Case Study 4 - Metasploit

Aloysius Egbo Ike

College of Arts & Sciences, Niagara University

CSO 550: Ethical Hacking

Dr. Glenn Papp

November 24, 2023

INTRODUCTION

The ongoing threat of cyberattacks necessitates constant innovation in defensive strategies in the dynamic field of cybersecurity. When it comes to penetration testing and vulnerability assessment, the Metasploit Framework is a particularly potent and adaptable toolkit that both malicious hackers and ethical hackers can use.

Enigbokan and Ajayi, 2017 in his article while discussing the results of interviews with cyber-security experts on how to manage cybercrimes, identified security measures that can be tailored to any organization, including vulnerability assessments, and penetration tests, and also discussed specific measures for managing common forms of cybercrimes, such as phishing.

Among the common themes identified as types of security assessments, the article discussed that infrastructure assessments as identified by the experts is one of the vulnerability assessments that can be conducted by organisations. The article further stated that they are two types of infrastructure assessments: internal and external. Internal assessments involve penetration testers going to clients' sites to conduct vulnerability scanning in an authenticated mode, while external assessments are usually carried out from outside an organization's network using penetration testing tools such as Metasploit to analyze the corporate network infrastructure. These assessments help determine the level of exposure of an organization's corporate network to external attacks.

Leszczyna, (2021) in his research paper reviewed the existing cybersecurity assessment methods and identified thirty-two methods but focused on their applicability in realistic contexts and environments. Along with nmap, nessus, and wireshark, Metasploit is also mentioned as one of the most common tools for penetration testing and also supports the assessments of SCADA systems.

The Metasploit Framework is a Ruby-based, modular penetration testing platform that enables users to write, test, and execute exploit code. The Metasploit Framework contains a suite of tools that can be used to test security vulnerabilities, enumerate networks, execute attacks, and evade detection.

For this project, we will me using the Metaspliot framework on Kalu to exploit our Metasploitable on IP address 192.168.1.9.

First, we start by shooting up Kali terminal and using Msfconsole command through Kali's terminal to invoke the interactive Metasploit console. It is the most common method to run the msfconsole command as seen in Fig.1 below.

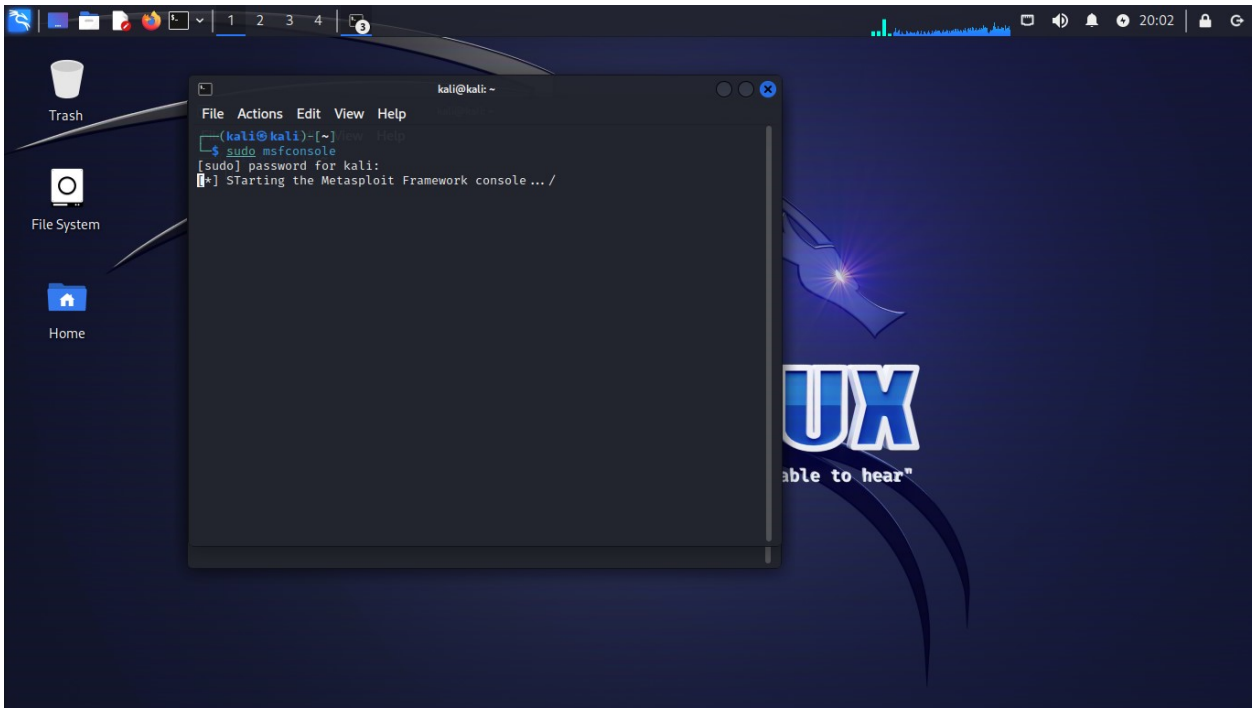

Fig. 1. Screenshot of Kali running msfconsole command.

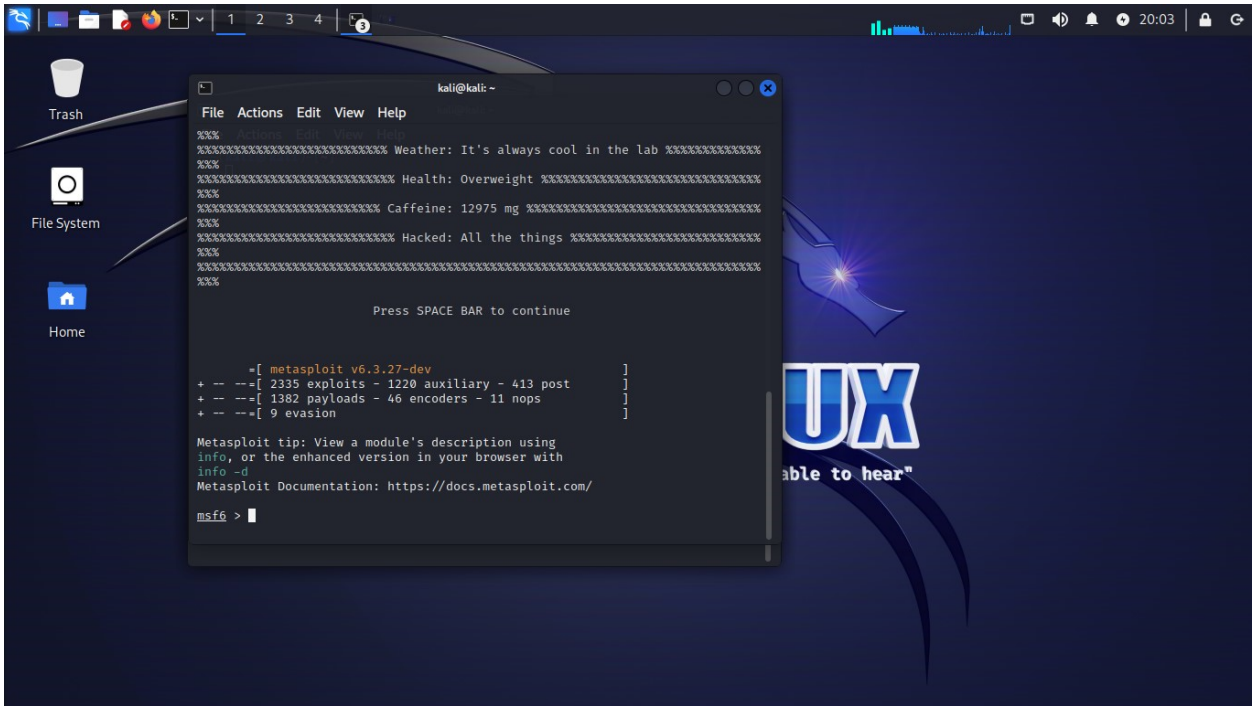

Fig. 2. Screenshot of msfconsole completed.

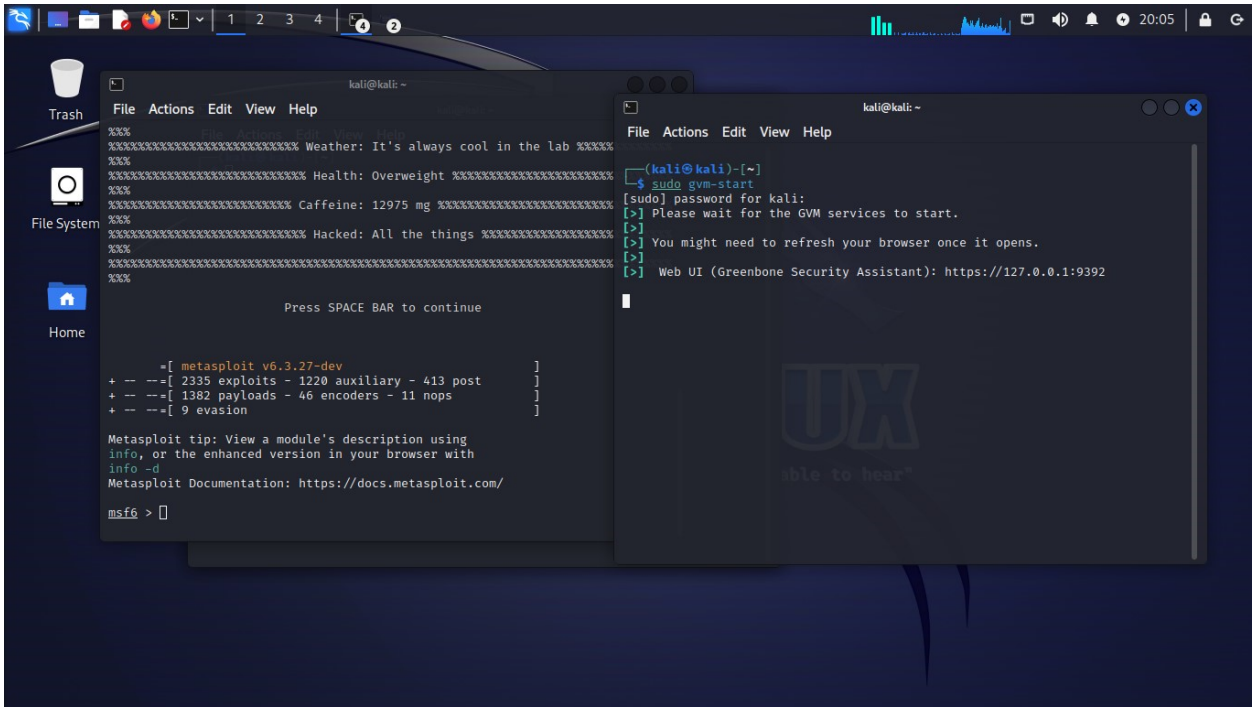

Fig. 3 Screenshot of sudo gvm-start command.

