Information Gathering & Vulnerability Assessment

Content

- Introducing Information Gathering Techniques
- Scanning and Discovering Target Clients
- Target System Scanning
- Target System Scanning with GUI

Introducing Information Gathering Techniques

Different Testing Methodologies

Often people get confused with the following terms and use them interchangeably without understanding that although there are some aspects that overlap within these, there are also subtle differences that needs attention:

- Ethical hacking
- Penetration testing
- Vulnerability assessment
- Security audits

Ethical Hacking

Hacking means different things to different people and more often, a hacker is thought of as a person sitting in a closed enclosure with no social life and with a *malicious intent*. Thus, the word *ethical* was prefixed to the term *hacking*. The term *ethical hacking* is used to refer to professionals who work to identify loopholes and vulnerabilities on systems, report it to the vendor or owner of the system, and also, at times, help them fix it. The tools and techniques used by an ethical hacker are similar to the ones used by a cracker or a Black Hat hacker, but the aim is different as it is used in a more professional way. Ethical hackers are also known as security researchers.

Penetration Testing

Penetration testing is a more professional term used to describe what an ethical hacker does. If you are planning for a career in hacking, then you would often see job posting with the title penetration tester. Penetration Testing is a way of identifying vulnerabilities in the systems and finding if the vulnerability is exploitable or not. Penetration testing is bound by a contract between the tester and owner of the systems to be tested. You need to define the scope of the test to identify the systems to be tested. The rules of engagement need to be defined, which decide the way in which the testing is to be done.

Vulnerability Assessment

Vulnerability assessments are broader than penetration tests. The end result of vulnerability assessment is a report prioritizing the vulnerabilities found, with the most severe ones on the top and the ones posing lesser risk lower in the report. This report is really helpful for clients who know that they have security issues but need to identify and prioritize the most critical ones.

Security Audits

Auditing is a systematic procedure that is used to measure the state of a system against a predetermined set of standards. These standards could be industry best practices or an in-house checklist. The primary objective of an audit is to measure and report on conformance. If you are auditing a web server for example, some of the initial things to look out for are the ports open on the server, harmful HTTP methods such as TRACE enabled on the server, the encryption standard used, and the key length.

Information Gathering

This is the process of collecting as much information as possible about the target system. For example, information about the **Domain Name System** (**DNS**) hostnames, IP addresses, technologies and configuration used, username organization, documents, application code, password reset information, contact information, and so on. During information gathering, every piece of information gathered is considered important.

Information gathering can be categorized in two ways based on the method used: **active** information gathering and **passive** information gathering.

In the active information gathering method, we collect information by introducing network traffic to the target network, while in the passive information gathering method, we gather information about a target network by utilizing a third party's services, such as the Google search engine.

Discovering Target System

Target Discovery

The purpose of this process is as follows:

- ✓To find out which machine in the target network is available. If the target machine is not available, we won't continue the penetration testing process on that machine and will move to the next machine.
- ✓To find the underlying operating system used by the target machine.

We can utilize the tools provided in Kali Linux for the target discovery process. Some of these tools are available in the **Information Gathering** menu. Others will have to be utilized from the command line.

NOTE: For the purposes of this section, an installation of Metasploitable has been completed and will be utilized as a target system (Linux).

Identifying the Target System

The tools mentioned here are used to identify the target machines that can be accessed by a penetration tester. Before we start the identification process, we need to know our client's terms and agreements. If the agreements require us to hide pen-testing activities, we need to conceal our penetration testing activities. Stealth techniques may also be applied for testing the **Intrusion Detection System** (**IDS**) or **Intrusion Prevention System** (**IPS**) functionality. If there are no such requirements, we may not need to conceal our penetration testing activities.

ping

The ping tool is the most famous tool that is used to check whether a particular host is available. The ping tool works by sending an **Internet Control Message Protocol** (**ICMP**) echo request packet to the target host. If the target host is available and the firewall is not blocking the ICMP echo request packet, it will reply with the ICMP echo reply packet.

Demo: ping ip_addr

arping

The arping tool is used to ping a host in the **Local Area Network** (**LAN**) using the **Address Resolution Protocol** (**ARP**) request. You can use arping to ping a target machine using its IP, host, or **Media Access Control** (**MAC**) address.

Demo: arping ip_addr –c 1

fping

The difference between ping and fping is that the fping tool can be used to send a ping (ICMP echo) request to several hosts at once. You can specify several targets on the command line, or you can use a file containing the hosts to be pinged.

In the default mode, fping works by monitoring the reply from the target host. If the target host sends a reply, it will be noted and removed from the target list. If the host doesn't respond within a certain time limit, it will be marked as unreachable.

