CST 620 Prevention of Cyber Attack Methodologies
Project 1 – Remote Access Controls

Security Control Implementation Report

Prepared By: *Abbrin Hoagland*

Version 1.0

Contents

1. Introduction

In the evolving landscape of cybersecurity, ensuring robust defenses against potential threats is imperative. According to Cooper (2014) and Denning (2015), cybersecurity should be a top priority for all organizations. This paper delves into the critical aspects of bolstering an organization's security posture through the implementation of remote access controls. Focused primarily on Remote Desktop Protocol (RDP) and Secure Shell (SSH), this project seeks to explore best practices, enact secure configurations, conduct meticulous testing, and document the implementation steps to mitigate the risk of cyberattacks effectively.

As the first step in this comprehensive project, we embark on an exploration of best practices for securing RDP and SSH access. These best practices will serve as the foundational principles upon which our implementation and testing phases will be built.

In the sections that follow, we will delve into the specific best practices for RDP and SSH, outlining the precise steps required to implement these controls securely. Screenshots will be included as tangible evidence of our work, reflecting the hands-on exercises carried out to ensure the security of remote access.

The conclusion will encapsulate the outcomes of our implementation and testing efforts, summarizing the effectiveness of the remote access controls and their alignment with the desired security posture.

Throughout this journey, our aim is to equip organizations with the knowledge and tools necessary to fortify their defenses and proactively prevent cyberattacks. The references section will provide valuable resources to further delve into the intricacies of remote access control methodologies.

2. Remote Desktop Protocol Best Practices

Remote Desktop Protocol (RDP) is a widely used technology that allows users to access a remote computer or server over a network connection. While RDP provides convenience and flexibility, it also presents security challenges. To ensure the secure use of RDP, it is essential to implement a set of best practices that mitigate risks and protect against potential cyber threats. In this section, we will outline key RDP best practices that organizations should consider when configuring and using this remote access technology.

1. Strong Authentication:

Multi-Factor Authentication (MFA): Implement MFA for RDP access. This adds an extra layer of security by requiring users to provide two or more forms of authentication before gaining access. Common factors include something you know (password), something you have (smartphone), and something you are (biometrics).

2. Regularly Update RDP Software:
Patch Management: Keep RDP software up to date with the latest security patches and updates. Vulnerabilities in RDP can be exploited by cybercriminals, making it crucial to address any known security issues promptly.

3. Network Level Authentication (NLA):

Enable Network Level Authentication (NLA) for RDP sessions. NLA requires users to authenticate themselves to the network before establishing a remote desktop connection. This adds an additional layer of security by preventing unauthorized access attempts.

4. Strong Password Policies:

Implement strong password policies for RDP accounts. Enforce complex passwords that include a combination of uppercase and lowercase letters, numbers, and special characters. Regularly remind users to update their passwords and avoid using easily guessable credentials.

5. Account Lockout Policies:

Configure account lockout policies to limit the number of login attempts. This helps prevent brute force attacks by locking out an account after a specified number of failed login attempts.

6. Least Privilege Principle:

Follow the principle of least privilege when granting RDP access. Only provide RDP permissions to users who require it for their specific tasks. Avoid granting unnecessary administrative privileges.

7. Network Segmentation:

Implement network segmentation to isolate RDP servers from other critical systems and networks. This reduces the attack surface and limits lateral movement in case of a breach.

8. Audit and Monitoring:

Enable logging and monitoring for RDP sessions. Regularly review logs for suspicious activities or login attempts. Implement intrusion detection systems (IDS) and intrusion prevention systems (IPS) to detect and prevent unauthorized access.

9. RDP Gateway:

Use an RDP gateway (Remote Desktop Gateway) for secure RDP connections. An RDP gateway acts as a secure proxy, allowing remote access to internal resources without exposing them directly to the internet.

10. Secure RDP Ports:

Change the default RDP port (TCP 3389) to a non-standard port. This can help deter automated scanning and brute force attacks. However, ensure that the chosen port is not commonly used for other services.

11. Regular Security Training:

Provide regular security training and awareness programs for employees who use RDP. Educate them about the risks associated with remote access and teach them safe practices.

By adhering to these RDP best practices, organizations can significantly enhance the security of their remote access infrastructure. Secure RDP configurations not only protect sensitive data but also help prevent unauthorized access and potential security breaches. It is essential to continuously monitor and update these practices to adapt to evolving cyber threats and maintain a robust security posture.

