Beatrice Antoniu

W1688177

6COSC002W SECURITY AND FORENSINCS ASSESSMENT

aBSTRACT:

You are hired as a penetration tester for a small equity firm that works with investors in cryptocurrency. Their trading platform allows customers to track their equity value. They can also decide if they want to sell or invest more in cryptocurrency. The platform is only open by invitation for few selected customers. Financial details are stored on the database of the platform. ID verification details are also stored in order to comply with financial regulations. Users credentials are stored on the database. Not all users have the same privilege.

# A. Information gathering

1. Identify the ports you found in the lab running on the server machine and briefly explain what threats those open ports bring to your scenario.
   * The main problem with most of the ports that I have found would be that if a hacker would try and hack into the server machine, they would easily gain access to personal information that belongs to the users such as emails and shared files. [Figure 1.1](#Services)
2. Identify two services running on the server machine that should be priority to protect.
   * **HTTP** and **IMAP** are two services that could cause real damage in my scenario. [Figure 1.2](#Versions)
3. Once you identify the services on your machine, research three internet vulnerabilities related to those services.
   * **IMAP** was designed to accept non-encrypted login credentials which could be a relief to future attackers as most people and machines can easily read this type of credentials. This service also does not allow multifactor authentication which means the attacker can easily gain access to an account that is using a common password and of course to its files and personal details. **(Loshin, 2019)**
   * **HTTP** sessions are not always secure and so the attacker can take advantage of that and steal temporarily the user’s identity in an active session. Another vulnerability of this service is that some object implementations can reveal references to important data such as database records or keys which can be easily manipulated by attackers to gain access to users’ personal information. Security misconfiguration is another issue for the **HTTP** service because the attackers can compromise the system completely due to the lack of maintenance. **(PureVPN, n.d.)**
4. Pick the four least secure services running on the server machine and explain the danger posed by each of them. Document your security concerns if you have any.
   * **HTTP**, as I mentioned before, this is one of the least secure not only for the server machine in general but especially in my scenario. The **HTTP** sessions are not secure, and the attacker could steal users’ identities in active sessions. **(PureVPN, n.d.)**
   * **HTTPS** is not always a secure connection; you are still being exposed to phishing and if the certificates for encryption are not renewed the same vulnerabilities for **HTTP** will apply to **HTTPS** as well. Entering any sort of sensitive data on a **HTTPS** website that has an expired certificate would lead to your data being stolen by potential attackers. **(Ryabova, 2018)**
   * **SSH** is a network protocol that operates network services securely over an unsecured network. The main problem with this service is that users are using keys rather than passwords to login and so the system is more vulnerable because those keys are being kept on the device. Another common issue with this service is related to the configuration and the settings because the admins can disable or change them and so the risks are being either decreased or highly increased. **(VENAFI, n.d.)**
   * **NETBIOS** is a network protocol that allows computer applications to be used on modern TCP/IP networks. This service can reveal incredible amounts of detailed and vital security information about an exposed network and allows attackers to map the network and freely navigate a compromised intranet. **(Wikipedia, n.d.)**

# B. Finding and exploiting vulnerabilities

1. Identify if the application is vulnerable to data tampering and exploit it if possible.
   * The application is vulnerable to data tampering and can be exploited. [Figure 2.1](#DataTampering) [Figure 2.2](#DataTampering2)
2. Identify if the application is vulnerable to SQL injection and exploit it if possible.
   * The application is vulnerable to SQL injection and can be exploited. [Figure 2.3](#SQLInjection) [Figure 2.4](#SQLInjection2) [Figure 2.5](#SQLInjection3)
3. Identify if the application is vulnerable to XSS vulnerability and exploit it if possible.
   * The application is vulnerable to XSS and can be exploited. [Figure 2.6](#XSS) [Figure 2.7](#XSS2) [Figure 2.8](#XSS3)
4. Can you identify any other vulnerability?
   * The application is also vulnerable to OS commands. [Figure 2.9](#OSCommands) [Figure 2.10](#OSCommands2) [Figure 2.11](#OSCommands3) [Figure 2.12](#OSCommands4)

# C. Man in the middle attacks and social engineering

1. If a client is connected to the server while you are testing the environment, identify what are the information that can be obtained from a packet capture of their communication.
   * If the system is **hijacked**, the attacker would have access to the login details that are being sniffed on when the user submits their credentials to log into the application. Besides the username and the password, the attacker does not get any other information from the packet. [Figure 3.1](#Ettercap) [Figure 3.2](#Spoofing) [Figure 3.3](#Ferret) [Figure 3.4](#Hamster) [Figure 3.5](#Hijacking) [Figure 3.6](#Hijacking2)
2. Identify a method that lure a normal user of the server to your computer instead of the server machine. What information you can get from this?
   * Both **cloning** and **phishing** would succeed in luring the user to the attacker’s computer, but through phishing they would reveal their personal information without getting suspicious, like it would if we would just clone the application. [Figure 3.7](#WebsiteCloning) [Figure 3.8](#WebsiteCloning2) [Figure 3.9](#WebsiteCloning3) [Figure 3.10](#Phishing) [Figure 3.11](#Phishing2) [Figure 3.12](#Phishing3) [Figure 3.13](#Phishing4)
3. If the server is protected, what can you do to penetrate the system from the client side?
   * To penetrate the system from the client side we need to use **Metasploit** to trick the user into executing programs so the attacker machine could take control over their system information and many other things. [Figure 3.14](#Metasploit) [Figure 3.15](#Metasploit2)

# D. Protecting your server. Now you have completed a first assessment of the network, what are your recommendations?

1. Based on your results, you have identified that “Port knocking” method is important to implement on your server. Explain.
   * Defeating **port** **knocking** requires brute force attacks to reveal even simple sequences and it would require the attacker to test and scan every port between attacks until they would find the right sequence. It is a flexible system add-in to incorporate into the script and it also protects against protocol vulnerability exploits. **(Wikipedia, n.d.)**
2. Hackers will attempt to scan a machine looking for suitable vulnerabilities to exploit. In your own words explain what false positives and false negatives are in relation to a Network Intrusion Detection System (NIDS).
   * A false positive is considered an acceptable attack that the **IDS** identifies, it can cause issues but can be solved. **(KristenS, et al., n.d.)**
   * A false negative is considered an actual attack that the **IDS** fails to identify and is hard to remove since no one knows the attack took place. **(KristenS, et al., n.d.)**
3. Explain the difference between Intrusion Detection System IDS and Intrusion Prevention System IPS. Suggest a recommendation for the scenario you have in a hand.
   * Both read network packets and compare the contents to a database of known threats. What differentiate them is what happens next: **IDS** do not act on its own, **IPS** accepts or rejects a packet based on the ruleset. Shortly, **IDS** needs a human or another system to act while **IPS** is controlled by a ruleset. **(Petters, 2020)**
   * Based on my scenario, **IPS** should be used as the clients’ financial details and id verification details are stored in the database and IPS offers more protection even for VPN connections, if the threats database is updated regularly. **(Petters, 2020)**
4. Evaluate the effectiveness of the following tools and specify which you will use. Justify your answers.
   * **Firewall** is the most effective for my scenario as it monitors the traffic and offers privacy for the user but also has an access policy that can block suspicious hosts. Trojan malwares are also detected and blocked by the firewall as well as any other malicious activities that can be done by a potential attacker. It is ideal for small organizations and fits my scenario the best. **(Roor, 2020)**
   * **Snort** is a software used for network IDS and IPS. It would be the least efficient tool for my scenario as it does not provide much security and privacy. It is useful for analyzing the traffic and detecting some possible attacks. **(Wikipedia, n.d.)**
   * **IPTABLES** is part of the firewall and so alone it would not make a good tool for my scenario. Iptables focuses the traffic of packets and requires privileges to operate and it must be executed by the root. **(Wikipedia, n.d.)**
5. Based on your findings, document any other recommendation based on vulnerabilities and weaknesses. Your recommendations should be based on the scenario you are working on and the type of data and services involved.
   * Based on my scenario not all the users using the platform have the same privileges and so their personal details are vulnerable in case of an attack. A better and more **secure cryptographic storage** should be considered so that all the users’ sensitive information would be kept encrypted and private. **Input/Output encoding** and whitelisting the input fields would solve most of the problems cause **by SQL injection** and **XSS**. **Two-Factor authentication** should be considered as well when logging into the platform to prove the users identify or at least a **unique request token** that only the user has access to. **VPN connections** would be a better solution for users that do not have the same privileges as the users that were invited to use the platform.

# References

KristenS, Wichers, Jkurucar & kingthorin, n.d. *OWASP.* [Online]   
Available at: https://owasp.org/www-community/controls/Intrusion\_Detection#  
[Accessed 05 April 2021].

Loshin, P., 2019. *TechTarget.* [Online]   
Available at: https://searchsecurity.techtarget.com/tip/Where-does-IMAP-security-fall-short-and-how-can-it-be-fixed  
[Accessed 29 March 2021].

Petters, J., 2020. *Varonis.* [Online]   
Available at: https://www.varonis.com/blog/ids-vs-ips/#:~:text=IPS%3A%20What%20is%20the%20Difference%3F,-Data%20Security&text=Intrusion%20Detection%20Systems%20(IDS)%20analyze,detects%20—%20helping%20stop%20the%20attack.  
[Accessed 05 April 2021].

PureVPN, n.d. *PureVPN.* [Online]   
Available at: https://www.purevpn.com/ddos/http-vulnerability  
[Accessed 29 March 2021].

Roor, M., 2020. *Hitechwizz.* [Online]   
Available at: https://www.hitechwhizz.com/2020/03/5-advantages-and-disadvantages-drawbacks-benefits-of-firewall.html  
[Accessed 05 April 2021].

Ryabova, Y., 2018. *Kaspersky.* [Online]   
Available at: https://www.kaspersky.com/blog/https-does-not-mean-safe/20725/  
[Accessed 05 April 2021].

VENAFI, n.d. *VENAFI.* [Online]   
Available at: https://www.venafi.com/education-center/ssh/security-and-vulnerabilities  
[Accessed 05 April 2021].

Wikipedia, n.d. *Wikipedia.* [Online]   
Available at: https://en.wikipedia.org/wiki/Snort\_(software)  
[Accessed 05 April 2021].

Wikipedia, n.d. *Wikipedia.* [Online]   
Available at: https://en.wikipedia.org/w/index.php?title=NetBIOS\_over\_TCP/IP&gettingStartedReturn=true  
[Accessed 05 April 2021].

Wikipedia, n.d. *Wikipedia.* [Online]   
Available at: https://en.wikipedia.org/wiki/Port\_knocking  
[Accessed 05 April 2021].

Wikipedia, n.d. *Wikipedia.* [Online]   
Available at: https://en.wikipedia.org/wiki/Iptables  
[Accessed 05 April 2021].

