ISSN:0975-9646

U. Palani et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (6) , 2014, 7146-7153

Network Security Testing Using Discovery Techniques

Mr.U. Palani
Associate Professor
IFET College of Engineering
Villupuram, Tamil Nadu

Mrs. S. Vanitha
Assistant Professor
IFET College of Engineering
Villupuram, Tamil Nadu

Mr. S. Lakshminarasimman
Assistant Professor
IFET College of Engineering
Villupuram, Tamil Nadu

I. ABSTRACT-

A network security assessment is the process of determining how effectively a network being assessed needs specific security objectives. In this paper we have covered the various challenges faced in providing network security, the threats to network security, the components of a network that are usually attacked and the security management.

This article provides an insight into the methodology and ethos behind penetration testing. It illustrates why security testing can provide substantial value to an organization by securing its IT infrastructure from real world attacks. It describes the need for penetration testing and analyses the requirements of the same before establishing the objectives to be covered in one run of the penetration test. It also provides background information on penetration testing processes and practices. It then discusses various types of threats in today's IT world.

Today's software penetration testing tools, practices, and (to some degree) staff have been developed and improved for an IT Security user base, primarily. However, to effectively make use of these elements in a software development environment takes careful thought and clear goals.

To get around this, and to get closer look on the implementation, challenges & practices discussed here , this document provides a description of and recommendations for a penetration testing process, models and methodology that is more suited to the needs of software developers than is typically found today.

Additionally, the document also provides information why penetration testing cannot be performed as a series of automation test runs by analyzing the methodology used for performing penetration testing. It also provides both a conceptual as well as a more specific survey of the practices, models and methodology available today for conducting penetration testing. This tool survey is then balanced against the need for trained, skilled, and highly motivated testing staff. Staff training is addressed and compared against mentoring or apprenticeship types of on the job training processes.

II. INTRODUCTION

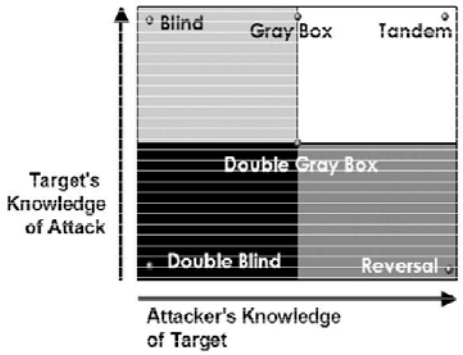
Network testing is the actual measurement and recording of a network's state of operation over a period of time. It involves recording the current state of network operation to serve as a basis for comparison or control. Network defenses are constantly under attack from cyber criminals, organized groups, and hacktivists. This threat landscape poses a danger not only to sensitive data, but to businesses as a whole. Data breaches alone have cost companies and individuals $139 billion in the past five years, and even a small vulnerability in network or data center infrastructures can lead to major damage from denial of service (DoS), malware, and other attacks.

III. NETWORK SECURITY CONCEPTS

Network security starts with authenticating the user, commonly with a username and a password. Since this requires just one detail authenticating the user name —i.e . the password, which is something the user 'knows'— this is sometimes termed one-factor authentication. With two-factor authentication, something the user 'has' is also used (e.g. a security token or 'dongle', an ATM card, or a mobile phone); and with three-factor authentication, something the user 'is'is also used (e.g. a fingerprint or retinal scan).

Once authenticated, a firewall enforces access policies such as what services are allowed to be accessed by the network users. Though effective to prevent unauthorized access, this component may fail to check potentially harmful content such as computer worms or Trojans being transmitted over the network. Anti-virus software or an intrusion prevention system (IPS)[3] help detect and inhibit the action of such malware. An anomaly-based intrusion detection system may also monitor the network and traffic for unexpected (i.e. suspicious) content or behaviour and other anomalies to protect resources, e.g. from denial of service attacks or an employee accessing files at strange times. Individual events occurring on the network may be logged for audit purposes and for later high-level analysis.

Communication between two hosts using a network may be encrypted to maintain privacy.



Honey pots, essentially decoy network-accessible resources, may be deployed in a network as surveillance and early-warning tools, as the honey pots are not normally accessed for legitimate purposes. Techniques used by the attackers that attempt to compromise these decoy resources are studied during and after an attack to keep an eye on new exploitation techniques. Such analysis may be used to further tighten security of the actual network being protected by the honey pot.

