INTRUSION DETECTION USING NETWORK MAPPING TOOL

Ms. Priya P. Ravale
Lecturer
I.T.Dept
W.I.T.Solapur. Solapur University
priyaravale@rediffmail.com

Ms.Shrutali V. Narkar
Lecturer
I.T Dept
W.I.T.Solapur Solapur University
shrutalivinay@hotmail.com

Abstract–*This paper introduces a prototype network mapping tool that can be used along with intrusion detection systems to provide, in real-time, a comprehensive picture of network topology. This software tool can generate descriptions for both physical and logical connectivity of network components. It also provides positive identification of the operating systems running on the networked machines, as well as state and configuration information about the hosts and their connectivity. We present the network mapping technique which is suitable for mapping a large network & also for hacking. This scheme allows system administrators to scan large networks to see which hosts are up and what services are running. Network mapping usually outputs a list of interesting ports with each ports number and protocol.*

1. INTRODUCTION

Network mapping or **Internet mapping** is the study of the physical connectivity of the Internet. Network mapping determines the servers and the operating systems run on them of internet-connected networks. It is not to be confused with the remote discovery of which characteristics a computer may possess (operating system, open ports, listening network services, etc), an activity which is called network enumerating and is more akin to penetration testing.

Network Mapping is an open source tool for network exploration and security auditing. It was designed to rapidly scan large networks, although it works fine against single hosts. Network Mapping uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running.

While Network Mapping is commonly used for security audits, many systems and network administrators find it useful for routine tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime.

Because of the complexity of networks and attacks on networks, analysts require as much information as possible about their own network infrastructure. Knowledge of host connectivity and equipment configuration is essential when one must investigate or react to computer or network attacks. In particular, managing network security is almost impossible without knowledge of network topology. Deploying network sensors and defensive tools, such as firewalls, cannot be done effectively without knowing where these components must be positioned.

Furthermore, existing Intrusion Detection Systems (IDSs) can detect a number of attacks, but they also generate false positives. It would be therefore helpful to provide additional information that can assist an analyst in correctly determining the nature of an attack (or concluding that no such attack is in progress)

For example, in indicating the likelihood that an attack is genuine or possibly the result of a misconfiguration.One of the problems with current IDSs is that they do not provide sufficient information about the network they monitor.

They do not present the "big picture" of what is going on in a complex network.It is beneficial to provide information such as: the type of operating system on each host, what types of network services are offered, and how these computers are interconnected. As well, a dynamic map is necessary to reflect changes in the network as they occur in real time.

1. TCP/IP:

Different implementations of the Transmission Control Protocol/Internet Protocol (TCP/IP) stack may respond slightly differently to network probes. The small differences in response to these probes can indicate what specific operating system is running on a networked computer.

This technique is sometimes called “OS fingerprinting” and is used by such applications as nmap (“network mapper”), a tool designed to scan networked systems for security auditing or simply for network investigation. Along with performing OS fingerprinting, nmap also tests a number of ports to see what networked services respond.

This reveals what services are running and which ports are open or closed. Through the use of nmap, CRC’s tool can provide information about software running on the network, including operating systems and servers.

1.2 SNMP:

The Simple Network Management Protocol (SNMP) is used in numerous tools for managing network components. Information about systems, performance, etc., is stored in Management Information Bases (MIBs); these MIBs can be queried to provide vast amounts of data (depending on which MIBs are supported), such as information on routing and operating systems.

2 Motivation:

A network mapping tool is a natural complement to intrusion detection systems. Analysts who need to make rapid response decisions need as much information as possible about the network so that attack information can be placed into perspective. A major limitation of IDSs that can be alleviated by network mapping is false positives, in which the IDS claims that an attack is taking place when in fact no such attack is occurring.

Dealing with this problem requires a spatial and temporal *context* for the attack, so that an analyst can more easily determine whether such an attack is real; for example, a false attack report may actually be due to aims configuration or harmless yet unusual activity between two hosts. Network mapping provides this context, through visually presenting network topology and communications.

Furthermore, dynamic network maps can reveal violations of network security policy, such as a wireless network being added in a secure area where wireless communications are forbidden.

Network mapping is a very useful method for securely managing complex networks, particularly those that are very dynamic or heterogeneous, such as a coalition network.

3. Mapping Tool: Creating the Network Map

The tools and protocols described in Section 3 are used in conjunction with the tool's own software to create a dynamic network map. The mapping takes place in two phases: first, network and host information is gathered and an initial map is drawn; second, if a new device is found, further information is gathered about this new component, and the map is updated.

3.1 Phase One: Initial Mapping

The prototype mapping tool works in an IP network: it is given a range of IP addresses to map, and proceeds to gather network and host information within that range. It is worth noting that the tool can run on any workstation within the designated IP range: because of the use of standard tools and protocols, the mapping software can be easily deployed anywhere in a network. Once initiated, the mapping tool proceeds in five distinct steps to gather its data. These steps form the basis of the protocol tool; they are described briefly in this section and are detailed later in the paper.