Demo: fping ip1 ip2 ip3

OS Fingerprinting

After we know that the target machine is alive, we can then find out the operating system used by the target machine. This method is commonly known as **Operating System** (**OS**) fingerprinting. There are two methods of doing OS fingerprinting: **active** and **passive**.

In the active method, the tool sends network packets to the target machine and then determines the OS of the target machine based on the analysis done on the response it has received. The advantage of this method is that the fingerprinting process is fast. However, the disadvantage is that the target machine may notice our attempt to get its operating system's information.

To overcome the active method's disadvantage, there is a passive method of OS fingerprinting. This method was pioneered by *Michal Zalewsky* when he released a tool called **pof**. The major advantage of passive OS fingerprinting is that it does the work while reducing the interaction between the testing machine and the target, greatly increasing the stealth of the fingerprinting. The most significant disadvantage of the passive method is that the process will be slower than the active method.

p0f

The p0f tool is used to fingerprint an operating system passively. It can be used to identify an operating system on the following machines:

- Machines that connect to your box (SYN mode; this is the default mode)
- Machines you connect to (SYN+ACK mode)
- Machines you cannot connect to (RST+ mode)
- Machines whose communications you can observe

The p0f tool works by analyzing the TCP packets sent during the network activities. Then, it gathers the statistics of special packets that are not standardized by default by any corporations. An example is that the Linux kernel uses a 64-byte ping datagram, whereas the Windows operating system uses a 32-byte ping datagram, or the **Time to Leave** (**TTL**) value. For Windows, the TTL value is 128, while for Linux this TTL value varies between the Linux distributions. This information is then used by pof to determine the remote machine's operating system.

p0f in action

1. To access pof, open a console and type pof -h. This will display its usage and options' description. Let's use pof to identify the operating system used in a remote machine we are connecting to. Just type the following command in your console: **pof –f /usr/share/pof/pof.fp -o pof.log**
2. This will read the fingerprint database from the usr/share/pof/pof.fp file and save the log information to the pof.log file. It will then display the following information:

 # pof -f /usr/share/pof/pof.fp -o pof.log
--- pof 3.07b by Michal Zalewski <lcamtuf@coredump.cx> ---
[+] Closed 1 file descriptor.
[+] Loaded 320 signatures from '/usr/share/pof/pof.fp'.
[+] Intercepting traffic on default interface 'etho'.
[+] Default packet filtering configured [+VLAN].
[+] Log file 'pof.log' opened for writing.
[+] Entered main event loop.

3. Next, you need to generate network activities involving a TCP connection, such as browsing to the remote machine or letting the remote machine connect to your machine. For the purposes of this demonstration, a connection to the HTTP site on the Metasploitable 2 machine was established.

Nmap

Nmap is a very popular and capable port scanner. Besides this, it can also be used to fingerprint a remote machine's operating system. It is an active fingerprinting tool. To use this feature, you can use the -O option to the nmap command.

For example, if we want to fingerprint the operating system used on the target machine, we use the following command: **nmap –O ip_addr**

Nmap was able to get the correct operating system information after fingerprinting the operating system of a remote machine.

Scanning Target System

Scanning Target System

This is a process that is used to find and collect information about ports, operating systems, and services available on the target machines. This process is usually done after we have discovered that the target machines are available. In penetration testing practice, this task is conducted at the time of the discovery process. The goal of performing the scanning process is to collect information about the services available on the target systems.

The Network Scanner

There are several tools that can be used to find open ports, fingerprint the remote operating system, and enumerate the services on the remote machine; however, I will use Nmap.

Network scanning is a method that is used to find the service version that is available on a particular port on the target system. This version information is important because with this information, the penetration tester can search for security vulnerabilities that exist for that software version.

While standard ports are often used, sometimes systems administrators will change the default ports for some services. For example, an SSH service may be bound to port 22 (as a convention), but a system administrator may change it to be bound to port 2222. If the penetration tester only does a port scan to the common port of SSH, it may not find that service. The penetration tester will also have difficulties when dealing with proprietary applications running on non-standard ports. By using the service enumeration tools, these two problems can be mitigated, so there is a chance that the service can be found, regardless of the port it binds to.

Nmap

Nmap is a port scanner that is comprehensive, feature- and fingerprint-rich, and widely used by the IT security community. It is a must-have tool for a penetration tester because of its quality and flexibility.

Besides being used as a port scanner, Nmap has several other capabilities, as follows:

Host discovery: Nmap can be used to find live hosts on the target systems. By default, Nmap will send an ICMP echo request, a TCP SYN packet to port 443, a TCP ACK packet to port 80, and an ICMP timestamp request to carry out the host discovery.

Service/version detection: After Nmap has discovered the ports, it can further check for the service protocol, the application name, and the version number used on the target machine.