I ran this command so as to access Greenbone and find the ports that are open on Metaspoitable. These ports are the ones that can be exploited using Metaslpoit.

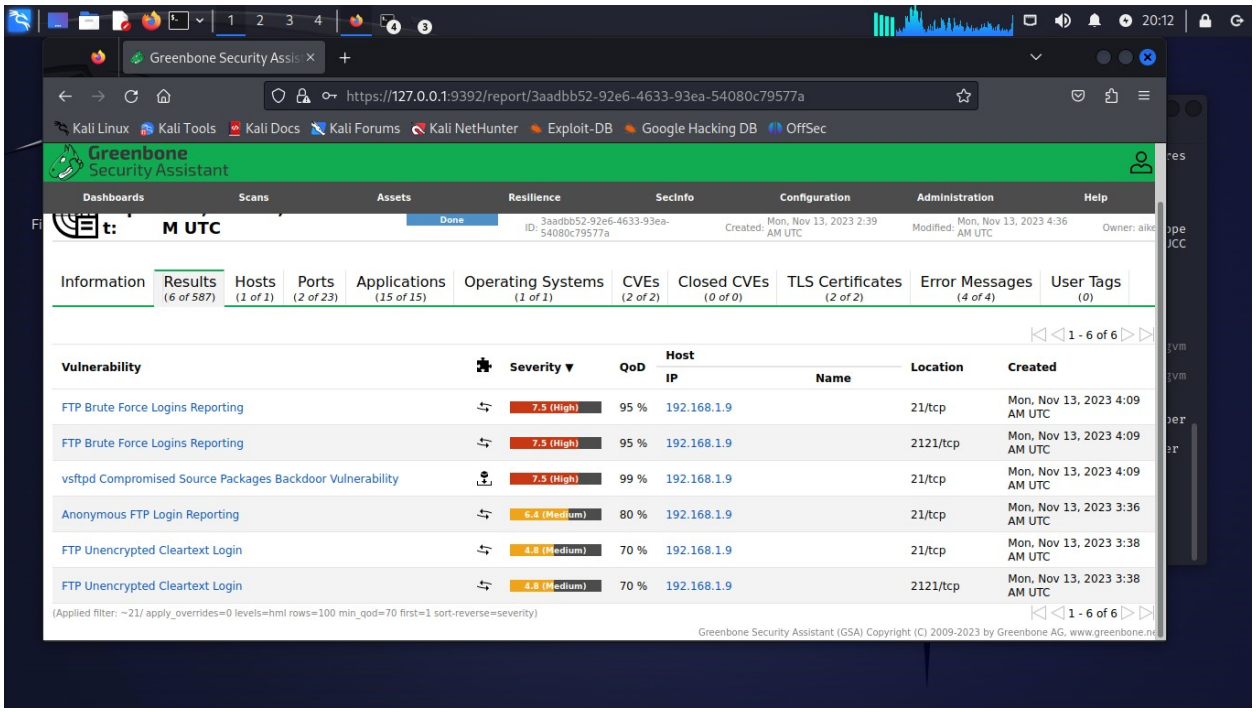

Fig. 3. Screenshot of port 21 that are open.

The screenshot above from Greenbone showed the type of vulnerability that's runs on port 21 which is FTP/TCP

Do I have any vulnerabilities in my results? Yes I did. To find these vulnerabilities in my results, I go to Scan, tasks and then click on the results tab. The high results shows the severity of the vulnerabilities.

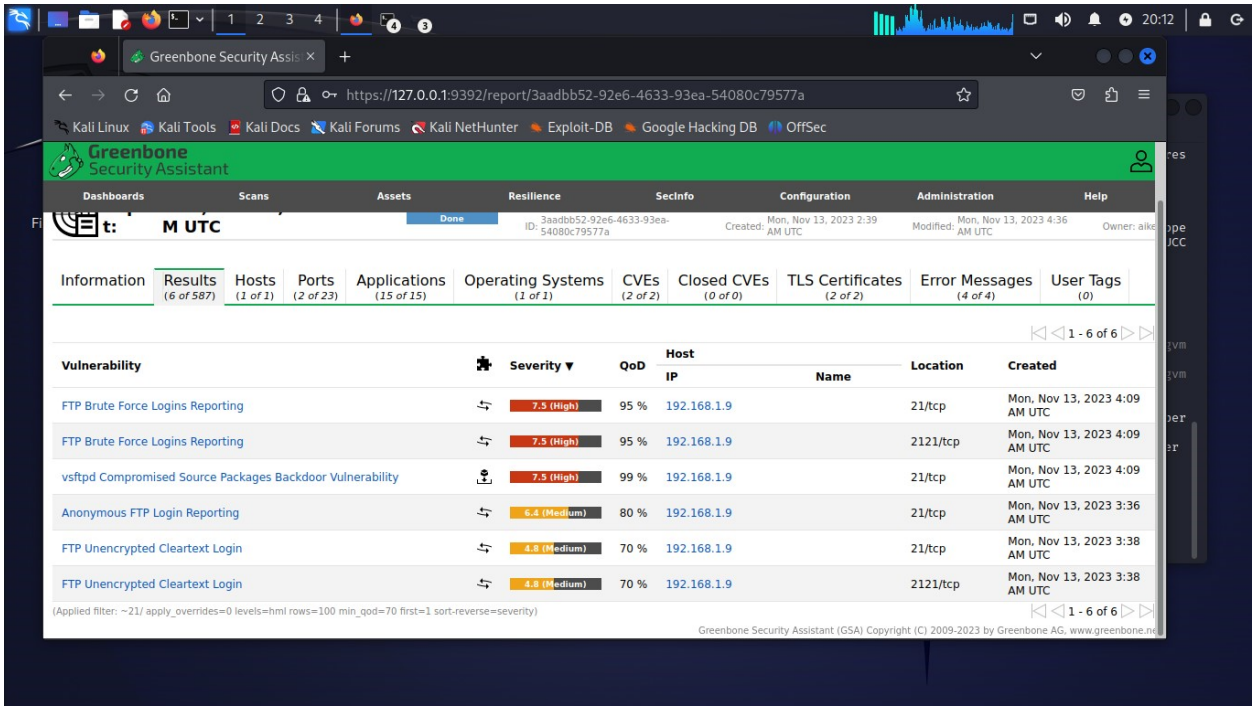

Fig. 4. Screenshot of port 21 that are open.

From the vulnerabilities listed from the search I ran for port 21, I found that one of the results is a backdoor vulnerability, which is the “vsftpd Compromised Source Packages Backdoor Vulnerability”.

A backdoor vulnerability is a unique method for avoiding standard authentication protocols in order to obtain unauthorized access to a system. Backdoor vulnerability usually entail the exploitation of vulnerabilities in the system or the installation of malicious software that opens a point of access for the attacker.

Miller et al., (2021), while discussing the evolution and significance of Industrial Control Systems (ICS) in critical infrastructure, introduced a paper that explores a set of publicly disclosed ICS cyberattacks, aiming to provide a comprehensive understanding of historical

attacks, analyze trends, and offer lessons learned with suggested actions to enhance security capabilities.

One of those attacks was the attack on a backdoor vulnerability for the Niagra AX ICS, which was posted online, allowing a user to connect to the system through an IP address with no authentication. Using this method, attackers gained access to the ICS used by an air conditioning company to control the organisations' heating and air conditioning.

This attack highlights not just the cost implications but also, the risk level of this type of vulnerability.

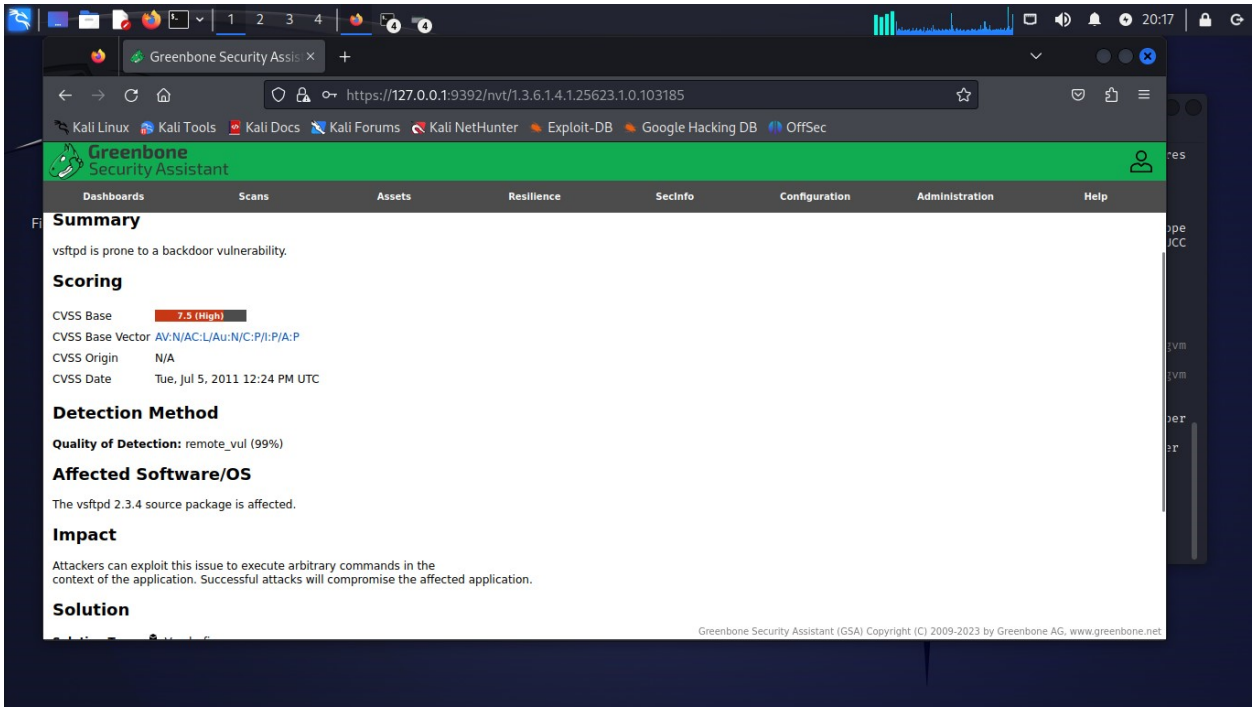

Fig 5. Screenshot showing the summary and impact of the vsftpd vulnerability.

