OULUN AMMATTIKORKEAKOULU

Janne Metso

PENETRATION TESTING

Ethical Hacking

PENETRATION TESTING

Ethical hacking

Janne Metso
Bachelor’s thesis
Autumn 2019
Information technology
Oulu University of Applied Sciences

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Tietojenkäsittelyn tutkinto-ohjelma, Järjestelmäasiantuntemus

Tekijä(t): Janne Metso
Opinnäytetyön nimi: Penetration testing
Työn ohjaaja: Jukka Kaisto
Työn valmistumislukukausi- ja vuosi: Syksy 2019 Sivumäärä: 27

Työn aiheena on penetraatiotestaus, joka tunnetaan myös eettisenä tai valkohattuhakkerointina. Tämä aihe kiehtoo monia, varsinkin tietoteknisellä alalla työskenteleviä ihmisiä. Tämän opinnäytetyön tekijä on halunnut tutustua penetraatiotestaukseen ja siihen liittyviin termeihin ja työkaluihin jo pidemmän aikaa. Alan ammattilaisen on hyvä kasvattaa tietämystään myös penetraatiotestauksen saralla. Tietotaidon lisääminen penetraatiotestauksesta ja järjestelmähaavoittuvuuksista antaa järjestelmistä vastuullisille henkilöille hyvät valmiudet dokumentoida ja valvoa näitä järjestelmiä. Samalla kun opinnäytetyön tekijä raottaa tietoturvallisuuden ja siihen liittyvien osa-alueiden verhoa, hän saa runsaasti lisää uutta tietoa siitä, miten tietoturvan ammattilaiset toimivat ja minkälaisia työkaluja he käyttävät työssään. Tämä opinnäytetyö käsittelee ainoastaan penetraatiotestauksen teoriaa ja testauksessa käytettäviä työkaluja, eikä puutu muihin tietoturvaan ja testaamiseen liittyviin osa-alueisiin. Tämä katsaus vain raapaisee pintaa työkalujen osalta. Tietoturvan kartoitukseen on käytettävissä valtava määrä erilaisia työkaluja ja lisää tietoa aiheesta löytää tutustumalla alan kirjallisuuteen ja internetistä löytyviin lähteisiin. Opinnäytetyössä itsessään on käytetty lähteinä alan kirjallisuutta sekä internetistä löytyviä lähteitä.

Asiasanat: Tietoturva, Kyberturvallisuus, Kybersodankäynti, Hakkerointi, Tietosuoja, Tietovuodot, Pääsynvalvonta

ABSTRACT

Oulu University of Applied Sciences
Barchelor of Business Administration, Degree Programme in Business Information Technology

Author(s): Janne Metso
Title of thesis: Penatration testing
Supervisor(s): Jukka Kaisto
Term and year when the thesis was submitted: Autumn 2019 Number of pages: 27

The subject of this thesis is Penetration testing (*pentesting*), also known as Ethical or White Hat Hacking. It is a subject that intrigues many people, especially the ones working in business information technology area. The author of this thesis has wanted to learn pentesting for a while now and the fine art of pentesting is a great asset to a system administrator. Gaining knowledge about pentesting gives good tools to control and document the security of the system one is responsible for. At the same time as learning pentesting opens a door to a whole new world, it will benefit the maker of this thesis by giving a good base knowledge for the world of penetration testing. This thesis will only be about theory of pentesting in general and introducing pentesting tools. It will not cover any other aspects of security and network testing.

Keywords: Cyber Attacks, Cyber Crime, Cyber Security, Data Security, Hacking, Data Break-In, Computer Crimes

CONTENTS

1 INTRODUCTION

The number of cyber security crimes is increasing all the time. Major companies are releasing more and more news about new cyber threats and information breaches made by different malicious groups or individuals. The threats of breaching the confidentiality, integrity and availability of networks, security systems, operating systems and data is increasing rapidly. While the knowledge about security threats is rising and specialists, as well ordinary people, want to know more about security and security related issues, so rises the knowledge of people that want to cause harm and chaos in cyber systems. The battle among so called *White Hat Hackers* and *Black Hat Hackers* is a never-ending circle where both parties play a cat and mouse game day after day. Unfortunately, the criminals executing these malicious acts are very clever and resourceful and they have the knowledge and the means necessary to penetrate even the robust security systems. This is a serious threat to every individual, community, company or government around the world. The growing need for security specialists and penetration testers has woken up politicians, media, education facilities and the ones in charge of information technology, giving the possibility to offer more cyber security related education and share more knowledge about cyber security related issues. Sharing and educating people about the ways of securely operate private data and systems will give less attack surface to the attackers and it will hopefully decrease the success rate of possible exploitations. Maybe one of the biggest security threats is the end-user that faces social-engineering attacks. The second biggest threat is the weak passwords on Internet-facing services, exposing personal and secure data to outsiders. The third biggest threat is all the unpatched vulnerabilities still available in software and operating systems.