What do we test in a network?

One of the first things to establish is what will be tested. Some important network characteristics are:

1) Utilization levels - Costs of communications networks are determined by the maximal capacities of those networks. The traffic carried by networks depends on how heavily those networks are used. Hence utilization rates and

utilization patterns determine the costs of providing services, and therefore are crucial in understanding the economics of communications networks.

A comparison of utilization rates and costs of various networks helps disprove many popular myths about the Internet.

Although packet networks are often extolled for the efficiency of their transport, it often costs more to send data over internal corporate networks than using modems on the switched voice network. Packet networks are not growing explosively because they utilize underlying transport capacity more efficiently, but because they provide much greater flexibility in offering new services. Study of utilization patterns shows there are large opportunities for increasing the efficiency of data transport and making the Internet less expensive and more useful.

2) Number of users - The number of users accessing the network contributes to the load on it. The load on the network determines it efficiency. A clear understanding of when and how many users access the network can help in managing it effectively.

3) Number of operational protocols - With the new standing as a utility, the stakeholders of the Internet demands five nines reliability; the network has to be operational 99.999% of the time. This new requirement means that the developers and manufacturers of network enabled devices and applications must do something they have never done before: deliver feature-rich products, on-time, with high quality. Meeting the new demand for high quality products is a disruptive change for most companies. It affects the culture, the organization, and even the compensation philosophy for the company.

4) Error statistics - The standard error is the standard deviation of the sampling distribution of a statistics.

5) Application utilization - The first major step is now taken in establishing effective network testing. Once you have begun network testing, you must stay the course. Undertaking a programme of network testing will be of no benefit if network testing is performed only after network modifications or problems.

When to perform network test?

• Network testing is not performed while troubleshooting a problem.

• In a perfect world, network test is performed in e ach segment or ring for an extended period of time, say one full week. A network testing schedule would then be determined by analysing the results of the long-term network testing. From this initial long-term network test, trouble spots would be selected and focused on. A trouble spot might consist of excessively high utilization or perhaps high levels of error conditions. Considering time and resources, you may be unable to network test every segment. In this case, you should select your critical or problematic networks for network testing.

IV. Challenges faced in Network Security Testing

Validating and securing network infrastructures requires a constant balance between security and performance. The only way to optimize both of them is to stress network security devices and systems with real-world conditions including stateful benign application traffic, DoS attacks, malware, and heavy user load. Proper network security testing must:

• Assess the performance and security of devices including firewalls, IPS devices, load balancers, and VPN equipment.

• Validate that all network security devices are capable of defending against the latest threats.

• Accurately predict how your network will perform under attack and what effects an attack will have on critical services.

Types of Attacks

Networks are subject to attacks from malicious sources. Attacks can be from two categories "Passive" when a network intruder intercepts data traveling through the network, and "Active" in which an intruder initiates commands to disrupt the networks normal operation.

Types of attacks include:

Passive Attack

Network Wire Tapping

- Telephone tapping is the monitoring of telephone and Internet conversations by a third party, often by covert means. The wiretap received its name because, historically, the monitoring connection was an actual electrical tap on the telephone line. Legal wiretapping by a government agency is also called lawful interception. Passive wiretapping monitors or records the traffic, while active wiretapping alters or otherwise affects it

- **Port scanner**

- A port scanner is a software application designed to probe a server or host for open ports. This is often used by administrators to verify security policies of their networks and by attackers to identify running services on a host with the view to compromise it.

- A port scan or portscan is "An attack that sends client requests to a range of server port addresses on a host, with the goal of finding an active port and exploiting a known vulnerability of that service."

Idle scan

- The idle scan is a TCP port scan method that consists of sending spoofed packets to a computer to find out what services are available. This is accomplished by impersonating another computer called a "zombie" (that is not transmitting or receiving information) and observing the behavior of the zombie system.

- This action can be done through common software network utilities such as nmap and hping. The attack involves sending forged packets to a specific machine target in an effort to find distinct characteristics of another zombie machine. The attack is sophisticated because there is no interaction between the attacker computer and the target: the attacker interacts only with the "zombie" computer.

