**BASICS OF THREAT DETECTION**

**UNIT – IV**

**M*ETASPOLIT BASICS TERMINOLOGY***

 *Metasploit itself is free , open source software , with many contributions in the security community , but two commercial Metasploit versions are also available . Metasploit isn’t just tool , it,s an entire framework that provide the infrastructure needed to automate mundane , routine and complex tasks . this allows you to concentrate on the unique or specialized aspects of penetration testing and on identifying flaws within your information security program .*

 *The majority of the following basics terms are defined in the context of Metasploit , but they are generally the same throughout the security industry .*

* *Exploit*
* *Payload*
* *Shellcode*
* *Module*
* *Listener*

***METASPLOIT UTILITIES***

*Having covered Metasploit’s three main interfaces , it,s time to few utilities . Metasploit’s are direct interfaces to particular features of the framework that can useful in specific situations , especially in exploit development .*

***MSFpayload***

*The msfpayload component of Metasploit allows you to generate shellcode , executables , and much more for use in exploits of the framework . Shellcode can be generated in many formats including C, Ruby , JavaScript, and even visual situations .*

***MSFencode***

*The shellcode generated by msfpayload is fully functional, but it contains several null characters that , when interpreted by many programs signify the end of a string and this will cause the code to terminate before completion .*

 *Metasploit contains a number of different encoders for specific situations . some will be useful when you can use only alphanumeric characters as parts of a payload , as is the case with many file format exploits or others applications that accept only printable characters as input , while others are great general purpose encoders that do well in every situation.*

***NASM SHELL***

*The nasm-shell utility can handy when you,re trying to make sense of assembly code , especially if , during exploit development , you need to identify the opcodes (the assembly instructions) for a given assembly command .*

***TARGETED SCANNING***

*When you are conducting a penetration test , there is no shame in looking for an easy win . A targeted scan looks for specific operating systems , services , program versions , or configurations that are known to exploitable and that provide an easy door into a target network . for example , it is common to scan a target network quickly for the vulnerability MSO8-067, as this is (still) an extremely common hole that will give you SYSTEM access much more quickly than scanning an entire target network for vulnerabilities .*

***Server message block scanning***

 *Metasploit can scour a network and attempt to versions of Microsoft windows using its module .*

 *We have discovered a system running windows XP without having to do a full scan of the network . This is a great way to target hosts quickly and quietly that are likely to be more vulnerable when our goal is avoid being noticed .*

***Hunting for poorly configured Microsoft SQL servers***

 *Poorly configured Microsoft SQL server (MS SQL) installations often provide an initial way into a target network . In fact , many system administration don’t even realize that they have MS SQL server installed as a prerequisite for some common software , such as Microsoft visual studio . These installations are often unused , unpatched or never configured.*

***SSH Server Scanning***

 *If during your scanning you encounter machines running secure shell (shell) , you should determine which version is running on the target , SSH is secure protocol , but vulnerabilities in various implementation have been identified. You never know when you might get lucky and come across an old machine that hasn’t been updated . you can use the framework’s ssh\_server module to determine the SSH version running on the target server .*

***FTP scanning***

 *FTP is a complicated and insecure protocol . FTP server are often the easiest way into a target network , and you should always scan for , identify , and finger-print any FTP servers running on your target . next , we scan our XP box for FTP services using the framework’s ftp\_version module*

***Simple Network Management Protocol Sweeping***

 *The Simple Network Management Protocol Sweeping (SNMP) is typically used in network devices to report information such as bandwidth utilization , collision rates , and other information . however , some operating system also have SNMP severs that can provide information such as GPU utilization , free memory , and other system-specific details .*

**INTELLIGENCE GATHERING**

 *Intelligence gathering follows the pre-engagement activities as the second step in a penetration test . your goals during intelligence gathering should be to gain accurate information about your targets without revealing your presence or your intensions , to learn how the organization operates , and to determine the best route of entry .*

 *Intelligence gathering is arguably the most important aspect of a penetration test because it provides the foundation for all work that follows . when recording your work , be methodical , accurate , and precise . And as stated earlier , be sure that before you fire off your exploits , you have learned all that you can about your target .*

***PASSIVE INFORMATION GATHERING***

 *By using passive and indirect information gathering you can use these techniques to identify network maintainers and even learn what operating systems and web server software is in use on the target network .*

 *Open source intelligence (OSINT) is a form of intelligence collection that uses open or readily available information to find , select , and acquire information about a target , Several tools make passive information gathering almost painless .*

***Whois lookups***

 *Let’s begin by using Backltrack’s whois lookup to find the names of sermaniar.net’s domain servers .*

 *The Domain Name System (DNS) severs are hosted by DOMAINCONTROL.COM , so this is good example of systems that would attack them . In most larger organization , the DNS servers are housed within the company and are viable attack vectors . zone transfers and similar DNS attcks can often be used to learn more about a network from both the inside and outside . In this scenario , because DOMAIMCONTROL.COM is not owned by secmaniac.net , we should not attack these systems and will instead move on to a different attack vector .*