3. Secure Shell Best Practices

Secure Shell (SSH) is a widely used protocol for securely accessing and managing remote servers and network devices. As a critical component of secure system administration and remote access, SSH demands careful configuration and adherence to best practices to minimize vulnerabilities and safeguard sensitive data. In this section, we will delve into essential SSH best practices that organizations should implement to strengthen their security posture.

1. Disable SSH Protocol Version 1 (SSHv1):

SSHv1 has known security vulnerabilities and should be disabled. Use SSHv2, which provides improved security features and is widely considered the more secure option.

2. Strong Authentication:

Public Key Authentication: Utilize public key authentication whenever possible. Public key authentication is more secure than password-based authentication, as it requires possession of the private key, making it extremely challenging for attackers to compromise.

3. Limit SSH Access:

Use SSH Keys for Access Control: Control SSH access by distributing and managing SSH keys. Assign SSH keys to specific users and revoke them when necessary, reducing the risk of unauthorized access.

4. Implement Multi-Factor Authentication (MFA):

Implementing MFA adds an extra layer of security to SSH access. Users are required to provide multiple authentication factors, such as something they know (password) and something they have (token or smartphone app).

5. Strong Password Policies:

If password authentication is necessary, enforce strong password policies for SSH accounts. Require complex passwords with a combination of uppercase and lowercase letters, numbers, and special characters. Regularly prompt users to update their passwords.

6. Disable Root Login:

Disable direct root login via SSH. Instead, allow users to log in as regular users and use the "su" or "sudo" command to gain root privileges when necessary. This reduces the risk of brute force attacks on the root account.

7. Limit SSH Port Access:

Change the default SSH port (TCP 22) to a non-standard port. While this does not provide robust security on its own, it can deter automated scans and reduce exposure to common threats.

8. Regularly Update SSH Software:

Keep SSH software up to date with the latest security patches and updates. Vulnerabilities in SSH implementations are periodically discovered and addressed through updates.

9. SSH Idle Timeout:

Implement an idle timeout for SSH sessions. Automatically log out users who have been inactive for a specified period. This prevents unauthorized access in case a user leaves an active session unattended.

10. Monitor and Log SSH Activity:

Enable SSH logging to capture authentication attempts and session activity. Regularly review logs for unusual or suspicious behavior. Implement intrusion detection systems (IDS) to detect and prevent SSH-related threats.

11. Restrict SSH Access via IP Whitelisting:

Implement IP whitelisting to restrict SSH access to specific trusted IP addresses or ranges. This limits access to authorized systems and reduces the risk of unauthorized connections.

12. Regular Security Training:

Conduct security awareness training for SSH users. Educate them about the risks associated with SSH access and provide guidance on secure practices.

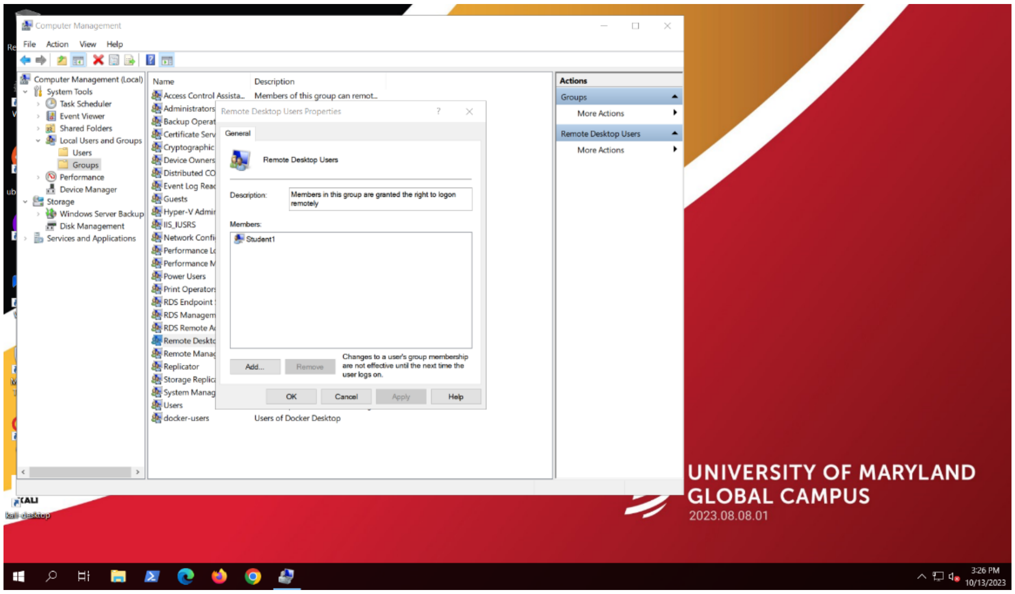
By adhering to these SSH best practices, organizations can significantly enhance the security of their remote access and server management. SSH plays a pivotal role in securing remote administration, and proper configuration and monitoring are essential to mitigating security risks. Regularly reviewing and updating these practices will help organizations adapt to evolving threats and maintain a strong security posture.