Active Attack

Denial-of-service attack

- A denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a computer or network resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the concerted efforts of a person, or multiple people to prevent an Internet site or service from functioning efficiently or at all, temporarily or indefinitely. Perpetrators of DoS attacks typically target sites or services hosted on

high-profile web servers such as banks, credit card payment gateways, and even root nameservers. The term is generally used relating to computer networks, but is not limited to this field; for example, it is also used in reference to CPU resource management.

Spoofing

- In the context of network security, a spoofing attack is a situation in which one person or program successfully masquerades as another by falsifying data and thereby gaining an illegitimate advantage.

Man in the middle

- In cryptography, the man-in-the-middle attack (often abbreviated MITM), bucket brigade attack, or sometimes Janus attack, is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker. The attacker must be able to intercept all messages going between the two victims and inject new ones, which is straightforward in many circumstances (for example, an attacker within reception range of an unencrypted Wi-Fi wireless access point, can insert himself as a man-in-the-middle).

ARP poisoning

- The principle of ARP spoofing is to send fake, or spoofed, ARP messages onto a LAN. Generally, the aim is to associate the attacker's MAC address with the IP address of another host (such as the default gateway).
- Any traffic meant for that IP address would be mistakenly sent to the attacker instead. The attacker could then choose to forward the traffic to the actual default gateway (interception) or modify the data before forwarding it (man-in-the-middle attack). The attacker could also launch a denial-of-service attack against a victim by associating a nonexistent MAC address to the IP address of the victim's default gateway.
- A denial-of-service attack may be executed if the attacker is able to use ARP snooping to associate an alternate MAC address with the IP address of the default gateway. Denied access to the gateway in this way, nothing outside the LAN will be reachable by hosts on the LAN.
- ARP spoofing attacks can be run from a compromised host on the LAN, or from an attacker's machine that is connected directly to the target LAN.

Smurf attack

A smurf attack is one particular variant of a flooding DoS attack on the public Internet.

It relies on misconfigured network devices that allow packets to be sent to all computer hosts on a particular network via the broadcast address of the network, rather than a specific machine. The network then serves as a smurf amplifier. In such an attack, the perpetrators will send large numbers of IP packets with the source address faked to appear to be the address of the victim. The network's bandwidth is quickly used up, preventing legitimate packets from getting through to their destination.

V. SECURITY MANAGEMENT

Security management for networks is different for all kinds of situations. A home or small office may only require basic security while large businesses may require high-maintenance and advanced software and hardware to prevent malicious attacks from hacking and spamming.

1) Homes & Small Businesses

- A basic firewall or a unified threat management system .
- For Windows users, basic Antivirus software. An anti-spyware program would also be a good idea. There are many other types of antivirus or anti-spyware programs out there to be considered.
- When using a wireless connection, use a robust password. Also try to use the strongest security supported by your wireless devices, such as WPA2 with AES encryption.
- If using Wireless: Change the default SSID network name, also disable SSID Broadcast; as this function is unnecessary for home use. (However, many security experts consider this to be relatively useless).
- Enable MAC Address filtering to keep track of all home network MAC devices connecting to your router.
- Assign STATIC IP addresses to network devices.
- Disable ICMP ping on router.
- Review router or firewall logs to help identify abnormal network connections or traffic to the Internet.
- Use passwords for all accounts.
- For Windows users, Have multiple accounts per family member and use non-administrative accounts for day-to-day activities.
- Raise awareness about information security to children.

2) Medium businesses

- Strong Antivirus software and Internet Security Software.
- For authentication, use strong passwords and change it on a bi-weekly/monthly basis.
- When using a wireless connection, use a robust password.
- Raise awareness about physical security to employees.
- Use an optional network analyzer or network monitor.
- An enlightened administrator or manager.
- Use a VPN, or Virtual Private Network, to communicate between a main office and satellite offices using the Internet as a connectivity medium. A VPN offers a solution to the expense of leasing a data line while providing a secure network for the offices to communicate. A VPN provides the business with a way to communicate between two in a way mimics a private leased line. Although the Internet is used, it is private because the link is encrypted and convenient to use. A medium sized business needing a secure way to connect several offices will find this a good choice
- Clear employee guidelines should be implemented for using the Internet, including access to non-work related websites, sending and receiving information.
- Individual accounts to log on and access company intranet and Internet with monitoring for accountability.
- Have a back-up policy to recover data in the event of a hardware failure or a security breach that changes, damages or deletes data.
- Assign several employees to monitor a group like CERT which studies Internet security vulnerabilities and develops training to help improve security.