3.1.1Host Discovery:

This step aims to find every active or shut-down computer that currently is, or has recently been, part of the network. To accomplish this task, several different scanning approaches are used (e.g., ICMP echo reply, ARP).

3.1.2 OS Discovery:

Once the devices that are part of the network are found, the tool tries to identify which operating system they are running, the role the device carries out (e.g., router, workstation), and the manufacturer name for networking devices.

3.1.3 Resource Discovery: This step finds out the services and shared resources offered by every computer located on the network.

3.1.4 Connectivity Discovery: Once the available resources and the services offered are identified, the tool uncovers how the hosts are connected together at the physical layer and at the network layer.

3.2 Phase Two: Updating Network Map

When a new device is found, its connectivity, state and configuration information must be added to the map. The tool repeats the last four steps, outlined in Section 3.1.1, for the new machine: that is, the operating system, resources, connectivity, and identity information is gathered for the newly-discovered host and the new map is generated in real time. The tool then returns to the listening phase, waiting to see if any new machines are added. This method ensures that the network map remains current.

4. Proposed Implementation:

The proposed implementation of the project will include following features: The project include the port scanning it starts with which port is open or close by using port scanning basics. This paper also details for the methods of finding out which operating system & application is running on that system.

4.1Port Scanning Basics:

While Network Mapping has grown in functionality over the years, it began as an efficient port scanner, and that remains its core function. The simple command nmap <target> scans more than 1660 TCP ports on the host <target>.

While many port scanners have traditionally lumped all ports into the open or closed states, Network Mapping is much more granular. It divides ports into six states: open, closed, filtered, unfiltered, open filtered, or closed filtered.

These states are not intrinsic properties of the port itself, but describe how Network Mapping sees them. For example, a Network Mapping scan from the same network as the target may show port 135/tcp as open, while a scan at the same time with the same options from across the Internet might show that port as filtered.**The six port states recognized by** Network Mapping

4.1.1 Open:-

An application is actively accepting TCP connections, UDP datagram's or SCTP associations on this port. Finding these is often the primary goal of port scanning. Security-minded people know that each open port is an avenue for attack. Attackers and pen-testers want to exploit the open ports, while administrators try to close or protect them with firewalls without thwarting legitimate users. Open ports are also interesting for non-security scans because they show services available for use on the network.

4.1.2 Closed:-

A closed port is accessible (it receives and responds to Network Mapping probe packets), but there is no application listening on it. They can be helpful in showing that a host is up on an IP address (host discovery, or ping scanning), and as part of OS detection. Because closed ports are reachable, it may be worth scanning later in case some open up. Administrators may want to consider blocking such ports with a firewall. Then they would appear in the filtered state, discussed next.

4.1.3 Filtered :-

Network Mapping cannot determine whether the port is open because packet filtering prevents its probes from reaching the port. The filtering could be from a dedicated firewall device, router rules, or host-based firewall software. These ports frustrate attackers because they provide so little information. Sometimes they respond with ICMP error messages such as type 3 code 13 (destination unreachable: communication administratively prohibited), but filters that simply drop probes without responding are far more common. This forces Network Mapping to retry several times just in case the probe was dropped due to network congestion rather than filtering.

This slows down the scan dramatically.

4.1.4 Unfiltered:-

The unfiltered state means that a port is accessible, but Network Mapping is unable to determine whether it is open or closed. Only the ACK scan, which is used to map firewall rulesets, classifies ports into this state. Scanning unfiltered ports with other scan types such as Window scan, SYN scan, or FIN scan, may help resolve whether the port is open.

4.1.5 Open filtered :-

Network Mapping places ports in this state when it is unable to determine whether a port is open or filtered. This occurs for scan types in which open ports give no response. The lack of response could also mean that a packet filter dropped the probe or any response it elicited. So Network Mapping does not know for sure whether the port is open or being filtered. The UDP, IP protocol, FIN, NULL, and Xmas scans classify ports this way.

4.1.6 Closed|filtered :-

This state is used when Nmap is unable to determine whether a port is closed or filtered. It is only used for the IP ID idle scan.

4.2 Port Scanning Techniques:

TCP SYN scanning:-

This technique is often referred to as "**half-open"scanning**".

-sS (TCP SYN scan):-

SYN scan is the default and most popular scan option for good reasons. It can be performed quickly, scanning thousands of ports per second on a fast network not hampered by restrictive firewalls. SYN scan is relatively unobtrusive and stealthy, since it never completes TCP connections. It also works against any compliant TCP stack rather than depending on idiosyncrasies of specific platforms as Network Mapping FIN/NULL/Xmas, Maim on and idle scans do. It also allows clear, reliable differentiation between the open, closed, and filtered states.