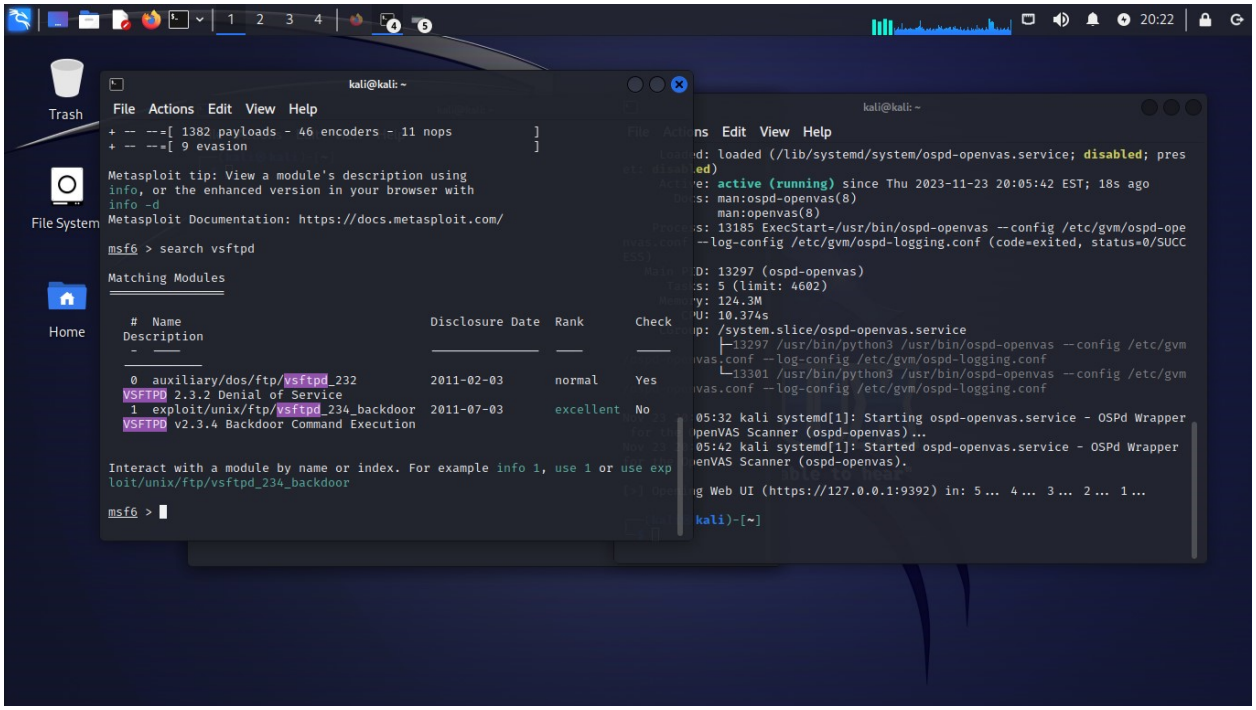

Fig 6. Screenshot of search command for vsftpd

I ran a search for the “vsftpd” backdoor vulnerability on the Metasploit framework and got two results as seen in Fig. 6 above but the one I should be interested in is the backdoor vulnerability found in the result which is the one with the Backdoor Command execution. I typed type the path that the result gave me which is “exploit/unix/ftp/vsftpd_234_backdoor”. I use the “use” command with the path from the screenshot in Fig. 6 and it returned the result in Fig 7. below.

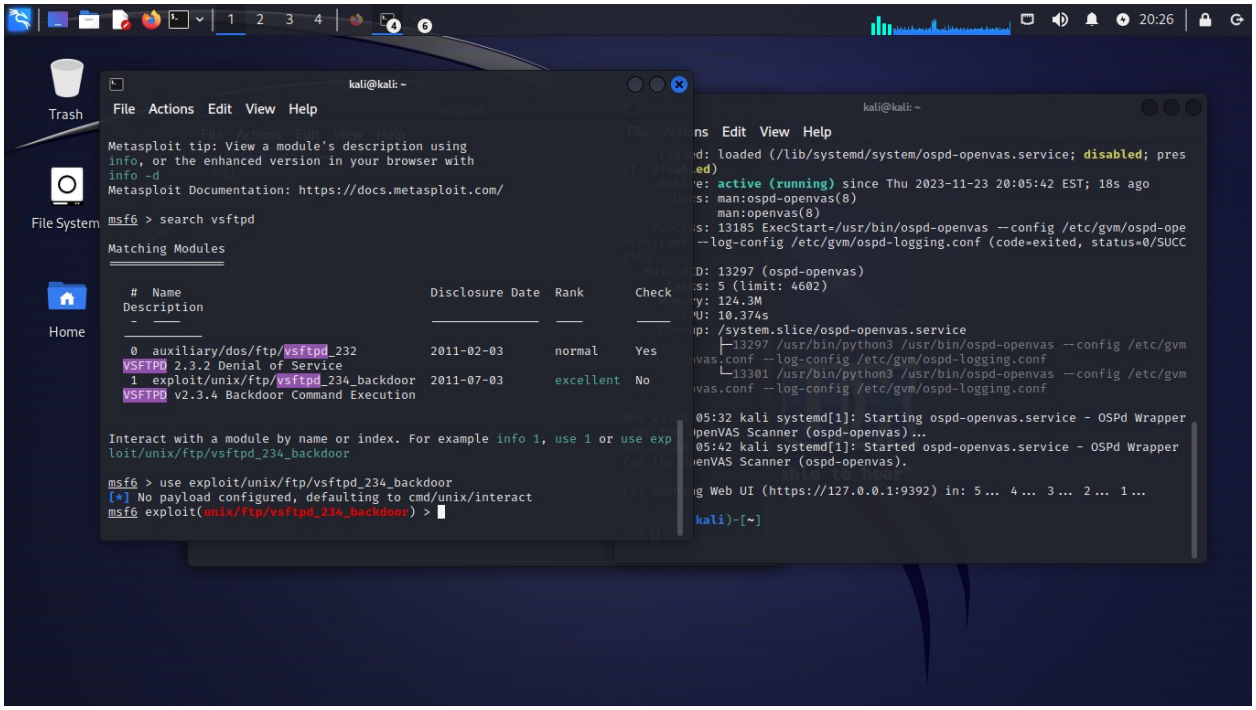

Fig. 7 Screenshot of use command on the backdoor command

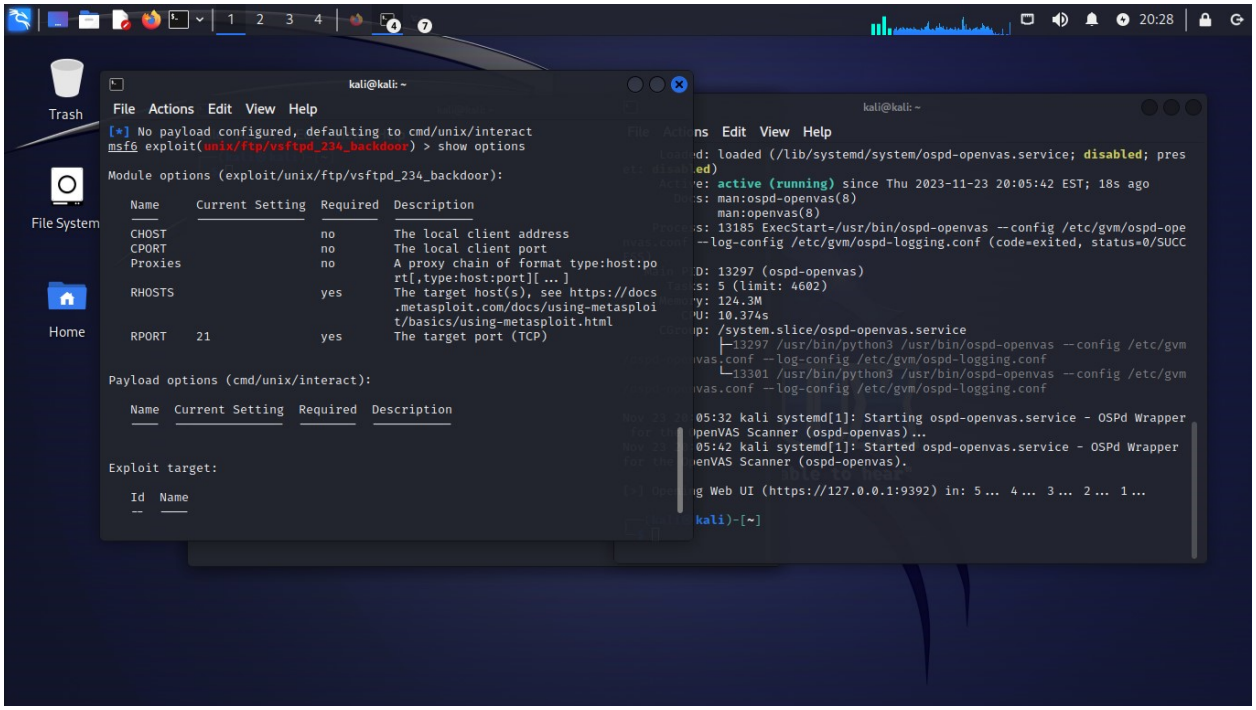

Fig 8. Screenshot of the result show options command

The show options command shows the available parameters for an exploit if used when the command line is in exploit context. The exploit contains a total of 5 options from which only 2 are required, the RHOSTS and the RPORT.