2 WHAT IS PENETRATION TESTING

Penetration testing, or *pentesting*, is a way of simulating attacks so the pentester can assess the risks that are associated with potential security breaches. Pentesting is not to be mixed with a vulnerability assessment. Vulnerability assessment discovers only vulnerabilities that could be used by attackers, but a pentester discovers vulnerabilities and exploits them where possible. Exploiting vulnerabilities gives the pentester a chance to assess the benefits that an attacker might gain after a successful exploitation. (Weidman. 2013, 1.)

A penetration tester is a so-called *White Hat Hacker*, or *ethical hacker*, who breaks into protected systems and networks to test them to assess their level of security. There are several ways to execute a penetration test. A client may want the tester to perform an *external penetration test* that simulates an attack from the Internet. Pentester performs client-side attacks and social-engineering to gain access to a client's internal network. Some tests require the pentester to perform an *internal penetration test* where the pentester is acting like a malicious employee or attacker who has already breached the perimeter. Sometimes the principal might want the pentester to test the security of the wireless network. (Weidman. 2013, 2.)

The *internal penetration* testing is called *white box* penetration testing (also known as clear box testing or glass box testing) and *external penetration* testing is categorized as a *black box* penetration testing. A white box penetration tester has access to the organization internal infrastructure and all the documents that organization may have from its systems. When the pentester has access to the internal systems, the testing process can be very deep and thorough. It maximizes the testing time and the pentester can utilize more recourses for the testing phase. However this is not a very realistic attack type, as the penetration tester is not in the same position as a *black hat hacker* (malicious hacker) would be. A black box penetration test does not require any previous information from the organization infrastructure, because the intention of this type of test is doing an external penetration test, where a pentester tries to find vulnerabilities from the internet facing services the organization has. The black box penetration test is a very realistic scenario for malicious attack, but it takes more time and some areas of the organization infrastructure might remain untested. (Secforce 2016, cited 1.3.2016.)

3 THE STAGES OF THE PENETRATION TEST

Penetration testing consists of seven different phases. Pentesting begins with the *pre-engagement* phase which means talking with the client and going through the goals, scope and reporting format of the penetration test. The *information-gathering* phase is a phase where the pentester uses publicly available tools and means to find information about the client, giving possible options for the tester to connect to the target. The *threat-modelling* phase is where the tester evaluates information gained from the previous phase. The pentester then determines the value of each finding and any possible security breaches found. This information is then used to evaluate and develop an action plan and methods for the attack. *Vulnerability analysis* is a phase where the pentester attempts to discover any vulnerabilities in the target systems. When vulnerabilities are found, they can be taken advantage of in the *exploitation* phase. If a successful exploit is performed, it may lead to a *post-exploitation* phase. In that phase the tester is trying to find access to additional systems, harvest sensitive data and so on. In the *reporting* phase, the findings are summarized and reported to the client. (Weidman. 2013, 2.)

3.1 Pre-engagement

Before the testing itself begins, it is important that both parties (pentester and organization) meet and discuss the methods and the goals of the test. This eliminates all the misunderstandings and puts both parties on the same page about the testing process. As a pentester, one should ask questions about client’s business, client’s worries and expectations. Does client expect a simple vulnerability scan or actual penetration testing which is more intrusive than just a scan. Does client have any fragile systems that pentester needs to be careful with, when testing the systems? What matters most to the client? What causes most damage to client’s reputation, financial losses or valuable information losses? (Weidman. 2013, 3.)

3.1.1 Scoping

When the scope of the pentesting is agreed with the organization, there are couple things that have to be taken into consideration. What will be the scope of IP addresses and hosts when the pentest is performed? Is the pentester allowed to execute exploitation when possible? Should the test be only about detecting possible vulnerabilities? Injecting exploitation may bring down a service or some other critical system (e.g. domain controller). The client has to understand that even a simple port scan may bring a server or router down. (Weidman. 2013, 3.)

