Sniffer Detection - A Network Security Measure

Nitin Ramesh Nimran

I. INTRODUCTION

The number of internet users has increased fourfold in the recent years. Increasingly users seemed to rely more on the applications, tools and services offered by the internet. While there is one group enjoying the benefits of these services, the other group (i.e. intruders) is engaged to crash the entire communication system. Various forms of attacks such as IP spoofing, IP session high jacking, Denial of Service (DoS) and spread of malicious agents have raised the concern for developing a robust communication system. Email and Web seems to be the most widely used and exploited applications of the date. The flaws and vulnerabilities of the existing system such as spam, phishing attacks demand for secure communication. Wireless networks provide convenience and flexibility to its users. Along with convenience follows network security issues which are more complex in wireless networks.

As IP does not provide any mechanism for host authentication the naïve approach adopted by most of the application developers is to provide this feature at application layer and thus minimize the intrusion. Other approaches fall under two categories- Intrusion detection and Intrusion prevention. At an organization level, Intrusion Detection Systems (IDS) are employed to protect the private network from the potentially dangerous outside world. These systems sitting on the network edge scan all the network traffic moving from or into the organization's network and thus detect malicious attacks from outside. However, intrusion detection systems fail to detect attacks originating from the internal network. Intrusion Prevention System (IPS) employs self defense method and takes an early action on suspicious packets before its propagation to large internet population. Thus, IPS sits inline with network traffic and takes preventive measures to minimize the impact of security breach. Most of the organizations increase their level of security by concurrent operation of IPS and IDS devices. Firewall is the most common example which acts as a filter to block traffic from a suspicious attacker and allow traffic from a legitimate user.

Denial of Service (Dos) is the most dangerous forms of network attack which seems to be uncontrollable using traditional network security measures. In such type of attacks a service station is bombarded with many requests making the service unavailable to legitimate users. Usage of IDS devices geographically distributed over various locations and collaborating with each other help in monitoring Distributed Denial of Service (Dos) attacks observed on wide-spread internet. However, in case of wireless networks the issues such as unstable signal strength, mobile stations, multiple channels, radio signal interference fail the idea of employing traditional IDS.

Most widely type of attacks involves spread of malicious agents which comes in various forms such as viruses, worms, spyware, Trojan. However, these types of attacks can be controlled by using anti-malware software. IP spoofing is other form of attack which is most commonly observed in internet. In this type of attack, the attacker modifies his IP address and masquerades to be the authorized user. Thus security breach is observed if application performs host authentication based on its IP address.

Most of the service providers' use overlay multicast networks to distribute contents such as Web pages, streamed audio and video over a large internet population. These types of networks are highly vulnerable to different type of message dropping attacks. A careful study of different types of attacks and optimal sampling based techniques helps in minimizing the effect of the same. Another cost effective solution is usage of network analysis tools (sniffers). This tool is used by the network managers to take pulse of larger network by gathering details of on-going network traffic. The low-level packet information helps the manager to identify network performance and security measures.

WLANs are highly exposed to security threats such as eavesdropping. Email is the most widely used form of communication but is vulnerable to eavesdropping attacks. The most common form of eavesdropping involves usage of sniffers. A sniffer is a program that operates the network adapter in promiscuous mode and helps in capturing, analyzing network traffic. Sniffers use protocol analysis mechanism to translate protocol data in human readable format and present it to its user. Email application which uses SMTP send message in plain text format, in this case the task is much more simplified for sniffer. Usage of security mechanisms such as WEP (Wired Equivalent Privacy) use cryptographic techniques to ensure privacy. However, sniffers still provide a work around solution for the hackers. Hackers often keep sniffer program running in background in order to sniff login information and passwords. As sniffers sit passively into a network and grab all on-going transmissions; the detection of sniffers becomes difficult.

