© July 2020| IJIRT | Volume 7 Issue 2 | ISSN: 2349-6002

Detection of Intrusion Using Packet Sniffer

Sameer Belbase
Department of Computing Science and Engineering, Galgotias University, Greater Noida, U.P., India

Abstract— **A computer program that targets packets of data transmitted over a network is generally said to be the packet sniffer. It captures the packets by setting the Network Interface Card into the unbridled mode and decodes them eventually. It converts the data flowing or information sharing in the network into a human-readable format so that people can read the traffic and understand it through which, we can detect network intrusion. Also, we can find out unsecured or malicious content passed on the network. In this project, the focus has been given on the fundamentals of packet sniffer and its operating. This project shows the event of the tool on UNIX platform and its use for Intrusion Detection. Also, this project describes strategies to notice the presence of such package on the network and to handle them expeditiously. Slight observation has been created on the operating behavior of already existing human package reminiscent of Wireshark, tcpdump, and psniffer, that provides a basis for the eventuality of our sniffer software. A library called libpcap has been used for capturing packets**

1. INTRODUCTION

The technique of monitoring every packet that crosses the network is known as packet sniffing. A packet sniffer is a computer software that monitors all network traffic. These do not only receive traffic sent specifically to them. Sniffers are capable of capturing all the incoming and outgoing traffic including clear-text password and usernames or other confidential content, which is one of the security threats dispensed by the sniffers. In General, it is impossible to detect those sniffing tools because they are passive in nature which means they only collect data. They can be detected, as though some can be fully passive some aren't. In this project, we have discussed the different packet sniffing methods and explained about how Anti-Sniff tries to detect these sniffing programs.

II. SNIFFING TOOLS

2.1 tcpdump*:* Tcpdump is an integral asset that permits us to sniff network packets and make some statistical analysis out of those dumps. One significant disadvantage to tcpdump is the size of the level record containing the content yield. In any case, tcpdump permits us to decisively observe all the traffic and empowers us to make statistical monitoring content

2.2 Ethereal*:* A free network protocol instrument for Windows and UNIX. It permits you to look at information from a live framework or a capture archive on disk.

2.3 sniff it*:* It is a robust packet sniffer with good filtering.

2.4Dsniff*:* Dsniff is assortment of apparatuses for network auditing and penetration testing. Filesnarf, msgsnarf, webspy ,urlsnarf, mailsnarf, urlsnarf and inactively show a tool for charming data (passwords, email, documents, so forth) arpspoof, dnsspoof, and macof encourage the capture endeavor of network traffic is usually inaccessible to an aggressor (e.g., attributable to layer-2 exchanging). Sshmitm and webmitm actualize active monkey-in-the-center assaults against diverted SSH and HTTPS conferences by misusing frailties in the impromptu PKI.

2.6 IP spoofing: At the point when the sniffing program is on a section between two communicating end points, the interloper can impersonate one end so as to capture the association. This event is regularly joined with a Denial of Service (DoS) assault against the produced address, so they don't meddle any longer.

2.5 Hunt*:* The fundamental objective of the HUNT venture is to create apparatuses for misusing notable shortcomings in the TCP/IP protocol suite.

III. TYPES OF SNIFFING

3.1Active Sniffing:Active sniffing is sniffing in the switch. A point to point network device is known as a switch. The switch controls the progression of information between its ports by effectively checking the MAC address on each port, which causes it to pass information just to its expected target.

3.2 Passive sniffing: Passive sniffing is the process of sniffing through the hub. Any traffic that is going through the non-switched or unbridged network portion can be seen by all machines on that fragment. Sniffers work at the data link layer of the system. Any information sent over the LAN is really sent to every single machine associated with the LAN. This is called passive since sniffers set by the aggressors inactively wait for the information to be sent and catch them.

IV. WIRESHARK

Wireshark presents captured packet data in as much detail as possible and is a network protocol analyzer. It was initially known as Ethereal. Numerous individuals use Wireshark in various manners.Network directors utilize it to analyze organize issues while organize security engineers utilize it to seem at security issues .QA engineers utilize it to substantiate organize applications. Engineers employ it to look out mistakes and check the convention executions. Some people utilize it to learn network protocol internals. Wireshark can moreover be valuable in various diverse circumstances.