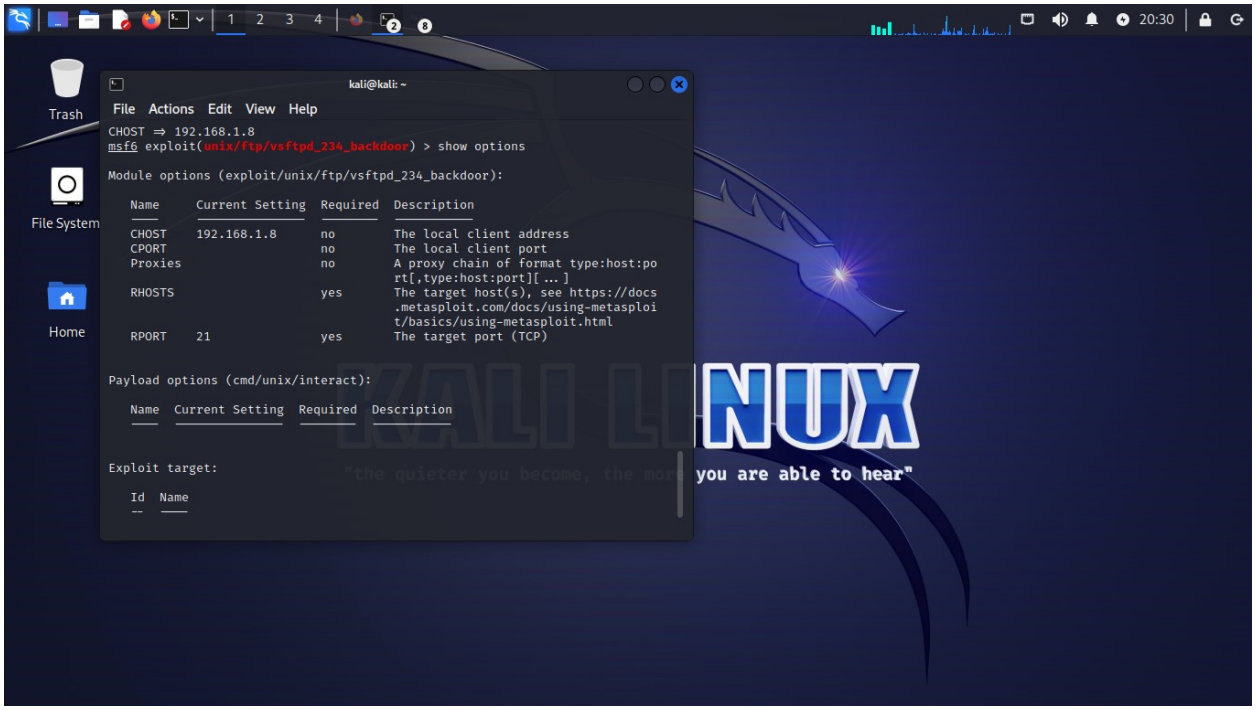

Fig. 9 Screenshot of the CHOST settings changed.

The CHOST setting, while not required was set to the IP address of Kali, the host system.

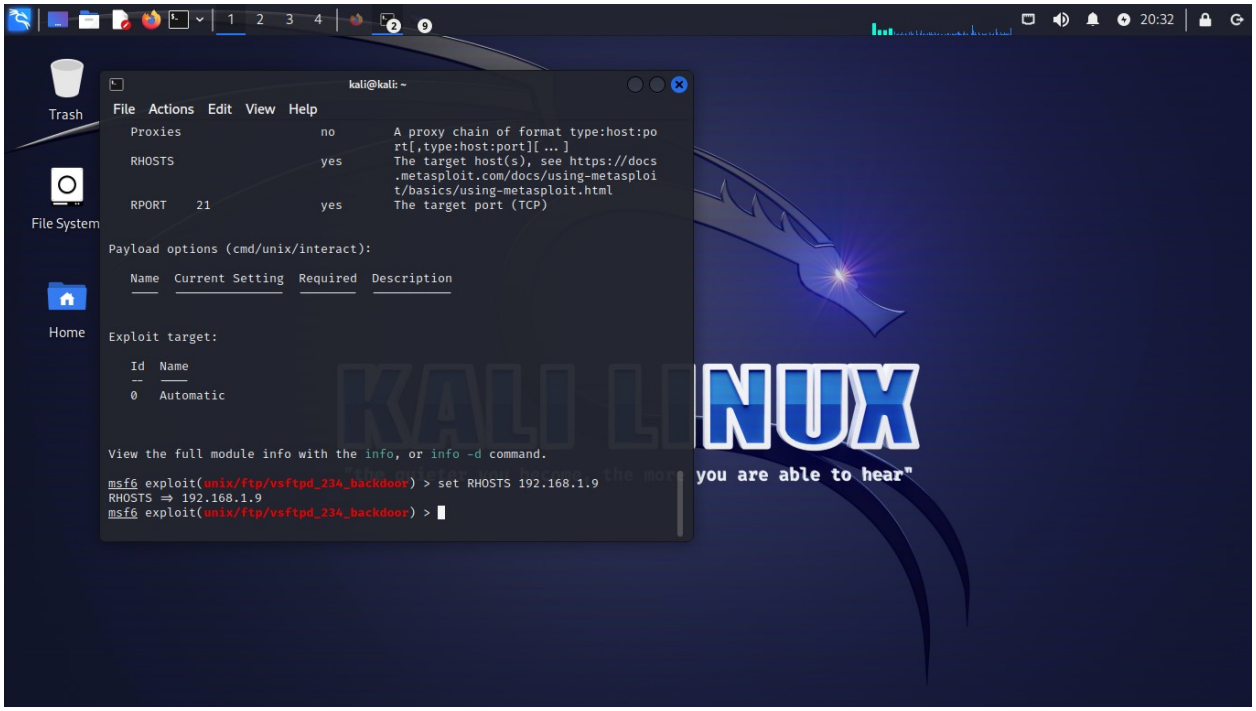

Fig 10. Screenshot of set command on RHOSTS to change the settings.

While discussing the importance of protecting critical infrastructures (CIs) such as water, electricity, and transportation systems, Gonzalez-Granadillo, et al (2021), in his journal, described an active attack against a water critical infrastructure where the attacker launches a Kali machine and uses PostgreSQL and Metasploit to execute a Modbus auxiliary attack against the victim database. The attacker then reads information from the database by specifying the target IP address using the RHOST, and target port number using RPORT. After executing the attack, the attacker succeeds in modifying the registers in the database, which creates an abnormal situation with temperature values falling outside the threshold. RHOSTS is the target system which in our case study, is the IP address of Metasploitable.