APPENDIX PORTFOLIOS SCREENSHOTS

# PORTOFOLIO TASK 1

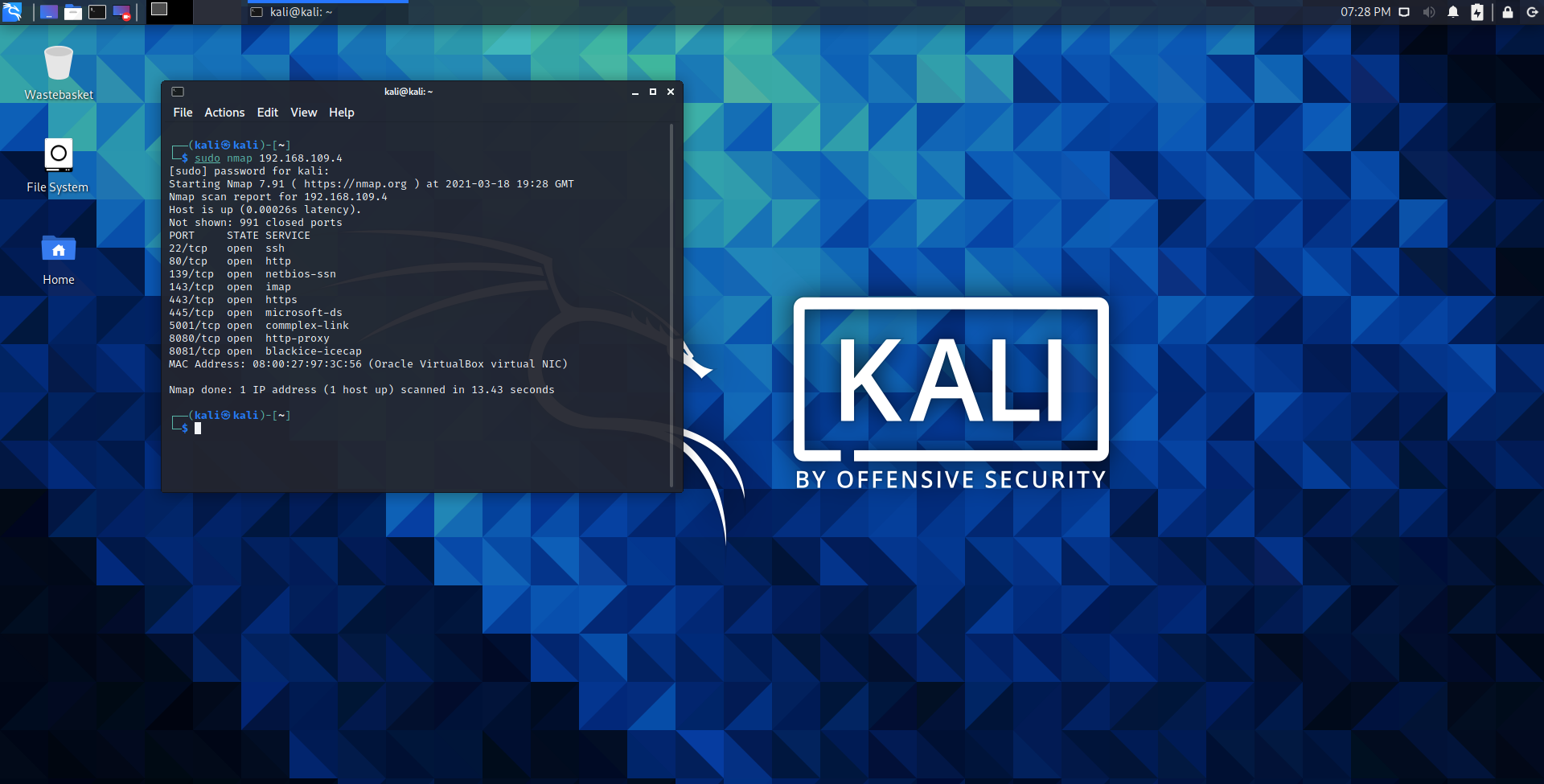


Figure 1. 1 – List of open services



Figure 1. 2 – The versions of the open services

# PORTOFOLIO TASK 2

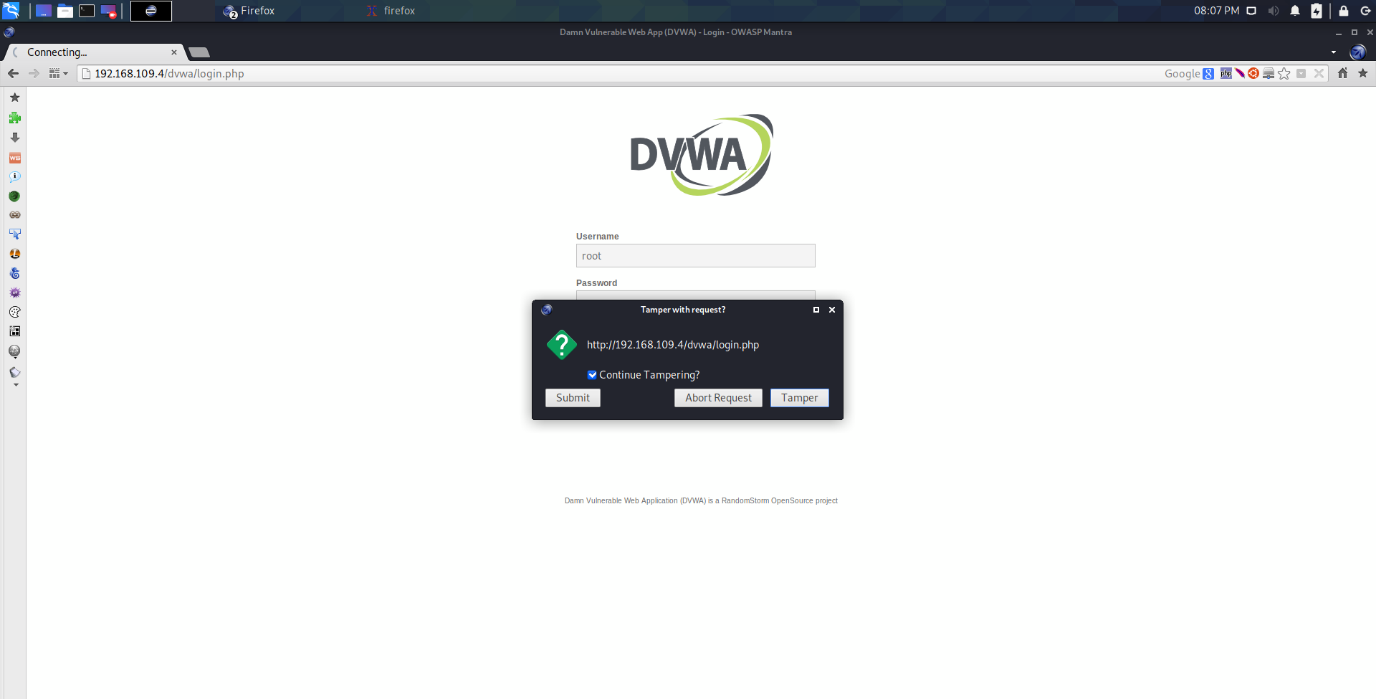


Figure 2. 1 – Data Tampering process