II. RELATED WORK

Most of the organizations employ layered security mechanisms to protect their internal network from intrusion attacks. The security infrastructure employed consists of Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). IDS is an defense system that watches packets traversing through the network, compares the network traffic with predefined rules and setting off an alarm if an attack is detected. IDS employ several methods to detect threats such as anomaly-based detection, signature-based detection and stateful protocol analysis. IDS can detect different types of attack such as attacks against services, unauthorized and malware attacks. As IDS sit on edge and control the traffic flow coming in and going out from organization's network, internal attacks are usually unidentified by IDS, in such cases IPS is used. IPS sits in line with network traffic and stops malicious traffic from invading the private network. IPS implements source-end defense method to block the intrusion. The intrusion detection and prevention system consists of categorizing packets into different types such as – normal packets, suspicious packets and Attack packets. Usually attack packets are blocked from entering internet and bandwidth for suspicious packets is restricted. On confirmation that suspicious packet has an attack action the source of packet is identified and edge router is informed to block further packets from the sender. IPS responds to detected threats many ways by either reconfiguring other security controls such as firewalls, deleting infected attachments from mail messages before forwarding it to users, and running patches in background to identify other security threats. As IPS and IDS occupy different positions in network, they can be used concurrently to increase the security level of an organization.

Recently Distributed Denial of Service (DDoS) attacks have demanded need for an effective security mechanism to limit the consequences of such attacks. To limit these attacks IDS devices are deployed over different geographic locations. These devices interact with each other using Intrusion Detection Message Exchange Format (IDMEF) and implement a co-operative system to find out the intruders from the widespread internet. This method saves the network bandwidth and resources. This system blocks the packets which are confirmed by a victim and thus reduces the false positive rate. This is done by making a signature for a suspicious packet by the edge router. This signature is verified by the intrusion detection system at the destination which sends a command to the edge router to block the suspicious packets. Through this signature information the attack source is located by the victim in the shortest time since it needs only one packet to locate attack source. This method is called IP based packet signature. The other two methods are cumulative sum and class based queuing. The intrusion prevention system used to detect the attack packet is composed of four subsystems edge router, firewall, intrusion detection system used at source end and intrusion detection system at the destination end. These are used to restrict the bandwidth and find the edge router of an attack source. In this procedure a host is classified as an attacker after victim confirms approximating the false positives to zero.

Once attacker is identified, IPS usually reconfigures settings in firewall. A firewall usually sits on the network edge and performs network traffic analysis. The function of firewall is to sit at the entry point of private network and provide secured access to and from the private network. Some predefined rules help the firewall to make decisions regarding incoming packet. Every incoming packet is analyzed with respect to the predefined rule. A rule specifies whether the packet is allowed to pass through or it should be discarded. Thus, all network traffic from suspicious users is blocked by firewall. The sequence of rules implemented in firewall should be complete (Every incoming packet should match with at least one rule), consistent (rules are correctly ordered) and compact (no redundant rules). A good Firewall Design Method employs different algorithms to generate, reduce and simplify firewall rules while maintaining completeness and consistency properties of original design.

Firewalls had many generations since its introduction. The earlier firewalls were mere packet filters, as they used to accept or discard packet by examining the contents of packet. This type of firewalls is called 'Stateless Firewalls'. Recently, firewalls make decision about every incoming packet not only by examining its contents but also by studying the packets it had accepted previously. Thus the firewalls maintain state information about all previously accepted packets. This type of firewalls is called 'Stateful Firewalls'. A stateful firewall helps to achieve finer access controls by tracking communication sessions between private network and outside world. A stateful firewall consists of two sections – stateful section and stateless section. For every incoming packet, the firewall first adds an additional field called 'tag' and uses the stateful section to calculate the value for this field in accordance with the current state of firewall. Later, the firewall examines the entire packet along with the tag field and compares with the rules defined in stateless section. The defined rule specifies whether the packet is to be accepted or discarded.

Overall security of a network depends on composing the measures of individual components which needs to needs to understand the interplay between components in a network. This is obtained by using attack graphs. These graphs allow composing individual measures of vulnerabilities, resources and configurations into a global measure of network security. This graph helps to get the information unknown between network components. To develop metrics for system principles like value assignment should be specific, hosts should be considered as a group of vulnerabilities are followed and significance of a resource is done on the basis of damaging caused by the compromising source, cost of reconfiguration and resistance of vulnerability. From

constructing attack graphs it can be observed that more vulnerabilities imply better security if vulnerabilities are all needed to reach a goal, number of vulnerabilities and least effort to reach a attack goal are not sufficient to find the security of a network and diversity in configuration causes either increase or decrease in the security of network. In an attack graph difficulty in execution of one exploit depends on the execution of another exploit. It shows that an imprecise argument can give misleading results making the network less secure.