Wireshark uses pcap to capture packets and runs on various operating systems such as Solaris, Unix and on Microsoft Windows. Wireshark supports the promiscuous mode for the Network Interface.

V. LINUX (KALI)

Kali Linux is a Debian-based Linux appropriation that accompanies plenty of pre-introduced apparatuses to help with data security undertakings like ethical hacking. Kali Linux was created by an eminent information security organization called Offensive Security.ns. It has more than 600 infiltration testing apparatuses pre-introduced including, nmap, Wireshark, Armitage, Aircrackng, John the Ripper and so on It underpins numerous dialects and is free of cost. Kali was created in a secure environment. It can run locally when introduced on a PC's hard drive, we can boot from a live CD or live USB, or it can run inside a virtual machine.

VI. ARP CACHE POISONING

Address Resolution Protocol (ARP) cache harming in addition is utilized for sniffing. Arp cache harming depends upon neighborhood Arp cache kept up by each have of framework. This cache contains IP(Internet Protocol) with comparing MAC addresses of as recently accessed hosts in the organize.

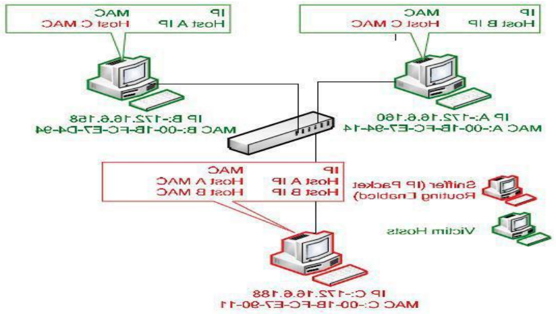


Fig. 1 : ARP Cache poisoning that would be used for the implementation.

Figure 1 clarifies Arp cache harming strategy that could be utilized for execution. inside the given chart, 'C' have performed Address Resolution Protocol cache harming assault. 'C' have sent Arp harm parcel to goal have 'A' that contains have 'C' mack address in supply mack address field and have 'B' IP address in the source IP address field of Arp harm parcel.

The native Arp cache price is poisoned either by adding false entry or change recent entry with new one once target host 'A' receives the toxic packet. The same procedure is perennial with the host 'B'. This procedure corrupts the native Arp caches of host 'A' and 'B' that are shown in Figure one. After the culmination of the poisoning methods, each host cannot communicate directly with one another.

The packet is currently sent to the person host wherever sniffer host reroutes the packet to the particular target destination . IP packet routing enabled should be enabled in person host in order that it may send packet back to actual destination once obtaining steer.

VII. SNIFFER COMPONENTS

7.1. The hardware: -Most product work from customary network adapters whereas some would possibly need especial hardware.

7.2. Capture driver:-This is that the most vital element. First, it captures the network traffic from the wire. Second, it filters for the actual traffic you would like. Lastly, it then stores the information in an exceedingly buffer.

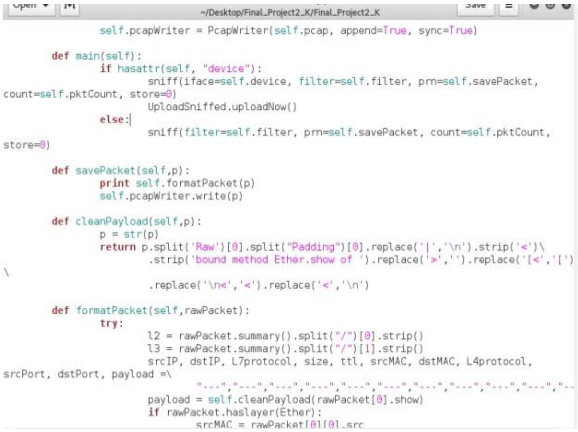
7.3. Buffer:-Buffer is employed to store the frames captured from the network.

7.4 Decoder:-Decoder displays the contents of network traffic with descriptive text in order that analysis will understand what's occurring.