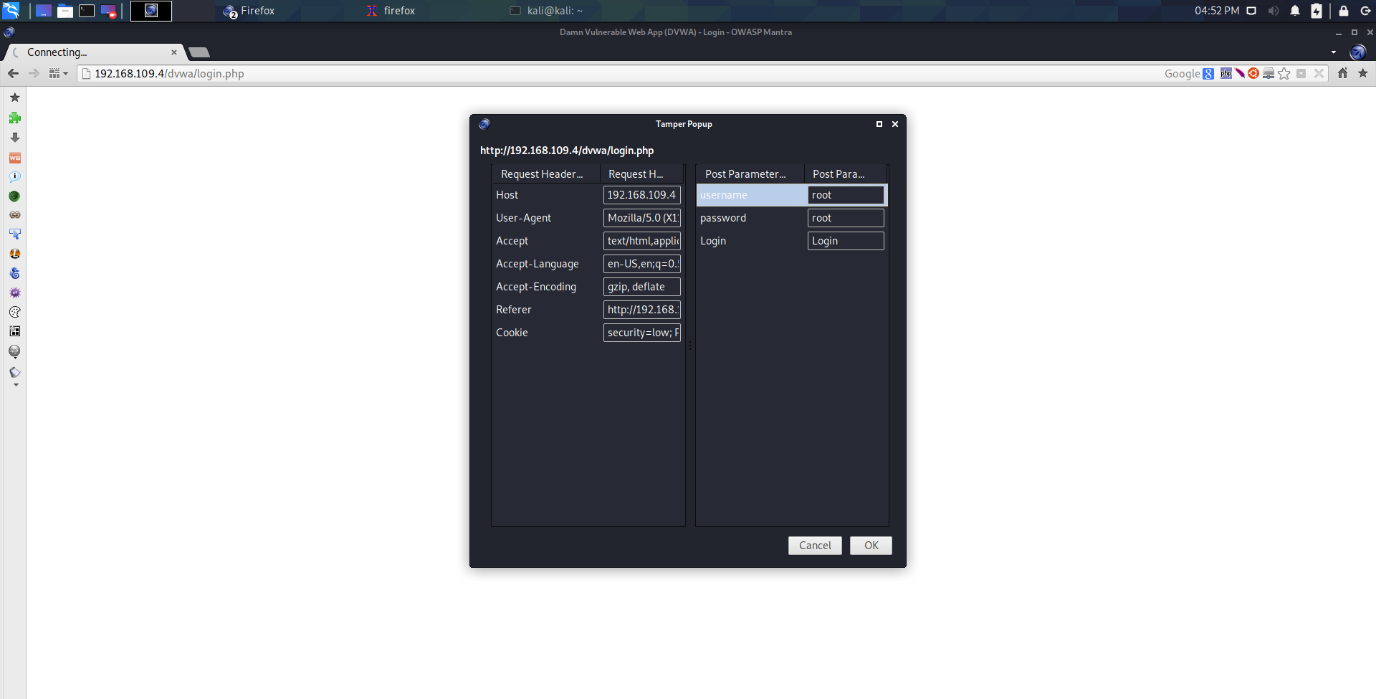


Figure 2. 2 – Data Tampering

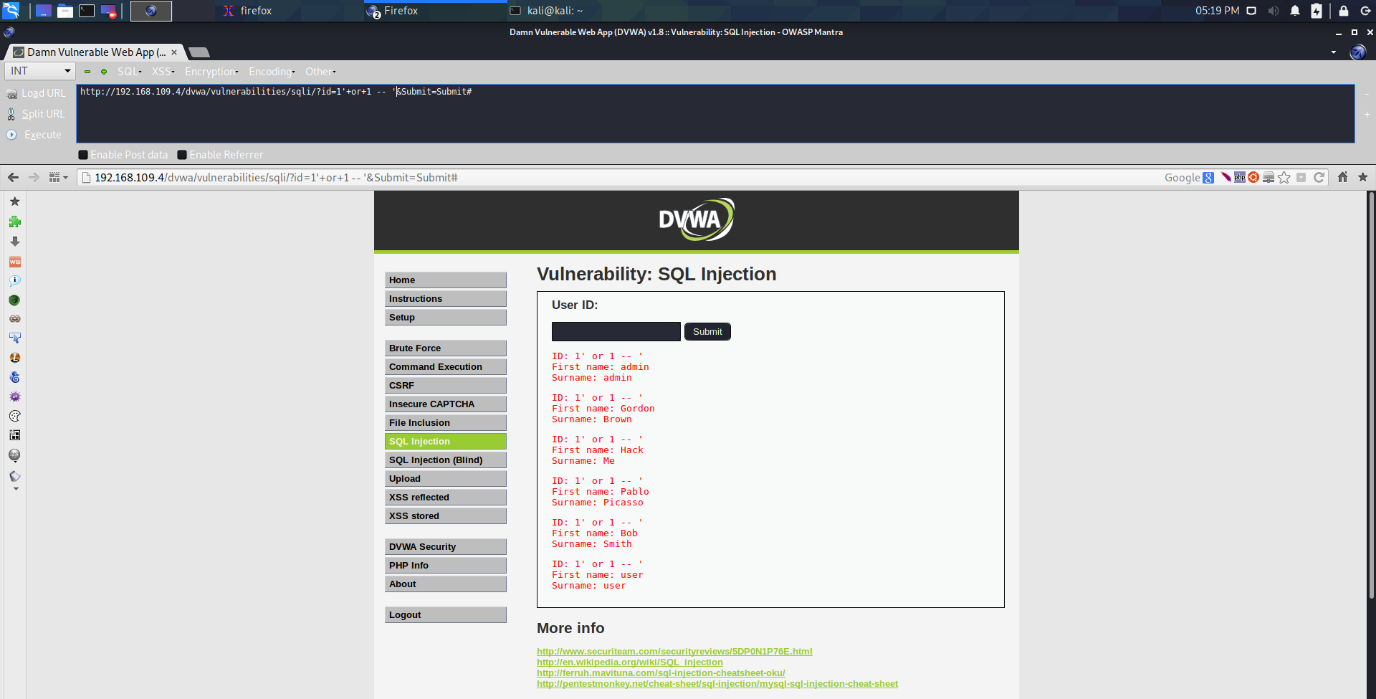


Figure 2. 3 – SQL injection

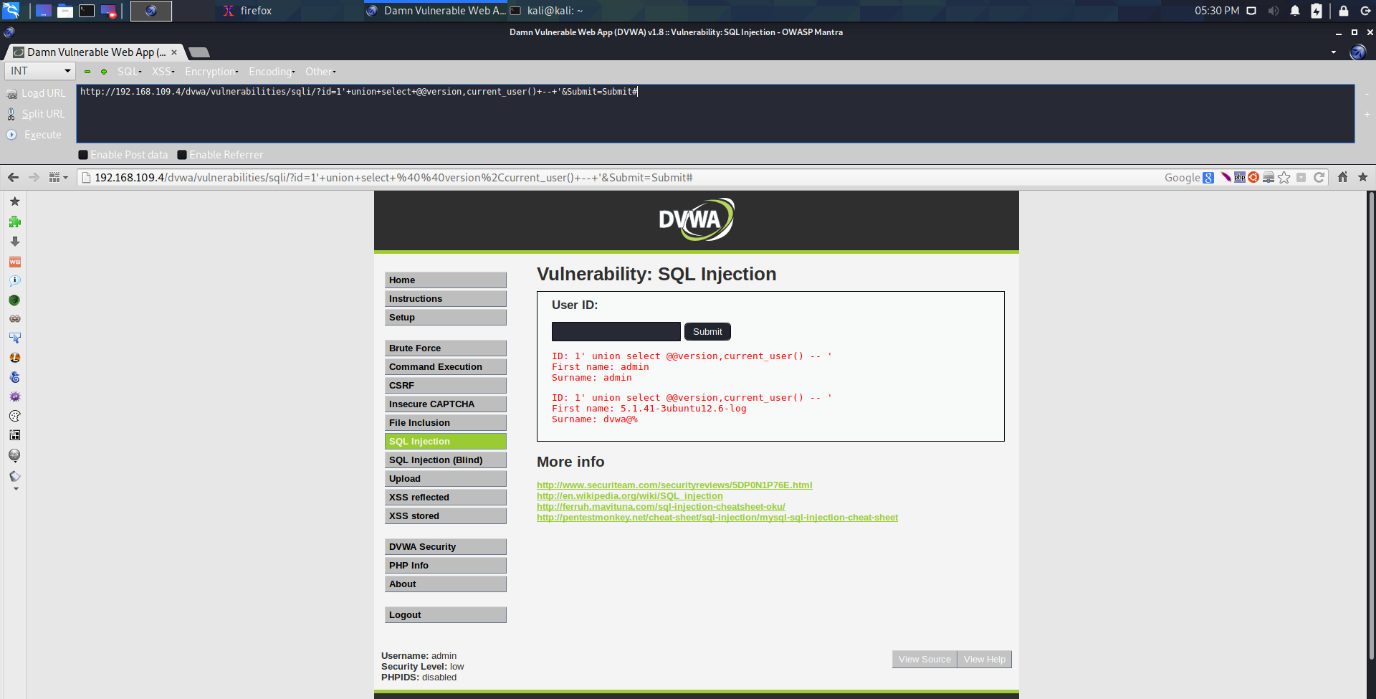


Figure 2. 4 – SQL injection

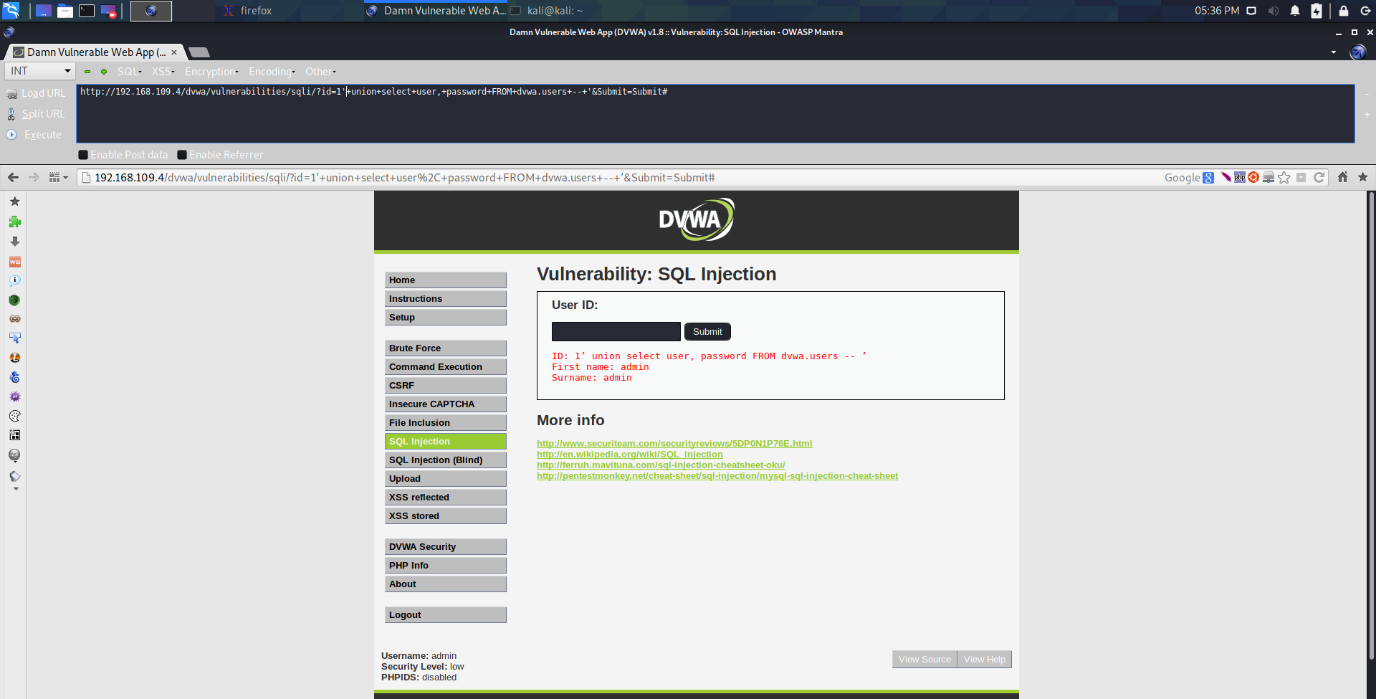


Figure 2. 5 – SQL injection

