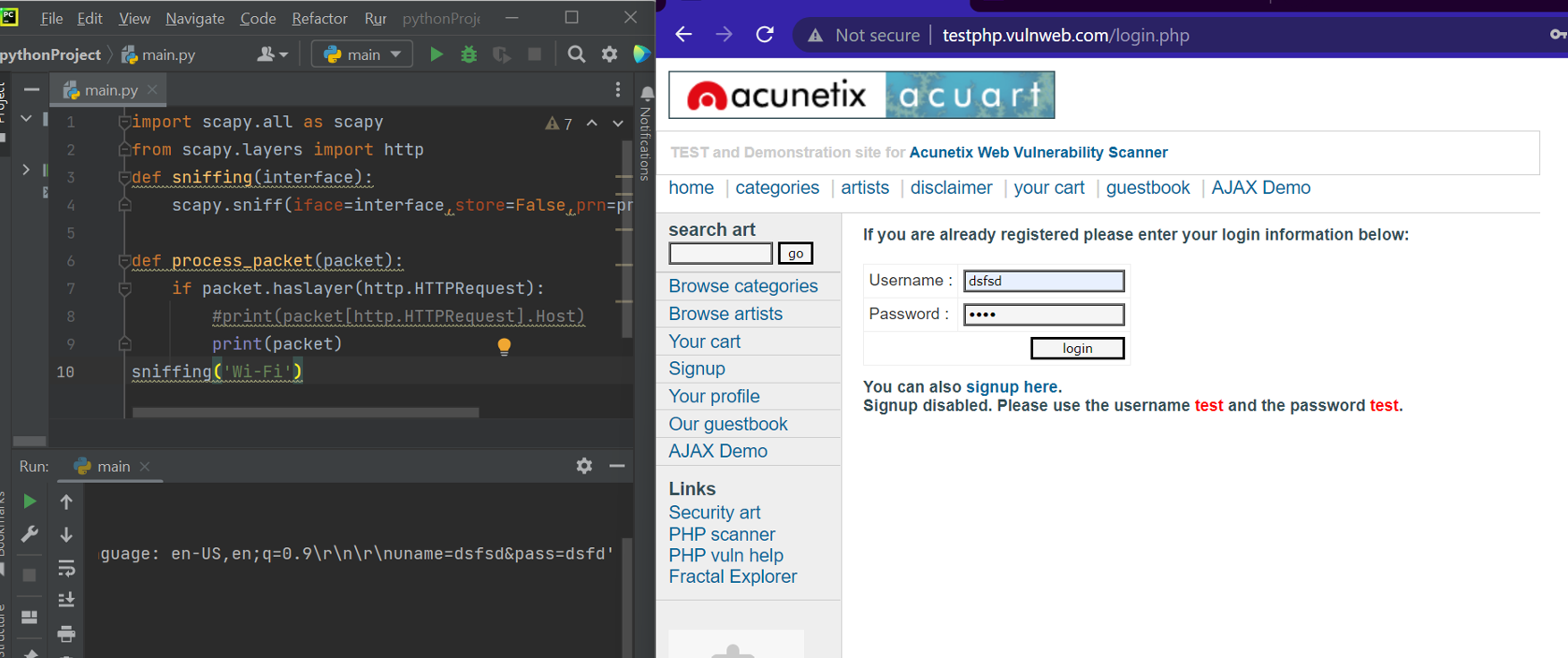
Ettercap is a flexible network manipulating tool. Many of its other features are launched from its simple man-in-the-middle (MITM) attack capability in a switched LAN context. Once ettercap has entered a switched connection, it is able to record and evaluate all communication between the two victim hosts and then utilise the following features:

Packet filtering: Automatically filter the TCP or UDP payload of packets in a live connection by searching for an arbitrary ASCII or hexadecimal string, replacing it with your own string, or simply dropping the filtered packet.

Character injection: It is the practice of introducing random characters into an active connection in either way to simulate instructions from the client or responses from the server. [10]

**V. Result And Discussion:-**

**Capturing Username and Password from Vulnerable site:**



**VI. Limitations:-**

There are various tools available for capturing network traffic, however some of them have limitations. The researcher must utilise additional tools for analysis to obtain the necessary traffic characteristics and take the memory size of the system in use into account because some tools merely record network traffic without analysis while others demand huge memory sizes for installation. The traffic that is transferred between the machine and the destination is encrypted by a VPN. Only encrypted data that is transferred to your VPN service provider would be visible to a packet sniffer. Insecure http sites are where we are collecting usernames and passwords. Unlike https sites, which encrypt data transmission to prevent hackers or others from monitoring the connection from seeing it.

**VII. Conclusion And Future Scope:-**

The packet sniffer is not merely a tool for hackers. In addition to other helpful uses, it may be used for network traffic analysis, monitoring, and troubleshooting. Recent study has also shown that this project may be applied successfully in the field of education in addition to being valuable for network monitoring. Because of their adaptability, packet sniffers may be used to: Diagnose network issues. Identify attempts at network penetration. Obtain knowledge to carry out a network intrusion. Filter network traffic for questionable content. The system intrusion will be found by the intrusion detection module. focuses on securing a network against intrusion by unauthorised users.

**References:**

[1] Mohan V. Pawar, Anuradha J. Network Security and Types of Attacks in Network. International Conference on Intelligent Computing, Communication & Convergence. (2015) 503 – 506.

[2]. S. Ansari, Rajeev S.G. and Chandrasekhar H.S, “Packet Sniffing: Brief Introduction”, IEEE Potentials, Dec 2002- Jan 2003, Volume: 21 Issue: 5, pp: 17 – 19.

[3] Dr. Aruna Varanasi, P. Swathi. Comparative Study of Packet Sniffing tools for HTTP Network Monitoring and Analyzing. IJCSET(www.ijcset.net) | December 2016 | Vol 6, Issue 12,406-409

[4] Shivam Kumar , Suryansh Jigyasu , Vikas Singh. Network Traffic Monitor and Analysis Using Packet Sniffer. International Journal of Research in Engineering and Science (IJRES). Volume:9 Issue:7 ǁ 2021 ǁ pp: 77-82.

[5] Ryan Spangler. Packet Sniffing on Layer 2 Switched Local Area Networks. Packetwatch Research. December 2003.

[6] Nimisha P. Patel, Rajan G. Patel, Dr. Dhiren R. Patel. Packet Sniffing: Network Wiretapping. IEEE International Advance Computing Conference (IACC 2009) Patiala, India, 6–7 March 2009.

[7] Jhilam Biswas, Ashutosh. An Insight in to Network Traffic Analysis using Packet Sniffer. International Journal of Computer Applications (0975 – 8887) Volume 94 – No 11, May 2014.

[8] Tom King. Packet Sniffing In a Switched Environment. GSEC Practical v1.4, Option 1 August 4th 2002, updated June/July 2006.

[9] Usha Banerjee, Ashutosh Vashishtha, Mukul Saxena. Evaluation of the Capabilities of WireShark as a tool for Intrusion Detection. International Journal of Computer Applications (0975 – 8887) Volume 6– No.7, September 2010.

[10] Duane Norton. An Ettercap Primer. GIAC Security Essentials Certification Practical Assignment Version 1.4b Option 1 April 14, 2004.