Operating system detection: Nmap sends a series of packets to the remote host and examines the responses. Then, it compares these responses with its operating system fingerprint database and prints out the details if there is a match. If it is not able to determine the operating system, Nmap will provide a URL where you can submit the fingerprint to update its operating system fingerprint database. Of course, you should submit the fingerprint if you know the operating system used on the target system.

Nmap

Network traceroute: This is performed to determine the port and protocol that is most likely to reach the target system. An Nmap traceroute starts with a high value of **Time to Live** (**TTL**) and decrements it until the TTL value reaches zero.

Nmap Scripting Engine: With this feature, Nmap can be extended. If you want to add a check that is not included with the default Nmap, you can do so by writing the check using the Nmap scripting engine. Currently, there are checks for vulnerabilities in network services and for enumerating resources on the target system.

It is good practice to always check for new versions of Nmap. If you find the latest version of Nmap available for Kali Linux, you can update your Nmap by issuing the following commands:

apt-get update
apt-get install nmap

Nmap

To start Nmap, you can navigate to **Applications** and then to **Information Gathering**. You can also start Nmap by going to the console to execute the following command:

nmap

This will display all of the Nmap options with their descriptions.

A new user to Nmap will find the available options quite overwhelming.

Fortunately, you only need one option to scan for the remote machine. That option is your target IP address or hostname, if you have set up the DNS correctly. This is done with the following command:

nmap ip_addr

Nmap: Operating System Detection

Nmap can also be asked to check the operating system used on the target machine. This information is very useful when you do the vulnerability identification process later on.

To use this feature, give Nmap the -O option.

The following is an example of this feature's usage. We want to find the operating system used on the target machine:

nmap -O ip_addr

Nmap: Other Options

Disabling host discovery : If a host is blocking a ping request, Nmap may detect that the host is not active; so, Nmap may not perform heavy probing, such as port scanning, version detection, and operating system detection. To overcome this, Nmap has a feature for disabling host discovery. With this option, Nmap will assume that the target machine is available and will perform heavy probing against that machine.

This option is activated by using the -Pn option.

Aggressive scan : If you use the -A option, it will enable the following probe:
Service version detection (-sV)
Operating system detection (-O)
Script scanning (-sC)
Traceroute (--traceroute)

It may take some time for this scan type to finish

Nmap: Options for IDS/Firewall Evasion

During penetration testing, you may encounter a system that is using firewall and IDS to protect the system. If you just use the default settings, your action may get detected or you may not get the correct result from Nmap.

The following options may be used to help you evade the firewall/IDS:

-f (fragment packets): The purpose of this option is to make it harder to detect the packets. By specifying this option once, Nmap will split the packet into 8 bytes or less after the IP header.

--mtu: With this option, you can specify your own packet size fragmentation. The **Maximum Transmission Unit** (**MTU**) must be a multiple of eight or Nmap will give an error, and exit.

Nmap: Options for IDS/Firewall Evasion

-D (decoy): By using this option, Nmap will send some of the probes from the spoofed IP addresses specified by the user. The idea is to mask the true IP address of the user in the log files. The user IP address is still in the logs. You can use RND to generate a random IP address or RND:number to generate the <number> IP address. The hosts you use for decoys should be up, or you will flood the target. Also remember that by using many decoys you can cause network congestion, so you may want to avoid that, especially if you are scanning your client network.

--source-port <portnumber> or –g (spoof source port): This option will be useful if the firewall is set up to allow all incoming traffic that comes from a specific port.

--data-length: This option is used to change the default data length sent by Nmap in order to avoid being detected as Nmap scans.

--max-parallelism: This option is usually set to one in order to instruct Nmap to send no more than one probe at a time to the target host.

--scan-delay <time>: This option can be used to evade IDS/IPS that uses a threshold to detect port scanning activity.

Scanning Target System with GUI

Zenmap

Zenmap is the graphical interface of Nmap. The advantages of Zenmap compared to Nmap are as follows:

- Zenmap is interactive; it arranges the scan results in a convenient way. It can even draw a topological map of the discovered network.
- Zenmap can do a comparison between two scans.
- Zenmap keeps a track of the scan results.
- To run the same scan configuration more than once, the penetration tester can use a Zenmap profile.
- Zenmap will always display the command that is run, so the penetration tester can verify that command.

To start Zenmap, navigate to **Kali Linux | Information Gathering | Network Scanners | Zenmap**, or use the console to execute the following command: **#zenmap**

This will display the main Zenmap window. Zenmap comes with 10 profiles that can be chosen. To find which command options are used on each profile, just click on **Profile** and the command options will be displayed in the **Command:** box

Thank you