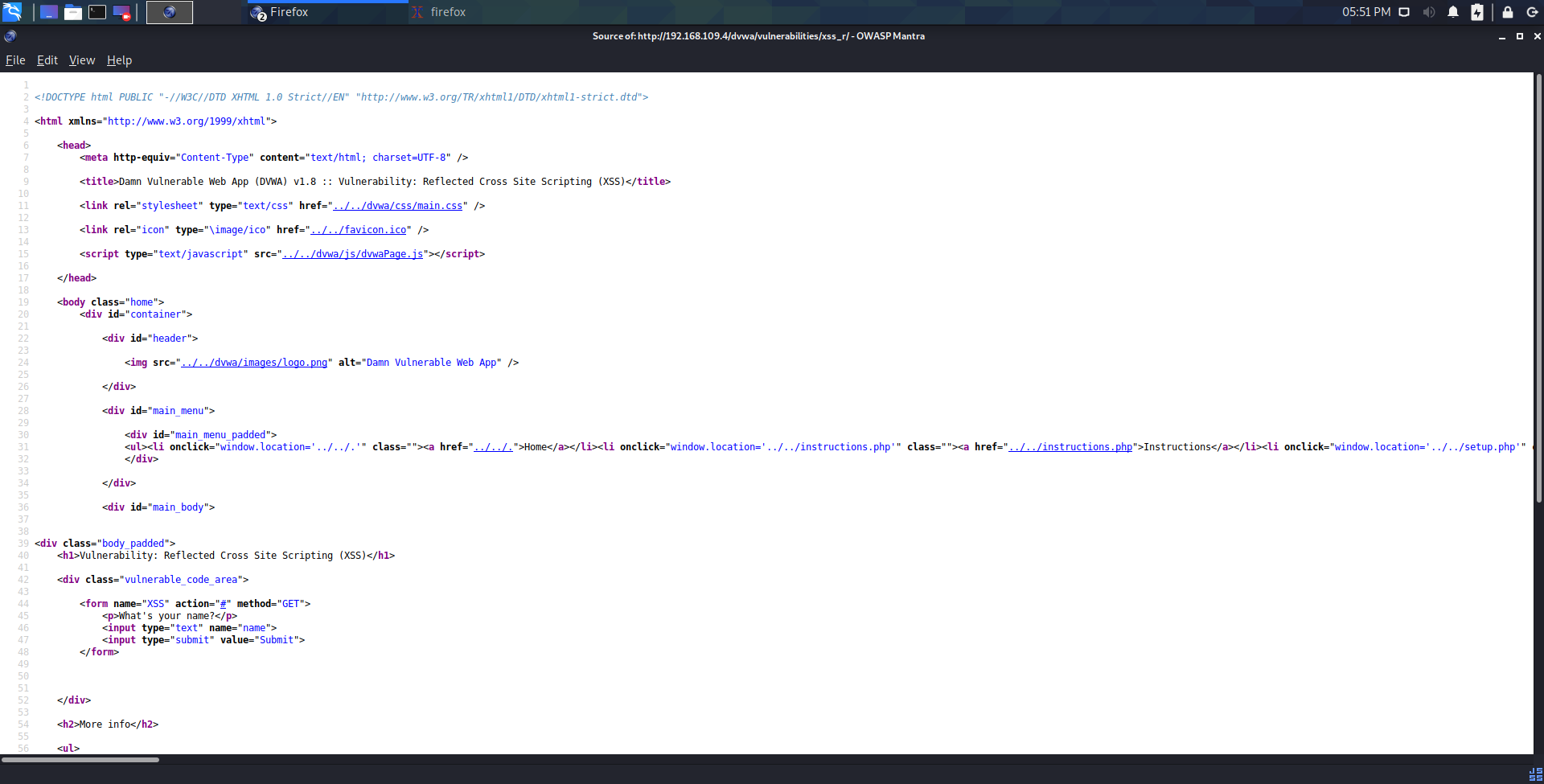


Figure 2. 6 - XSS

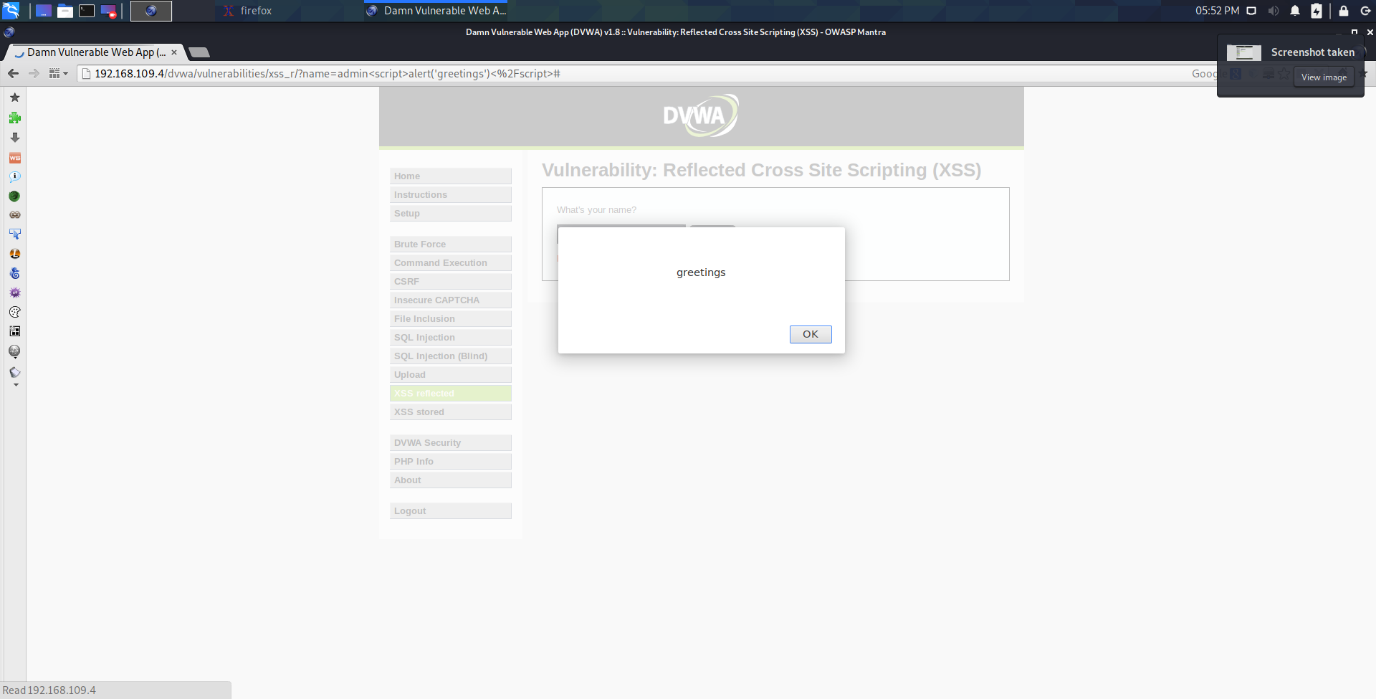


Figure 2. 7 - XSS

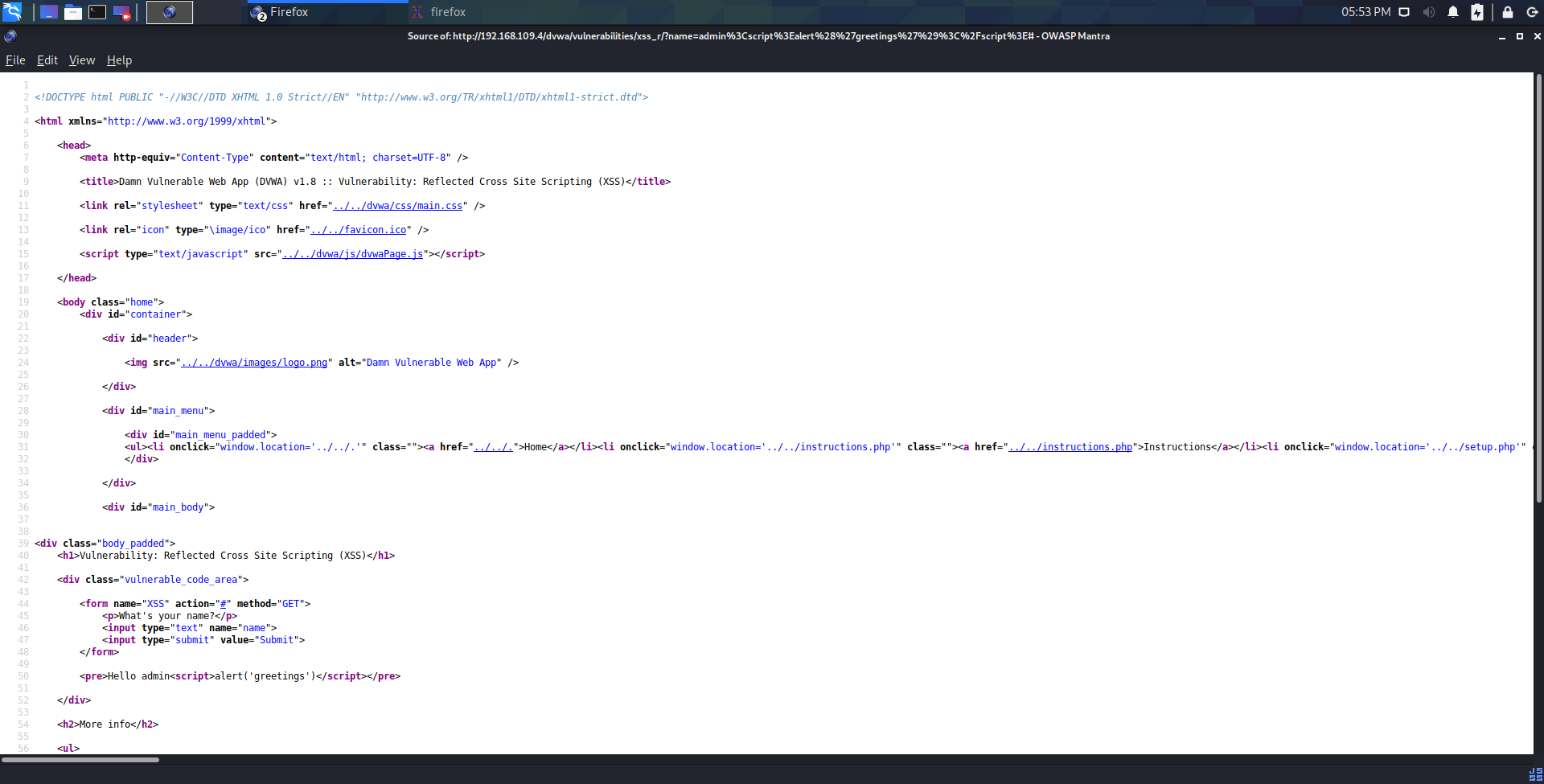


Figure 2. 8 - XSS

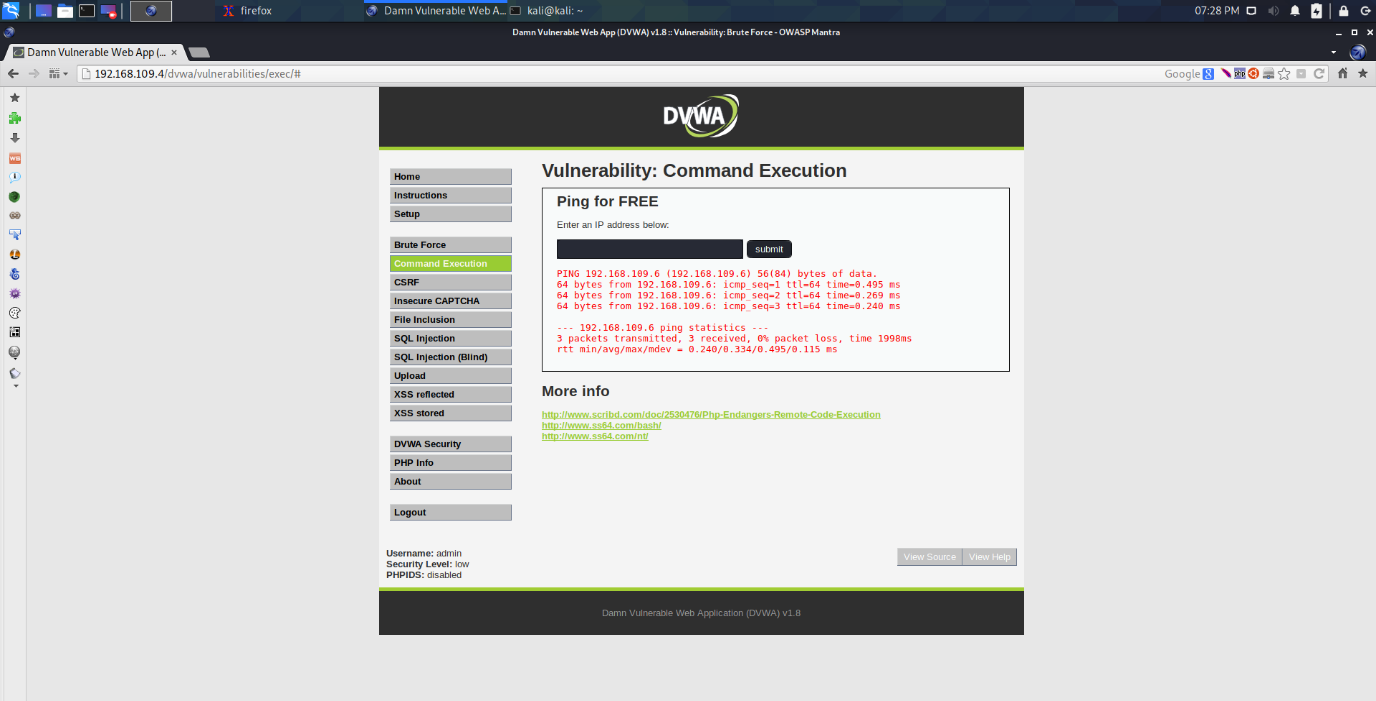


Figure 2. 9 – OS commands

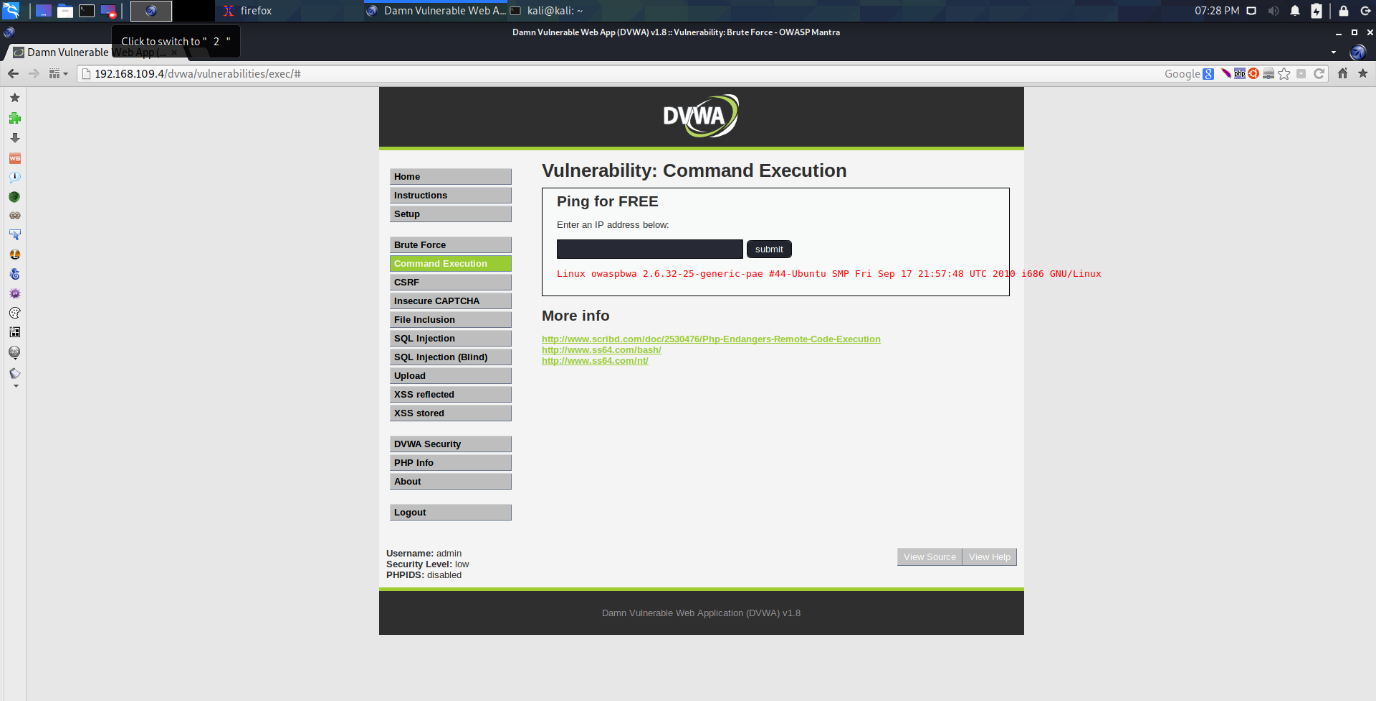