4. Remote Desktop Protocol Control Implementation and Testing

FICBANK, an organization deeply committed to the robust security of its remote access infrastructure, undertook a meticulously planned and executed process of implementing and validating RDP control measures. These measures were thoughtfully designed to align closely with recognized industry best practices, thereby ensuring the establishment of a strong and secure RDP access system.

One of the pivotal facets of this initiative was the implementation of the 'Least Privilege' doctrine, which fundamentally underscored the careful assignment of RDP access permissions. In this approach, the cornerstone principle was to grant RDP permissions exclusively to individuals with a direct operational need. This strategic move effectively curtailed unnecessary administrative privileges, significantly reducing the potential attack surface.

To manifest the implementation of these access control measures, the creation of the "Student1" user account was undertaken with utmost care. This account was fortified by the assignment of a robust and intricately designed password. To maintain the integrity of access control, authorization was rigorously administered, ensuring that RDP access was exclusively extended to authorized users within the designated "Remote Desktop Users" group.



Remote Desktop Users Properties

Remote Desktop Users

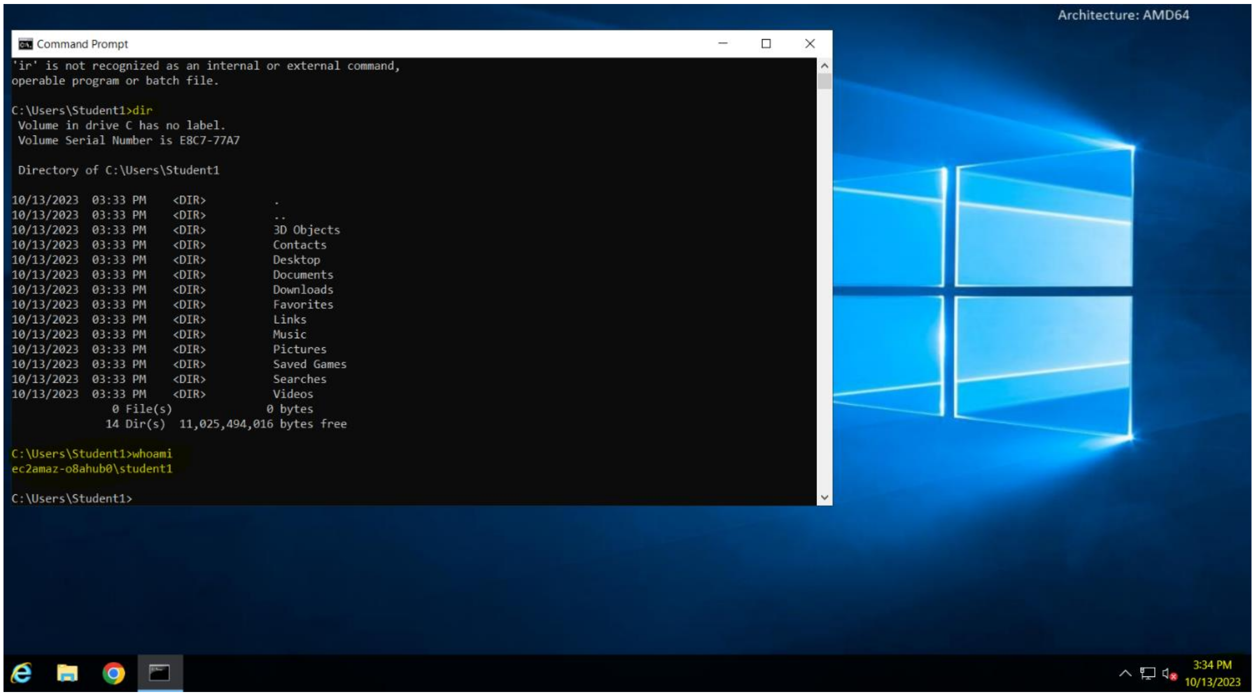
Description: Members in this group are granted the right to logon remotely

Members:
Student1

Changes to a user's group membership are not effective until the next time the user logs on.

