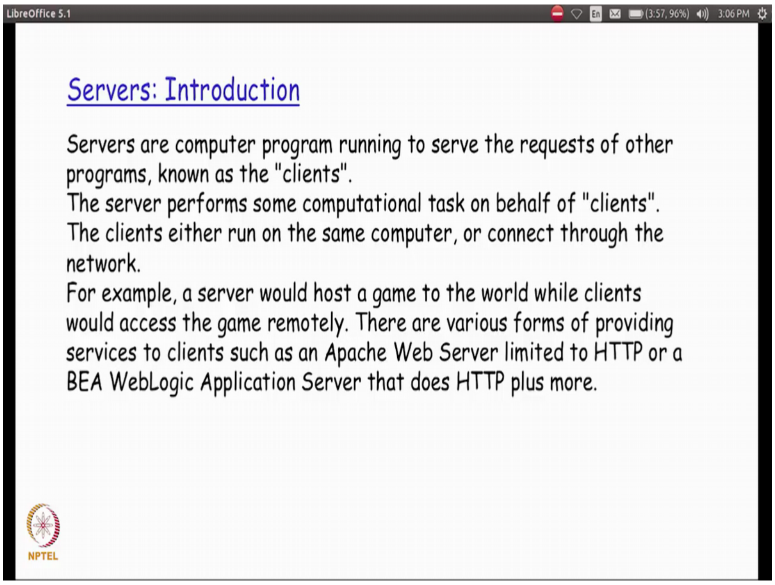
Information security - IV
Prof. Vasan
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module – 17
Lecture - 17
Service Side Attacks Tools in Kali Linux

Starting from this session onwards we are going to look at a there the different kind of tools that are available in Kali Linux for doing a server-side attack by a penetration tester. Till the previous session we actually had been looking at the different a type of techniques that other penetration tester would typically do, as part of the initial reconnaissance action.

Once having this detail from the reconnaissance step on whatever target that is actually trying to attack. The penetration tester would actually go ahead and try to employee the next set of tools if the target is going to be a server. So, from the session on words for the next few sessions, were going to look at the different type of tools that are typically available for the penetration tester to make use of when the target that is actually expecting to attack and bring it down is a server.

(Refer Slide Time: 01:11)



Servers: Introduction

Servers are computer program running to serve the requests of other programs, known as the "clients".
The server performs some computational task on behalf of "clients".
The clients either run on the same computer, or connect through the network.
For example, a server would host a game to the world while clients would access the game remotely. There are various forms of providing services to clients such as an Apache Web Server limited to HTTP or a BEA WebLogic Application Server that does HTTP plus more.

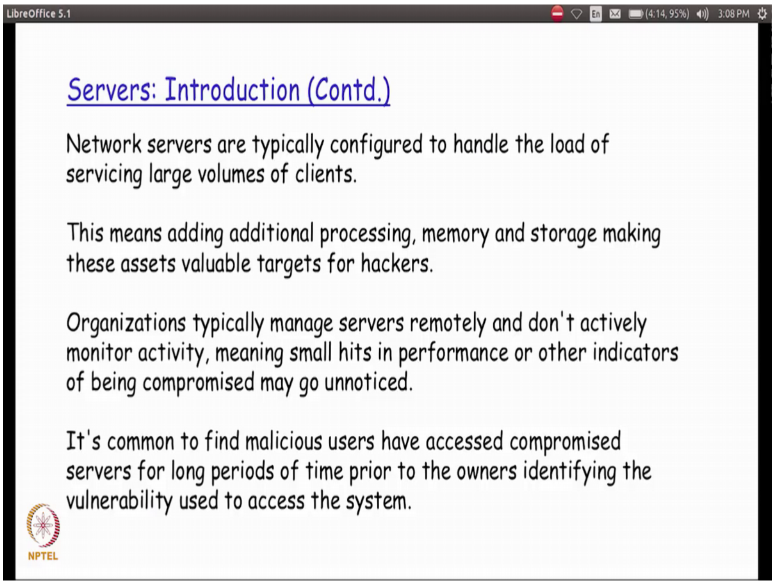
As we all know by now, our servers are basically systems are computer programs that are actually running, waiting for request to come in from the clients right.

So, the clients are the end nodes that will typically initiate some kind of a severs request to be done by sever. And by the servers are expected to be always brought up first as compared to clients. And then the server will basically perform whatever task is actually been submitted to it by the client. Typically, the clients and servers although technically they could actually be running on the same system for all practical purposes you would find them actually running on two different systems connected over a network. So, the server for example, code post the messaging application or it could actually host a game to which the clients from all over the internet would possibly access remotely and try to make use of the particular server application.

So, there are different forms and mechanism by which the clients can talk to the server and one of the most common base for triggering any kind of a server application is so, what is called as a web server. Among the web server is there are different web servers are very popularly used in a internet today and a few of the very popular ones is apache web server or we web logic application a server. Or it could even be something like the internet information server from Microsoft and so on and so forth so.