Figure 2. 10 – OS commands

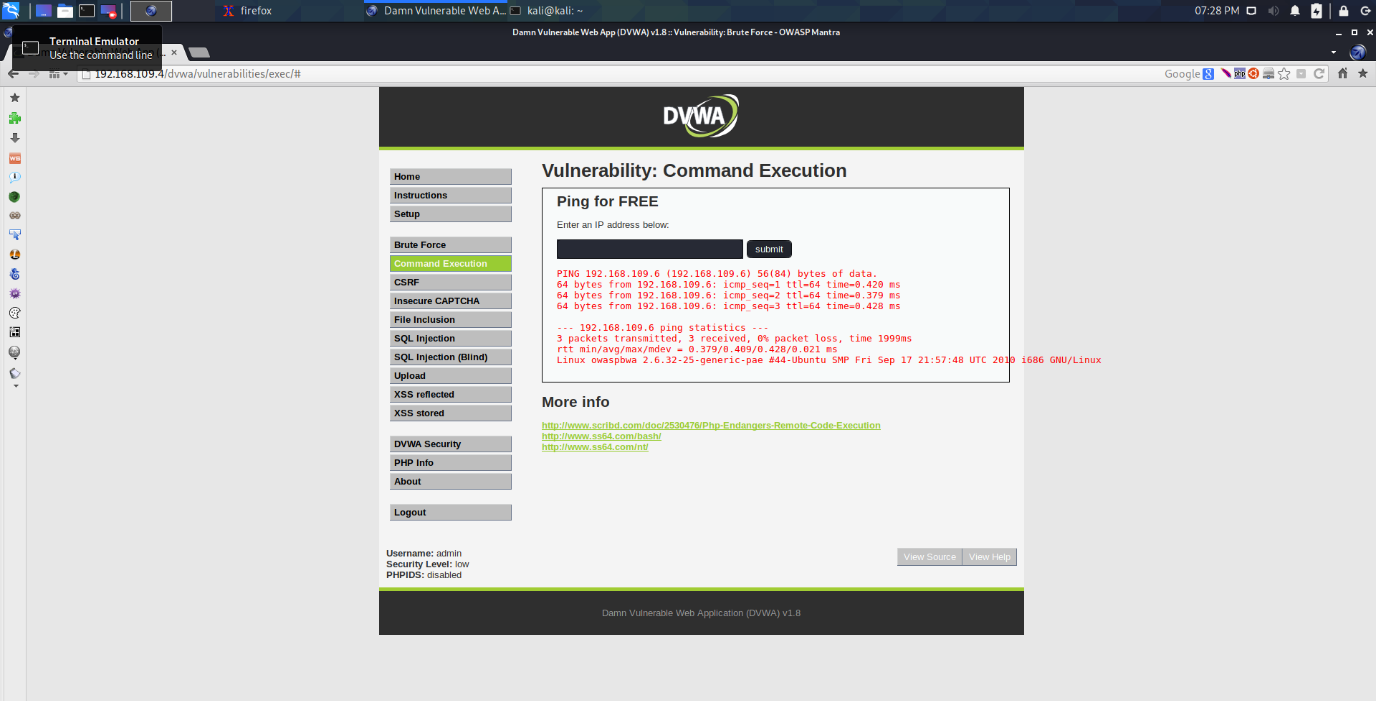


Figure 2. 11 – OS commands

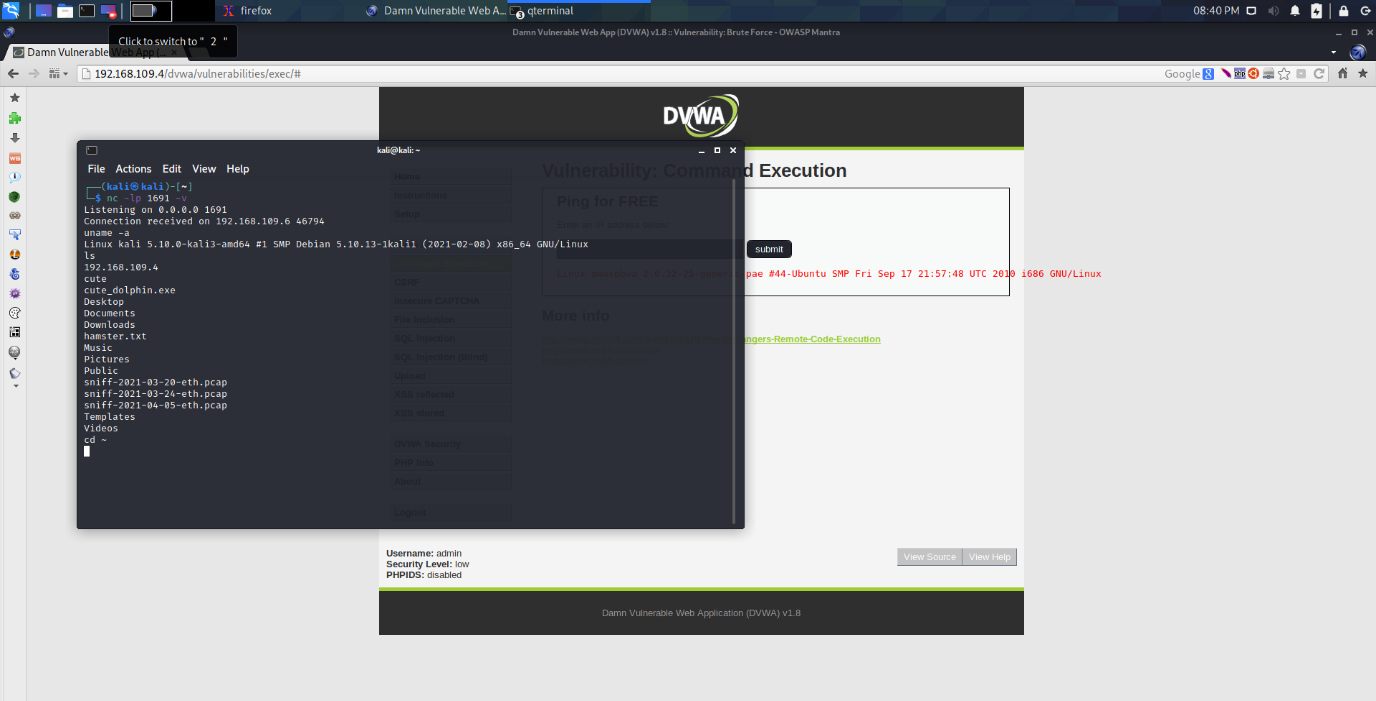


Figure 2. 12 – OS commands

# PORTOFOLIO TASK 3

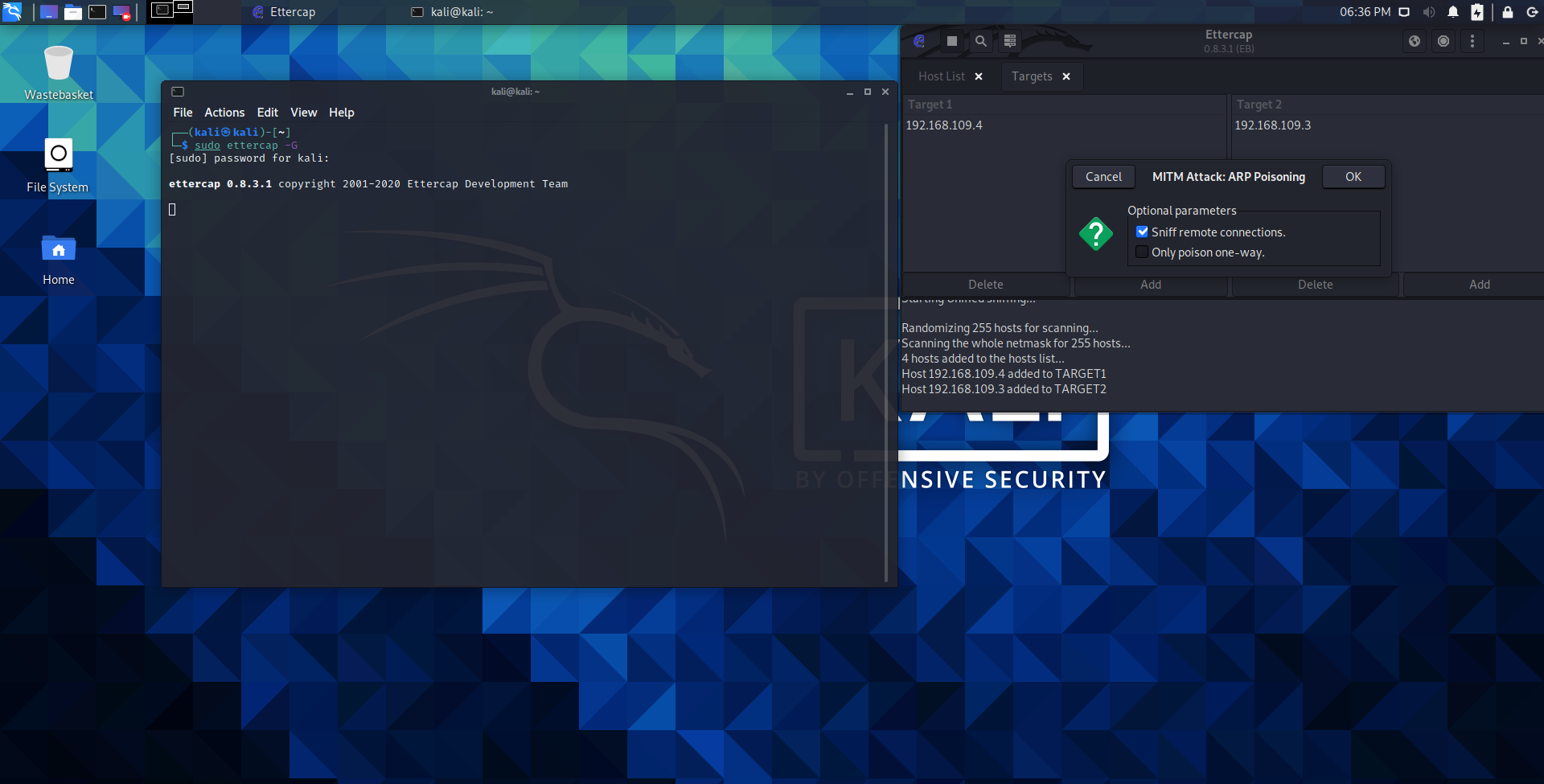


Figure 3. 1 - Ettercap