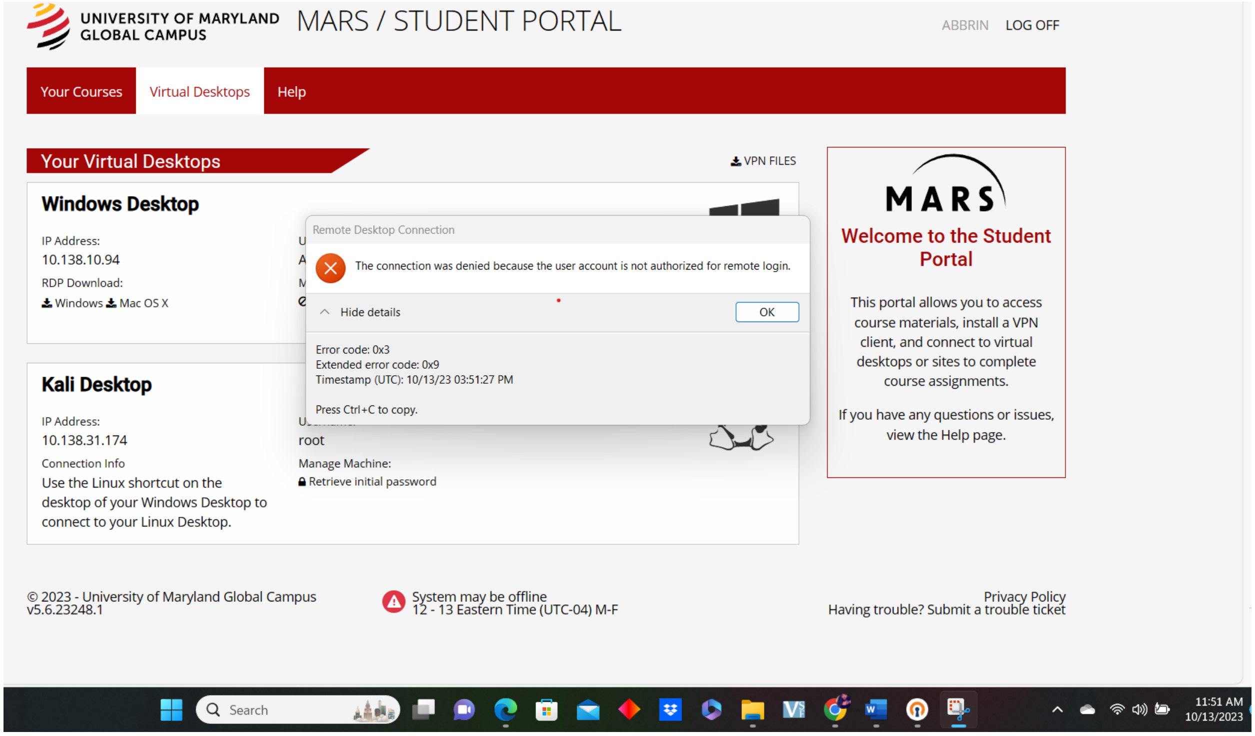
Name	Description
Access Control Assista...	Members of this group can remot...
docker-users	Users of Docker Desktop