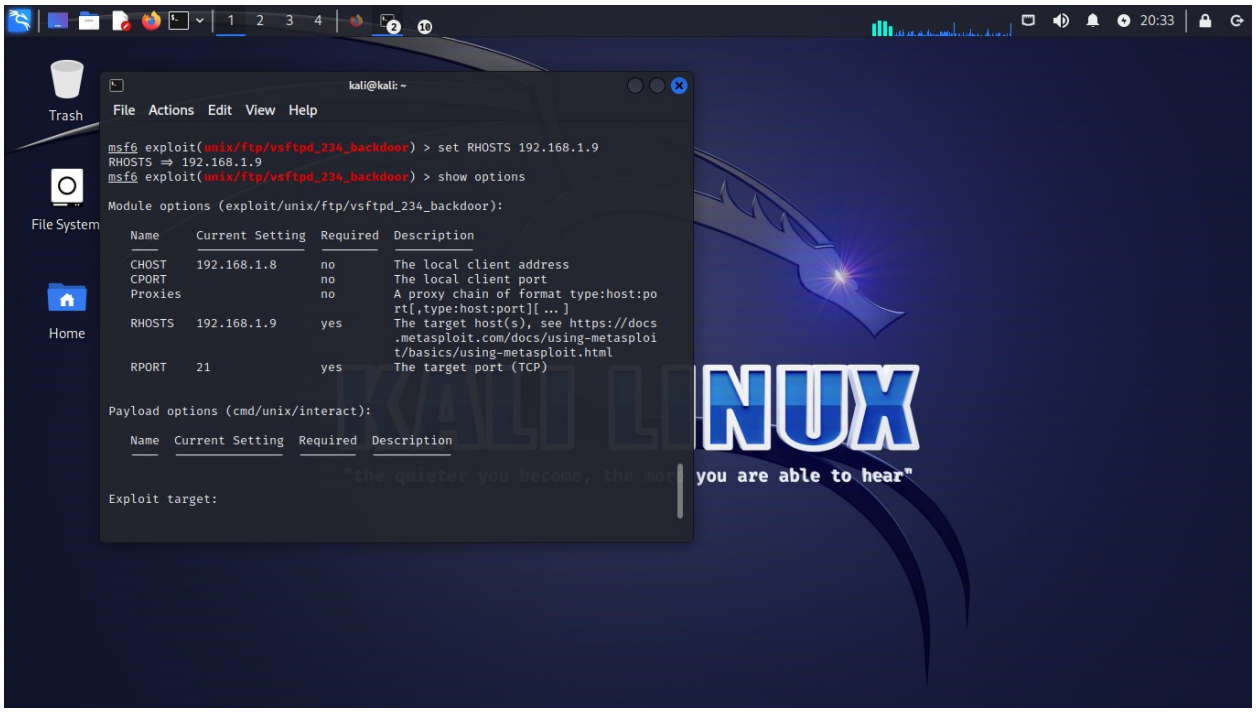

Fig 11. Screenshot showing the complete settings on vfstpd

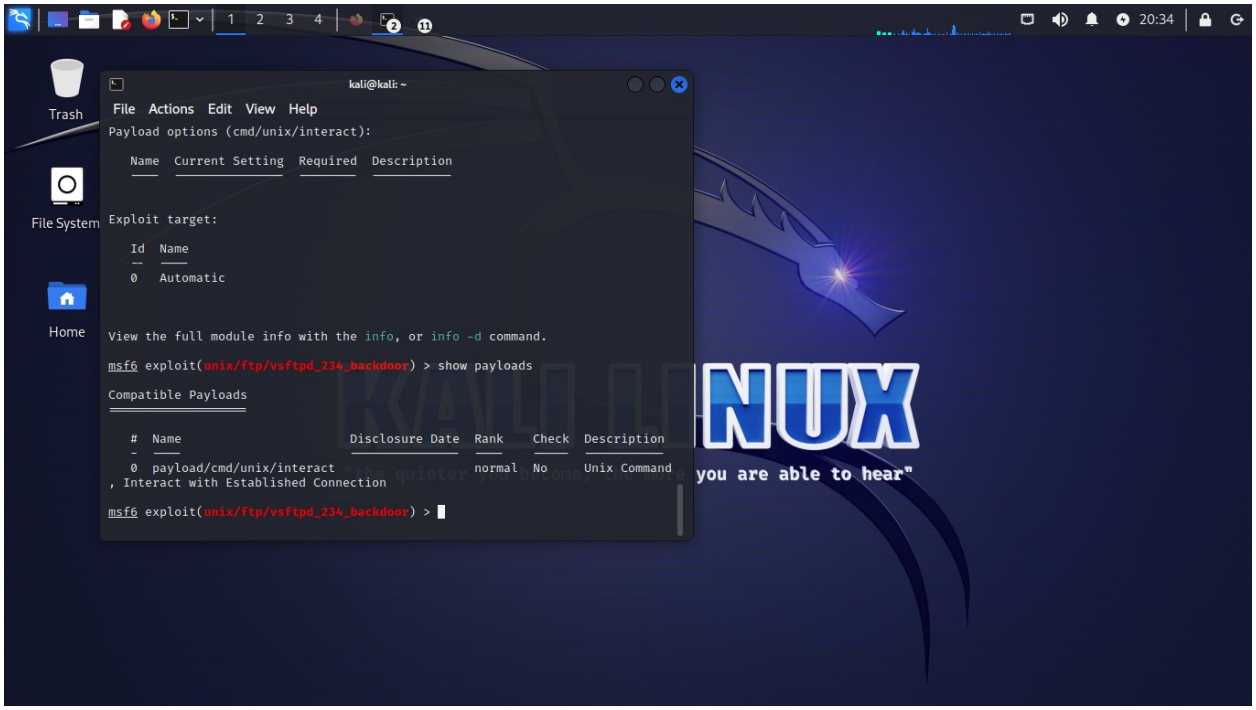

Fig 12. Screenshot of show Payload command

The show payloads command in the msfconsole will return a list of compatible payloads for this exploit. In our vfstpd exploit case study, it return the compatible payloads as seen in the screenshot.

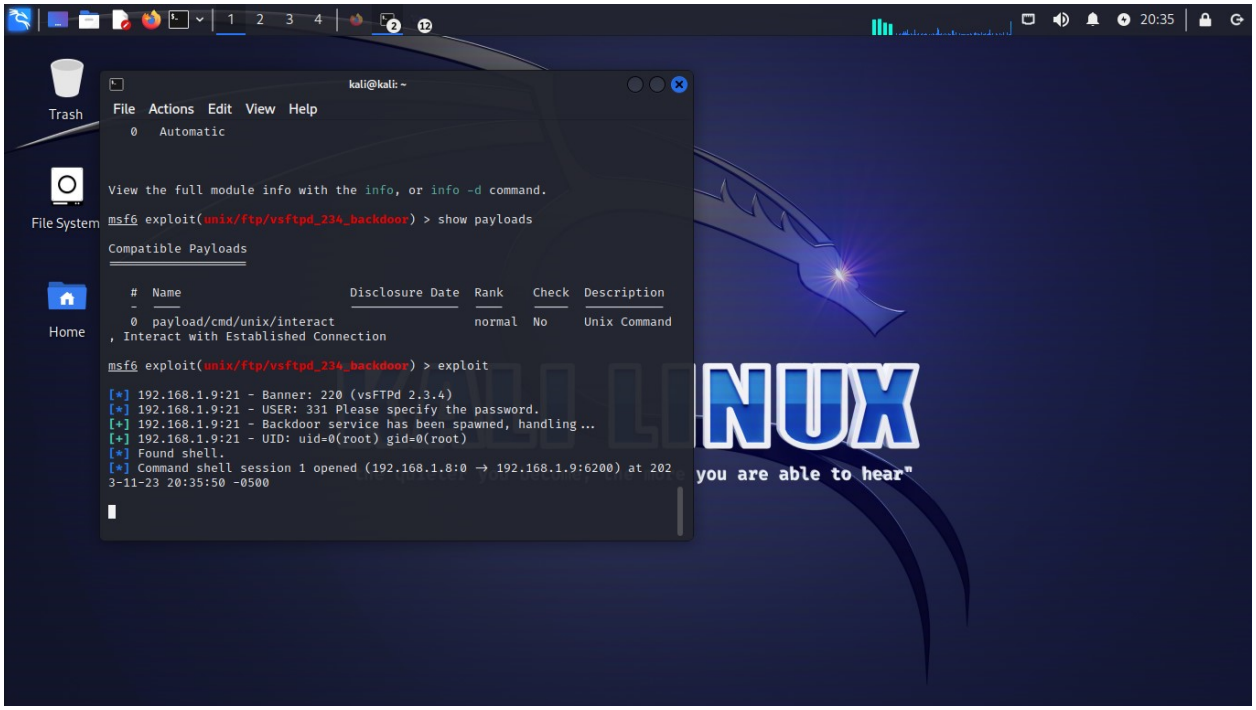

Fig 13. Screenshot showing successful exploit.

Exploit successful!!

When all the required options have been set for the exploit, including a payload the exploit is ready to be executed. The exploit is then executed using two commands: run and exploit. By typing run or exploit in the msfconsole and the exploit will run.

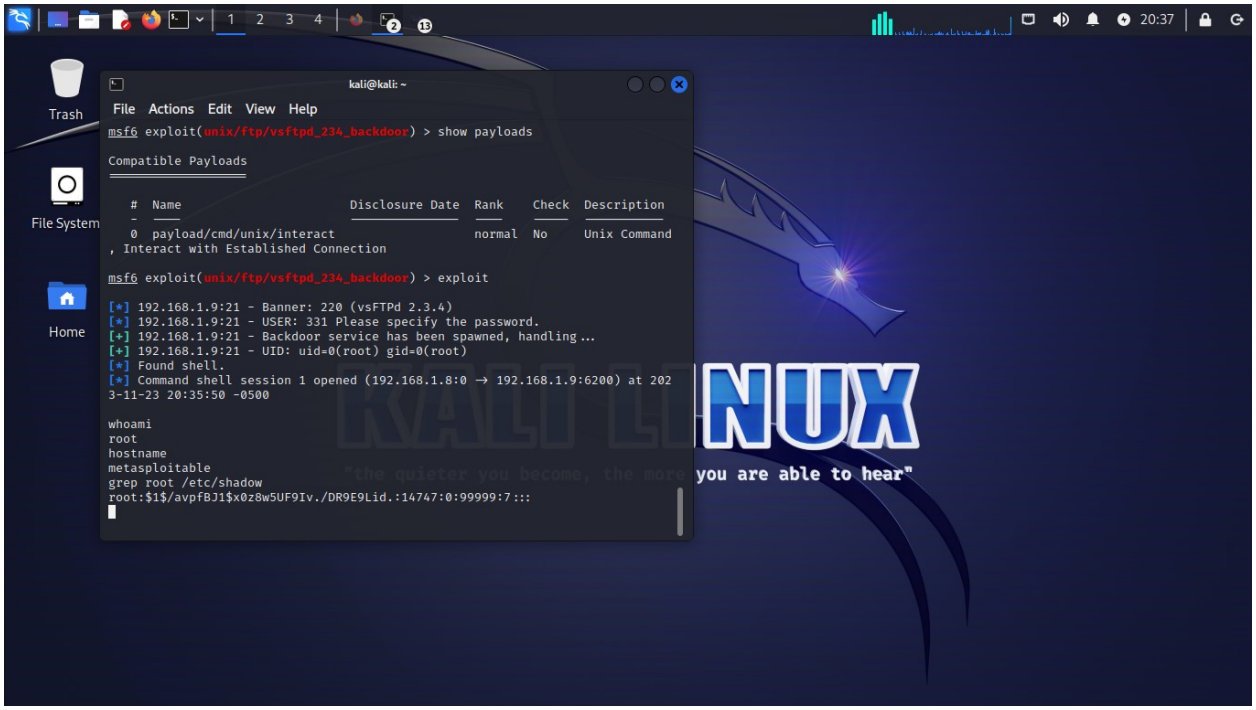

Fig. 14 Screenshot showing the whoami command

The whoami command allows the users to see the authentication level of the currently logged-in user. The output displays the username of the effective user in msfconsole.