In static overlay networks the multicasting functionality is done at the end hosts. Here each host participates in forwarding the messages to the other hosts. Application of multicast include sending data to a large number of subscribe by the service provider. This data also called as packets reach every node in the absence of packet losses and attacks. Here each node has a security credential authorized by the service provider to join the network. So any compromised nodes which cause the denial of service attack must be the node inside the network. This problem of attack detection in a static overlay, where nodes will be online always is done by implementing sampling schemes. There are two types of sampling schemes. Simple random sampling scheme is a sampling scheme where group controller checks every node with equal probability. Every node replies with an acknowledgement of 1 if it receives the packet and zero if it does not receive. Another scheme is the group based sampling scheme where group controller selects a subset of nodes for sampling. An attacker tries to place the compromised nodes as sparsely as possible and group controller tries to include tries to include more compromised nodes in its sampling path by keeping sampling groups as sparsely as possible and at the same time it needs sampling density. So a group controller makes a balance between these two. The latter scheme has high detection rate compared to the former. These two schemes can provide the probability of message dropping attack. Attacker identification is done based on the reasoning of the sampling results. This scheme constructs a spanning tree, resolves it according to the status of the sampled leaves and derives suspicious sub paths to detect a compromised node. It provides each node with a reputation and sets a threshold. If the nodes reputation falls below this threshold it is classified as a compromised node. Due to the presence of malicious nodes which may report false root path information there occurs two types of attacks root path falsification attacks and receiving status change attacks. To prevent this group controller verifies the authenticity of root path and checks the consistency of the reported receiving status. In dynamic environment the node is either in online state or offline state. Here Attack detection is done by dynamic group based algorithm where group controller uniformly samples member nodes and attacker identification is done by using spanning tree construction. These schemes have high detection and identification rates but low false positive rates.

Wireless monitoring is used in both wireless research and commercial WLAN management. This wireless monitoring system consists of sniffers to observe traffic characteristics on wireless medium. From this system PHY/MAC characteristics can be inferred. Wireless monitor is faced with challenges like limited capability of each sniffer, sniffer placement and data collection. The capturing capability of single sniffer is limited in terms of measurement loss and careful selection of location for the placement of sniffers. Merging multiple sniffers is used to reduce the measurement loss and it also gives a complete picture of WLAN traffic. Methods like time synchronization between multiple traces is used to correctly merge multiple sniffers and merging procedures are used to convert the timestamp of each frame captured by the sniffer to reference time. Sniffers placement method is used to locate the sniffer correctly so that it obtains acceptable capturing performance. Several techniques using MAC information are applied for security monitoring.

The widespread usage of wireless LAN has raised new concerns for security and privacy. The traditional intrusion detection systems provide reasonable security for wired networks however, for wireless networks the system collapses. Wireless networks share an open medium among the stations. Most of the antennas used in wireless networks are Omni-directional leading to problems such as signal interference and poor signal strength. These features fail the effectiveness of detecting intrusions using traditional IDS and IPS. As the stations in wireless networks are mobile, the intrusion detection sensor is unable to catch the signals with unstable strength. Thus, responding to an attack by knowing the location of attacker cannot be realized. One of the cost effective network security measure involves usage of sniffer as a network analysis tool to help identify anomalies in the network. This tool uses network interfaces of computer to capture packets from the network which are destined for other systems. The gathered information helps the network manager not only diagnose performance problems in the network but also to detect security threats. However, this tool is now widely used by hackers to sniff the ongoing communication. Sniffing is the most common type of attacks observed in wireless networks. As sniffer passively intrudes network, its detection is even more difficult than active attacks conducted by any station. It is also used to run as a patch in background and gather information about user login information, password and other secret information.