All these web servers typically wait for a request coming in from any of the clients on the internet. Especially, the client that are actually authorised to connect to this particular server as, soon as the request comes in they basically have the job done on the server side and then respond back to the client appropriately.

(Refer Slide Time: 03:13)



Servers: Introduction (Contd.)

Network servers are typically configured to handle the load of servicing large volumes of clients.

This means adding additional processing, memory and storage making these assets valuable targets for hackers.

Organizations typically manage servers remotely and don't actively monitor activity, meaning small hits in performance or other indicators of being compromised may go unnoticed.

It's common to find malicious users have accessed compromised servers for long periods of time prior to the owners identifying the vulnerability used to access the system.

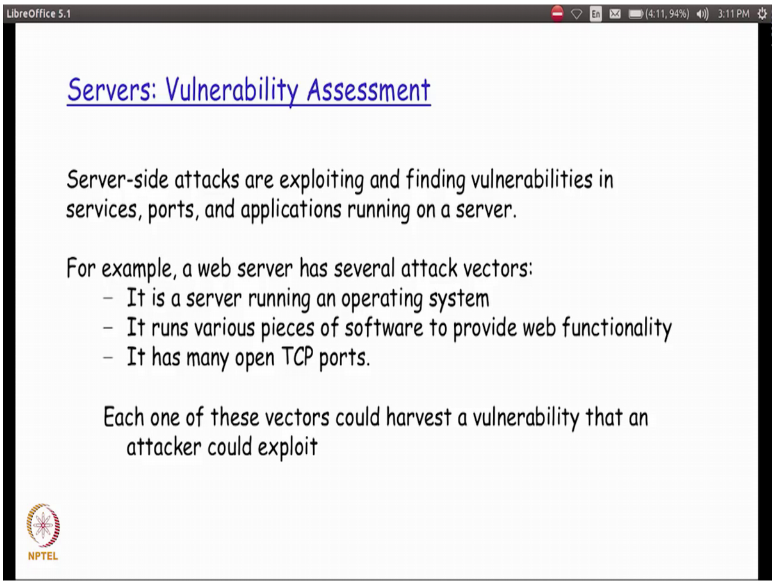
So, servers a could be a web server it could be a mail sever, it could be a file sever, and so on. And so, forth are there are there are potentially lot of applications that I could actually run as a server making the services of that available to all the clients across the internet. So, because depending on the number of clients a server could potentially be contactor where a the server configuration from the hardware perspective we will basically see that it has a different loads. So, I could actually have a server which is actually going to do lot of processing capability; so the applications that is going to run the server. For example, is going to actually do lot of computation which essentially means that it requires lot of processing power.

So, I will actually have the server configured with higher processing capacity and I if at all the application is actually going to be consuming lot of memory then the server will actually have much more memory configured and made available on that, because otherwise the applications are going to be failing; so likewise on the storage aspect also.

Now, because the fact that the service typically today is internet are very, very high and servers having lots of information available with at these, basically become a very, very good entry point that a penetration tester will actually try to make use of because more the amount of data. For example, or more amount of prosinaries done in a particular server the penetration tester will think that he actually has a better chance of getting some very useful data out of this particular system. And thereby this start this system this server system would actually be targeted by the tester to sort of a successfully penetrate.

So, because of the benefits penetration tester would get by a trying to attack this kind of a system will always find that the server which are compromise. That is basically already successfully a penetrated by the tester for very long periods of time are left as it is. Before, the owners basically even get to understand or realise that a particular server system is actually been compromised.

(Refer Slide Time: 05:40)



Servers: Vulnerability Assessment

Server-side attacks are exploiting and finding vulnerabilities in services, ports, and applications running on a server.

For example, a web server has several attack vectors:

- It is a server running an operating system
- It runs various pieces of software to provide web functionality
- It has many open TCP ports.

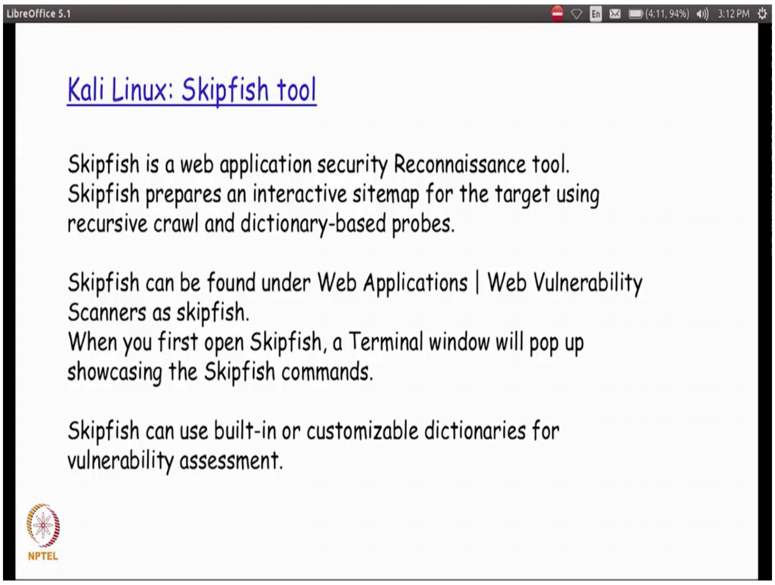
Each one of these vectors could harvest a vulnerability that an attacker could exploit

What could be the different kinds of vulnerabilities that are there typically on a server. So, for example, if you take a web server the attacker would possibly concentrate from different perspectives as we will see with few of the tools down below in some of our are sessions or. So, it could actually be a server with is running in operating system.