Validation was a critical component of this process. A comprehensive evaluation of the access control measures was executed to ensure their efficacy. This validation process was exhaustive, involving the re-establishment of RDP access and a meticulous examination of various facets. Among these assessments were the scrutiny of desktop background changes, the careful inspection of command prompt accessibility, and the verification of user context through the 'whoami' command.



```
'ir' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Student1>dir
 Volume in drive C has no label.
 Volume Serial Number is E8C7-77A7

 Directory of C:\Users\Student1

10/13/2023  03:33 PM    <DIR>          .
10/13/2023  03:33 PM    <DIR>          ..
10/13/2023  03:33 PM    <DIR>          3D Objects
10/13/2023  03:33 PM    <DIR>          Contacts
10/13/2023  03:33 PM    <DIR>          Desktop
10/13/2023  03:33 PM    <DIR>          Documents
10/13/2023  03:33 PM    <DIR>          Downloads
10/13/2023  03:33 PM    <DIR>          Favorites
10/13/2023  03:33 PM    <DIR>          Links
10/13/2023  03:33 PM    <DIR>          Music
10/13/2023  03:33 PM    <DIR>          Pictures
10/13/2023  03:33 PM    <DIR>          Saved Games
10/13/2023  03:33 PM    <DIR>          Searches
10/13/2023  03:33 PM    <DIR>          Videos
               0 File(s)              0 bytes
              14 Dir(s)  11,025,494,016 bytes free

C:\Users\Student1>whoami
ec2amaz-o8ahub0\student1

C:\Users\Student1>
```

Furthermore, the significance of an access revocation process should not be underestimated. As a critical facet of the implementation, an elaborate procedure to revoke RDP access for the "Student1" user was undertaken. This process involved the meticulous removal of the "Student1" user account from the "Remote Desktop Users" group, confirming the successful revocation of access.



Windows Desktop

IP Address: 10.138.10.94

RDP Download: Windows Mac OS X

Kali Desktop

IP Address: 10.138.31.174

Connection Info: Use the Linux shortcut on the desktop of your Windows Desktop to connect to your Linux Desktop.

Username: root

Manage Machine: Retrieve initial password

Remote Desktop Connection

The connection was denied because the user account is not authorized for remote login.

Hide details OK

```
Error code: 0x3
Extended error code: 0x9
Timestamp (UTC): 10/13/23 03:51:27 PM

Press Ctrl+C to copy.
```

Welcome to the Student Portal

This portal allows you to access course materials, install a VPN client, and connect to virtual desktops or sites to complete course assignments.

If you have any questions or issues, view the Help page.

© 2023 - University of Maryland Global Campus
v5.6.23248.1

System may be offline
12 - 13 Eastern Time (UTC-04) M-F

Privacy Policy
Having trouble? Submit a trouble ticket

Adding another layer of security to the infrastructure, network segmentation measures were introduced. This strategic move effectively isolated RDP servers from other critical systems and networks, thereby significantly reducing the potential attack surface. The implications of this approach are far-reaching, minimizing the potential for lateral movement in the event of a security breach.

Logging and monitoring were another instrumental aspect of this initiative. By enabling comprehensive logging and monitoring for RDP sessions, the organization proactively positioned itself to identify suspicious activities or unauthorized login attempts. Furthermore, the addition of intrusion detection systems (IDS) and intrusion prevention systems (IPS) further fortified the system, effectively detecting and mitigating unauthorized access attempts.

In summation, the culmination of these meticulously implemented RDP best practices not only fortified the security of the remote access infrastructure but also exemplified a proactive approach to safeguard sensitive data. The strategic plan aimed at fortifying the infrastructure against unauthorized access and potential security breaches. Through a commitment to ongoing monitoring and adaptability, FICBANK's security posture remains agile and robust, aligning with the organization's unwavering dedication to protecting its digital assets.