Prior to engaging the testing phase, the pentester and organization have to agree on the rules of engagement. Here are some examples of considerations that may be included in the rules of engagement:

- When is the pentester allowed to execute testing?
- If the organization has legacy systems that have known issues with automated scanning, this has to take into consideration.
- Who will be the contact person if the pentester encounters issues during the engagement?
- Does the entity (client, organization or a department inside the organization) want any updates regarding the ongoing exploitation during the test? The entity may want to implement its own countermeasures or response plan during an exploit.
- If there are any security controls (e.g. anti-virus programs, firewalls) detecting or preventing the testing, consider whether these should be disabled or configured so as to not interfere with the testing process.
- If the testing reveals sensitive data (passwords, secret documents), does the entity want them to be disclosed in the documentation.
- If the pentester utilizes personal equipment, what steps must be taken to ensure the safety of the equipment (updates, service packs, etc.) inside the entity network.
- Does the pentester use external connections or IP addresses during the testing phase? Do they need to be provided by the organization?
- What happens to the sensitive data obtained? How it is secured?
- If a previous or active compromise of systems is discovered, what are the actions? (E.g. activate response procedures to stop the ongoing malicious act and stop the penetration test until the compromised situation is resolved).

(Penetration Test Guidance Special Interest Group 2015, 12.)

3.1.2 Documentation

The pentester should have access to all detailed documents of any components within the scope of the test. It should include at least some kind of a network diagram, information about the services, servers and other network equipment that are in the scope of the test. If the penetration test is supposed to be launched as a black box penetration test, the documentation is not relevant. (Penetration Test Guidance Special Interest Group 2015, 11.)

3.1.3 Third-Party-Hosted / Cloud environments

When the organization has a hosted environment in a cloud, some additional precautions have to be in place. The organization needs a written approval from a third party (i.e., hosting provider, etc.), so the penetration testing can be executed. The scope of the penetration testing has to exclude other infrastructure than the provided network to the organization from the third party service provider. (Penetration Test Guidance Special Interest Group 2015, 12.)

3.1.4 Success criteria

When determining the success criteria, the pentester and the organization have to agree, how far can the attacker (pentester) penetrate into the environment. If there is not an agreed point at which the penetration test is considered complete, then there is a risk of exceeding the expectations and boundaries when the client considers the pentest to be over. (Penetration Test Guidance Special Interest Group 2015, 13.)

3.2 Information gathering

Information-gathering phase is the next phase after pre-engagement. In this phase the pentester gathers information and analyzes freely available sources of information. The process is known as *open source intelligence* (OSINT). This is when the pentester begins to use different tools such as port scanners to map the network and gain understanding from the internal and/or external network. In this phase the pentester will explore also the different software found from the network. (Weidman. 2013, 4.)

3.3 Threat modelling

Threat modelling is based on the information gathered in the information-gathering phase. In this phase the pentester thinks like an attacker. The plans of attack will be developed as will be the strategies on how to penetrate the organization systems. (Weidman. 2013, 4.)

3.4 Vulnerability analysis

In the vulnerability Analysis phase, the pentester begins actively discovering different vulnerabilities. When vulnerabilities are found, the pentester can determine how successful different exploitation strategies will be. Vulnerability scanners use vulnerability databases and a series of active checks that try to determine available vulnerabilities in the client systems. If this stage fails, the risk of injecting wrong exploits grows and when exploits fail, they can crash services and set off intrusion-detection alerts. One other important tool that a pentester should have, is critical thinking. Manual analysis and result verification is an important part of vulnerability analysis. (Weidman. 2013, 4.)

3.5 Exploitation

Exploitation is a phase, where the pentester runs exploits against the vulnerabilities being found. Some of the vulnerabilities can be easy to exploit and run, e.g. a simple logging in to a system with a default password. (Weidman. 2013, 4.)

3.6 Post exploitation

In this phase, the pentester tries to dig a little deeper by gathering information e.g. searching for interesting files and trying to elevate the pentesters privileges. The pentester may try gaining access to additional systems via the exploited machine dumping password hashes, or use the exploited machine to successfully attack systems that were not previously available. (Weidman. 2013, 4.)

3.7 Reporting

In the final phase, the pentester makes a report to the organization. It should be provided in a meaningful way, telling what needs to be improved, how the pentester got in to the system, what was found in the organization network, how to fix the problems discovered, and so on. The report should include a so called *executive summary* and a *technical report*. (Weidman. 2013, 5.)