3) Large businesses

- A strong firewall and proxy to keep unwanted people out.
- A strong Antivirus software package and Internet Securit y Software package.
- For authentication, use strong passwords and change it on a weekly/bi-weekly basis.
- When using a wireless connection, use a robust password.·
- Exercise physical security precautions to employees.
- Prepare a network analyzer or network monitor and use it when needed.
- Implement physical security management like closed circui t television for entry areas and restricted zones.
- Security fencing to mark the company's perimeter.
- Fire extinguishers for fire-sensitive areas like se rver rooms and security rooms.
- Security guards can help to maximize security.

4) School

- An adjustable firewall and proxy to allow authorized users access from the outside and inside.
- Strong Antivirus software and Internet Security Software packages.
- Wireless connections that lead to firewalls.
- Children's Internet Protection Act compliance. (Only schools in the USA)
- Supervision of network to guarantee updates and changes based on popular site usage.
- Constant supervision by teachers, librarians, and administrators to guarantee protection against attacks by both internet and sneaker net sources.
- An enforceable and easy to understand acceptable use p olicy which differentiates between school owned and personally owned devices
- FERPA compliance for institutes of higher education

5) Large government

- A strong firewall and proxy to keep unwanted people out.
- Strong antivirus software and Internet Security Software suites.
- Strong encryption.
- Whitelist authorized wireless connection, block all else.
- All network hardware is in secure zones.
- All hosts should be on a private network that is invisible from the outside.
- Host web servers in a DMZ, or a firewall from the outs ide and from the inside.
- Security fencing to mark perimeter and set wireless ra nge to this.
- Inventory controls of government owned mobile .

VI. Network Discovery

Network discovery uses a number of methods to discover active and responding hosts on a network, identify weaknesses, and learn how the network operates.

Types of Network Discovery Techniques:

· Passive techniques use a network sniffer to monitor network traffic and record the IP addresses of the active hosts, and can report which ports are in use and which operating systems have been discovered on the network. Passive discovery can also identify the relationships between hosts—including which hosts communicate with each other, how frequently their communication occurs, and the type of traffic that is taking place—and is usually performed from a host on the internal network where it can monitor host communications. This is done without sending out a single probing packet. Passive discovery takes more time to gather information than does active discovery, and hosts that do not send or receive traffic during the monitoring period might not be reported.

Active techniques send various types of network packets, such as Internet Control Message Protocol (ICMP) pings, to solicit responses from network hosts, generally through the use of an automated tool. One activity, known as OS fingerprinting, enables the assessor to determine the system's OS by sending it a mix of normal, abnormal, and illegal network traffic. Another activity involves sending packets to common port numbers to generate responses that indicate the ports are active. The tool analyzes the responses from these activities, and compares them with known traits of packets from specific operating systems and network services—enabling it to identify hosts, the operating systems they run, their ports, and the state of those ports. This information can be used for purposes that include gathering information on targets for penetration testing, generating topology maps, determining firewall and IDS configurations, and discovering vulnerabilities in systems and network configurations. Tools should definitely be part of penetration testing.

Network Discovery Tools:

Network discovery tools have many ways to acquire information through scanning. Enterprise firewalls and intrusion detection systems can identify many instances of scans, particularly those that use the most suspicious packets (e.g., SYN/FIN scan, NULL scan). Assessors who plan on performing discovery through firewalls and intrusion detection systems should consider which types of scans are most likely to provide results without drawing the attention of security administrators, and how scans can be conducted in a more stealthy manner (such as more slowly or from a variety of source IP addresses) to improve their chances of success.

Assessors should also be cautious when selecting types of scans to use against older systems, particularly those known to have weak security, because some scans can cause system failures. Typically, the closer the scan is to normal activity, the less likely it is to cause operational problems. Network discovery may also detect unauthorized or rogue devices operating on a network. For example, an organization that uses only a few operating systems could quickly identify rogue devices that utilize different ones. Once a wired rogue device is identified,12 it can be located by using existing network maps and information already collected on the device's network activity to identify the switch to which it is connected. It may be necessary to generate additional network activity with the rogue device—such as pings—to find the correct switch. The next step is to identify the switch port on the switch associated with the rogue device, and to physically trace the cable connecting that switch port to the rogue device. A number of tools exist for use in network discovery, and it should be noted that many active discovery tools can be used for passive network sniffing and port scanning as well. Most offer a graphical user interface (GUI), and some also offer a command-line interface. Command-line interfaces may take longer to learn than GUIs because of the number of commands and switches that specify what tests the tool should perform and which an assessor must learn to use the tool effectively. Also, developers have written a number of modules for open source tools that allow assessors to easily parse tool output.