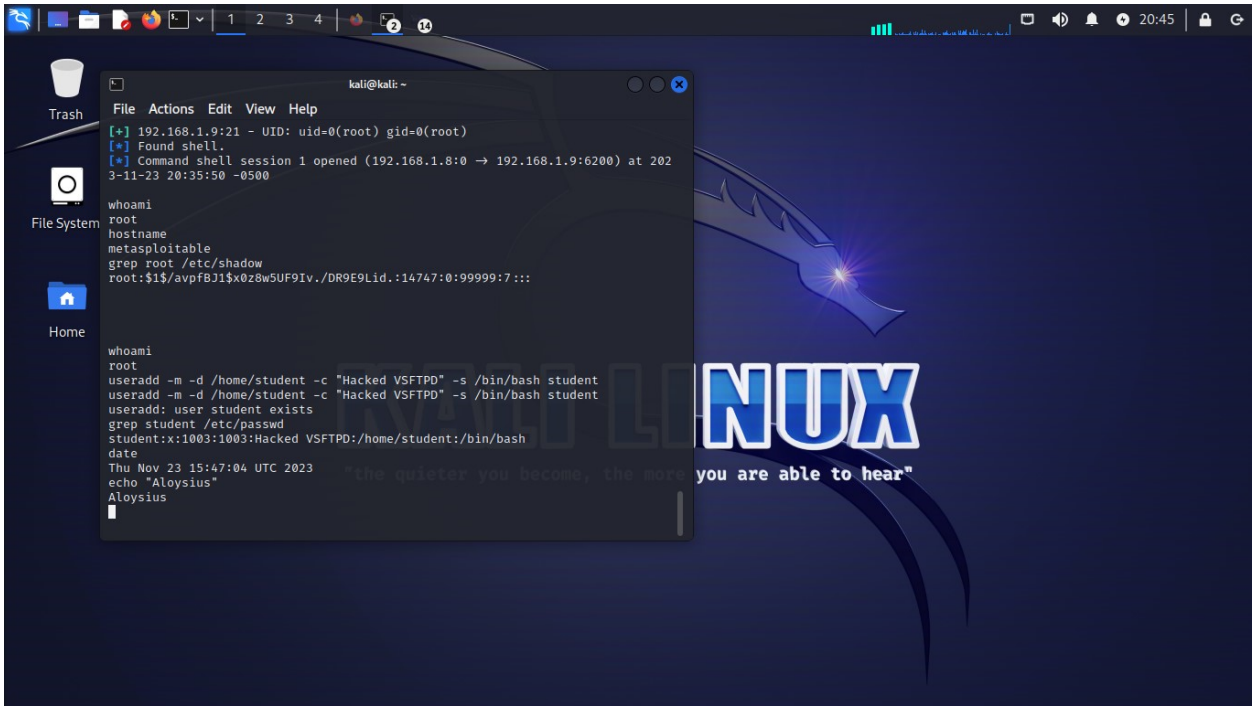

Fig. 15 Screenshot showing that I have root access.

To confirm this, I ran the sudo shutdown 0 command to remotely shutdown the target system and it was successful from the screenshot Fig 16 below

What does the grep root /etc/shadow do? It essentially extracts a list of usernames from the "/etc/shadow" file. The "/etc/shadow" file stores the encrypted passwords of the users on the system and some additional properties.

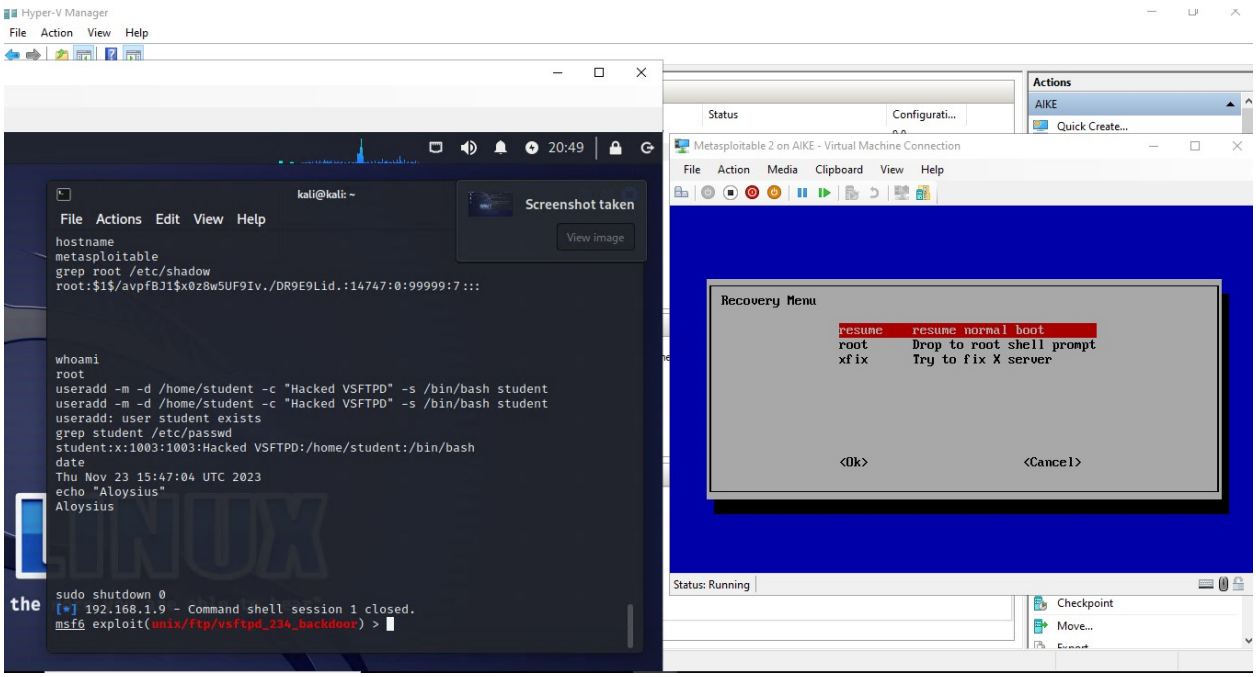

Fig 16. Screenshot showing a successful shutdown of Metasploitable from Kali

The screenshot above shows the successful exploit of the target system using the vfstpd cyber threat, back door vulnerability. To mitigate risks such as this, Rizvi et al., (2020) emphasizing the critical need to address security controls proactively in various domains to safeguard against cyber threats, particularly in the context of Internet of Things (IoT) devices, specifically highlighted vulnerabilities related to backdoors in IoT devices, acknowledging that some devices are delivered with intentional backdoors for remote access and debugging purposes. The proposed solution involves establishing a comprehensive policy and procedure infrastructure, documentation, and defined limits for developers regarding backdoors, along with the implementation of security measures such as monitoring authentication attempts and disabling unused Telnet/SSH ports.

The discussion extended to the security challenges associated with IP cameras and smart hubs, emphasizing the importance of implementing security controls to combat vulnerabilities. For IP

cameras, the proposal suggests the adoption of a layered approach, including hardened password implementation, multi-factor authentication, and restricting login attempts. Regarding smart hubs, the focus is on addressing vulnerabilities introduced by third-party applications through internal vetting, defensive programming methodologies, and proper application permissions.

Additionally, the text addresses the prevalence of Man-in-the-Middle (MITM) attacks in the IoT infrastructure and recommends a protocol change from unencrypted protocols like Trivial File Transfer Protocol (TFTP) to encrypted protocols such as Datagram Transport Layer Security (DTLS). The design philosophy of DTLS, its features, and its application for securing communication in IoT devices are discussed as part of the proposed solution.

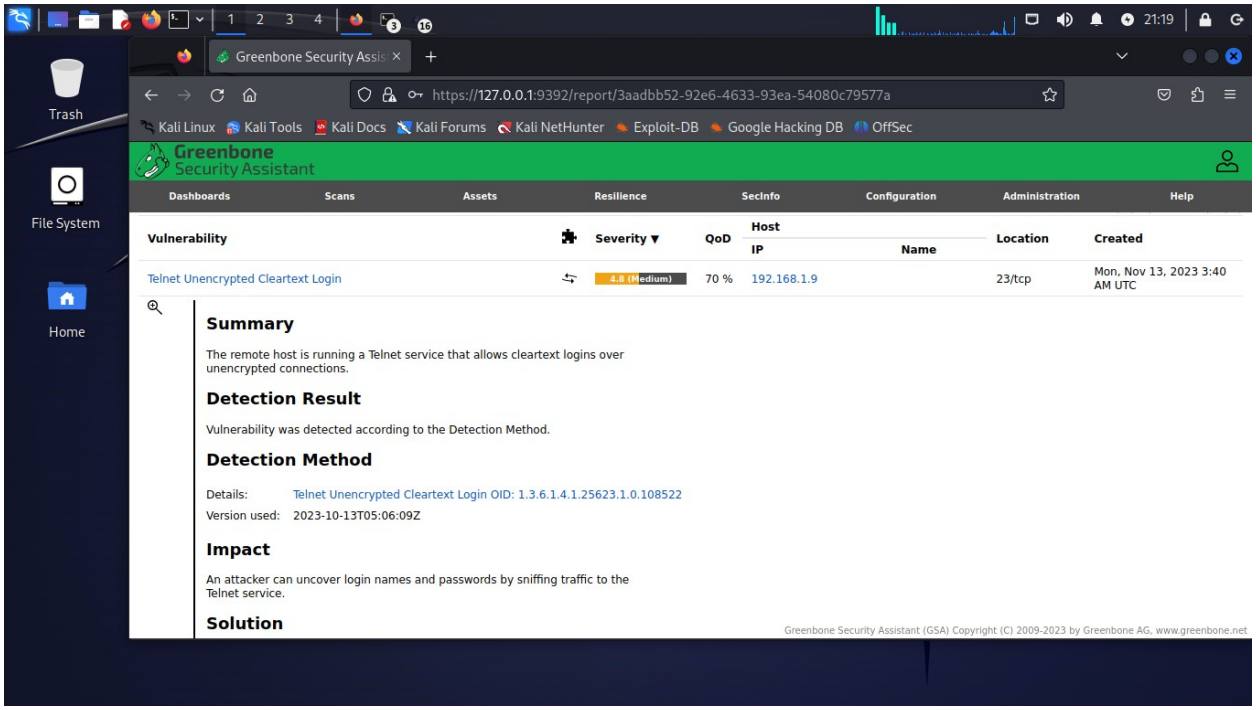

Fig 20. Screenshot showing the Telnet Unencrypted Cleartext Login

To figure out how to run an exploit on Port 23 using the telnet login exploit, first I did a search for the vulnerability in Greenbone, which I did from the screenshot above.