III. PROBLEM STATEMENT

A packet sniffer is a powerful network analysis tool. It plugs into the network and eavesdrops on network traffic. Sniffers were primarily by network managers to evaluate and diagnose performance problems based on the low-level information gathered by sniffing network traffic.

Features of sniffer:

Apart from its primary use a Sniffer has following features:

1. A sniffer passively intrudes into the network.
2. Sniffer use protocol analysis feature to decode binary network traffic into human readable format.
3. Sniffers can be plugged into any network. Usually they are used in networks using shared medium.
4. Sniffer has plug-in for different protocol such as Ethernet, IP, TCP, UDP, PPP, SMPTP, POP3 and many more.
5. Some sniffer products contain editing feature which allow captured packets to be edited and transmitted back to the network.

The above features make Sniffer a handy tool for hackers or intruders. A sniffer used to sniff passwords and other information is considered to be passive attack. By nature passive attacks are difficult to detect as the intruder does not give any indication of its activity. So sniffing is potentially more dangerous than active attack before it goes undetected.

Components of sniffer:

1. Capture driver - It works with network card to grab packets from the network and store in buffer.
2. Buffer - It is used to store all captured packets.
3. Decode - It is used to decode the binary network traffic with descriptive information for the analyst.
4. Editing - It is used to edit captured packets and transmit the same onto the network.

Working of sniffer:

A sniffer must be located on the same network segment of which it wants to capture network traffic. Within the network segment the sniffer can be placed anywhere.

When users transmit message using any of the network applications, the message is broken up into small units called packets. These packets make their way through the network to the segment of the destination computer. As the packet reaches destination segment, every computer on that segment observes that packet. The network interface card within each computer examines the destination address in the header field of the packet and grabs the packet if it matches with its host machine. Thus, network card performing the task of address comparison captures packets matching with the host address and discards the other packets. A sniffer program running on a system operates the network card in some other mode called Promiscuous mode. In this mode, the network adapter grabs a copy of every packet that it encounters with.

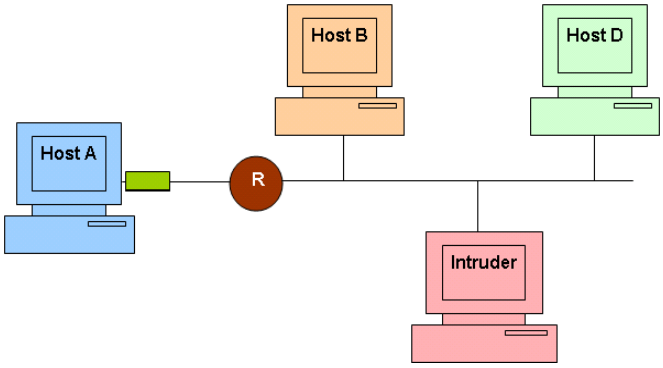

Figure: A network segment with intruder running sniffer

Thus, in the above figure it can be observed that when source machine A forwards packet destined for host D to router R, the router immediately routes the packet to destination segment. In normal mode network card in machines Host B will reject packet and Host D will pick up the packet. Now when an intruder taps into the network segment, the network card in the intruders will grabs a copy of packet destined for Host D.

Now, if a sniffer is required to capture all network traffic then for burst data huge buffer is required to store all of the packets. So to store required traffic and limit the buffer size a sniffer can be usually configured in one of the two modes

1. Unfiltered - capture all of the network packets
2. Filtered - capture packets containing specific information

Protocols vulnerable to sniffing:

1. Telnet - Sniffer can be used to capture keystrokes and thus sniff sensitive information such as user name and password.
2. HTTP - Many web sites that require user authentication transport passwords in plain-text format, which can be easily captured by sniffer.
3. NNTP, POP, FTP, IMAP, SNMP, SMTP -
   All of the above protocols sent data in plain-text format unless mechanisms like S/MIME, PGP and SSL are employed

Thus, as sniffing is a passive form of attack which usually goes undetected there is a need for security mechanisms to detect the sniffer running in the network segment.