7.5. Packet editing/transmission:-This is often to transmit and edit your own network packets in the network .

VIII. WORKING

The acceptance of packets within the network relies on matching of the MAC address of the Network interface card therewith of packets. The comparison is made between the MAC address and if the comparison is productive the packets is accepted or else it's discarded out. This can be done as a result of NIC as NIC only acknowledges the matched MAC address of the packets with its own. NIC behaves in 2 modes according to the need out of that non promiscuous mode implies that the NIC is in don’t care condition and it solely reads the info that is directed to that and doesn't interfere with the opposite networks. This mode is sometimes not used and thus promiscuous mode is employed, which implies that NIC is not meant and also the packets are circulated all around the network. Packet sniffers that are used for sniffing sets the NIC in the promiscuous mode of its own system because of this, the all packets are received regardless of packets being directed to a specific network. Thus for efficacious capturing and analysis of packets, NIC is ready in the promiscuous mode. The NIC of the node that is employed to receive packet is set within the Promiscuous mode that is that the basic requirement for the packet transfer from one node to another within the network. Memory is created out there known as driver memory, within which the packet which is detected by the NIC at that individual node is stored. This memory content is directly passed to buffer referred to as kernel buffer. This kernel buffer can be used for numerous applications per the packets in it. Sniffing method will be classified into 3 steps per the method as. Packet sniffer collects raw binary information from the wire which can be principally tired as the promiscuous mode. The Captured binary information is born-again into a readable type. Then the packet takes the captured network information so the verification is finished of its protocol, supported the knowledge extracted, and accordingly its analysis will start. It is then supported by protocol specific options.



```
		self.pcapWriter = PcapWriter(self.pcap, append=True, sync=True)

	def main(self):
		if hasattr(self, "device"):
			sniff(iface=self.device, filter=self.filter, prn=self.savePacket, 
count=self.pktCount, store=0)
			UploadSniffed.uploadNow()
		else:
			sniff(filter=self.filter, prn=self.savePacket, count=self.pktCount, 
store=0)

	def savePacket(self,p):
		print self.formatPacket(p)
		self.pcapWriter.write(p)

	def cleanPayload(self,p):
		p = str(p)
		return p.split('Raw')[0].split("Padding")[0].replace('|','\n').strip('<')\
			.strip('bound method Ether.show of ').replace('>','').replace('[<','[')
\

			.replace('\n<','<').replace('<','\n')

	def formatPacket(self,rawPacket):
		try:
			l2 = rawPacket.summary().split("/")[0].strip()
			l3 = rawPacket.summary().split("/")[1].strip()
			srcIP, dstIP, L7protocol, size, ttl, srcMAC, dstMAC, L4protocol, 
srcPort, dstPort, payload =\
				"---","---","---","---","---","---","---","---","---","---","--
			payload = self.cleanPayload(rawPacket[0].show)
			if rawPacket.haslayer(Ether):
				srcMAC = rawPacket[0][0].src
```

Fig. 2: Project code snippet that was used

The Project Intrusion detection using packet sniffer was made using coding on python which uses platform namely Wireshark and KALI (LINUX).

IX. CONCLUSION

There isn't an issue that can't be found and settled utilizing packet sniffer innovation. Sniffers can be utilized as the principal technique for assault on various issues that fluctuate from over-burden systems to lethargic changes to lost bundles. As various systems and hubs proceed to develop and as system speeds quicken, it turns out to be increasingly harder to screen a LAN by utilizing conventional instruments, for example, RMON (Remote Monitoring) tests. Bundle sniffers, on the other hand, screen traffic on organizing directly down to the Header data on every arrangement of information. This implies you can follow information from the beginning stage to its endpoint. Packet sniffers can likewise be utilized to distinguish the kinds of bundles on a system and find whether the particular packet has any mistakes.

REFERENCES

[1] Packet Sniffer [Online], Available: https://www.scribd.com/document/326488470/Packet-Sniffer-Project-Document

[2] Network Intrusion Detection System [Online], Available: https://www.researchgate.net/publication/232625696_Network_Traffic_Analysis_and_Intrusion_Detection_Using_Packet_Sniffer

[3] Packet Sniffing [Online], Available: http://irjes.com/Papers/vol4-issue6/G-136-138.pdf

[4] LINUX Journal [Online], Available: https://www.linuxjournal.com/article/5201

[5] Daiji Sanai, “Detection of Promiscuous Nodes Using ARP Packet”,http://www.securityfriday.com/promiscuous_detection_01.pdf

[6] Security Informatics [Online], Available http://fs.unm.edu/SecuritateaInformatica.pdf