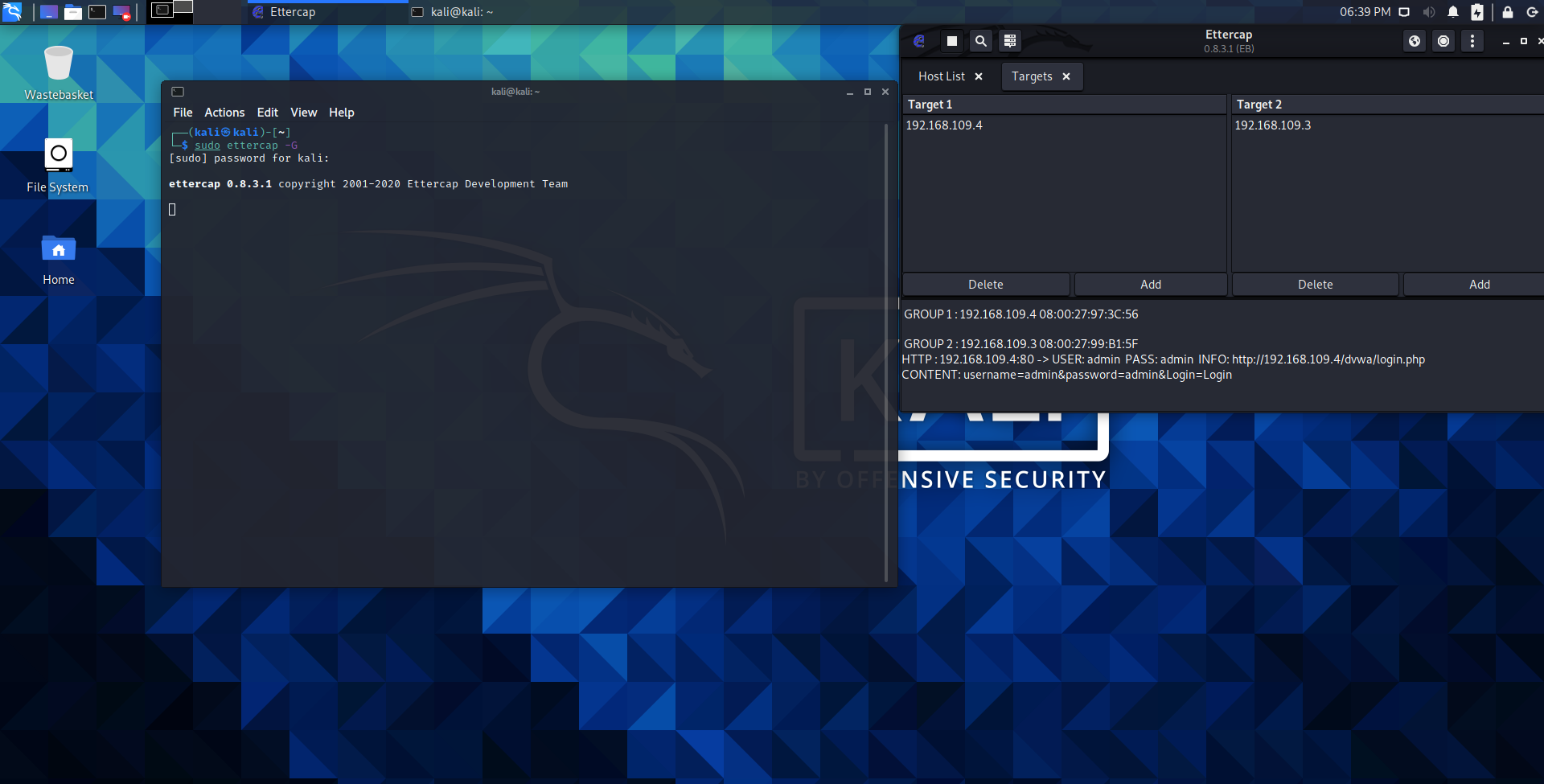


Figure 3. 2 – Spoofing

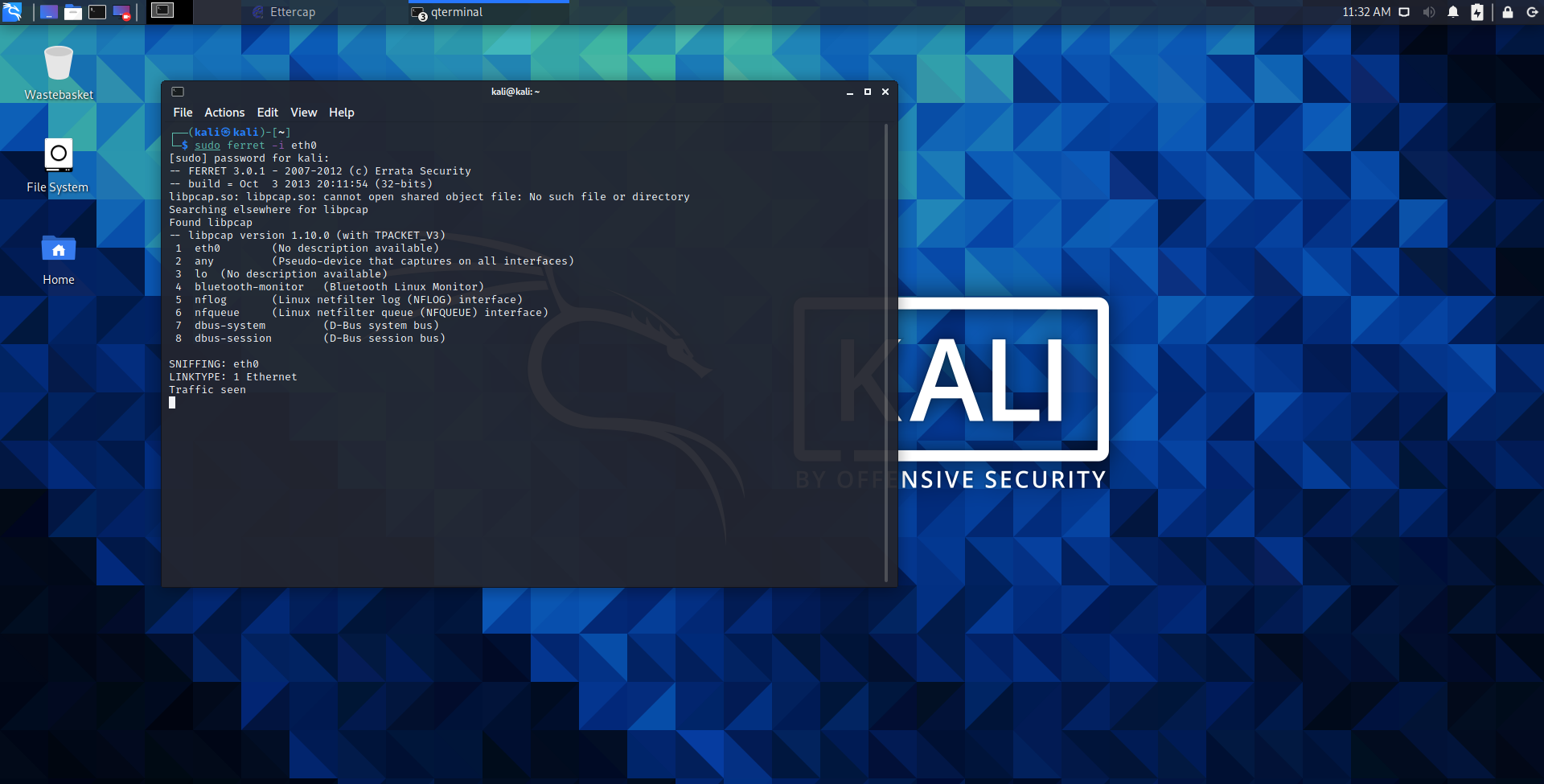


Figure 3. 3 - Ferret



Figure 3. 4 - Hamster

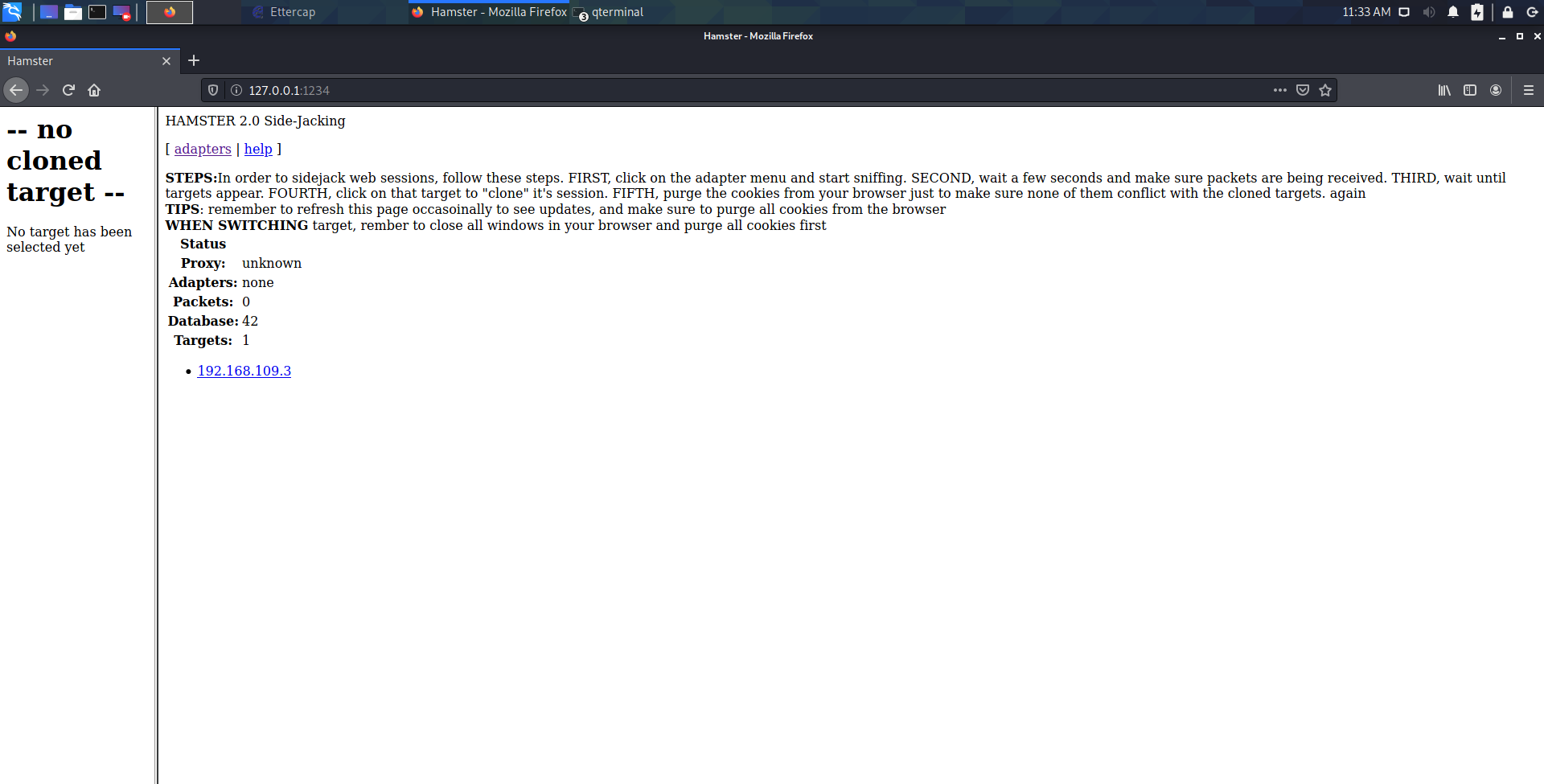


Figure 3. 5 – Hijacking

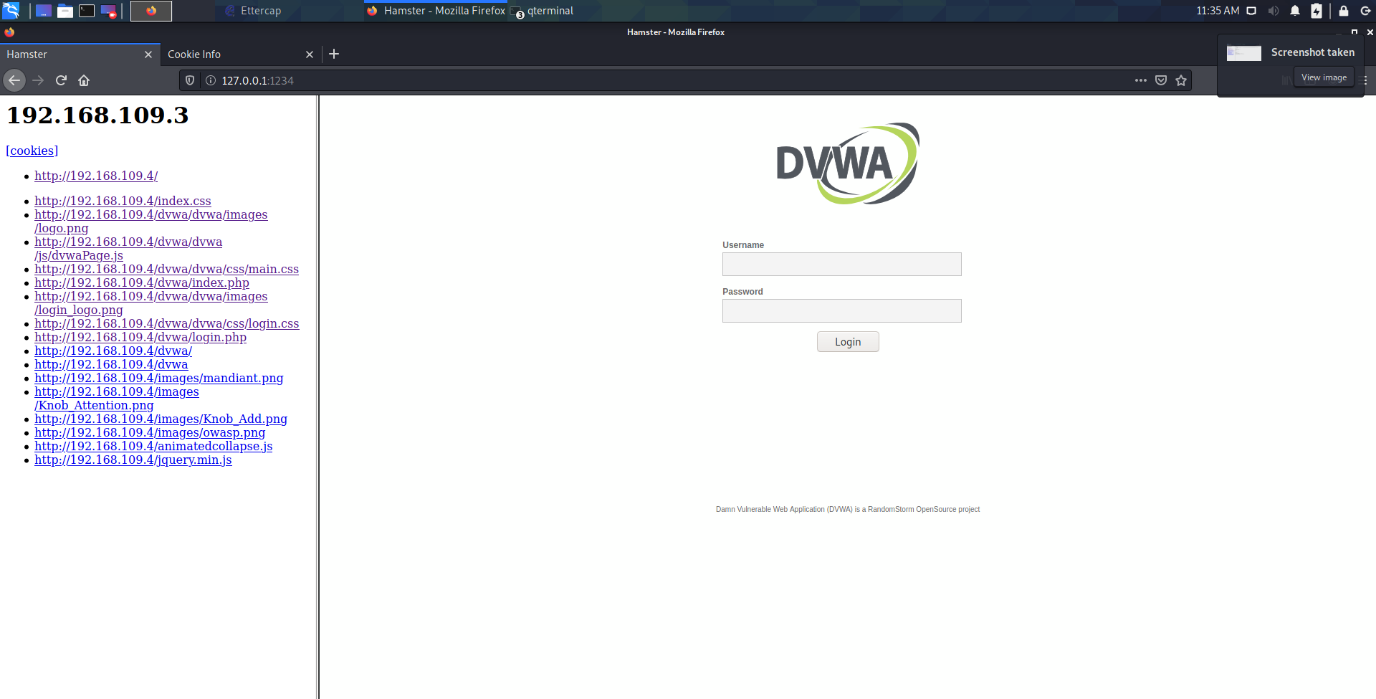


Figure 3. 6 – Hijacking