So, the moment I know what operating system is running, I could potentially know what are all the different a attack points that I can use on that particular operating system. So, for providing the a web functionality a to be accessible as a server a depending on the technologies that is used, application that is used if I can get access to that list I will basically be able to find out possible vulnerabilities that are there and then if I call. So, get to know the different type of TCP ports are the right now open again based on the TCP port numbers the attacker would basically come to know on what he could potentially leverage on to get access into the system.

So, a each of these a different types of vectors a would help possibly the penetration tester to a penetrate the system successfully.

(Refer Slide Time: 06:55)



Kali Linux: Skipfish tool

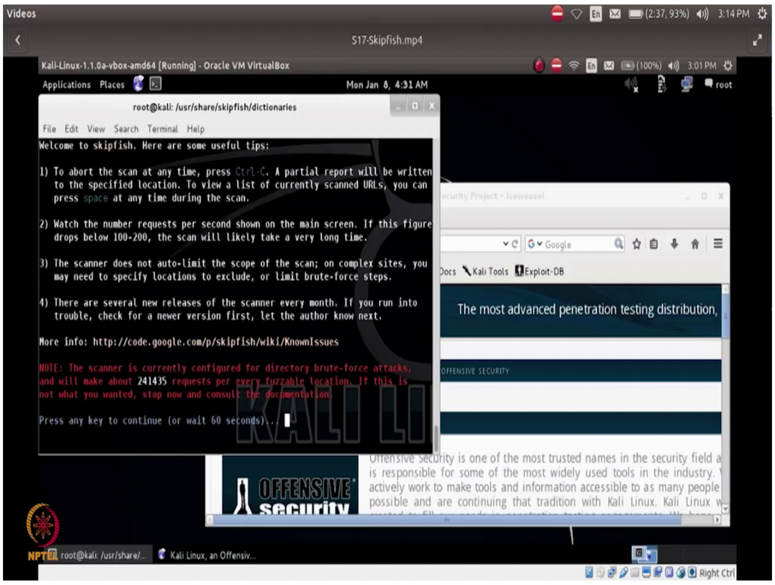
Skipfish is a web application security Reconnaissance tool. Skipfish prepares an interactive sitemap for the target using recursive crawl and dictionary-based probes.

Skipfish can be found under Web Applications | Web Vulnerability Scanners as skipfish.
When you first open Skipfish, a Terminal window will pop up showcasing the Skipfish commands.

Skipfish can use built-in or customizable dictionaries for vulnerability assessment.

Now, there are few tools that we are going to start looking at. So, one of the first tools is what is called the Skipfish tool. So, this is basically an web application security reconnaissance tool. So, this tool basically actually tries to get a complete sitemap for the target using what is called is recursive crawl a techniques. Where and looking at the initial index page it tries to find out what are all the different subpages that are actually possible and available in the tool in that particular portal and then try to get access to the contents of those individual pages. So, with a very small demo let us take a look at how this particular Skipfish tool is actually working.

(Refer Slide Time: 07:41)



Videos
S17-Skipfish.mp4
Kali-Linux-1.1.0a-vbox-amd64 [Running] - Oracle VM VirtualBox
Applications Places
Mon Jan 8, 4:31 AM
root@kali: /usr/share/skipfish/dictionaries
File Edit View Search Terminal Help

```
Welcome to skipfish. Here are some useful tips:

1) To abort the scan at any time, press Ctrl-C. A partial report will be written
   to the specified location. To view a list of currently scanned URLs, you can
   press space at any time during the scan.

2) Watch the number requests per second shown on the main screen. If this figure
   drops below 100-200, the scan will likely take a very long time.

3) The scanner does not auto-limit the scope of the scan; on complex sites, you
   may need to specify locations to exclude, or limit brute-force steps.

4) There are several new releases of the scanner every month. If you run into
   trouble, check for a newer version first, let the author know next.

More info: http://code.google.com/p/skipfish/wiki/KnownIssues

NOTE: The scanner is currently configured for directory brute-force attacks,
and will make about 241435 requests per every fuzzable location. If this is
not what you wanted, stop now and consult the documentation.

Press any key to continue (or wait 60 seconds)...
```

The most advanced penetration testing distribution,
Offensive Security is one of the most trusted names in the security field a
is responsible for some of the most widely used tools in the industry.
actively work to make tools and information accessible to as many people
possible and are continuing that tradition with Kali Linux. Kali Linux w

So, Skipfish this comes with the a default set of dictionaries available with it and if you really look at there are different dictionaries a like complete what less extensions, only medium, minimal and so on. So, depending on whatever is the file that we are actually going to be taking up here and giving as an argument to Skipfish. The amount of time to the Skipfish will take to run to completion will become difference of for example, if you actually select the word is file complete audible because it basically contains this dictionary contains a lot of words.