3.7.1 Executive summary

Executive summary offers a high-level overview of the penetration test. It is intended for the executives in charge of the organization’s security. It should include the following parts:

- **Background** - description of the test and its purpose. Defining the terminology being used in the pentesting.
- **Overall posture** - overall view of the effectiveness of the test. A description of the issues found and the general issues that affect the security of the organization e.g. are there systematic patch delivery plans.
- **Risk profile** - What is the overall stage of security in the organization compared to other similar organizations? It could be measured as high, moderate, or low, but there should also be an explanation of the ranking included.
- **General findings** - a synopsis of the findings discovered. There could be some statistics and metrics included on the effectiveness of possible countermeasures being deployed.
- **Recommendation summary** - a summary of the tasks required to remediate the issues found in the pentest, as a high-level overview.
- **Strategic read map** - short- and long-term goals for the organization on how to improve their security posture. For example, the pentester might address issues and fixes with certain patches for a short-term solution, but in the long term the patches get old without a solid plan on how to manage the future patch releases.

(Weidman. 2013, 5.)

3.7.2 Technical report

The technical report that will be delivered to the organization offers the technical details from the penetration test. It should include the following details:

- **Introduction** - detailed inventory of scope, contacts and so on.
- **Information gathering** - details from the information gathering phase, especially the organization Internet footprint.
- **Vulnerability assessment** - details from the vulnerability assessment phase of the penetration test.
- **Exploitation/vulnerability verification** - details from the exploitation phase of the penetration test.
- **Post exploitation** - details from the post exploitation phase of the penetration test.
- **Risk/exposure** - details from the risks discovered. If the identified vulnerabilities are exploited by an attacker, this section estimates the possible losses.
- **Conclusion** - this section gives the final overview of the test.

(Weidman. 2013, 5.)

4 PENETRATION TESTING TOOLS

There are several different testing tools available both open and closed source that a pentester can utilize. Open source testing tools are a good option to use and most of them are updated frequently. A pentester can perform professional pentesting with open source tools, so there is no need to use the closed source (also known as proprietary software) programs. Basically open source means, the software provided by open source license is publicly available to everybody. It also means that the source code can be modified or enhanced by anybody. Closed source programs are the author's property and the author is the only one that is legally allowed to modify or enhance the program. Open source does not automatically mean it is free software, it means that the source code is publicly available and it can be modified to suit the programmers needs and the modified software can then be distributed as a commercial software. When the open source code is used to build a commercial program, the author of the software is obligated to provide the source code publicly available to everybody. There are many levels of open source license types, but they are not covered here, because they do not belong to the scope of this thesis. (EDU.fi 2013, cited 12.3.2016.)

In this section, the author of this thesis introduces some tools used in the pentesting process. Tools are just introduced on a *good-to-know* basis because this thesis does not cover any practical testing. There are some tools that cover more ground and fall into more than one category, but the introduction of the tool will explain the purpose and how the tool works.

4.1 Kali Linux

Kali Linux is a debian-based Linux distribution and it is used for advanced penetration testing purposes. It can also be used as Security Auditing tool. Kali Linux is a fully functional operating system that includes over 600 penetration testing tools that are integrated to the system itself. Keeping the system updated is extremely easy and actually the developers of Kali Linux recommend that the tools will be updated by updating the entire system, not one program at a time. Kali Linux is so well rebuilt, that a pentester has all the tools needed to do a full penetration test, no other tools are necessary. Kali Linux is a free system which is developed, funded and maintained by Offensive Security. Offensive Security is a company focused on information security training.

With Kali Linux, a pentester is able to use similar tools and techniques for security testing as a black hat hacker would use for hacking the organization’s infrastructure. Kali Linux 2 has several tools included to aid in penetration testing and this thesis covers mostly tools available in Kali Linux 2. (Kali Linux Official Documentation 2016, cited 14.3.2016.)

4.2 Metasploit

Metasploit is available as a commercial program, Metasploit Pro and Express version. It is also available as a limited free Metasploit Community version and as a free Metasploit Framework version. The Pro version is a full version with a GUI (Graphical User Interface). Express version is for baseline penetration testing and comes with a GUI. The Community version is meant for students and small businesses, as it is free but with limited features but still equipped with a GUI. Framework version is a full set of tools and exploits but it runs only in a command line interface. Framework version is considered as “de-facto” standard for penetration testing. (RAPID7 2016, cited 22.3.2016.)

