© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 13 Issue: 03 | NCASIT-2022 | April 18th 2022
Paper is available at http://www.jrps.in | Email : info@jrps.in
Refereed & Peer Reviewed

Packet Sniffer Cyber Security Tool

Vishal Injewar, Ayush Gupta, Yash Kamone, Swapnil Thawale, Shrinidhee Pande
Student, St. Vincent Pallotti College of Engineering & Technology, Nagpur

Abstract – Packet Sniffing is a way to tap each packet as it flows through the network. Packets sniffers are applications which is used to read data packets traversing network inside the Transmission Control Protocol/Internet Protocol (TCP/IP) layer and OSI layer of networking. This paper discusses about packet sniffing and packet sniffer tool, how packets are sniffed, traced and type of information that is being collected from packet

Introduction

In Computer Networking, whenever the data flows, it flows in packets. A network packet or IP packet may be designated as a data unit traveling from a source address to a destination address across the networks over the Internet. Vinton Cerf and Robert Kahn proposed the Internet communication protocols used today and the system we are referring to[1]. A protocol is a defined set of rules that determine how data is transmitted between different devices in the network. The Scapy is a powerful python module designed by Philippe Biondi in 2003[2]. It can manipulate packets of a wide number of internet protocols. Packet sniffing is the practice of monitoring and capturing all the packets passing over a computer network[3]. It is one of the many networking tasks that Scapy can accomplish. The intercepted packets provide crucial details such as which website a person visits, what material they view, what they download, and nearly everything else. Usually, the collected packets are saved for later study. There are two types of sniffing: active sniffing and passive sniffing. Flooding the switch content address memory (CAM) table with address resolution protocols is known as active sniffing (ARPs). Below is a collection of Active Sniffing Techniques.

MAC Flooding

A Media Access Control Attack, also known as MAC flooding, is a technique developed to enhance the security of network switches.

DHCP Attacks

Attacks on DHCP, or Dynamic Host Configuration Protocol, occur when an attacker attempts to respond to DHCP inquiries by listing themselves.

DNS Poisoning

DNS cache poisoning is a security flaw. They inject fake information into a DNS cache, leading DNS requests to return an incorrect response and users to be directed to the wrong websites. DNS spoofing is another term for DNS cache poisoning.

Spoofing Attacks

A Spoofing attack is a situation in which a person or program successfully identifies as another by falsifying data, to take an illegal advantage.

© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 13 Issue: 03 | NCASIT-2022 | April 18th 2022
Paper is available at http://www.jrps.in | Email : info@jrps.in
Refereed & Peer Reviewed

ARP Poisoning

ARP poisoning is an attack that allows an attacker to intercept communication between network devices.

Related Work

In the present world internet runs on the basis of two network models, the Open System Interconnection (OSI) model and the TCP / IP model, and different layers of these models[4]. These two models contain different levels of work. The OSI model has seven layers: 1. Physical layer 2. Data link layer 3. Network layer 4. Transport layer 5. Session layer 6. Presentation layer 7. Application layer. For the TCP / IP model, there are four layers. The first network access layer, the second Internet layer, the third interhost layer, and the fourth application layer.

Sniffing can be performed on the physical and network access layers of the OSI and TCP/IP models, respectively, after analysing each layer. The physical layer specifies how a data stream of raw bits is transmitted through a physical data link between network nodes. The physical layer of a network is composed of the network's electronic circuit transmission methods. It is a foundational layer that underpins a network's higher-level functionalities and can be implemented using a variety of hardware technologies with significantly differing features. So, to sniff packets attach sniffer tool to the physical layer.

Packet sniffer can differentiate packets based on their protocols. A network protocol is a set of rules that govern how data is delivered between network devices[5]. Network protocols are responsible for allowing computers to connect with one another through the internet, and hence play an important part in modern digital communications.

There are many types of protocols:

Internet Protocol (IP)

IP was designed with the aim of being used as an addressing protocol. It's most frequently associated with TCP. Packets with IP addresses are routed through multiple nodes in a network until they reach their destination. TCP/IP is the most widely used protocol for linking networks.