For example, combining a tool's Extensible Markup Language (XML) output capabilities, a little scripting, and a database creates a more powerful tool that can monitor the network for unauthorized services and machines. Learning what the many commands do and how to combine them is best achieved with the help of an experienced security engineer. Most experienced IT professionals, including system administrators and other network engineers, should be able to interpret results, but working with the discovery tools themselves is more efficiently handled by an engineer.

VII. Network Discovery Techniques

Network discovery uses a number of methods to discover active and responding hosts on a network, identify weaknesses, and learn how the network operates.

Active Vs Passive Discovery:

Some of the advantages of active discovery, as compared to passive discovery, are that an assessment can be conducted from a different network and usually requires little time to gather information. In passive discovery, ensuring that all hosts are captured requires traffic to hit all points, which can be time-consuming—especially in larger enterprise networks.., A disadvantage to active discovery is that it tends to generate network noise, which sometimes results in network latency. Since active discovery sends out queries to receive responses, this additional network activity could slow down traffic or cause packets to be dropped in poorly configured networks if performed at high volume. Active discovery can also trigger IDS alerts, since unlike passive discovery it reveals its origination point. The ability to successfully discover all network systems can be affected by environments with protected network segments and perimeter security devices and techniques. For example, an environment using network address translation (NAT)—which allows organizations to have internal, non-publicly routed IP addresses that are translated to a different set of public IP addresses for external traffic—may not be accurately discovered from points external to the network or from protected segments. Personal and host-based firewalls on target devices may also block discovery traffic. Misinformation may be received as a result of trying to instigate activity from devices. Active discovery presents information from which conclusions must be drawn about settings on the target network.

For both passive and active discovery, the information received is seldom completely accurate. To illustrate, only hosts that are on and connected during active discovery will be identified—if systems or a segment of the network are offline during the assessment, there is potential for a large gap in discovering devices. Although passive discovery will only find devices that transmit or receive communications during the discovery period, products such as network management software can provide continuous discovery capabilities and automatically generate alerts when a new device is present on the network. Continuous discovery can scan IP address ranges for new addresses or monitor new IP address requests. Also, many discovery tools can be scheduled to run regularly, such as once every set amount of days at a particular time. This provides more accurate results than running these tools sporadically.

Network Port and Service Identification:

Network port and service identification involves using a port scanner to identify network ports and services operating on active hosts—such as FTP and HTTP—and the application that is running each identified service, such as Microsoft Internet Information Server (IIS) or Apache for the HTTP service. Organizations should conduct network port and service identification to identify hosts if this has not already been done by other means (e.g., network discovery), and flag potentially vulnerable services. This information can be used to determine targets for penetration testing.

All basic scanners can identify active hosts and open ports, but some scanners are also able to provide additional information on the scanned hosts. Information gathered during an open port scan can assist in identifying the target operating system through a process called OS fingerprinting. For example, if a host has TCP ports 135, 139, and 445 open, it is probably a Windows host, or possibly a Unix host running Samba. Other items—such as the TCP packet sequence number generation and responses to packets—also provide a clue to identifying the OS. But OS fingerprinting is not fool proof. For example, firewalls block certain ports and types of traffic, and system administrators can configure their systems to respond in nonstandard ways to camouflage the true OS.

Some scanners can help identify the application running on a particular port through a process called service identification. Many scanners use a services file that lists common port numbers and typical associated services—for example, a scanner that identifies that TCP port 80 is open on a host may report that a web server is listening at that port—but additional steps are needed before this can be confirmed. Some scanners can initiate communications with an observed port and analyse its communications to determine what service is there, often by comparing the observed activity to a repository of information on common services and service implementations. These techniques may also be used to identify the service application and application version, such as which Web server software is in use—this process is known as version scanning. A well-known form of version scanning, called banner grabbing, involves capturing banner information transmitted by the remote port when a connection is initiated. This information can include the application type, application version, and even OS type and version. Version scanning is not fool proof, because a security-conscious administrator can alter the transmitted banners or other characteristics in hopes of concealing the service's true nature. However, version scanning is far more accurate than simply relying on a scanner's services file.