The Metasploit Framework (MSF) is included in the Kali Linux 2 and it is one among of the most used free auditing tool available to any security professionals around the world. It is not just a collection of exploits, it is an infrastructure that one can utilize for own custom needs. The Metasploit Framework provides a full set of tools from commercial grade exploits and exploit development environment, to web vulnerability plugins and network information gathering tools. (Offencive Security 2016, cited 29.3.2016.)

4.3 Information gathering and reconnaissance tools

Information gathering plays an important role in penetration testing. If we want to be able to launch an attack to the target organization, we need to gather some basic information about the target. Below is a short introduction from the tools and resources tested by author in the process of making this thesis.

4.3.1 Recon-NG

Recon-NG is included in Kali Linux 2, and it can be referred as Metasploit for information collection. It is a powerful tool that allows the pentester to perform automated information gathering and network reconnaissance. Recon-NG interface has been designed to have the same look and feel with Metasploit and the command use and functions are similar. Recon-NG runs from the command prompt. (Dieterle. 2014, 67.)

Recon-NG can automate a lot of the steps required for information gathering phase. It has copious features for user information collecting for social engineering attacks. It can be used to execute network mapping and information gathering. It can be used to automatically gather passive information on the target websites and even be used for active probing towards the data obtained from the target website. For example if we need to find out all the sub-domains from the target, we could use Google search engine to enumerate site sub-domains. We would have to use switches like *"site:"* and *"inurl:"* to find the sub-domains and then remove all sub-domains (*-inurl*) found in the search field to get other subdomains to appear. This is a slow process but it can be automatically executed with Recon-NG and the results will be recorded for a later view. (Dieterle. 2014, 69.)

Recon-NG can also be used in many other ways with different modules. Some require a program API key (application programming interface key) like Twitter for tweets, Shodan (search engine for Internet-connected devices), LinkedIn or Google. Using API key can give specific information from the sites that can be linked to the target. (Dieterle. 2014, 71.)

4.3.2 DMitry

DMitry is a whois lookup tool included in Kali Linux 2 that can be used for quickly finding out multiple pieces of information about a site. The tool can be used for example to gather sub-domain information, email addresses, whois lookups, tcp port scan and uptime information. The tool runs from command prompt. (Kali Tools 2014, Dmitry, Cited 29.3.2016.)

4.3.3 Netdiscover

Netdiscover is a tiny tool for network scan and it is included in Kali Linux 2. It runs from the command prompt, and it scans network devices for IP- and MAC addresses. If needed, it runs also in steath “passive” mode where it only sniffs traffic without sending any data out. (Dieterle. 2014, 72.)

4.3.4 Nmap and Zenmap

Nmap is included in Kali Linux 2. It runs from command prompt and it is a free open source utility. It can be used for security auditing and network discovery. It scans the network for hosts, services, operating systems, packet filters/firewalls and many other characteristics. Nmap can easily scan large networks or just one single host. It is a very powerful tool and it is well supported by a vibrant community that includes users and developers. Nmap package includes several tools that can be used for scanning and query purposes. (Kali Tools 2014, Nmap, Cited 29.3.2016.)

Zenmap is a graphical version of Nmap. It can be used for the same purposes as the command line utility. It may be easier to use for a beginner, but it is recommended to get accustomed to the command prompt, because it is much more diverse than the GUI version. (Kali Tools 2014, Nmap, Cited 29.3.2016.)

4.3.5 theHarvester

theHarvester is a command prompt tool made for external pentesting. It is a Python tool for quickly scour through the customer footprint on the internet. It can harvest information about emails, subdomains, hosts, employee names, open ports and banners using different public sources, for example different search engines, PGP, LinkedIn and Twitter. (Weidman. 2013, 118.)

4.3.6 Shodan

Shodan is a search engine for computers and other network devices. It is sometimes called the “*Hacker’s Google*” or “*Dark Google*”. With the right keywords and filters the pentester can quickly assess what systems on the organization’s network are publicly visible to all users. Using Shodan might be a good idea because there are a large number of important systems in the public network that really should not be out there. Shodan can find open, outdated and insecure systems, for example phone systems, network storage, routers, security cameras, building controls, open network printers and security systems. There are many systems available that are either open (no passwords), or are set up with a default password that can be easily found, or simple password that is easily cracked. While the world of IOT (Internet Of Things) is rapidly growing, at the same time the number of potential exploitation targets is rapidly growing too. (Dieterle. 2014, 76.)

Shodan searches can be also executed via Metasploit. One has to just sign up for a free Shodan user account and download API key from their website. With the API key, the pentester can automate Shodan searches directly from the Metasploit command prompt. (Dieterle. 2014, 84.)