5. Secure Shell Control Implementation and Testing

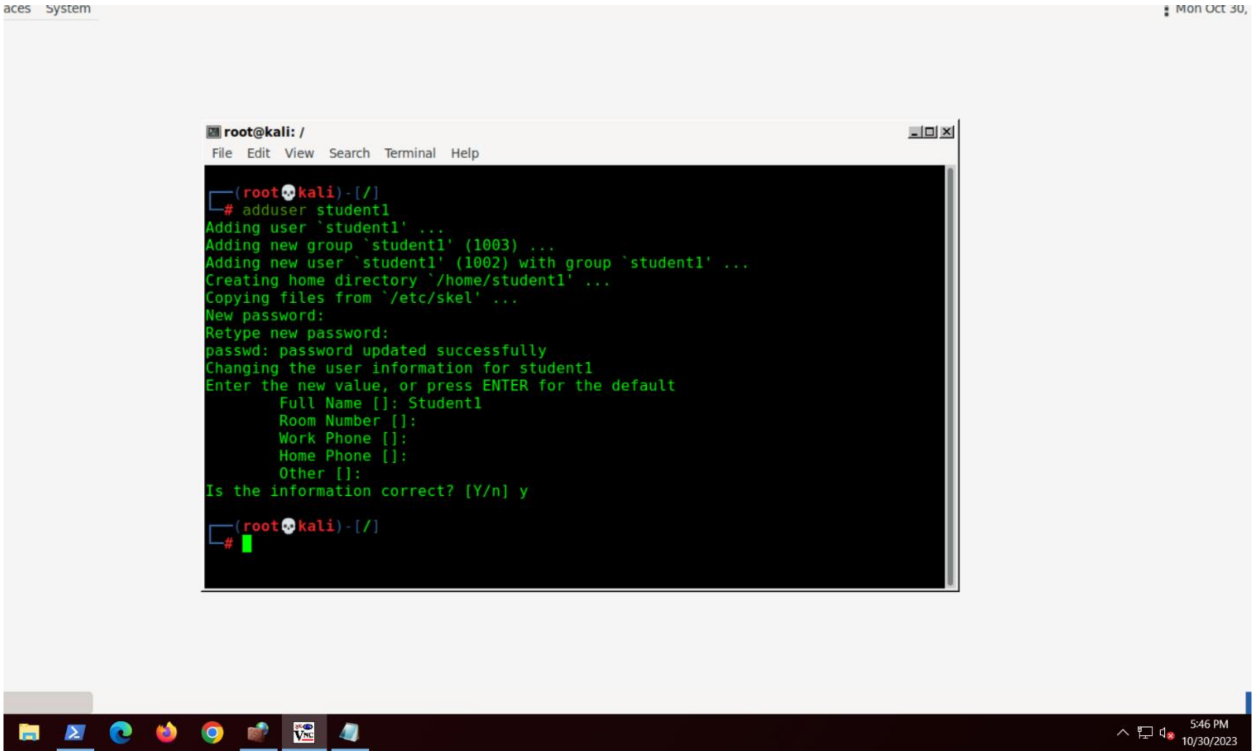
In our ongoing commitment to fortify our remote server management, the FICBANK team undertook a comprehensive journey to implement and validate the best practices surrounding Secure Shell (SSH) control. SSH, an integral technology for secure remote server management, was subjected to rigorous implementation and validation. Our approach was deliberate and methodical, underpinned by a series of critical steps and configurations, and supported by a suite of versatile tools and software.

The initiation of our SSH control implementation was marked by our attempt to establish SSH connections from our Windows VM to the Kali Linux VM. It is worth noting that this initial endeavor met with access denial for both the Administrator and "root" users, effectively underscoring the urgency of implementing robust SSH access controls.

To facilitate root user login via SSH, we embarked on the pivotal task of editing the 'sshd_config' file. This file, a linchpin of SSH configuration, is typically situated in the '/etc/ssh/' directory within most Linux distributions. The process involved the nuanced modification of specific settings, specifically "PermitRootLogin" and "PasswordAuthentication." These parameters were adjusted to "yes" to permit root user access. Upon saving the changes, the SSH service was dutifully restarted to activate the newly configured settings.