Scanner models support the various scanning methods with strengths and weaknesses that are normally explained in their documentation. For example, some scanners work best scanning through firewalls, while others are better suited for scans inside the firewall. Results will differ depending on the port scanner used. Some scanners respond with a simple open or closed response for each port, while others offer additional detail (e.g., filtered or unfiltered) that can assist the assessor in determining what other types of scans would be helpful to gain additional information.

Network port and service identification often uses the IP address results of network discovery as the devices to scan. Port scans can also be run independently on entire blocks of IP addresses—here, port scanning performs network discovery by default through identifying the active hosts on the network. The result of network discovery and network port and service identification is a list of all active devices operating in the address space that responded to the port scanning tool, along with responding ports. Additional active devices could exist

that did not respond to scanning, such as those that are shielded by firewalls or turned off. Assessors can try to find these devices by scanning the devices themselves,placing the scanner on a segment that can access the devices, or attempting to evade the firewall through the use of alternate scan types (e.g., SYN/FIN or Xmas scan).

It is recommended that if both external and internal scanning are to be used and the assessors are intentionally performing the testing "blind," that external scanning be performed fi rst. Done in this order, logs can be reviewed and compared before and during internal testing. When performing external scanning, assessors may use any existing stealth techniques to get packets through firewalls while evading detection by IDS and IPS.14 Tools that use fragmentation, duplication, overlap, out-of-order, and timing techniques to alter packets so that they blend into and appear more like normal traffic are recommended. Internal testing tends to use less aggressive scanning methods because these scans are blocked less often than external scans. Using more aggressive scans internally significantly increases the changes of disrupting operations without necessarily improving scan results. Being able to scan a network with customized packets also works well for internal testing, because checking for specific vulnerabilities requires highly customized packets. Tools with packet-builder ability are helpful with this process. Once built, packets can be sent through a second scanning program that will collect the results. Because customized packets can trigger a denial of service (DoS) attack, this type of test should be conducted during periods of low network traffic—such as overnight or on the weekend.

Although port scanners identify active hosts, operating systems, ports, services, and applications, they do not identify vulnerabilities. Additional investigation is needed to confirm the presence of insecure protocols (e.g., Trivial File Transfer Protocol [TFTP], telnet), malware, unauthorized applications, and vulnerable services. To identify vulnerable services, the assessor compares identified version numbers of services with a list of known vulnerable versions, or perform automated vulnerability scanning as discussed in Section 4.3. With port scanners, the scanning process is highly automated but interpretation of the scanned data is not.

Although port scanning can disrupt network operations by consuming bandwidth and slowing network response times, it enables an organization to ensure that its hosts are configured to run only approved network services. Scanning software should be carefully selected to minimize disruptions to operations. Port scanning can also be conducted after hours to cause minimal impact to operations.

Vulnerability Scanning:

Like network port and service identification, vulnerability scanning identifies hosts and host attributes (e.g., operating systems, applications, open ports), but it also attempts to identify vulnerabilities rather than relying on human interpretation of the scanning results. Many vulnerability scanners are equipped to accept results from network discovery and network port and service identification, which reduces the amount of work needed for vulnerability scanning. Also, some scanners can perform their own network discovery and network port and service identification. Vulnerability scanning can help identify outdated software versions, missing patches, and mis-configurations, and validate compliance with or deviations from an organization's security policy. This is done by identifying the operating systems and major software applications running on the hosts and matching them with information on known vulnerabilities stored in the scanners' vulnerability databases.

Vulnerability scanners can:

- Check compliance with host application usage and security policies
- Provide information on targets for penetration testing
- Provide information on how to mitigate discovered vulnerabilities.

Vulnerability scanners can be run against a host either locally or from the network. Some network-based scanners have administrator-level credentials on individual hosts and can extract vulnerability information from hosts using those credentials. Other network-based scanners do not have such credentials and must rely on conducting scanning of networks to locate hosts and then scan those hosts for vulnerabilities. In such cases, network-based scanning is primarily used to perform network discovery and identify open ports and related vulnerabilities—in most cases, it is not limited by the OS of the targeted systems. Network-based scanning without host credentials can be performed both internally and externally—and although internal scanning usually uncovers more vulnerabilities than external scanning, testing from both viewpoints is important. External scanning must contend with perimeter security devices that block traffic, limiting assessors to scanning only the ports authorized to pass traffic.