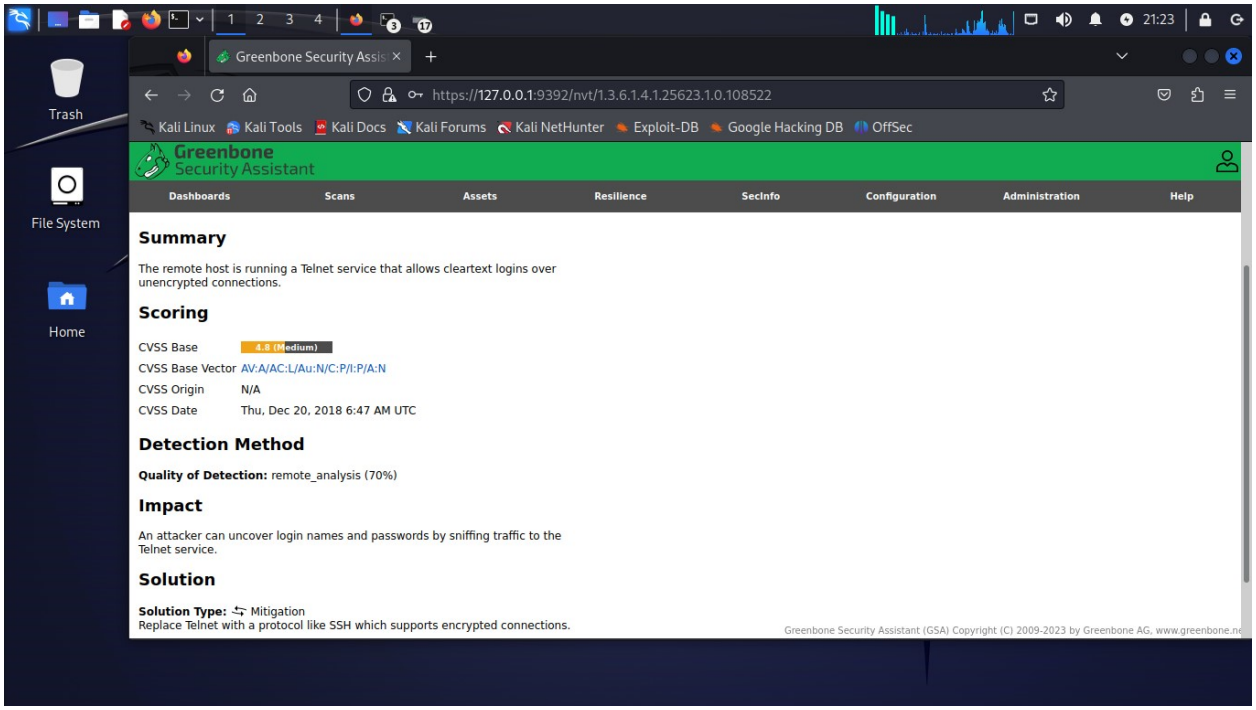

Fig 21. Screenshot of the description of the Telnet vulnerability with its impact and solution

From Greenbone, the impact of this vulnerability is that an attacker can uncover login names and passwords by sniffing traffic to the Telnet Service and to mitigate this risk, we replace Telnet with a protocol like the SSH which supports encrypted connections will have to be replaced.

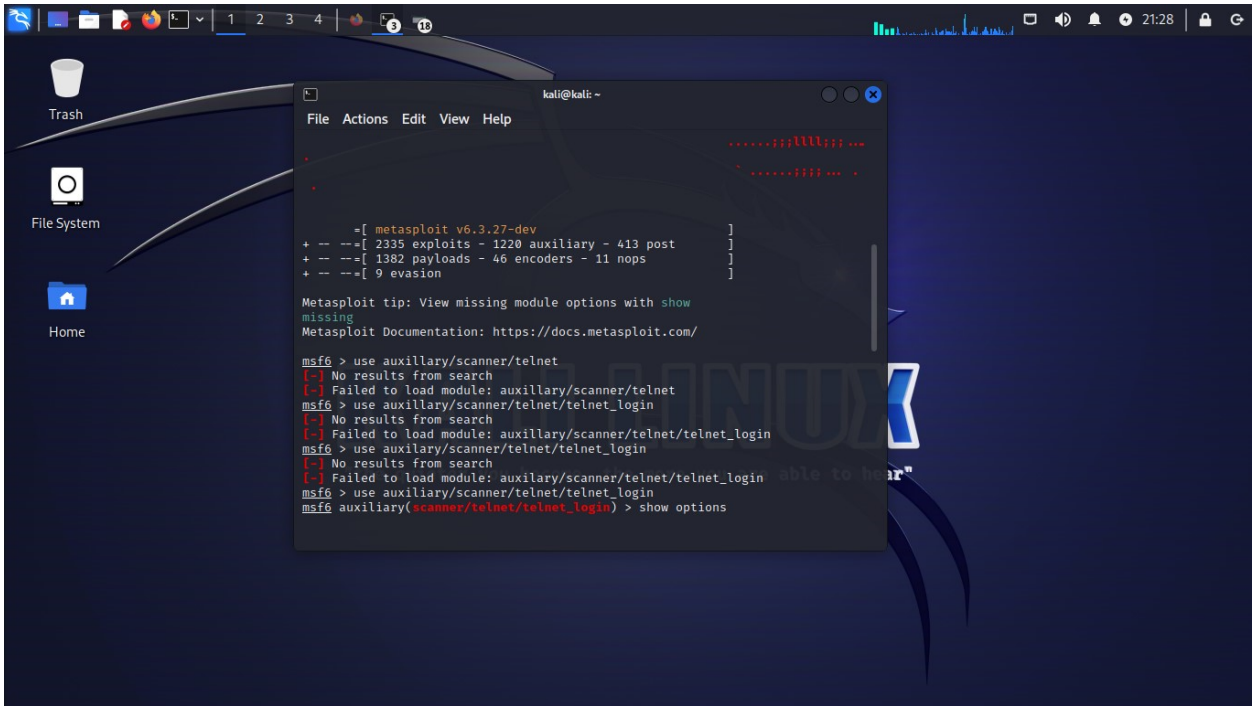

Fig 22. Screenshot showing the use command on auxiliary/scanner/telnet

The telnet_login module is designed to attempt login to Telnet servers using provided credentials and a range of specified IP addresses. This auxiliary module offers flexibility by allowing the user to pass credentials in various ways, such as setting a specific username and password, iterating through lists of usernames and passwords, or providing a file with username-password pairs. In practical application, the user configures the scanner to use specific files containing short usernames and passwords, then sets it to run against a designated subnet.

