Concordia Institute for Information System Engineering (CIISE)

Concordia University

INSE 6961 – Graduate Seminar in Information and Systems Engineering

Graduate Seminar Report -- 3

Udemy Course: Learning Ethical Hacking from Scratch

Section Covered: Website Hacking - Information Gathering

Total Time covered for this Segment: 50 Mins

Submitted to:

Professor Ayda Basyouni

Submitted By:

Nithish Reddy Yalaka – 40164619

Social Engineering:

Introduction - What Is A Website ?

A website is nothing more than a computer program. That computer's specifications are likely to be superior to ours. It still functions as a computer because it has an operating system and programs such as a webserver and a database in the form of Apache and SQL. The webserver is responsible for understanding and running the web application. We should develop the web application in any programming language, such as java, PHP, or Python, to better comprehend it. The database is where the web application's data is stored. All of this information is kept on a computer known as the server. Anyone can visit and ping this machine because it is linked to the internet and has a legitimate IP address. As a result, whenever a user requests a page or web application, it is served by the web server installed on the target server, rather than on the client's computer. When it runs on the webserver, it sends a ready-to-read HTML page to the users who have requested it. Consider a client's request for youtube.com, which will be translated to an IP address by a DNS server. Then it will travel to youtube.com's IP address and run the page that the customer requested. As a result of the execution, the client receives mark-up written in HTML.

If we want something to be done on the webserver in the future, for example, to receive a shell or reverse shell, the web shell we give should be written in a language that the server understands, independent of who sees the page. This will happen on the server rather than on the client's machine. We'll be able to log on to the server as a result of this. If we can identify a website that permits us to run JavaScript code, on the other hand, the code will be executed on the client's machine. The code, on the other hand, can be injected into the webserver, run on the client side, and you can work on the client machine rather than the server. The difference between the languages of a client and a server is substantial.

How To Hack a Website?

The focus of this report is on web application security testing. So, according to the course video, our target is a metasploitable computer, the IP address is 10.20.14.204, and all of our website files are saved in the directory "var/www/." So, if we open a browser on any system on the same network and type in 10.20.14.204, we'll see a website named metasploitable.

We can attack the website since it is installed on a computer, just like any other computer. To hack a website, there are three different sorts of attacks. The following are some of them:-

Pen-testing a web application: This is where we can try to test the online application. As a result, our aim may not be the web application, but rather the user of that website. We will never gain access to a person's computer using this method, but you will be able to go to his website, hack into it, and obtain sensitive information about that individual.

Server-Side Attacks: the computer has a web server, an operating system, and numerous apps. We can exploit any flaws in those applications and acquire access to the machine if

we can locate them.

Client-Side Attacks: Websites are managed and maintained by humans. We'll probably be able to gain their account and password if we've been able to hack any of the website administrators, and we'll be able to log in or SSH into whatever services they use to maintain their website.

Gathering Basic Information Using Whois Lookup:

We'll concentrate on technologies that are commonly found on websites, such as domain names and DNS data. We use whois Lookup to discover information about the target's owner, such as IP address, server, or domain. We're simply pulling data from an online database of owners' information. When we register a domain name, we must provide information such as our name, contact information, and address so that others may know who owns it. We may use a whois Lookup to retrieve this information. This service is available on a variety of websites. We'll use whois.domaintools.com and enter your target domain name, for example, https://whois.domaintools.com/isecurity.org; as you can see, there's a lot of information about our target domain name available, such as email, the registered company's address, IP address, domain name creation date, history, server type, Operating System name and version, and name servers that are being used. When the domain name's privacy protection is enabled, we can't see the registered company's phone number or address.

Discovering Technologies Used On the Website:

We will learn about the technologies that the target website employs in this part. To do so, we use the Netcraft tool and go to the target website, such as https://sitereport.netcraft.com/? URL=isecurity.org. Web trackers will show you what third-party resources or applications are being used on our target website, in this case, Google Analytics, Google CDN, and other services. As a result, we may be able to find or access the target computer using this method. It provides us with the most important details regarding the technology in use. It's running on an Apache web server, as we can see. It employs PHP on the server side. We know that websites can comprehend and execute PHP code based on this. If we can deliver payloads with Metasploit and Veil-Evasion, we can write them in PHP and have them run on our target website. We can also observe that the website is JavaScript compatible. If the JavaScript code can't run on the website, it won't run on the website. Because JavaScript is a client-side language, it will run on the computers of users who visit the site. The website's web apps will also be displayed by Netcraft. If our target website runs on WordPress, we can exploit any existing vulnerability in that version of WordPress on the target site.

Discovering Websites on the Same Server:

Many websites can be found on a single server and gaining access to one can help you acquire access to others. Assume we obtain the target's IP address but are unable to gain access to the target's website with apps installed. In that scenario, we can use the same IP to find other websites installed on the same server, and if we can hack one of those

websites, we can utilize our metre Peter shell, PHP shell, or whatever shell we're using to navigate to the target website. When we use robtex.com to look for zaid.com, we may find a lot of websites with the same IP address. Another method was to find websites with the same IP address by searching with IP: [target Ip] in search engines.

Discovering Subdomains:

Our aim is always in target.com, e.g., google.com, which is referred to as the domain name. Many websites include subdomains that appear directly before the domain name, such as mail.google.com, where google.com is the domain name and mail is the subdomain name linking to various online apps. As a result, it's always important to discover all the subdomains associated with the domain name because it expands our attack surface and provides us with more information about the new web application, the beta version of the web application, which is still in development and has a higher chance of having bugs and gain access to a subdomain before moving on to the target website. We can find subdomains with the help of a tool called knock, which is installed on a modified Kali Linux and allows us to use the command knockpy [target website] to acquire a list of all subdomains.

Discovering Sensitive Files:

This step will search the target website for files and directories that may contain passwords, config files, or information about the server that we can use to abuse our target. Webserver files are often stored under /var/www, which contains a number of files and directories. A directory called Mytilidae, an online program designed to hack, can be found. So, open a browser window, type the IP address, and go to the Mutillidae online application. The URL is var/www/Mutillidae/; the forward slash at the end indicates that we are in its directory, and the IP address will take the place of var/www. There are always hidden files and directories, and we can see how to acquire the URLs of these files and access the content by using this method. We utilize a tool called dirb for this, and the command to use is dirb [target URL] [wordlist]. According to the course, world list is a dictionary containing various names, and dirb performs a brute force search to find the file if one exists. Only files with the names we specified in the wordlist will be found.

Analyzing Discovered Files:

Many files and directories from the previous part can be found, such as the login page and the PHP file, which is quite important because it contains a wealth of information about the PHP interpreter. The robot.txt file is another crucial file that tells search engines like Google how to treat webpages. We can see that the web administrator does not want Google to see password, config.inc, and other directories. So, we can look at the passwords file and get the usernames and passwords, but we don't know where to utilize them, so we'll try to use them for the login page. We can see information that allows us to connect to the database in the config.inc file since it contains details such as dbuser, dbpass, dbhost, and dbname.

References:

https://concordia.udemy.com/course/learn-ethical-hacking-from-scratch/learn/lecture/5369454#overview

Instructor : Zaid Sabih

Segments Covered:

Section 19: Website Hacking and Section 20: Website Hacking – Information Gathering