IV. PROBLEM ANALYSIS AND DESIGN IDEA

In theory, it is impossible to detect sniffing programs because they are passive in nature. At any point the intruder does not give any indication of his activity. Sniffers only collect packets and don't transmit anything. However, in practice it is sometimes possible to detect sniffing programs.

Sniffers are available in two broad varieties:

1. Standalone: The standalone variation uses capture driver and wire-tap devices to capture network packets. It also provides 'protocol analysis' feature to decode the binary network traffic and present it in comprehensible format to the users. A stand-alone packet sniffer doesn't transmit any packets.
2. Non-standalone: The non-standalone variation runs on a system running TCP/IP stack. Thus, the overhead of protocol analysis is now taken care by the protocol stack running on the system. Sniffer when installed non-standalone on a normal computer, often generates traffic

When crackers/hackers invade machines, they often install sniffing programs. Most of the sniffers run on machines with default TCP/IP stack. Thus, when such machines are queried with normal requests then tend to respond.

V. NEW PROPSALS:

Our solution targets detection of packet sniffers executing on networked machines with default TCP/IP stack. Our sniffer detection tool consists of following cases:

Sniffer detection on a network segment:

In this case, we perform various tests to identify sniffer running on a host which is part of small network segment. The tests involve flooding invalid traffic over the network segment and identifying sniffer based on the observations.

Step I. Identifying Suspicious Host on LAN

The first step of our tool is to identify host on share network which is suspicious of running sniffer. It has been observed that the hosts operating in promiscuous mode do not have low level hardware filtering. Thus in such hosts, all the sniffed network traffic is directed to OS kernel for processing. Hence, the latency observed for such type of hosts increases considerably as the OS is busy handling sniffed network traffic.

The basic approach is as follows:

1. Start a timer and send ICMP Echo request messages to all the hosts on the shared network.
2. The request message should be destined to the machine correctly.
3. When the destination hosts responds with ICMP Echo reply turn off the timer.
4. The time difference when request was sent and response was received will give us an estimate of round-trip time for the particular hosts.
5. The above steps can be repeated number of times to determine average response time.
6. Note down the average response time for the all hosts which sent ICMP Echo reply messages.
7. Now create large amount fake network traffic by specifying invalid destination MAC address in all the frames.
8. A normal machine not operating in promiscuous mode should ignore such traffic but, a machine operating promiscuous mode sniffs all such frames.
9. After sending out fake traffic send ICMP echo messages to all the hosts once again to identify the average response time.
10. The system busy with capturing sniffed traffic does the frame filtering at OS kernel level, hence the response time observed for sending ICMP Echo reply is high.
11. The average response observed for ICMP Echo replies for all the hosts is compared with the previous history. If the difference is above the threshold limit then the host is suspected to run sniffer.
12. However, as the response time in these cases are dependent on the current network load. The hosts identified as suspicious need not necessarily be running sniffer.
13. To make our results more realistic ICMP message types Timestamp request and Timestamp reply can be used to correctly identify the processing time for systems operating in promiscuous mode.
14. Also, it can be observed that if the network is flooded with invalid traffic, the number of ICMP echo request messages dropped by a system running sniffer also increases considerably.
15. Once the suspicious host is identified we can run a series of tests as specified next step and verify our results.

Step II. Sniffer detection on identified suspicious host

In this method, the machine suspected of running sniffer is queried with invalid request messages. The message is made invalid by specifying correct IP address but invalid MAC address. If the machine passes our tests then it is confirmed that sniffer is running on that machine.

For illustration consider a network segment as shown below:

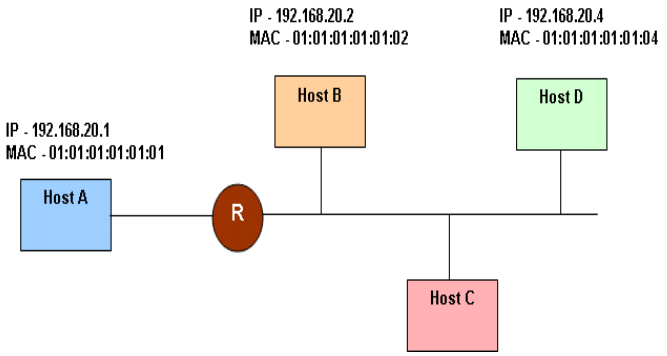

Figure: Sniffer detection on suspicious host

For example, let us assume that Host C is identified as suspicious host from our previous test and we are executing tests from Host A. We construct a unicast ARP frame on Host A specifying source IP address as 192.168.20.1, source MAC address as 01:01:01:01:01:01 and destination MAC address as some invalid address. This frame directed on to the network segment will be sniffed by Host C without comparing destination MAC in the frame with its own MAC address. Now, the sniffed ARP frame contains IP address - MAC address mapping of Host A. Host C maintains this mapping information in ARP cache for some time out period. The entries of ARP cache are flushed out after the time out period expires. Now, Host A constructs ICMP frame with source IP address as 192.168.20.1, source MAC address as some fake address let us say 01:01:01:01:01:0F, destination IP address as 192.168.20.3, destination MAC address as 01:01:01:01:01:03. The constructed ICMP Echo request message is directed on to the network segment. As the message is destined of Host C, it will capture the message and send an ICMP Echo response to indicate that it is alive. But, one thing to note in this scenario is that the captured ICMP Echo message shows fake address for Host A but as Host C had previously sniffed ARP packet it maintained the IP address - MAC address mapping of Host A. Thus, Host C directly responds with ICMP Echo response message without sending ARP packet to retrieve the MAC address of Host A. Hence, it confirms that sniffer running on Host C helped it to capture previous ARP message which was actually destined for non-existing machine on the network segment.

Sniffer detection on huge network:

In this case, we perform various tests to identify sniffer running on a host which is part of huge network. The tests involve flooding invalid traffic over the entire network and identifying sniffer based on the observations.

The protocol analysis feature of sniffer programs is flawed with the reverse DNS lookups they do for the new IP addresses they observe. Our strategy works on the principle of flooding the network with invalid packets and observing reverse DNS lookup requests encountered on the local DNS server. For illustration consider a network as shown below:

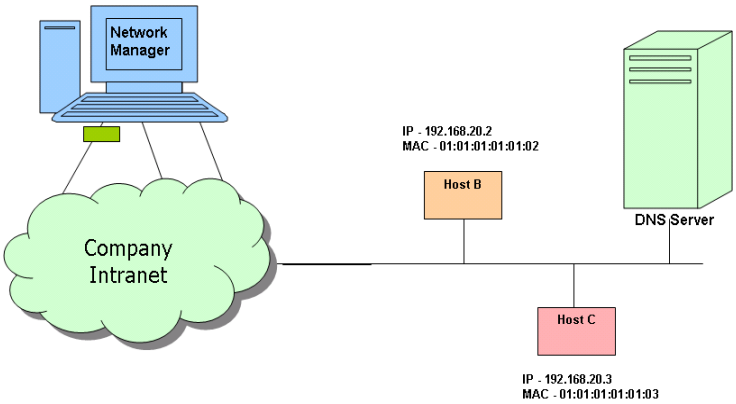

Figure: Sniffer detection on intranet

Suppose the network manager who is remotely monitoring the organization's network needs to identify sniffer running on any of the hosts within the intranet, then the executed sniffer detection tool works as follows:

1. Construct ICMP Echo request messages with invalid destination IP addresses.
2. Closely monitor all the incoming DNS queries on the local DNS servers.
3. The constructed ICMP Echo request messages (ping) are flooded across the entire network.
4. All the machines receiving the ping messages shall not reply as no one have their IP address corresponding to the destination IP address. Thus the tool pinging for machines which does not exist in the company's network is expected to receive replies indicating destination is not reachable.
5. However, a host running in promiscuous mode sniffs as usual all ICMP request messages.
6. The sniffer running on the hosts then attempts to identify domain name associated with the IP address they observe from the captured ICMP Echo request messages.
7. Accordingly, the sniffer hosts queries local DNS server asking for domain name associated with the IP address.
8. The tool already monitoring local DNS server shall catch the reverse DNS lookup for bad address asked by machines executing sniffer programs.