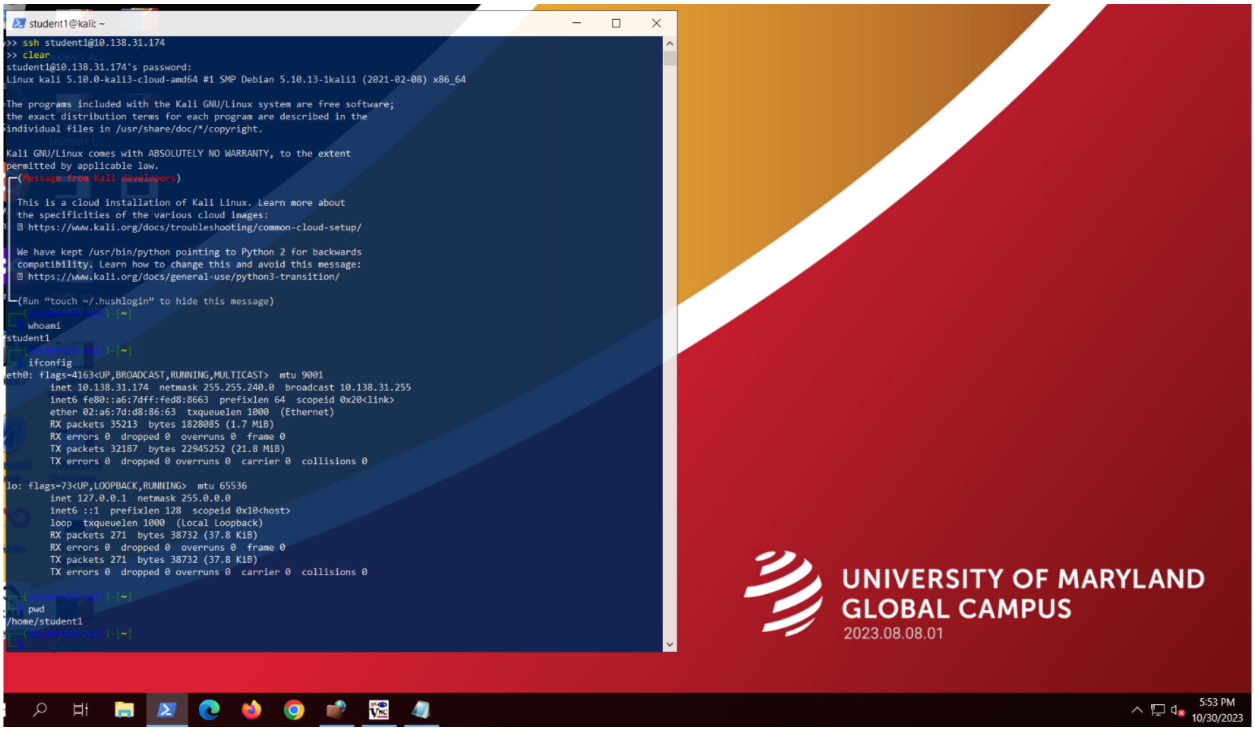
The resultant configuration allowed us to execute SSH connections into the Kali Linux VM as the root user. An important realization in this process was the case-sensitivity of user accounts in Linux. Our SSH sessions clearly distinguished between "root" and "Root."

Having successfully established root user access, our next endeavor revolved around the creation of a fresh user account, thoughtfully named "Student1," on the Kali Linux VM. This operation was facilitated through the utilization of the 'adduser' command, ensuring the provision of requisite information, inclusive of a robust password.



```
┌──(root💀kali)-[/]
└─# adduser student1
Adding user `student1' ...
Adding new group `student1' (1003) ...
Adding new user `student1' (1002) with group `student1' ...
Creating home directory `/home/student1' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for student1
Enter the new value, or press ENTER for the default
        Full Name []: Student1
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y

┌──(root💀kali)-[/]
└─# 
```

Subsequently, SSH access was tested using the newly created "Student1" account, producing a distinctive login response that unequivocally affirmed the success of the access attempt under the "Student1" user context.



student1@kali: ~

```
>> ssh student1@10.138.31.174
>> clear
student1@10.138.31.174's password:
Linux kali 5.10.0-kali3-cloud-amd64 #1 SMP Debian 5.10.13-1kali1 (2021-02-08) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
┌─(Message from Kali developers)
│
│ This is a cloud installation of Kali Linux. Learn more about
│ the specificities of the various cloud images:
│ ⇒ https://www.kali.org/docs/troubleshooting/common-cloud-setup/
│
│ We have kept /usr/bin/python pointing to Python 2 for backwards
│ compatibility. Learn how to change this and avoid this message:
│ ⇒ https://www.kali.org/docs/general-use/python3-transition/
│
└─(Run “touch ~/.hushlogin” to hide this message)
┌──(student1㉿kali)-[~]
└─$ whoami
student1
┌──(student1㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.138.31.174  netmask 255.255.240.0  broadcast 10.138.31.255
        inet6 fe80::a6:7dff:fed8:8663  prefixlen 64  scopeid 0x20<link>
        ether 02:a6:7d:d8:86:63  txqueuelen 1000  (Ethernet)
        RX packets 35213  bytes 1828085 (1.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 32187  bytes 22945252 (21.8 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 271  bytes 38732 (37.8 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 271  bytes 38732 (37.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

┌──(student1㉿kali)-[~]
└─$ pwd
/home/student1
┌──(student1㉿kali)-[~]
└─$
```

To augment the overall security apparatus, we turned to IPTables, the renowned Linux firewall. Here, our primary objective was to configure IPTables in a manner that would exert precise control over SSH port access, which conventionally occupies port 22. Through the careful crafting of rules within the IPTables framework, we gained the capacity to either grant or withhold access to incoming SSH connections based on various criteria, including port assignments and originating IP addresses. This phase of the project effectively demonstrated how the strategic deployment of IPTables could serve as an additional layer of protection, reinforcing the security of SSH access.