Transmission Control Protocol (TCP)

TCP is a very well communication protocol for sending and receiving data over a network. It breaks down any communication into packets that are transferred from source to destination before being reassembled at the destination.

User Datagram Protocol (UDP)

UDP is a loss-tolerant and low-latency communication protocol that works as a replacement for the Transmission Control Protocol.

The packet sniffer tool is developed on the basis of modules that are being used in scripted networking tools. The module that chosen for this tool is the scapy module and using the sniff function of this module.

Scapy module:

It is used by importing in Python with its own Command Line Interpreter (CLI), which allows for the creation, modification, transmission, and capture of network packets. It falsifies or decodes packets, sends them across the wire, captures them, and compares requests and responses. Scanning, tracerouting, probing, assaults,

© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 13 Issue: 03 | NCASIT-2022 | April 18th 2022
Paper is available at http://www.jrps.in | Email : info@jrps.in
Refereed & Peer Reviewed

and network discovery are all possible with it.

Sniff() Function:

The Sniff() function returns information about all the packets sniffed[6]. This function's intended purpose is to allow you to customize how the packet prints out in the console, allowing you to replace the usual packet printing display with a custom format. The 'prn' argument in Sniff allows us to pass a function that runs with each intercepted packet and also allows us to perform custom actions on sniffed packets. Sniff uses the counter parameter to limit the number of packets it captures; otherwise, sniff() listens indefinitely. Filter packets when sniffing with the filter option. One can filter depending on source, destination, IP address, port number, and protocol using the BPF (Berkeley Packet Filter) syntax. The Iface argument is used to pass an interface.

Functions for creating network scanner:

ARP (): We can use this function to create ARP packets (requests or responses). By default, it generates an ARP packet when called.

Summary (): It provides us the information of the packet like the type of packet and destination by creating an object of ARP class.

Show (): It's similar to summary (), but it provides more information about the packet by using it as an ARP class object.

Is(): It's used to see what fields we have access to for a given packet. For instance, an ARP packet is used to determine which fields are available for a specific packet.

Advantages of scapy over other networking modules:

1. Other network modules, such as Wireshark and Nmap, were developed for specific tasks like as sniffing and network scanning. The functionalities of Scapy can be used to create new, high-level functionalities based on the needs of the user.

2. Other networking tools filter their output by interpreting the incoming packet, however scapy's outputs are fully decoded and left to the user's interpretation.

Methodology

Sniffing is done with the sniffer tool. Sniffing is the technique of tracking and recording all data packets as they go over the network. The packet sniffer tool first ask the user for network interface. The program will begin scanning for packets that are passing across the specific network interface, extracts the information and print it in command line.

About the code:

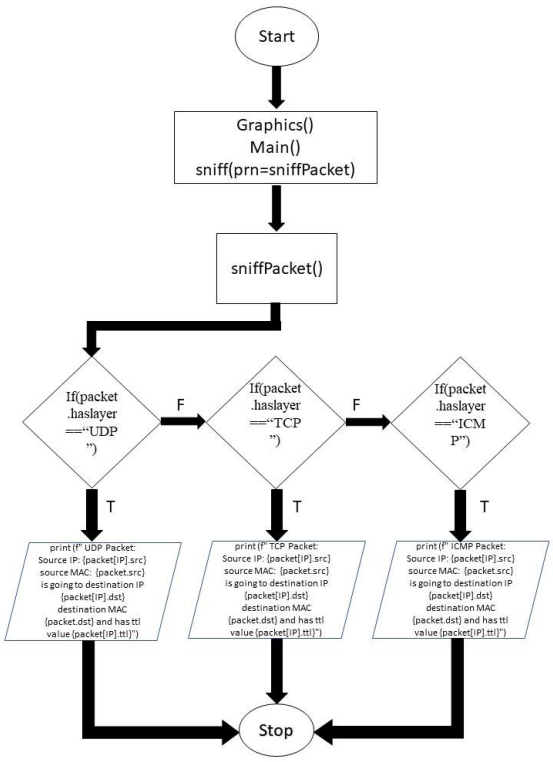
This tool is coded in Python and uses the sniff function of the scapy module to perform the scan. The graphics(), is called at the beginning of the program. In the command line interface, this function will print text graphics. The main() function is the second function being called, the program will prompt the user for the network interface. The sniff function from the scapy module will be invoked after the user enters the interface. The interface is passed by iface parameter, and ‘prn' argument passes a function sniffPacket() to execute with each sniffed packet.

© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 13 Issue: 03 | NCASIT-2022 | April 18th 2022
Paper is available at http://www.jrps.in | Email : info@jrps.in
Refereed & Peer Reviewed

These tools are built around the sniffPacket() function. This function will look for protocols, source and destination MAC addresses, as well as source and destination IP addresses in packets.

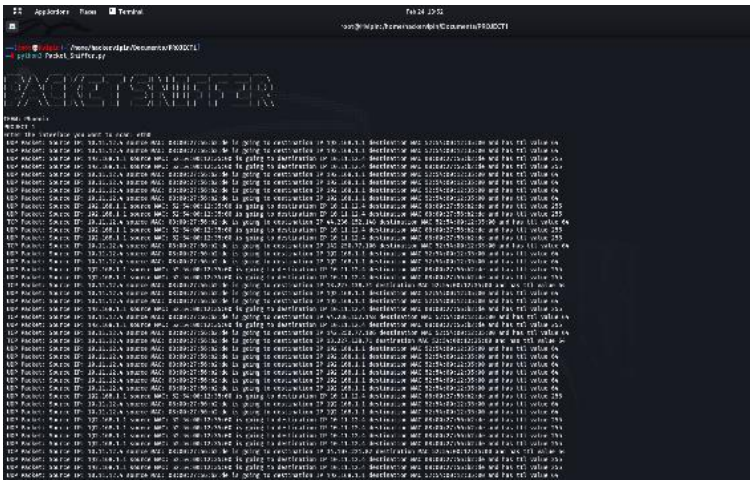
The '.haslayer' function scan the packet protocol, 'packet[IP].src' and 'packet[IP].dst' functions scan the source and destination IP addresses and 'packet.src' and 'packet.dst' funtions scan source and destination MAC addresses.

This function will print the protocol type of the packet, as well as the source and destination IP and MAC addresses of the sniffed packets.



Results and Observation

Packets sniffer extract’s information such as source and destination IP addresses and source and destination MAC addresses. The network administrator could determine whether the packet was incoming or outgoing, whether it came from a genuine source or not.



Conclusion:

Sniffing can be done at the physical layer. The source and destination IP addresses, as well as the source and destination MAC addresses, are all included in the packet. Packer sniffer can be used to hunt down suspicious activity in our network's backend.

References:

[1] Vinton G. Cerf and Robert E. Kahn, Member, IEEE “A Protocol for Packet Network Intercommunication”, Vol Com-22, No 5 May 1974.

[2] Philippe Biondi and the Scapy community, “Scapy Documentation Release 2.4.5.” Apr 12, 2022.

[3] Rupam, Atul Verma, Ankita Singh, “An Approach to Detect Packets Using Packet Sniffing”, Department of Computer

Science, Sri Ram Swroop Memorial Group of Professional Colleges Tiwari Gang Faizabad Road, Lucknow, Uttar Pradesh, India. International Journal of Computer Science & Engineering Survey (IJCSES) Vol.4, No.3, June 2013

[4] Umeh Innocent Ikechukwu, "Network Models and Design Modelling the Design of Computer Networks for Effective Management", Proc. of The Second Intl. Conf. On Advances In Computing, Control And Networking - ACCN 2015.

[5] Anders Berglund, "On the Understanding of Computer Network Protocols", March 2002.

[6] Scapy 2.4.5. documentation, link: https://scapy.readthedocs.io/en/latest/usage.html