Sniffer detection on detector host:

The proposed sniffer detection tool shall work satisfactorily to identify networks of different sizes. The great advantage of the tool is that it can be executed on any host connected to the network and detect sniffers running on the neighboring hosts. However, it can quite be possibility that the hosts on which sniffer detection tool is executed is also running sniffer program in background. It has been observed that when hackers intrude machine then often install sniffer program which runs as a patch in background also in some cases these programs are a part of the other programs such as trojan which invade our machines through mail attachments. Unknowingly, when the program gets launched it also loads the sniffer program in background. To counter attack this possibility our tool before executing any of the tests on the neighboring hosts or on the network first runs self-detection test. In this test, the tool constructs a in-valid message loopback message which is destined for my IP address but not for my MAC address. Based on the observation of the type of reply the tool can make out if the adapter on the host is executing in promiscuous mode. Once the tool confirms the non-existence of the sniffer after performing self-detection test, the host acts as detector and can now be configured to run any of our above specified tests based on the topology of the network.

CONCLUSION:

Sniffer a powerful network analyzer tool can be handy tool for hackers/crackers if proper security mechanisms are not adopted. Sniffer programs which are freely available can be executed on any of the networked hosts for malicious purposes. Sniffing network traffic without giving any indication of the activity categorizes sniffing as passive form of attack. In theory, passive attacks are impossible to detect. The existing IDS and IPS mechanisms fail considerably to detect sniffer. The existing solutions raise encryption (using WEP, SSL and VPN) as the best form of defense against sniffers. However, a sniffer executing as a patch in background on source or destination machines shall break the strength of encryption as they can capture information before it is sent out in encrypted format. Thus, in such scenario detection of sniffer becomes primary concern for all the organizations to enhance the strength of their security system.

This paper proposes a new sniffer detection tool which is cost-effective and can be executed on any system to help identify sniffer executed on any of the networked hosts. The tool executes on the principle that sniffers are normally executed on hosts with normal TCP/IP stack so when systems are queried with normal TCP/IP requests they tend to respond. The analysis of the responses helps our tool identify sniffers executing on those hosts. The tool runs different test starting with self-detection, then detection on network segment and thereafter on the entire organization's network. The tool can be configured to run on any network of any topology and produce effective results in comprehensible format with good response time.

REFERENCES

[1] Liang Xie and Sencun Zhu on 'Message Dropping Attacks in Overlay Networks: Attack Detection and Attacker Identification'
From ACM Digital Library ISSN:1094-9224

[2] Jihwang Yeo, Moustafa Youssef, Ashok Agrawala on 'A framework for wireless LAN monitoring and its applications'
From ACM Digital Library ISBN:1-58113-925-X

[3] Bouzida, Y. Cuppens, F. Gombault, S. on 'Detecting and Reacting against Distributed Denial of Service Attacks'
From Communications, 2006 IEEE International Conference

[4] Lih-Chyau Wuu , Yen-Hung Chen , Chih-Chieh Ma and I-Tao Lung on 'A practice of the intrusion prevention system' TENCON 2007 - 2007 IEEE Region 10 Conference

[5] Gouda, M.G. Liu, A.X. on 'A model of stateful firewalls and its properties'
From Dependable Systems and Networks, 2005. DSN 2005. Proceedings. International Conference

[6] Gouda, M.G.; Liu, X.-Y.A. on 'Firewall design: consistency, completeness, and compactness'
From Distributed Computing Systems, 2004. Proceedings. 24th International Conference

[7] Zhiqi Tao Ruighaver, A.B. 'Wireless Intrusion Detection: Not as easy as traditional network intrusion detection'
From TENCON 2005 2005 IEEE Region 10

[8] Anh, N.T. Shorey, R. on 'Network sniffing tools for WLANs: merits and limitations'
From Personal Wireless Communications, 2005. ICPWC 2005. 2005 IEEE International Conference

[9] Lingyu Wang, Anoop Singhal, Sushil Jajodia on 'Toward measuring network security using attack graphs'
From Proceedings of the 2007 ACM workshop on Quality of protection