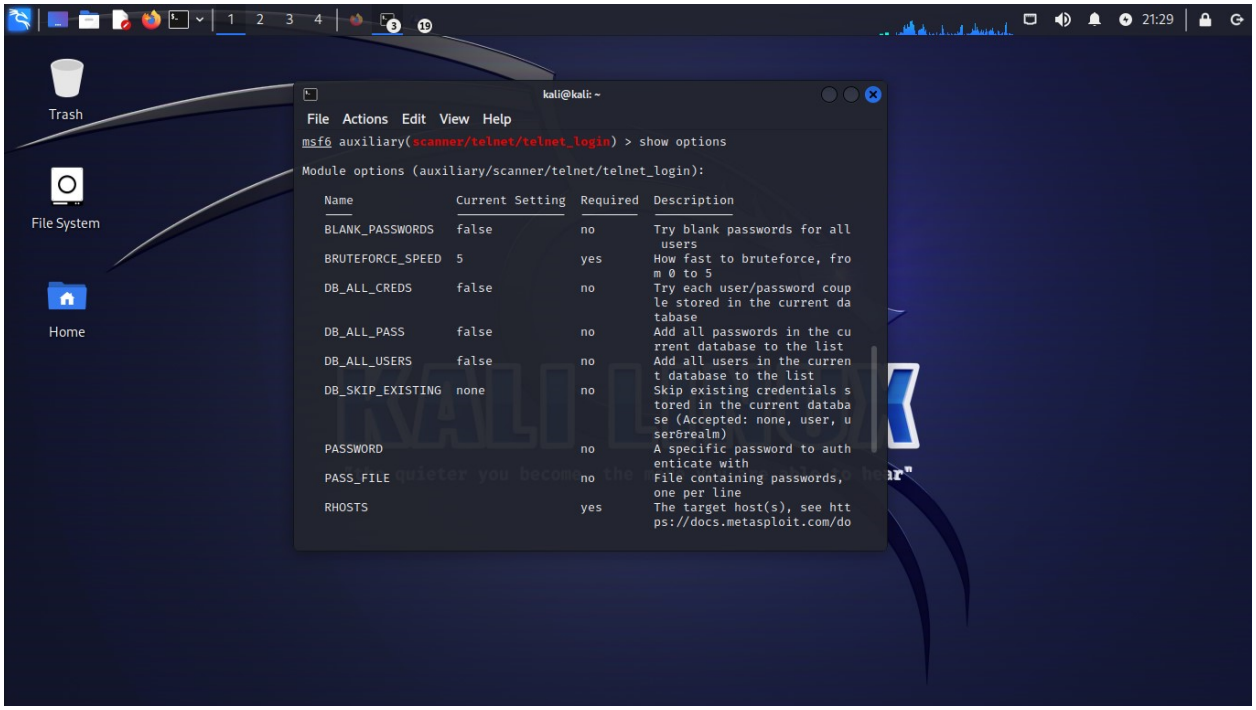

Fig 23. Screenshot showing the show option command

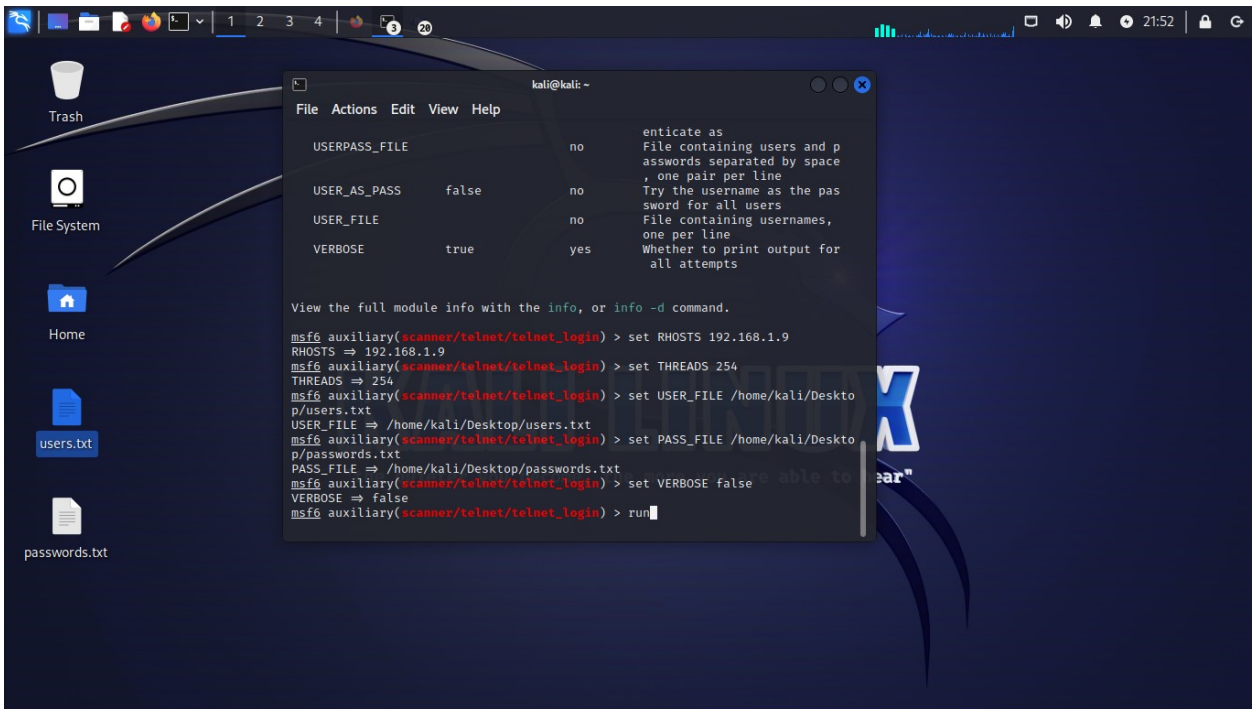

Fig 24. Screenshot showing how I changed the settings using set command

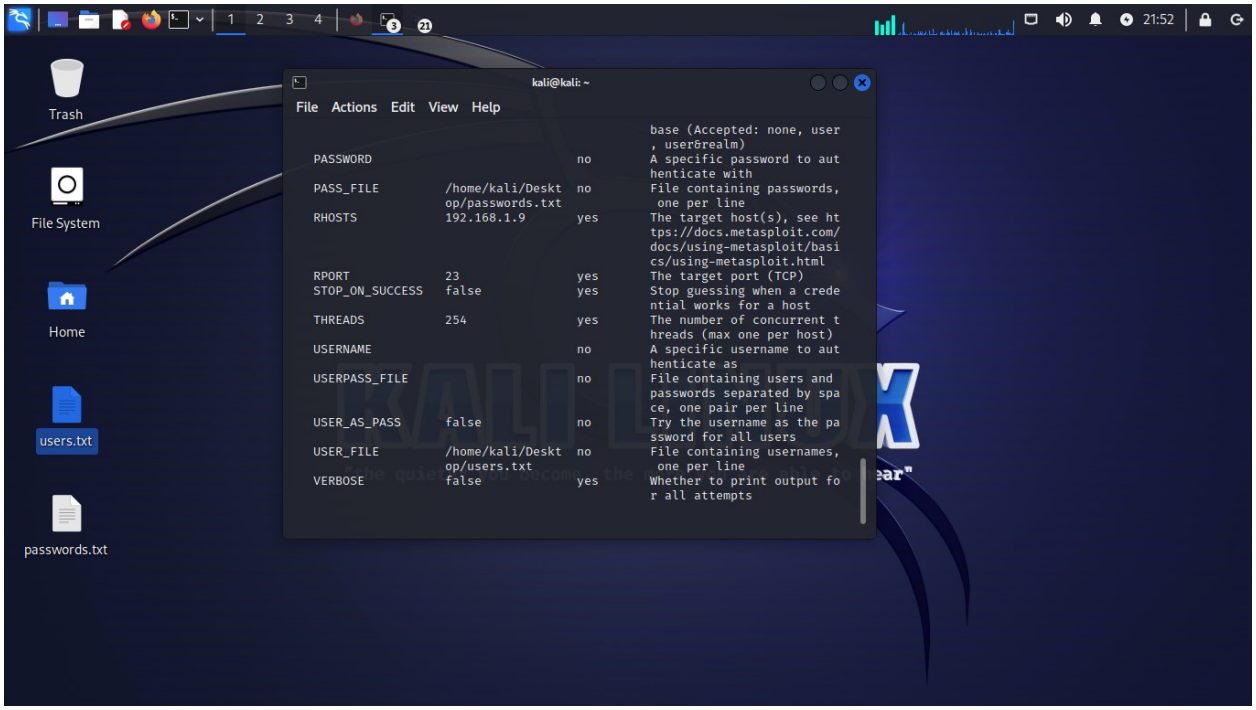

Fig 25. Screenshot showing the show options command with the parameters all set

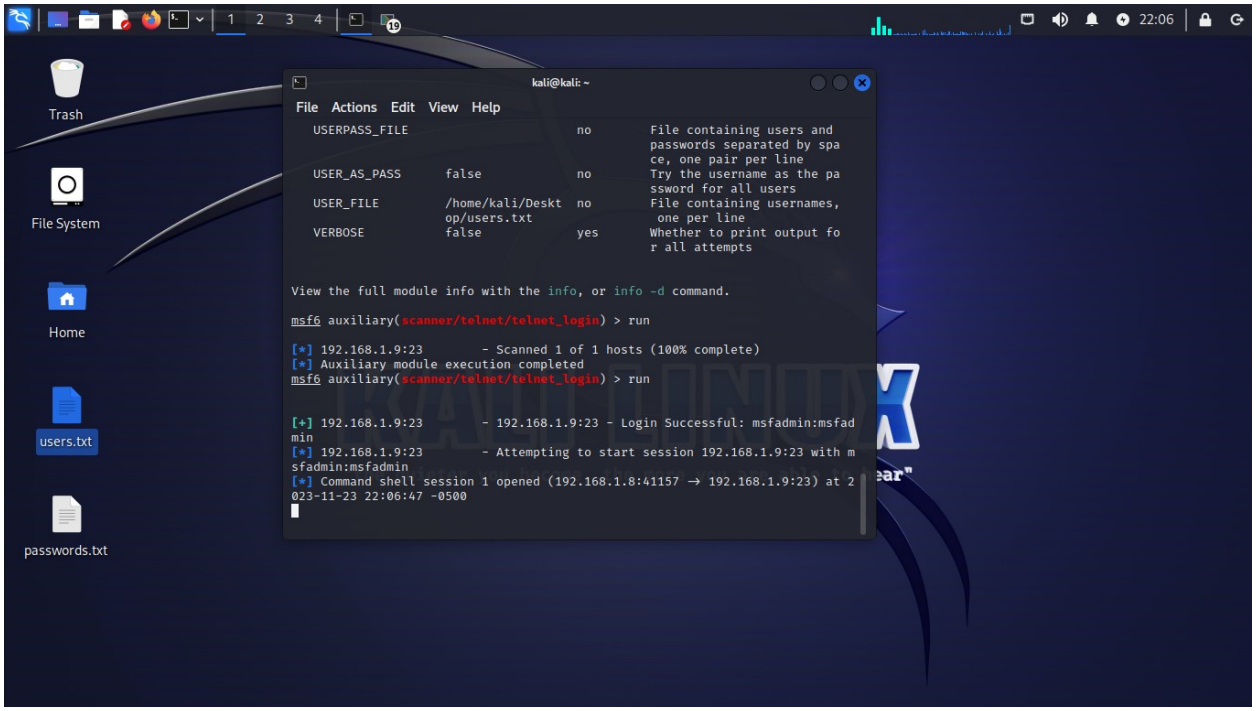

Fig 26. Screenshot showing the run command

While trying to run the “run” command, I observed that if you don’t put on your metsalpoitable, the target system, you will not be able to obtain your password and user text. The user text is a file on my Kali desktop. The path is home/kali/Desktop/users.text and home/kali/Desktop/passwords.txt

Conclusion

This case study discussed cybersecurity and vulnerability assessments using Greenbone and Metasploit Framework. The purpose of the Greenbone tool in the Metasploit case study is to identify the open ports on the Metasploitable machine that can be exploited using Metasploit. Greenbone helped to identify vulnerabilities in the open ports by providing information on the type of vulnerability that runs on each port. It also allowed us to find vulnerabilities in the results and the severity of the vulnerabilities. The vulnerability found on port 21 was FTP/TCP. This vulnerability was discovered through a scan using Greenbone, which showed the open ports that could be exploited using Metasploit. The impact of this vulnerability is that an attacker can gain access to login names and passwords by sniffing traffic to the Telnet Service. To mitigate this risk, the Telnet protocol should be replaced with a protocol like SSH, which supports encrypted connections. Additionally, a backdoor vulnerability was also found on port 21, which is the "vsftpd Compromised Source Packages Backdoor Vulnerability". This type of vulnerability allows attackers to gain unauthorized access to a system by exploiting vulnerabilities in the system or installing malicious software that opens a point of access for the attacker.. Overall, Greenbone is a useful tool for conducting vulnerability assessments and identifying potential security risks in a network.

References

Enigbokan, O., & Ajayi, N. (2017). Managing Cybercrimes Through the Implementation of Security Measures. *Journal of Information Warfare*, *16*(1), 112–129. https://www.jstor.org/stable/26502879

Gonzalez-Granadillo, G., Diaz, R., Caubet, J., & Garcia-Milà, I. (2021). CLAP: A Cross-Layer Analytic Platform for the Correlation of Cyber and Physical Security Events Affecting Water Critical Infrastructures. *Journal of Cybersecurity and Privacy, 1*(2), 365. https://doi.org/10.3390/jcp1020020

Leszczyna, R. (2021). Review of Cybersecurity Assessment Methods: Applicability Perspective. *Computers & Security*, *108*, 102376. https://doi.org/10.1016/j.cose.2021.102376

Rizvi, S., Pipetti, R., McIntyre, N., Todd, J., & Williams, I. (2020). Threat model for securing internet of things (IOT) network at device-level. *Internet of Things*, *11*, 100240. https://doi.org/10.1016/j.iot.2020.100240