It will take more amount of time to complete the com[complete] the tool from grading process. So, one of the ways by which I could actually specify Skipfish is to run the tool from the command line by saying Skipfish with the option of what dictionary file to be given. Where should command be output finally, and what is the URL on which this particular details is required. So, based on all these details it actually starts running and then it tells you initially what is the kind of the initial help that we required for the user to understand.

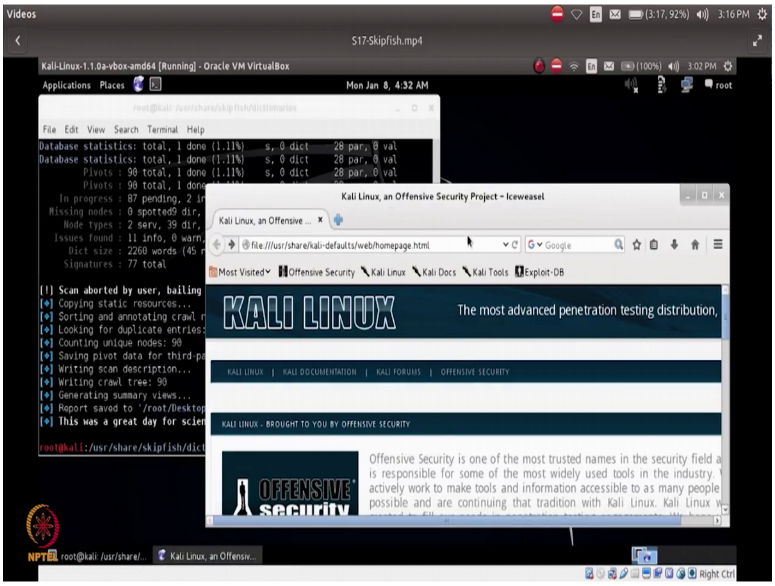
So, once the person is actually going through the user has actually gone through this set of initial details is provided. The user can basically go head in press any key or automatically at the end of 60 seconds the tool will actually start running. So, if you see the output here it basically tries to do the scanning of this particular URL that we have given. So, we have actually specify at the URL as www dot Facebook dot com and it basically goes ahead and tries to do different kinds of scans and tries to finds out the different faults that could possibly be there like HTTP fault TCP fault and so on and finally, it tries to accumulate the statistics.

So, it will basically go on for whatever amount of time it takes to complete or alternatively the user can also press control C to sort of abort this particular tool run, after a certain period of time. If the user believes that he would have actually got all the details by this period of by this period of time right. So, the tool is right now running continuously getting all the details and then it basically tells at the end of it if it is actually been aborted by the user it tells that this scan has been aborted by the user.

So, if you believe that it is actually taking too much time and whatever basic set of information that as a user we would have already got the user can go ahead and actually go ahead go ahead and press control C to abort the running of the tool also.

So, the end of it basically tells where is the place the output final output is going to be present and it is basically an index chart html file that recreates

(Refer Slide Time: 10:39)



```
Database statistics: total, 1 done (1.11%)    s, 0 dict    28 par, 0 val
Database statistics: total, 1 done (1.11%)    s, 0 dict    28 par, 0 val
          Pivots : 90 total, 1 done (1.11%)    s, 0 dict    28 par, 0 val
          Pivots : 90 total, 1 done
     In progress : 87 pending, 2 in
   Missing nodes : 0 spotted9 dir,
      Node types : 2 serv, 39 dir,
    Issues found : 11 info, 0 warn,
       Dict size : 2260 words (45 r
      Signatures : 77 total

[!] Scan aborted by user, bailing
[+] Copying static resources...
[+] Sorting and annotating crawl r
[+] Looking for duplicate entries:
[+] Counting unique nodes: 90
[+] Saving pivot data for third-pa
[+] Writing scan description...
[+] Writing crawl tree: 90
[+] Generating summary views...
[+] Report saved to '/root/Desktop
[+] This was a great day for scien

root@kali:/usr/share/skipfish/dict
```