4.3.7 Wireshark

Wireshark is a monitoring tool that allows the pentester to capture network traffic and analyze it. It has a GUI interface. With Wireshark the pentester can capture for example, wireless, Ethernet and Bluetooth traffic. Although monitoring traffic is useful, Wireshark can be used for other purposes too. It can be used to perform a so-called *man-in-the-middle* attack by doing Address Resolution Protocol (ARP) poisoning also known as ARP spoofing. ARP is a protocol that resolves IP addresses to MAC addresses for traffic routing purposes. The pentester uses Wireshark to trick the target machine or switch into believing that the traffic belongs to the machine the pentester is using. By doing so, all the traffic intended to go to the target, redirects via Wireshark allowing pentester to capture traffic that is not supposed to interact with the pentesters system. (Weidman. 2013, 156.)

4.4 Attacking and exploiting tools

Kali Linux 2 includes various attacking tools. Some tools can do more than just reconnaissance or inject exploits etc. Tools like Metasploit and Wireshark can be used in information gathering and after that move forward to inject exploits or execute other types of attacks as discussed below. In this section there is a short introduction from tools other than Metasploit. Arpspoof and Dnsspoof are not used while executing operation system pentesting, but they are good tools that deserve to be mentioned.

4.4.1 Arpspoof

Arpspoof is a tool used for ARP cache poisoning. It is a command prompt-based tool that is easy to use. Arpspoof can be used to execute a man-in-the-middle attack between 2 devices or it can be used as impersonating the default gateway while all the network traffic intended to go to internet, flows through the pentesters system. The traffic flowing through the bogus gateway can then be easily captured with Wireshark. When using such a blatant method for capturing network traffic, one has to be very careful, as using a Kali Linux laptop/desktop (or even worse, virtual machine) as an organization gateway, it can suffocate the network bandwidth and inflict network issues. (Weidman. 2013, 164.)

When executing ARP spoofing to trick the organization network thinking the pentesters machine is the gateway, it is very important to remember to turn on IP forwarding. When IP forwarding is turned on, the pentester tells Kali Linux machine to redirect any irrelevant packets received, to their proper destination. If this phase is not performed, the pentester creates a *denial-of-service* (DoS) condition to the organization network, where the legitimate clients are unable to access required services. (Weidman. 2013, 163.)

4.4.2 Dnsspoof

DNS resolves domain names (such as www.gmail.com) to IP addresses to route network traffic in the internet. DNS (Domain Name Service) cache poisoning can be executed in the same way as ARP spoofing. The pentester sends a bunch of bogus DNS resolution replies that point to the wrong IP addresses. When Arpspoof is used to execute a man-in-the-middle attack in the Kali Linux device, making the Kali Linux device act as a gateway, the DNS cache poisoning can be injected, forcing the end users to navigate to Kali Linux web server impersonating, as for example, a gmail web site. Dnsspoof can be used via the command prompt. (Weidman. 2013, 164.)

4.4.3 Msfvenom

Msfvenom is a standalone command prompt Metasploit payload generator that can be used to create shellcode payloads. Shellcode is a piece of code that can be executed in the target machine and the target machine then creates a remote shell back to the shellcode creator. This type of exploitation is usually executed via social engineering attacks, where the hacker booby-traps a file and then the end user receives, for example, a specifically targeted e-mail with the malicious file attached to the e-mail. Another way to distribute the malicious shellcode can be a legitimate program where the shellcode is embedded. When the program is installed or run, the shellcode creates a backdoor to the end user computer giving the hacker a remote access or control over the exploited machine. In the new Kali Linux 2 Metasploit tool, Msfvenom utility replaces the Msfpayload and Msfencode utilities. (Dieterle. 2015, 54.)

4.4.4 Shellter

Shellter is a command prompt tool used for obfuscating a legit Windows .exe file. Shellter can add shellcode to the file modifying it in a way it will be ignored by antivirus programs. This tool is intended for pentesters not hackers, so it does not retain the integrity of the .exe file making it inoperable; the resultant .exe package with the shell embedded to the file works great. Using Shellter is very easy because of its automatic mode that makes the process painless. In Shellter version 4.0 there is an increased obfuscation through a custom encoder and polymorphic decoder, making AV programs capability to detect malicious embedded code very hard, almost impossible. (Dieterle. 2015, 89.)