The conclusive phase of our validation process centered on renewed SSH connection attempts, this time facilitated from the Windows VM to the Kali Linux VM. It was during this phase that the impact of correctly configured IPTables was prominently featured. A well-implemented IPTables configuration effectively precluded the establishment of SSH connections, affirming the robustness of our security measures. Subsequent removal of the IPTables rule promptly restored SSH access, concluding our validation exercise.

In summary, the comprehensive endeavor to implement and validate SSH control best practices at FICBANK underscores our unwavering commitment to secure remote server management. Throughout this journey, a diverse toolkit of tools, configuration files, and firewall settings were employed to meticulously reduce the risk of unauthorized access, reinforcing the security framework of the systems under consideration. These initiatives are an integral part of FICBANK's proactive approach to maintaining a resilient and secure digital environment. (NIST Publications, 2017)

6. Conclusion

In the dynamic landscape of cybersecurity, the effective implementation of remote access controls stands as a critical line of defense against potential threats. This report has taken a comprehensive approach to secure Remote Desktop Protocol (RDP) and Secure Shell (SSH) access by emphasizing best practices, hands-on implementation, and rigorous testing.

Our journey began with a foundation of best practices, ranging from multi-factor authentication, updates, and network segmentation to strong passwords, least privilege, and advanced firewall configurations. These best practices served as the bedrock upon which our secure configurations were built.

FICBANK, an organization deeply committed to the robust security of its remote access infrastructure, undertook a meticulously planned and executed process of implementing and validating RDP and SSH control measures, with our guidance. These measures were thoughtfully designed to align closely with recognized industry best practices, thereby ensuring the establishment of a strong and secure RDP and SSH access system.

Focusing on RDP, we meticulously detailed the creation of the "Student1" user account, optimized for RDP access, and validated the efficacy of our access controls. Furthermore, we bolstered security with the strategic deployment of Windows Defender Firewall.

In the case of SSH, we shared insights into enabling root login, creating the "Student1" account, and successfully establishing SSH access, all of which significantly enhanced FICBANK's remote server management.

This report, rooted in practical application and FICBANK's experience, offers an invaluable reference for organizations seeking to fortify their remote access security. It provides a comprehensive roadmap to navigate the ever-evolving landscape of cyber threats, ensuring that the principles and practices outlined here remain adaptable to emerging vulnerabilities and challenges.

In an era marked by dynamic and evolving cyber threats, the secure implementation of remote access controls is not just a necessity but a strategic imperative. These measures protect sensitive data, uphold system integrity, and deter unauthorized access. (NIST Publications, 2016 & 2023) We extend an open invitation to organizations and cybersecurity professionals to utilize this report, inspired by our collaboration with FICBANK, as a guiding resource to reinforce their defenses against cyberattacks, thereby contributing to a more secure digital environment.

7. References

- Cooper, W. H. (2014). *Cybersecurity strategy: A practical guide for leaders*. Syngress.
- Denning, D. E. (2015). The future of cybersecurity: Challenges and opportunities. *Communications of the ACM*, 58(3), 28-31.
- National Institute of Standards and Technology. (2023, October 30). Cybersecurity Framework (CSF). Retrieved from https://www.nist.gov/cybersecurity-framework
- NIST Publications (2016). *NIST Special Publication 800-46 Revision 2: Guide to Enterprise Telework, Remote Access, and Bring Your Own Device (BYOD) Security*. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-46r2.pdf
- NIST Publications (2017). *NIST Special Publication 800-53 Revision 5: Recommended Security Controls for Federal Information Systems and Organizations*. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf
- SANS Institute. (2023). *The 2023 SANS Cybersecurity Survey Report*.