***Netcraft***

 *Netcraft (*[*http://searchdns.netcraft.com/*](http://searchdns.netcraft.com/)*) is a web-based tool that we can use to find the IP address of a server hosting a particular website , having identified secmaniac.net’s IP address as 75,118,185,142, we do another whois lookup on that IP address*

***NSLookup***

 *To get additional server information , we’ll use back|track to leverage nslookup , a tool built into most operating systems , to find information about secmaniac.net.*

***ACTIVE INFORMATION GATHERING***

 *In active information gathering , we interact with a system to learn more about it . we might , for example , conduct port scans for open ports on the target or conduct ports scans for open ports on the target or conduct scans to determine what services are running . Each system or running service that we discover gives us another opportunity for exploitation . But beware : If you get careless while active information gathering , you might be nabbed by an IDS or intrusion prevention system (IPS) not a good outcome for thr covert penetration tester.*

***Part scanning with Nmap***

 *Having identified the target IP range with passive information gathering as well as the secmaniac.net target IP address , we can begin to scan for open ports on the target by port scanning a process whereby we meticulously connect to port on the remote host to identify those that are active . (Obviously , in a larger enterprise , we would have multiple IP ranges and things to attack instead of only one IP ).*

 *For example , let’s leave secmaniac.net behind and trun to the virtual machine described in Appendix A , with IP address 172.16.32.131 . before we get started , take a quick look at the basic nmap syntax by by entering nmap from the command line on your back|track machine .*

***Working with database in Metasploit***

 *When you’re running a complex penetration test with lot of target , keeping track of everything can be challenge . Luckily , Metasploit has you covered with expansive support for multiple database systems .*

 *To ensure that database system is available for your system , you should first decide which database system you want to run . Metasploit supports MySQL and PostgreSQL : because PostgreSQL is the default , we’ll stick with it in the this discussion .*

***Importing Nmap results into Metasploit***

 *When you are working with other team members , with various individuals scanning at different times and form different locations , it helps to know how to run nmap on its own and then import its results into the framework . Next , we’ll nmap’s-ox option into the framework .*

***Part scanning with Metasploit***

 *In addition to its ability to use third-party scanners . Metasploit has several port scanners built into its auxiliary modules that directly integrate with most aspects of the framework . we’ll use these port scanners to leverage compromised systems to access and attack his process , often called pivoting , allows us to use internally connected system to route traffic to a network that would otherwise be inaccessible .*

***VULNERABILITY SCANNING***

 *A vulnerability scanner is an automated program designed to look for weakness in computers, computer system , networks , and applications . the program and analyzing the response received , in an effort to enumerate any vulnerabilities present on the target by using its vulnerability database as reference .*

 **The Basic Vulnerability Scan**

 *Let’s look at how a scan works at the most basic level. In the following listing, we use netcat to grab a banner from the target 192.168.1.203. Banner grabbingis the act of connecting to a remote network service and reading the service identification (banner) that is returned. Many network services such as web, file transfer, and mail servers return their banner either immediately upon connecting to them or in response to a specific command. Here we connect to a web server on TCP port 80 and issue a GET HTTP request that allows us to look at the header information that the remote server returns in response to our request.*

**Scanning with NeXpose**

 *NeXpose is Rapid7’s vulnerability scanner that scans networks to identify the devices running on them and performs checks to identify security weak-nesses in operating systems and applications. It then analyzes the scan data and processes it for inclusion in various reports*

 *Our target for scanning will be a default installation of Windows XP SP2 as configured in Appendix A. We will first perform a basic overt scan of our target and import the vulnerability scan results into Metasploit. We will close out this section by showing you how to run a NeXpose vulnerability scan directly from msfconsole rather than using the web-based GUI, eliminating the need to import a scan report.*

**Scanning with Nessus**

*The Nessus vulnerability scanner from Tenable Security (http://www.tenable.com/ ) is one of the most widely used vulnerability scanners. Metasploit’s Nessus plug-in lets you launch scans and pull information from Nessus scans via the console, but in the example that follows, we’ll import Nessus scan results independently. Using Nessus with a free Home Feed, we’ll run this scan against the same target we’ll use throughout this chapter, with known credentials. In these early stages of a penetration test, the more tools you can use to fine-tune your future attacks, the better.*

**Specialty Vulnerability Scanners**

 *Although many commercial vulnerability scanners are available on the market, you are not limited to them. When you want to run a scan for a specific vul-nerability across a network, Metasploit’s many auxiliary modules can help you accomplish such tasks.*

**Using Scan Results for Autopwning**

*Let’s take a quick diversion into exploitation. Metasploit’s Autopwn tool auto-matically targets and exploits a system using an open port or using the results of a vulnerability scan export. You can use Autopwn to harness the results of most vulnerability scanners, including NeXpose, Nessus, and OpenVAS.*

*For example, here’s how we could use a Nessus results import to target a system and autopwn it. Create a new database with db\_connect and use db\_importto import the scan report. In the next example, we run db\_autopwn with a series of switches to launch attacks against all targets (e), show all matching modules (t), use a reverse shell payload (r), select exploit modules based on vulnerability (x), and also select based on open ports (p). Once db\_autopwnlaunches, Metasploit begins launching exploits at the targets. Successful exploits return a shell to the attacking machine.*