4.4.5 Windows Privilege Escalation by Bypassing UAC with Metasploit

User Access Control (UAC) is a security feature that was introduced in Windows 7 operating system and also the feature is integrated to newer versions of Windows. It can be a drawback to a possible attacker trying to penetrate the organization systems, but when right conditions are met, the UAC can be easily bypassed.

In Windows as in Linux systems there is a superuser that has unrestricted access to the system. In Linux the superuser is “Root” and in Windows the superuser is the user “System”. If the attacker can access the superuser account, the possibilities for information gathering and exploitation against the hacked operating system are limitless. If the attacker gets a remote administrator rights to the system, UAC prevents, for example, obtaining the password hashes. Metasploit has a built in UAC bypass module that allows the pentester (and attacker) to bypass the UAC giving a system level access to the operating system. This means that the UAC can be easily bypassed if the attacker can access any account with admin rights, making the UAC futile as a defense mechanism against possible attacks. (Dieterle. 2014, 122.)

Many companies have given local administrator rights to their employees without ever thinking of the possible risks it involves. That being said, there are companies that have even disabled the UAC feature thinking it is just a public nuisance and disturbs users. Both actions are potential security risks and they should be considered thoroughly before adopting.

4.5 Post exploitation tools

After pentester has gained access to the target machine via command line, Meterpreter or PowerShell or by any other means possible, the pentester starts the post exploitation phase. This is when they try to get a foothold of the system so that it can be accessed and exploited to gather intelligence or execute tools to get access even deeper to the organization's systems.

4.5.1 Metasploit & PowerShell

Metasploit exploitation with Meterpreter PowerShell connection enables a remote connection to the target with PowerShell interface up and running. The PowerShell code or script can also run from the regular Meterpreter shell but every command will not run correctly. A second limitation is the PowerShell execution policy that disables running the scripts by default. The execution policy can be bypassed simply by typing "**powershell.exe –executionpolicy bypass**" –command to the front of the PowerShell code that is executed. The solution for ensuring the code works, is to encrypt the code. The encryption is made from a .txt file and the file can contain a full PowerShell script, which is encrypted to series of random text. The output can be then copied from the command line and pasted behind the execution policy bypass command. After this the PowerShell code is ready to be executed with a high success rate. With a working PowerShell remote connection to the target machine, the pentester can accomplish many tasks. The pentester could dig system recources, copy information, examine network connections and so on. The pentester could execute a custom pop-up window to the target machine, letting the user know that the machine has been exploited. If the intruder would be a criminal with malicious intentions, the results could be more devastating, because in the wrong hands, Windows PowerShell is a powerful tool that can cause a lot of harm. (Dieterle. 2015, 105.)

4.5.2 PowerSploit – PowerShell Payloads

PowerSploit is a collection of security-related modules and functions written in PowerShell used for security testing purposes. The modules are used in command prompt. Using PowerShell scripts is a good idea, because when the scripts are executed in the target machine, they usually never touch the target machines hard drive (unless the actual scripts are downloaded to the target machines hard drive). PowerShell scripts can avoid Anti-Virus software very efficiently, because the scripts usually run in windows service contexts like the PowerShell service, allowing the PowerShell script to bypass the Anti-Virus scanner undetected. The PowerSploit scripts come pre-installed in Kali in the “**/usr/share/powersploit**” directory. (Dieterle. 2015, 121.)

4.5.3 Metasploit Meterpreter Post Modules

Meterpreter has many Post modules, that are add-on Ruby scripts that can be executed via command line interface. They can be run after the pentester gets a successful Meterpreter shell connection, giving the pentester effective tools to manipulate a compromised system for example to recover data and account credentials. One example would be the use of “**firefox_creds.rb**” module. This module gives the pentester the possibility to pull Firefox internet browser credential information to the **.bin** database files that can be opened in Kali Linux SQLite for viewing all passwords saved in the Firefox browser. (Dieterle. 2015, 96.)

4.5.4 IRB Railgun

Railgun extends Meterpreter even further. It is a command prompt tool and using Railgun with success, gives the pentester full control of Windows API (Application Programming Interface). Windows API is an interface that allows programs to interact with Windows. Railgun allows the pentester to load DLLs, and after that remotely call Windows functions against the operating system. One example would be password decryption from a certain program. It can be done by grepping sensitive information from memory or calling the DLL that is responsible for the program’s password decryption and execute the decryption via its own decryption engine. (Github 2014, Railgun, cited 15.4.2016.)