Kali Linux, an Offensive Security Project - Iceweasel

file:///usr/share/kali-defaults/web/homepage.html

KALI LINUX The most advanced penetration testing distribution,

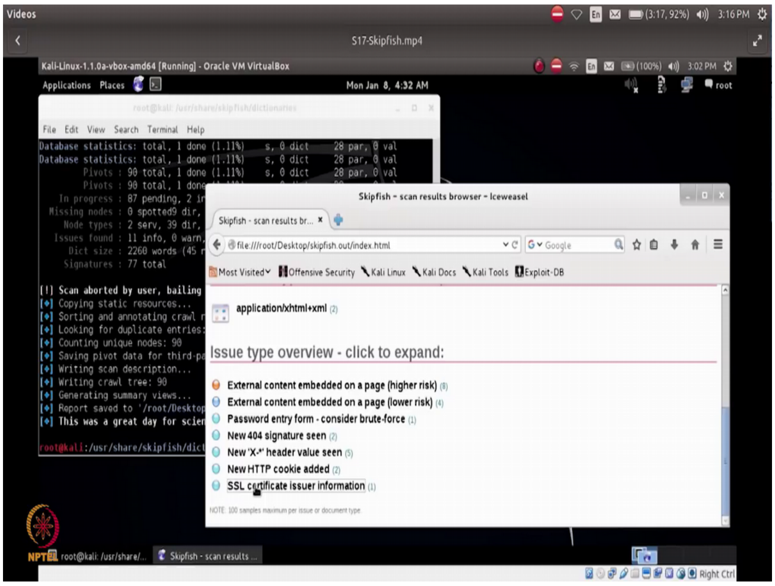
KALI LINUX - BROUGHT TO YOU BY OFFENSIVE SECURITY

Offensive Security is one of the most trusted names in the security field a is responsible for some of the most widely used tools in the industry. actively work to make tools and information accessible to as many people possible and are continuing that tradition with Kali Linux. Kali Linux w

Videos
S17-Skipfish.mp4
Kali-Linux-1.1.0a-vbox-amd64 [Running] - Oracle VM VirtualBox
Applications Places
Mon Jan 8, 4:32 AM
File Edit View Search Terminal Help
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB
KALI LINUX | KALI DOCUMENTATION | KALI FORUMS | OFFENSIVE SECURITY
OFFENSIVE security
NPTEL
root@kali: /usr/share/...
Kali Linux, an Offensiv...
Right Ctrl

And that file you could potentially open it in your browser. If you see here, it basically has all the details of whatever it could gather.

(Refer Slide Time: 10:50)



Skipfish - scan results browser - Iceweasel

file:///root/Desktop/skipfish.out/index.html

application/xhtml+xml (2)

Issue type overview - click to expand:

- External content embedded on a page (higher risk) (8)
- External content embedded on a page (lower risk) (4)
- Password entry form - consider brute-force (1)
- New 404 signature seen (2)
- New 'X-*' header value seen (5)
- New HTTP cookie added (2)
- SSL certificate issuer information (1)

NOTE: 100 samples maximum per issue or document type.

Videos
S17-Skipfish.mp4
Kali-Linux-1.1.0a-vbox-amd64 [Running] - Oracle VM VirtualBox
Applications Places
Mon Jan 8, 4:32 AM
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB
NPTEL
root@kali: /usr/share/...
Skipfish - scan results ...
Right Ctrl

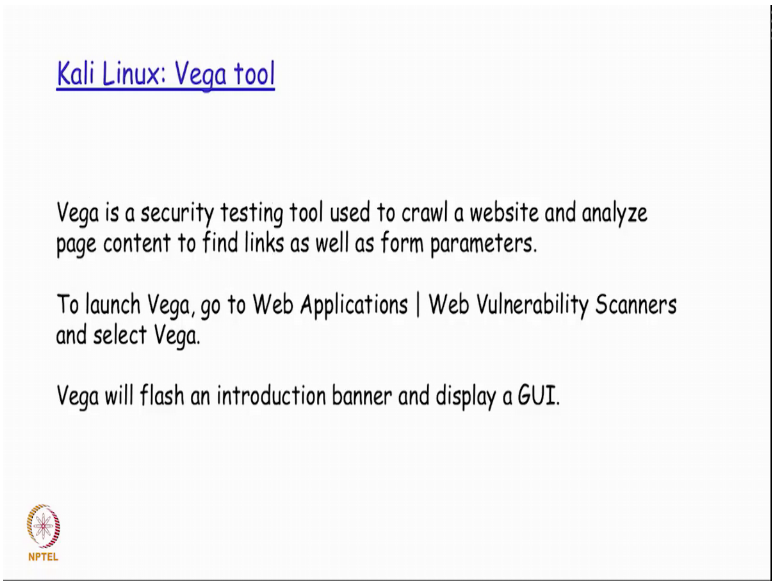
From the scanned run that was actually happening. So, it basically tells; what are the different possible problems that it is actually found out. Each of this could be actually real issue or it could be just a warning for the user to take care of to address that particular issue right.

So, password entry form if you look at it for example, it is a considered brute force. So, it basically gives an idea that there is a password entry form that is here. And I could potentially considered brute force mechanism to sort of use this form. And using the brute force technique that is actually available to find out what is the login; and the password combination that will successfully enable the penetration tester to get inside the particular site right.