Assessors performing external scanning may find challenges similar to those faced with network discovery, such as the use of NAT or personal and host-based firewalls. To overcome the challenges of NAT and conduct successful network-based scanning, assessors can ask the firewall administrator to enable port forwarding on specific IP addresses or groups of addresses if this is supported by the firewall, or request network access behind the device performing NAT. Assessors can also request that personal or host-based firewalls be configured to permit traffic from test system IP addresses during the assessment period. These steps will give assessors increased insight into the network, but do not accurately reflect the capabilities of an external attacker—althou gh they may offer a better indication of the capabilities available to a malicious insider or an external attacker with access to another host on the internal network. Assessors can also perform scanning on individual hosts.

For local vulnerability scanning, a scanner is installed on each host to be scanned. This is done primarily to identify host OS and application mis-configurations and vulnerabilities—both network-exploitable and locally exploitable. Local scanning is able to detect vulnerabilities with a higher level of detail than network-based scanning because local scanning usually requires both host (local) access and a root or administrative account. Some scanners also offer the capability of repairing local mis-configurations.

A vulnerability scanner is a relatively fast and easy way to quantify an organization's exposure to surface vulnerabilities. A surface vulnerability is a weakness that exists in isolation, independent from other vulnerabilities. The system's behaviors and outputs in response to attack patterns submitted by the scanner are compared against those that characterize the signatures of known vulnerabilities, and the tool reports any matches that are found. Besides signature-based scanning, some vulnerability scanners attempt to simulate the reconnaissance attack patterns used to probe for exposed, exploitable vulnerabilities, and report the vulnerabilities found

when these techniques are successful.

One difficulty in identifying the risk level of vulnerabilities is that they rarely exist in isolation. For example, there could beseveral low-risk vulnerabilities that present a higher risk when combined. Scanners are unable to detect vulnerabilities that are revealed only as the result of potentially unending combinations of attack patterns. The tool may assign a low risk to each vulnerability, leaving the assessor falsely confident in the security measures in place. A more reliable way of identifying the risk of vulnerabilities in aggregate is through penetration testing, which is discussed in previous section.

Another problem with identifying the risk level of vulnerabilities is that vulnerability scanners often use their own proprietary methods for defining the levels. For example, one scanner might use the levels low, medium, and high, while another scanner might use the levels informational, low, medium, high, and critical. This makes it difficult to compare findings among multiple scanners. Also, the risk levels assigned by a scanner may not reflect the actual risk to the organization—for example, a scanner might label an FTP server as a moderate risk because it transmits passwords in cleartext, but if the organization only uses the FTP server as an anonymous public server that does not use passwords, then the actual risk might be considerably lower. Assessors should determine the appropriate risk level for each vulnerability and not simply accept the risk levels assigned by vulnerability scanners.

Network-based vulnerability scanning has some significant weaknesses. As with network sniffing and discovery, this type of scanning uncovers vulnerabilities only for active systems. This generally covers surface vulnerabilities, and is unable to address the overall risk level of a scanned network. Although the process itself is highly automated, vulnerability scanners can have a high false positive error rate (i.e., reporting vulnerabilities when none exist). An individual with expertise in networking and OS security should interpret the results. And because network-based vulnerability scanning requires more information than port scanning to reliably identify the vulnerabilities on a host, it tends to generate significantly more network traffic than port scanning. This may have a negative impact on the hosts or network being scanned, or on network segments through which scanning traffic is traversing. Many vulnerability scanners also include network-based tests for DoS attacks that, in the hands of an inexperienced assessor, can have a marked negative impact on scanned hosts. Scanners often allow all DoS attack tests to be suppressed so as to reduce the risk of impacting hosts through testing.

Another significant limitation of vulnerability scanners is that, like virus scanners and IDSs, they rely on a repository of signatures. This requires the assessors to update these signatures frequently to enable the scanner to recognize the latest vulnerabilities. Before running any scanner, an assessor should install the latest updates to its vulnerability database. Some vulnerability scanner databases are updated more regularly than others—this update frequency should be a major consideration when selecting a vulnerability scanner. Most vulnerability scanners allow the assessor to perform different levels of scanning that vary in terms of thoroughness. While more comprehensive scanning may detect a greater number of vulnerabilities, it can slow the overall scanning process. Less comprehensive scanning can take less time, but identifies only well-known vulnerabilities. It is generally recommended that assessors conduct a thorough vulnerability scan if resources permit.