4.6 Tools for maintaining access

After a successful remote shell connection to the target, the pentester wants to be able to connect back to the exploited machine (if the rules of engagement permit maintaining access to target). That is what the possible attacker would do, so in this section the author reviews a couple methods for connecting back to the target at a later time. Gaining a persistent access is more than just creating a backdoor access to the exploited machine. Persistence means many options and basically it is executed by whatever means necessary to gain access to the target machine at a later time. There are multiple reasons for maintaining access to a target machine. For example the attacker would want to add new users, modify the user permissions, enable a service, create a new share or create a backdoor access. (Dieterle. 2015, 134.)

4.6.1 Metasploit Meterpreter “Persistence” Script

The built in Meterpreter “*Persistence*” module is one of the most common ways used when the pentester wants to maintain access to the target machine. For the module to work, the pentester has to have active meterpreter session to the target machine. The session has to be open with administrator privileges that are then elevated to system level with the “*bypassuac*” module. After these conditions are met, the “*Persistance*” module can be executed. Depending on the settings, the module injects a PowerShell script to the target machine, commanding the machine to connect back to the source with a fixed interval. If the target machine is restarted or shut down, the script executes again when the target machine is up and running. The “Persistance” module can be remotely removed when it is no longer needed. (Dieterle. 2015, 134.)

4.6.2 Metasploit S4u_persistence – Scheduled Persistence

This Metasploid module runs as a scheduled task. It can also be tagged to different system events, for example workstation lock, logon, logoff and so on. “S4u_*persistence*” module requires active meterpreter session and payload to work correctly. (Dieterle. 2015, 141.)

4.6.3 Metasploit Vss_Persistence – Volume Shadow Copy Persistence

The “Vss_Persistence” module uses Windows Volume Shadow copy service for creating a persistent backdoor access to the exploited machine. This exploit needs elevated system access to run without problems. (Dieterle. 2015, 142.)

4.6.4 Netcat Backdoor

Netcat is a Windows executable file that has to be uploaded to the target machine system32 folder. For Netcat to work, the pentester needs an elevated session open in Meterpreter (system level). After that the program details are added to the target machines registry and then the program needs to be remotely started. After the target machine is rebooted (it can be done via meterpreter shell), the program should enable connection to be established from Kali Linux with nc program. (Dieterle. 2015, 143.)

REFERENCES

Dieterle, D, W. 2014. Basic Security Testing With Kali Linux. San Bernardino, CA: Cyberarms.

Dieterle, D, W. 2015. Intermediate Security Testing With Kali Linux. Lexington, KY: Cyberarms.

EDU.fi, 19.9.2013. Avoimen lähdekoodin määritelmä. Cited 12.3.2016, http://www.edu.fi/valo_opas/avoin_lahdekoodi_maaritelma/.

Github, 2014. How to use Railgun for Windows post exploitation. Cited 15.4.2016, https://github.com/rapid7/metasploit-framework/wiki/How-to-use-Railgun-for-Windows-post-exploitation.

Kali Linux Official Documentation, 2016. What is Kali Linux. Cited 14.3.2016, http://docs.kali.org/introduction/what-is-kali-linux/.

Kali Tools, 2014. DMitry Package Description. Cited 29.3.2016, http://tools.kali.org/information-gathering/dmitry/.

Kali Tools, 2014. Nmap Package Description. Cited 29.3.2016, http://tools.kali.org/information-gathering/nmap/.

Offencive Security, 2016. Introduction to Metasploit, Metasploit Unleashed. Cited 29.3.2016, http://docs.kali.org/introduction/what-is-kali-linux/.

Penetration Test Guidance Special Interest Group 2015. Information Supplement: Penetration Testing Guidance. PCI Security Standards Council. Cited 1.3.2016, https://www.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf/.

Pritchett, W, L. & De Smet, D. 2013. Kali Linux Cookbook. Birmingham: Packt Publishing Ltd.

RAPID7, 2016. Metasploit: Penetration testing software editions. Cited 22.3.2016, https://www.rapid7.com/products/metasploit/editions.jsp/.

Secforce Ltd blog, 2016. Black box penetration testing vs white box penetration testing. Cited 1.3.2016, https://www.secforce.com/blog/2008/11/black-box-penetration-testing-vs-white-box-penetration-testing/.

Singh, A. 2013. Instant Kali Linux. Birmingham: Packt Publishing Ltd.

Weidman, G. 2014. Penetration testing: a hands-on introduction to hacking. San Francisco: No Starch Press, Inc.