So, we are going to be looking at for example, different kind of brute force technique send the different tools that are there typically for logging in by using a password, from pre-established dictionary files. So, there are lot of tools and actually available for that which we will be seeing in the subsequent session.

So, likewise it basically goes and print the different kinds of possibly issues that are there which the user can basically go ahead and try to look at and see as a penetration tester if any of that could actually be made use of. So, for example, here I have the details of what are all the different response and request message headers of HTTP from, which I could actually get quite a few details possibly. So, the next tool that we are going to be actually looking at is what is called as a Vega?

(Refer Slide Time: 12:22)



Kali Linux: Vega tool

Vega is a security testing tool used to crawl a website and analyze page content to find links as well as form parameters.

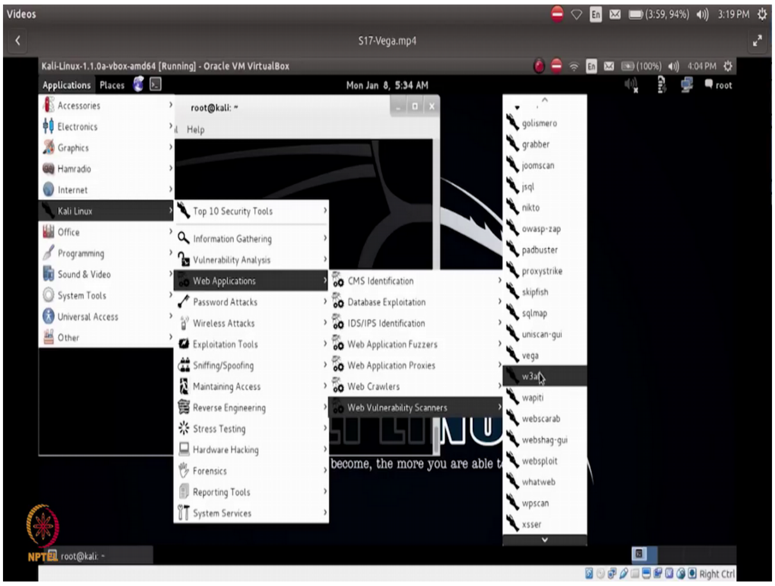
To launch Vega, go to Web Applications | Web Vulnerability Scanners and select Vega.

Vega will flash an introduction banner and display a GUI.

So, Vega is also a basically tool that is actually available in Kali Linux under the web applications banner and it is that typically a security testing tool again which is used to crawl a website and sort of analyse the page content as well as a possible form parameters.

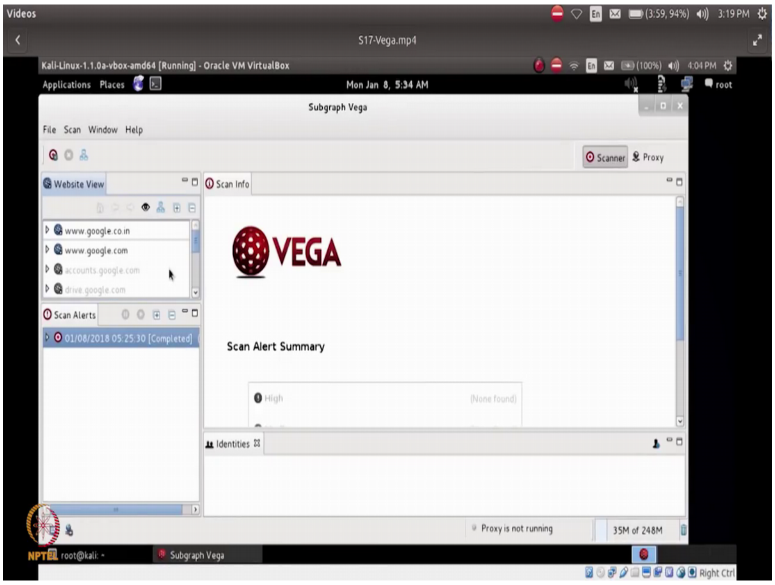
So, if at all it could actually gather details about the form parameters it will basically be able to display that as well. So, let us see with the very quick demo of how this particular tool actually works. So, this is actually a tool that is actually available under the Kali Linux menu option under the web applications banner.

(Refer Slide Time: 13:00)



So, under web applications you have a web vulnerability scanners in which this particular tool Vega is actually available for us to just click on it and then start the application. So, once application starts it opens up with the banner and displays me the opening menu.