Vulnerability scanning is a somewhat labour-intensive activity that requires a high degree of human involvement to interpret results. It may also disrupt network operations by taking up bandwidth and slowing response times. Nevertheless, vulnerability scanning is extremely important in ensuring that vulnerabilities are mitigated before they are discovered and exploited by adversaries.

As with all pattern-matching and signature-based tools, application vulnerability scanners typically have high false positive rates. Assessors should configure and calibrate their scanners to minimize both false positives and false negatives to the greatest possible extent, and meaningfully interpret results to identify the real vulnerabilities. Scanners also suffer from the high false negative rates that characterize other signature-based tools—but vulnerabilities that go undetected by automated scanners can potentially be caught using multiple vulnerability scanners or additional forms of testing. A common practice is to use multiple scanners—this provides assessors with a way to compare results.

Breaking Point Network Security Testing Solution:

Breaking Point provides customers advance insight into how to harden and secure network security. By re-creating the behavior of millions of connected users, Breaking Point products ensure that enterprises, government organizations, service providers, and equipment vendors can maintain network resiliency by subjecting their security equipment to extreme real-world conditions.

Breaking Point products allow users to:

- Stress network infrastructures with a choice of more than 4,500 security attacks and 28,000 pieces of malware—including more than 100 pieces of mobile malware—plus obfuscations and evasions.
- Measure and harden the performance, security, and stabili ty of load balancers, firewalls, IPS devices, and other equipment with up to 120 gigabits per second of application, attack, and malformed traffic.
- Re-create more than 150 application protocols—with multicast support.
- Emulate sophisticated, large-scale DoS and mobile-initiated botnet attacks to uncover previously hidden weaknesses.

VIII. CONCLUSION

No matter what the threat, a professional network security test should accurately model the attack characteristics of the profiles discussed. Also one should be able to use the appropriate network security and discovery type. A methodical and scientific approach should be used to successfully document a test and create reports that are aimed at different levels of management within an organization. Finally, network security testing should never be regarded as a one-off service. Systems change, threats emerge and business strategies evolve. Testing should be repeated at frequent intervals and particularly following major changes to an IT infrastructure. It's also important to remember that network security testing is but just one form of testing and any organization should develop an overall security testing strategy that is tailored to the threat models and security policies of their organization.

REFERENCES

[1] Allen, J.; Barnum, S.; Ellison, R.; McGraw, G.; Mead, N. Software Security Engineering: A Guide for Project Managers. Boston, MA: Addison-Wesley, 2008 (ISBN 978-0-321-50917-8).

[2] Arkin, B.; Stender, S.; & McGraw, G. "Software Penetration Testing." IEEE Security & Privacy Magazine 3, 1 (Jan-Feb 2005): 84–87.

[3] Farmer, D. & Venema, W. "Improving the Security of Your Site by Breaking Into it."http://www.fish2.com/security/admin-guide-to-cracking.html"

[4] Garfinkel, S.; Schwartz A.; & Spafford, E. Practical Unix and Internet Security, 3rd edition. Sebastopol, CA: O'Reilly and Associates, 2003.

[5] Wysopal, C.; Nelson, L.; Zovi, D.; Dustin, E. The Art of Software Security Testing: Identifying Software Security Flaws. Boston, MA: Addison-Wesley, 2008 (ISBN 0-321-30486-1).

[6] Mark blackburn, robert busser, aaron nauman, t-vec "model-based approach to security test automation", national institute of standards and technology.

[7] Anthony De loreto, Software Engineer, IBM DB2, "Test Automation Theory"

[8] McGraw, Gary. *Software Security: Building Security In*. Boston, MA: Addison-Wesley Professional, 2006 (ISBN 0-321-35670-5).

[9] http://en.wikipedia.org/wiki/Application_security

[10] Srinath Anantharaju, Security Team, "Automating web application security testing"

[11] http://en.wikipedia.org/wiki/Security_testing

[12] http://googleonlinesecurity.blogspot.com/2007/07/auto mating-web-application-security.html

[13] http://en.wikipedia.org/wiki/Test_automation

[14] http://smartbear.com/products/qa-tools/automated-testing