This technique is often referred to as half-open scanning, because you don't open a full TCP connection. You send a SYN packet, as if you are going to open a real connection and then wait for a response. A SYN/ACK indicates the port is listening (open), while a RST (reset) is indicative of a non-listener. If no response is received after several retransmissions, the port is marked as filtered. The port is also marked filtered if an ICMP unreachable error (type 3, code 1, 2, 3, 9, 10, or 13) is received.

sA (TCP ACK scan):-

This scan is different than the others discussed so far in that it never determines open (or even open|filtered) ports. It is used to map out firewall rulesets, determining whether they are stateful or not and which ports are filtered.

The ACK scan probe packet has only the ACK flag set (unless you use --scan flags).When scanning unfiltered systems, open and closed ports will both return a RST packet. Network Mapping then labels them as unfiltered, meaning that they are reachable by the ACK packet, but whether they are open or closed is undetermined. Ports that don't respond, or send certain ICMP error messages back (type 3, code 1, 2, 3, 9, 10, or 13), are labeled filtered.

5 Data Flow Diagrams:-

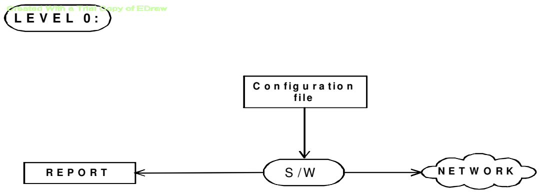

Fig 5.1: LEVEL 0

This fig describes the following: The configuration files are provided to the software. Software maps the network and provides the information to the server.

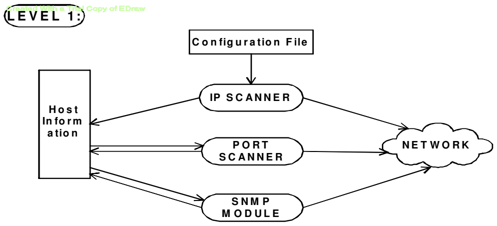

Fig 5.2: LEVEL 1

This fig describes the following: The configuration files are provided to the IP Scanner. Using that file IP Scanner maps the network and provides the IP addresses to the host.

The port scanner uses the IP addresses provided by the host , scans the network and assigns the information to the host .This available information from host is used by SNMP module to identify OS & application running on them.

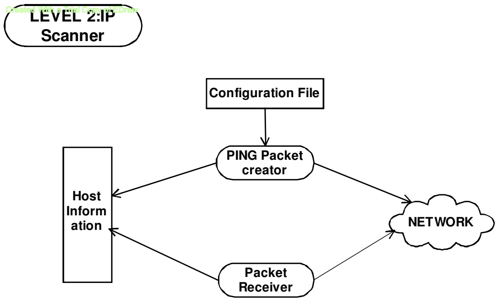

Fig 5.1: LEVEL 2.1

This Fig describes at Level 2 : IP scanner uses the configuration file for PING packet creator for mapping the network. Packet receiver receives the results and provides the information to Host.

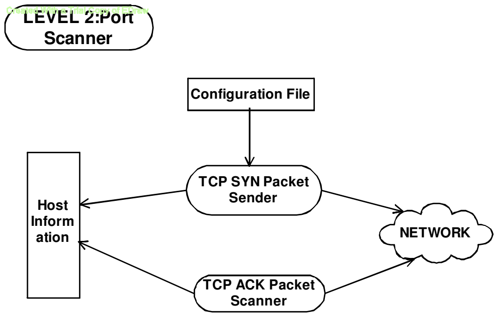

Fig 5.2: LEVEL 2.2

This fig. describes the following: The Port Scanner configuration file is provided to the TCP SYN packet Sender that sends packet to network. TCP ACK packet scanner scans the network & provides the result to the host.

Conclusions

This paper has presented a software prototype of a new network mapping tool that assists network monitoring, providing accurate information and detailed maps. The tool is versatile and can be used in a variety of security applications. Several projects are underway to develop it further, including the integration of this tool with intrusion detection systems and vulnerability detection systems in order to support real-time network anomaly display, and integration with traffic visualization systems.

References

[1] Ipswich Inc.'s What sup Gold. http://www.ipswitch.com/Products/network-management.html

[2] Network Associates Cyber Cop Scanner. http://www.cybercop.co.uk/cybercop/scanner/default.htm

[3] Cheops-nag. http://cheops-ng.sourceforge.net

[4] Fluke Networks LAN Maps hot. http://www.flukenetworks.com/us/LAN/Monitoring+Analysis+Diagramming/LAN+MapShot/Overview.htm

[5] Microsoft Visio Enterprise Network Tools. http://www.microsoft.com/office/visio/evaluation/indepth/network.asp

[6] HP OpenView Network Node Manager. http://www.openview.hp.com/products/nnm/index.asp