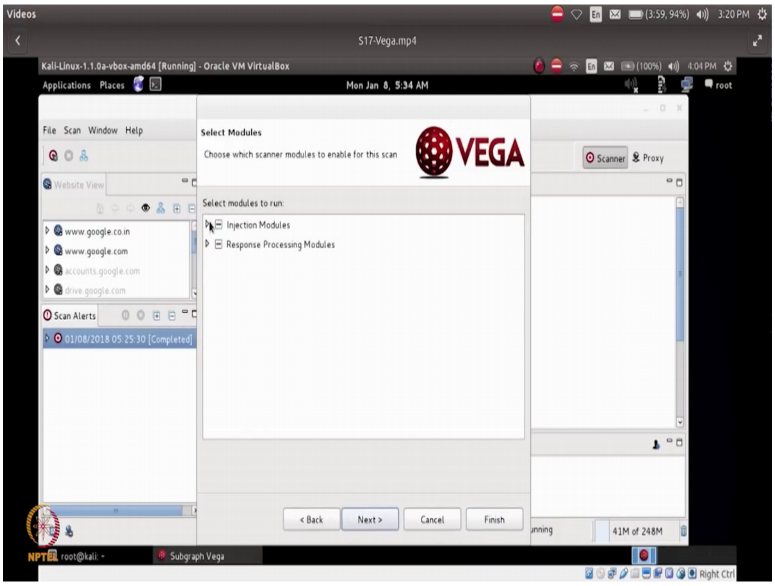
(Refer Slide Time: 13:22)



So, the Vega can actually be used either has a scanner or as a proxy. So, I if I basically go ahead and start a new scan I basically have to enter the URL that I am intending to do scan.

So, let us say that we type www dot Google dot com and then say next,

(Refer Slide Time: 13:43)



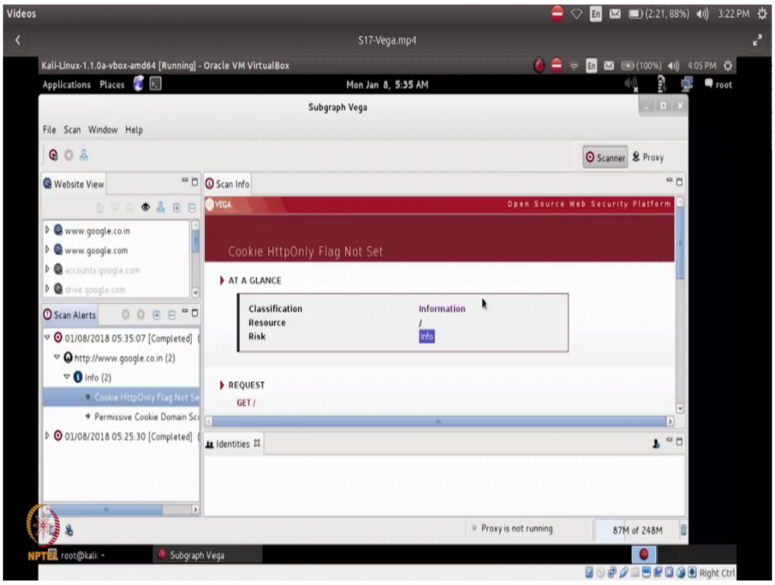
So, here we find that there are 2 types of modules that actually present one mode what is called as an injection module and among the injection modules Vega. Actually got a few of the setting that is possible could be made use of and all those things are actually mentioned here which could optionally be enabled also by selecting the corresponding button. So, it basically list down the different modules that actually available for that which is what it will actually try to do the testing on this particular site.

So, the moment I basically press the finished and the button I get a redirect message because I had actually typed www dot Google dot com and it now, tells me that it is actually getting redirected to another site called Google dot co dot in and Vega ask for a confirmation from the user, on whether the scan means to be performed on this redirected site also. Because in quite a few cases we would actually find out that it is basically taking us to into a different site on which we never intended the direction to be done.

In which case, the I could actually say that I do not want to redirection to be done on the redirected I do not want this scanning to be done on the redirected site by pressing no here. Right because in this particular case possibly since the load balancing has been enabled on the web server of Google's site it has detected that this particular request is actually come from India. And so, it is automatically redirected from Google dot com to Google dot co dot in which is basically in India based Google's (Refer Slide Time: 15:17) server.

If we go and decide yes, I want the scanning to be done on the redirected site also Vega goes ahead and complete the scan by doing all the scan of the modulus that has been enabled as part of the initial setting before we started of the scan. So, tend to the scan it basically tells that there are a couple of information messages alerts that we need to possibly look at and these get printed here; so one of them if you see is a message called cookie HTTP only flag not set right. So, if we go and click on the left bar and then try to get more details of that it tells us on the

(Refer Slide Time: 16:02)