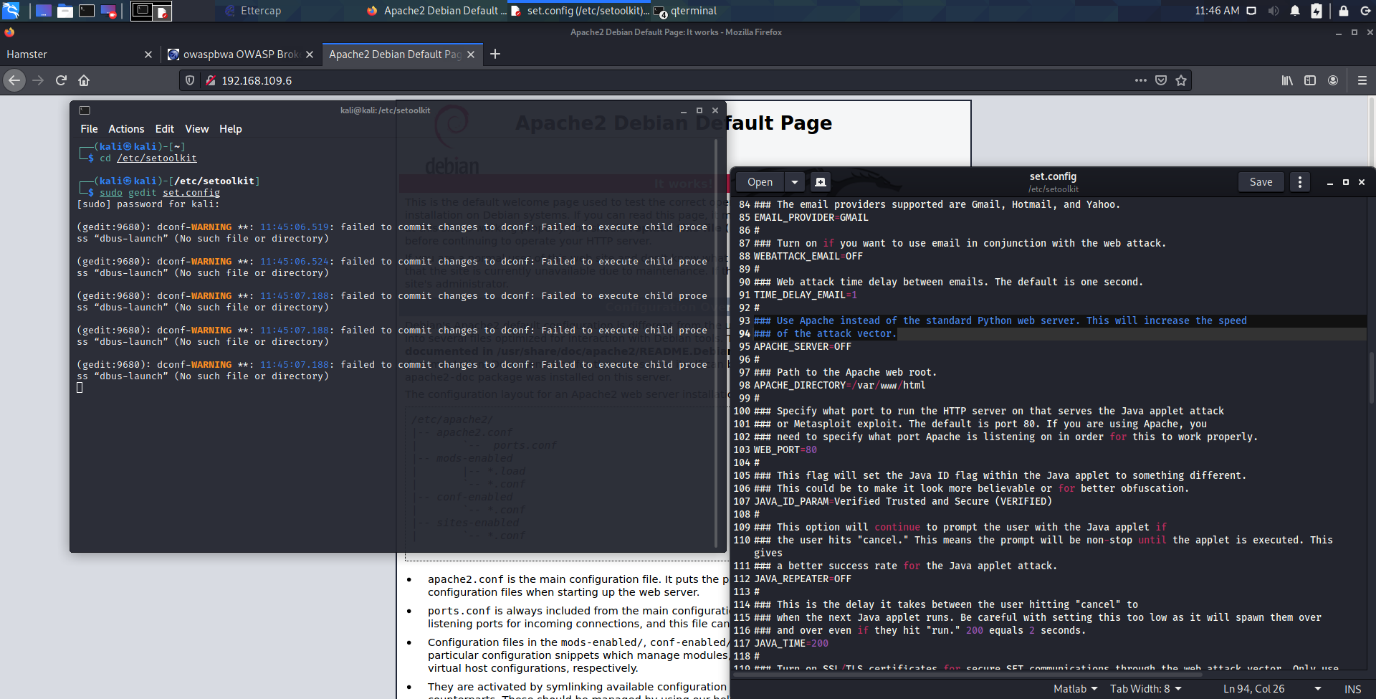


Figure 3. 7 – Website Cloning

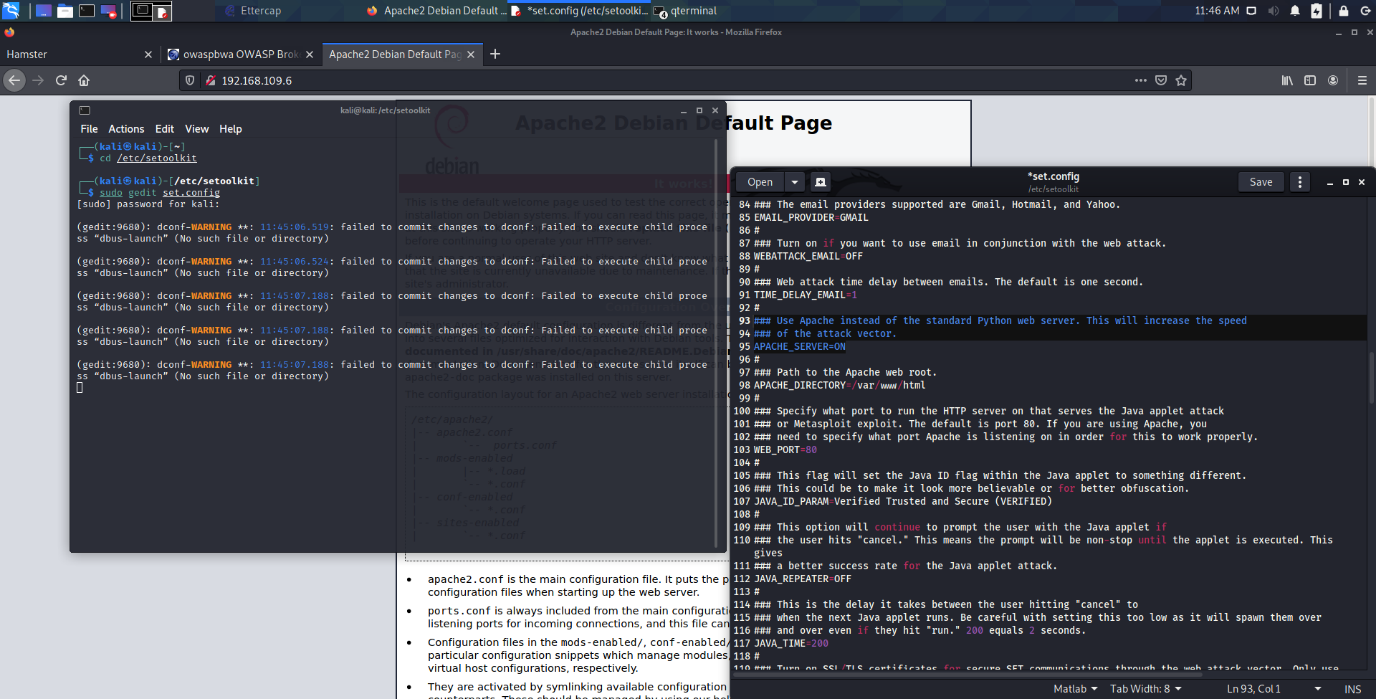


Figure 3. 8 – Website Cloning

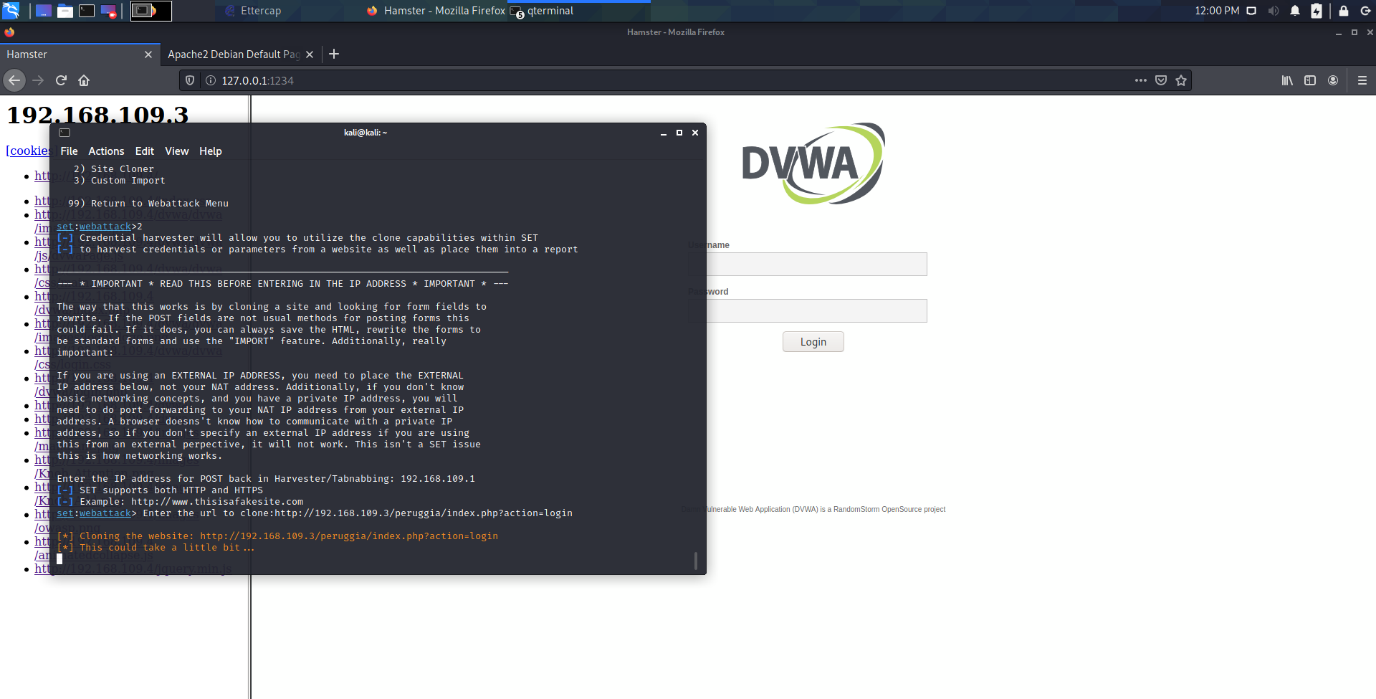


Figure 3. 9 – Website Cloning



Figure 3. 10 – Phishing

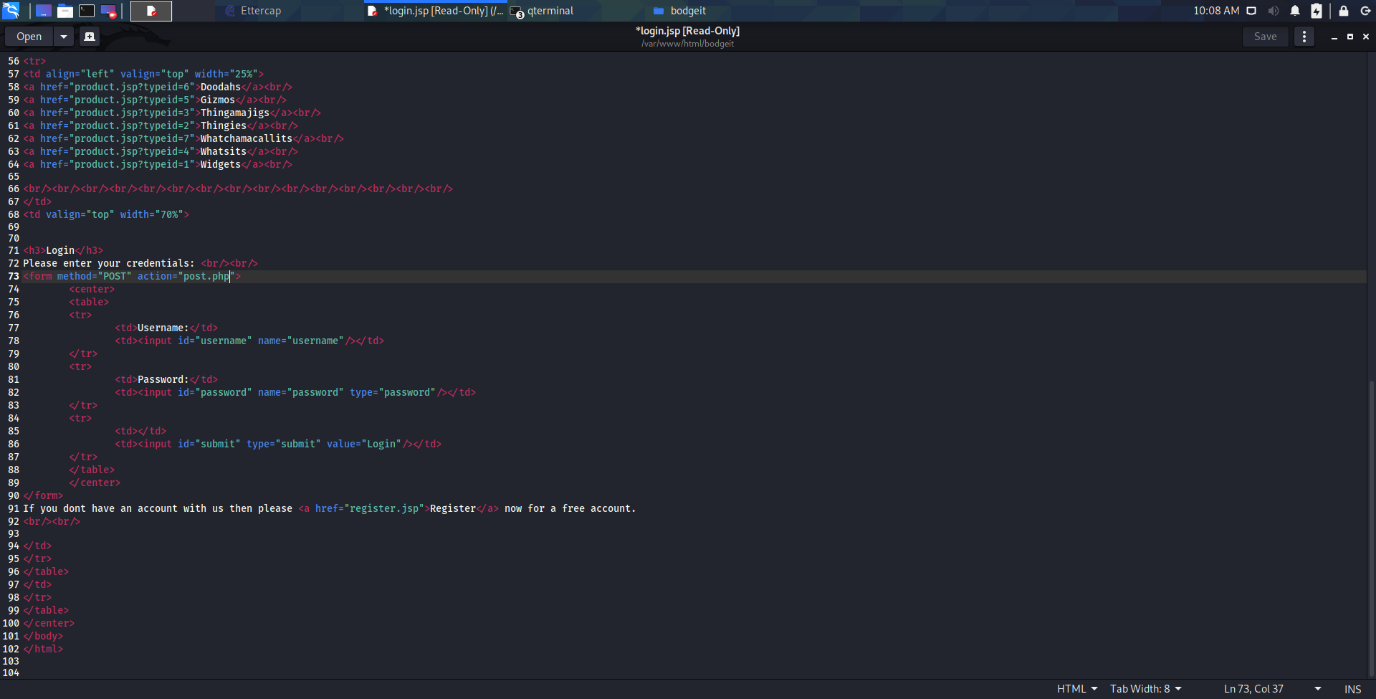