Cookie HttpOnly Flag Not Set

AT A GLANCE

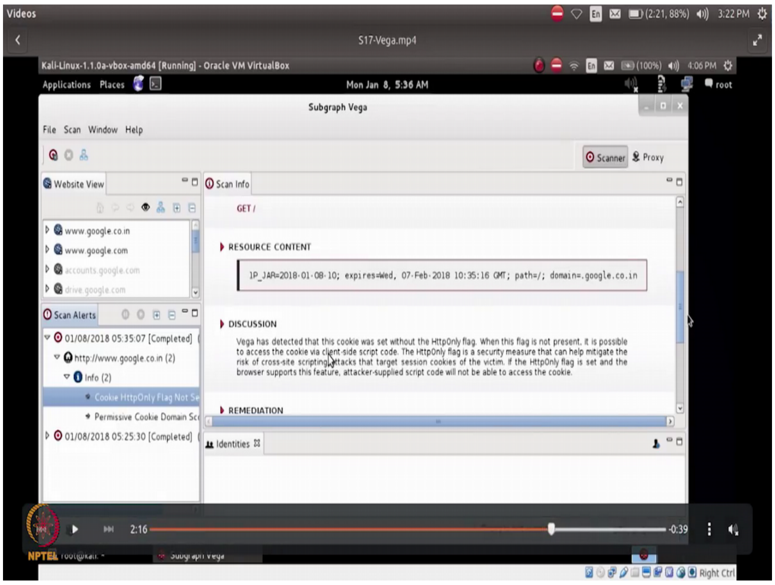
Classification	Information
Resource	/
Risk	Info

REQUEST

GET /

Right side right frame what specifically it meant by that particular message.

(Refer Slide Time: 16:09)



GET /

RESOURCE CONTENT

```
1P_JAR=2018-01-08-10; expires=Wed, 07-Feb-2018 10:35:16 GMT; path=/; domain=.google.co.in
```

DISCUSSION

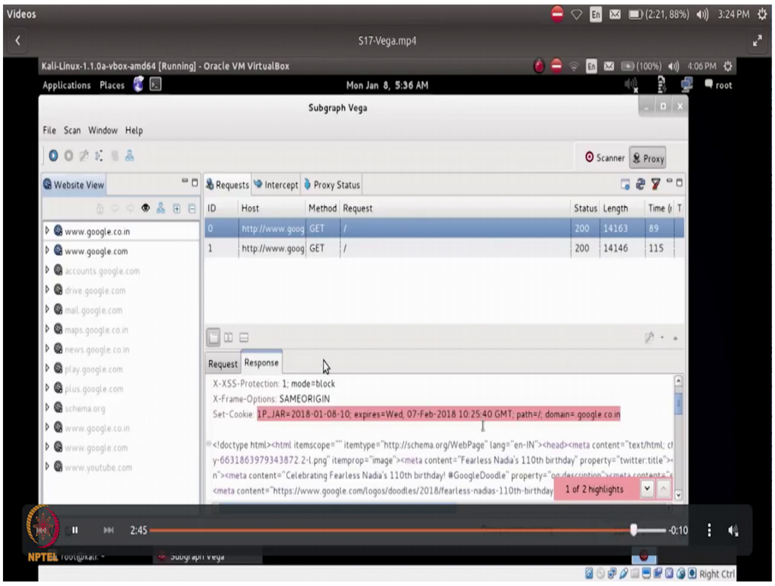
Vega has detected that this cookie was set without the HttpOnly flag. When this flag is not present, it is possible to access the cookie via client-side script code. The HttpOnly flag is a security measure that can help mitigate the risk of cross-site scripting attacks that target session cookies of the victim. If the HttpOnly flag is set and the browser supports this feature, attacker-supplied script code will not be able to access the cookie.

REMEDIATION

So, it basically tells you here that it has detected a cookie that could be set without HTTP only flag. So, what we mean by HTTP only flag is that with the HTTP only flag the cookie will be set on the server only when it is actually come from a browser HTTP based browser.

But when the flag is not present it is also possible to access this particular cookie via a script code that something like a java script code, that client side could possibly triggered. So, this is just a security measure that has actually been done to prevent cookies being sort of misused by the client side script; for example, and since it has been detected here in this particular server Vegas actually given as a particular detail. So, likewise if I enable other models injection modulus possibly and if those are actually available as part of the vulnerability is our present as part of the particular web server that is getting stand.

(Refer Slide Time: 17:11)



ID	Host	Method	Request	Status	Length	Time (i	T
0	http://www.goog	GET	/	200	14163	89	
1	http://www.goog	GET	/	200	14146	115	

```
X-XSS-Protection: 1; mode=block
X-Frame-Options: SAMEORIGIN
Set-Cookie: 1P_JAR=2018-01-08-10; expires=Wed, 07-Feb-2018 10:25:40 GMT; path=/; domain=.google.co.in

<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang="en-IN"><head><meta content="text/html; c
y-6631863979343872.2-l.png" itemprop="image"><meta content="Fearless Nadia's 110th birthday" property="twitter:title">
n"><meta content="Celebrating Fearless Nadia's 110th birthday! #GoogleDoodle" property="og:description"><meta content="
<meta content="https://www.google.com/logos/doodles/2018/fearless-nadias-110th-birthday
```

[scaned] and then we will find that Vega is able to capture those findings also and then present it to it was as part of this particular scans. So, similarly it could also run as a proxy in which case it will basically display the details on the proxy related information like cash in the URL access and so on and so forth. So this again another tool that potential penetration tester could actually make use of to sort of successfully find out more details about a particular web server or a web portal by getting the details from the pages that has been actually hosted as part of the web portal.

Thank you.