Figure 3. 11 – Phishing

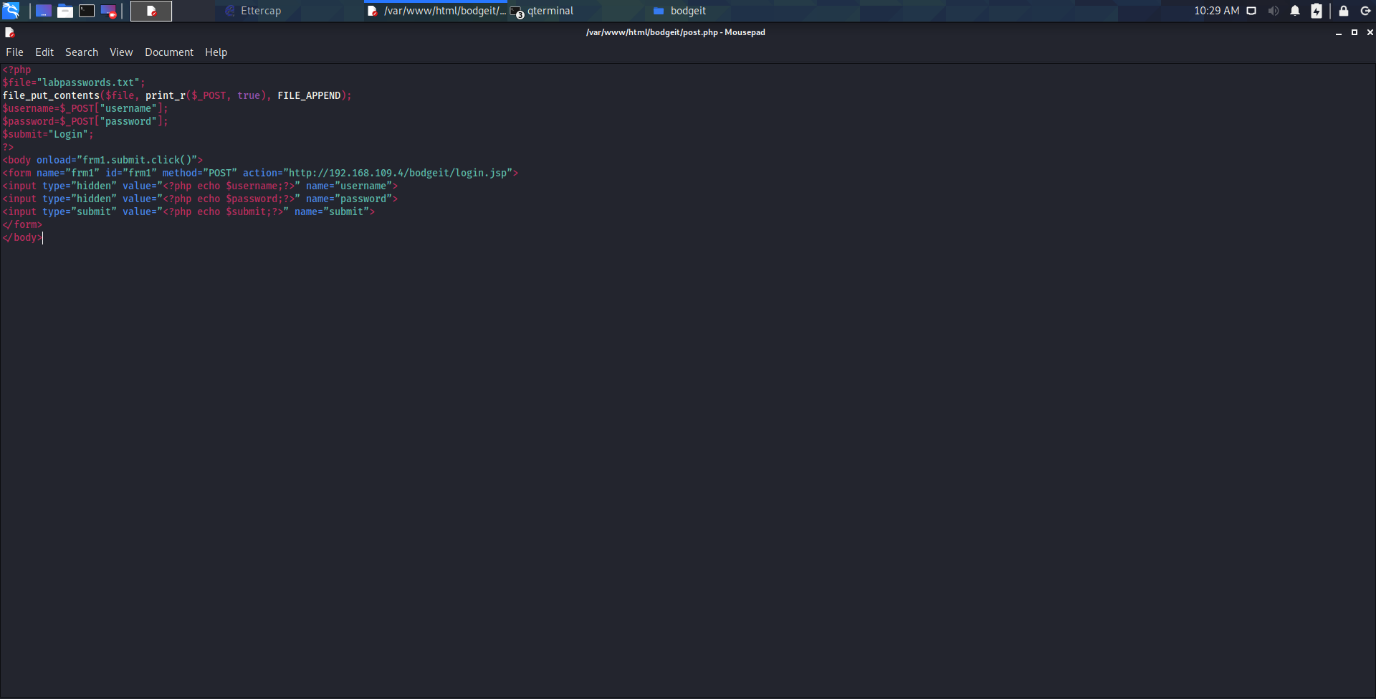


Figure 3. 12 – Phishing

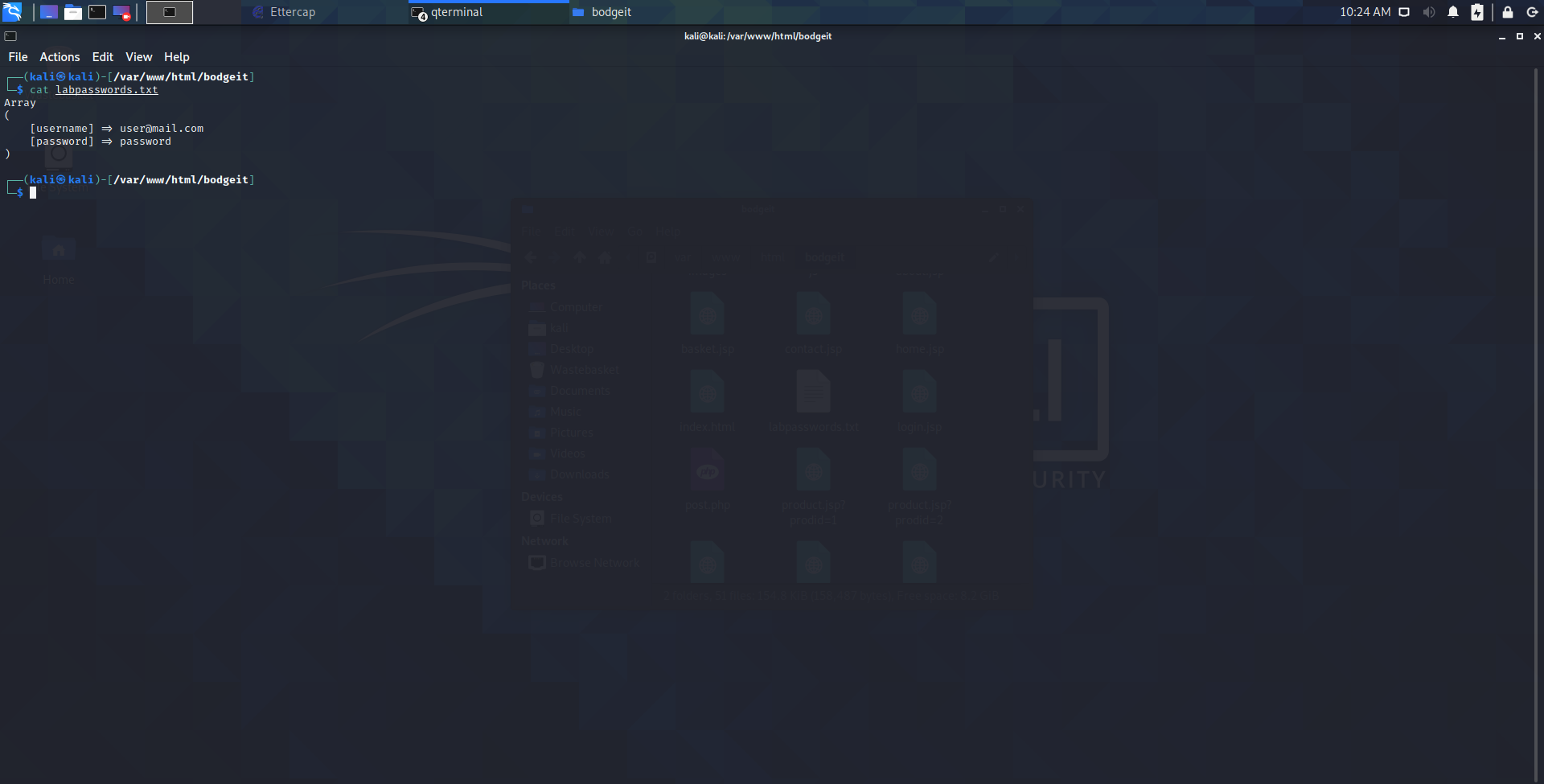


Figure 3. 13 – Phishing

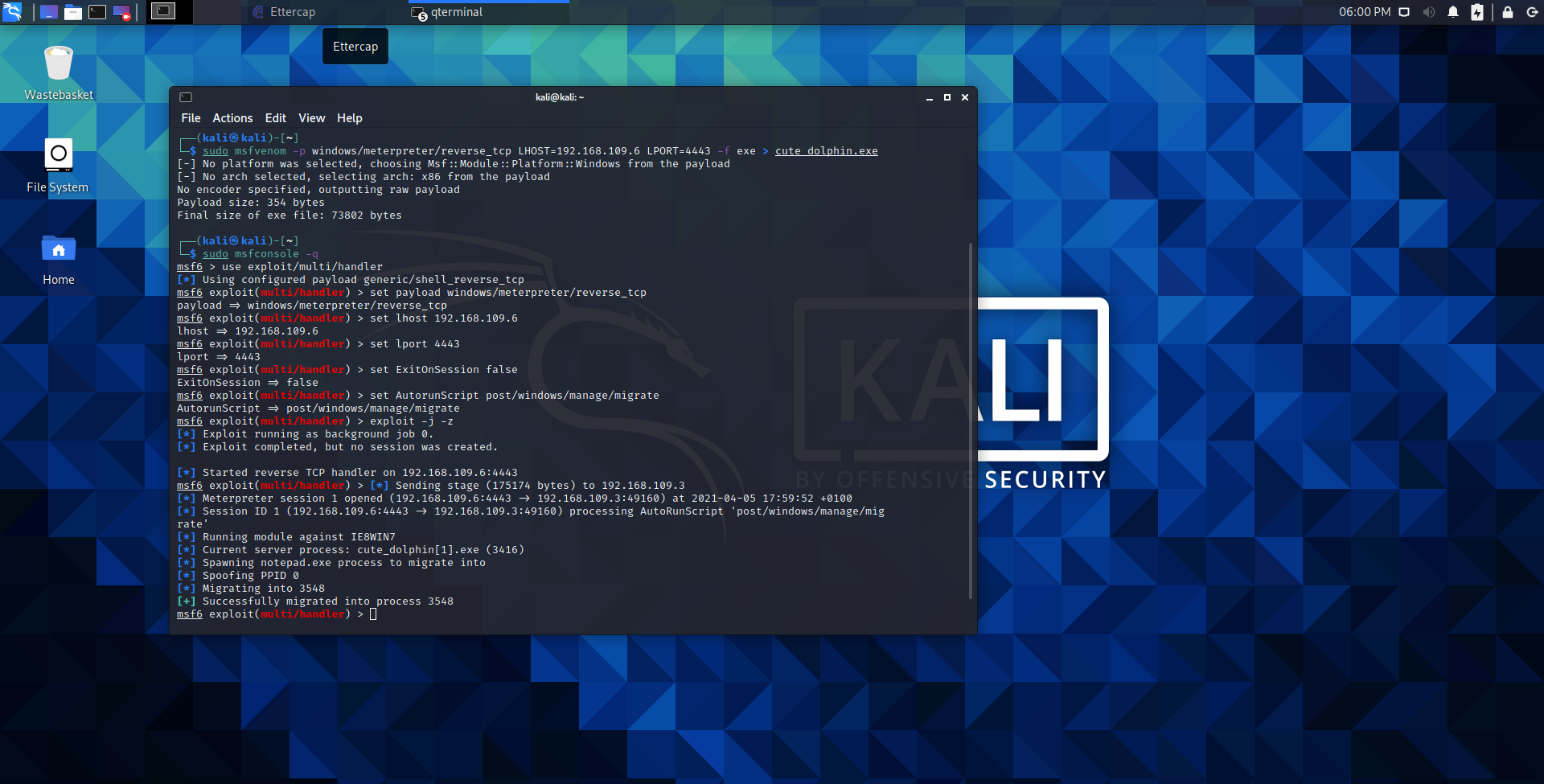


Figure 3. 14 – Metasploit

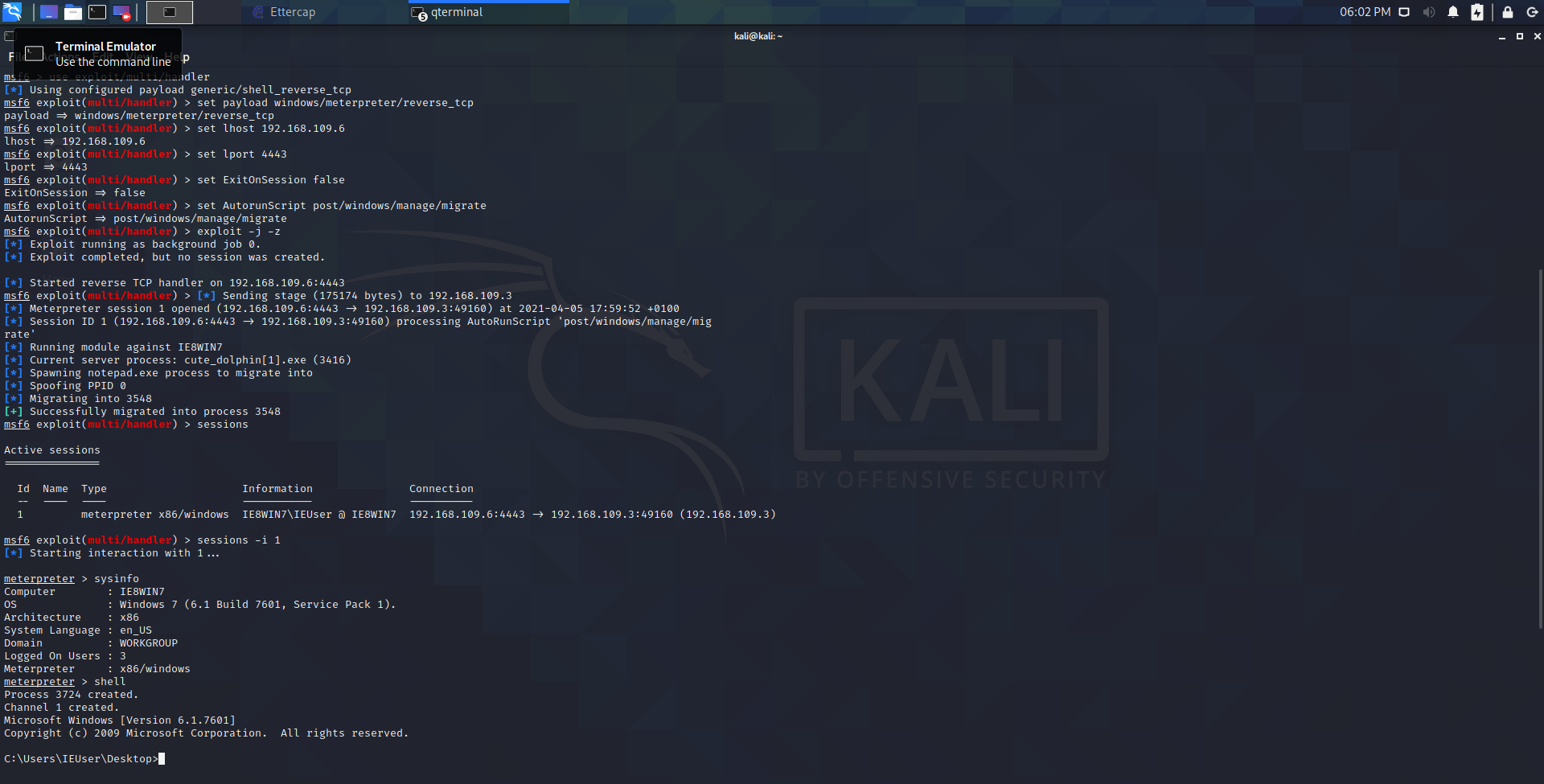


Figure 3. 15 – Metasploit