TASK 2

Cyber Security Internship Report

Presented to:

Digital Empowerment Network

Presented by:

Ali Aitazaz [LinkedIn]

☎ +92 332 5890142

🌐 www.digitalempowermentnetwork.org

Contents

Implementing Multi-Factor Authentication

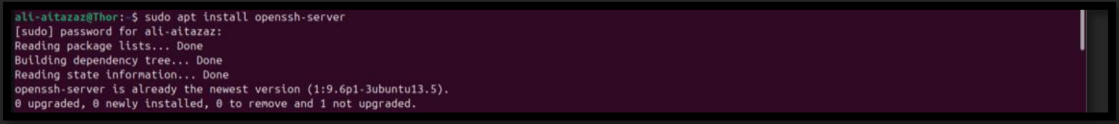
1. Selecting an MFA solution compatible with the system.

In this task we will set-up MFA for our Ubuntu System with Google Authenticator. You can use any type of authenticator (Microsoft Authenticator, Google Authenticator, DUO Mobile, Yubico etc)

After selecting the Authenticator power up your System in this case (Ubuntu), and install openssh-server which we are going to be using to test out our configuration of MFA.

Openssh-Server

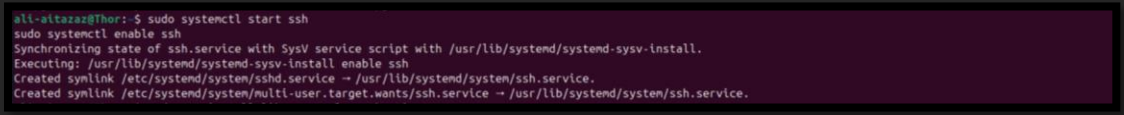
sudo apt install openssh-server



```
ali-aitazaz@Thor:~$ sudo apt install openssh-server
[sudo] password for ali-aitazaz:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssh-server is already the newest version (1:9.6p1-3ubuntu13.5).
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
```

Once openssh-server is installed you can run the ssh service and enable ssh on startup by using commands

sudo systemctl start ssh
sudo systemctl enable ssh

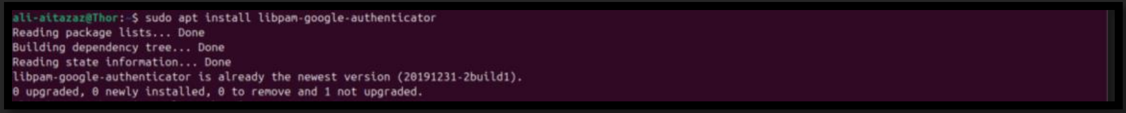


```
ali-aitazaz@Thor:~$ sudo systemctl start ssh
sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh
Created symlink /etc/systemd/system/sshd.service → /usr/lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /usr/lib/systemd/system/ssh.service.
```

Google Authenticator

Then you can install Google Authenticator into Ubuntu

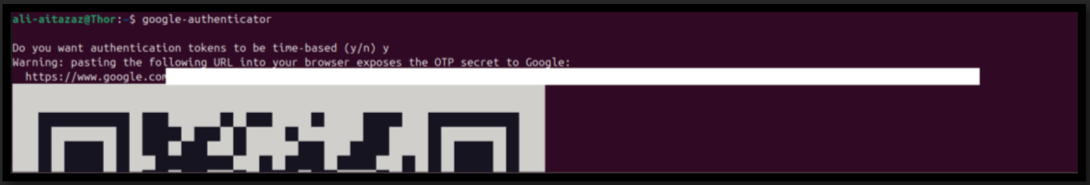
sudo apt install libpan-google-authenticator



```
ali-aitazaz@Thor:~$ sudo apt install libpam-google-authenticator
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libpam-google-authenticator is already the newest version (20191231-2build1).
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
```

Now, you can run Google Authenticator by using command

google-authenticator



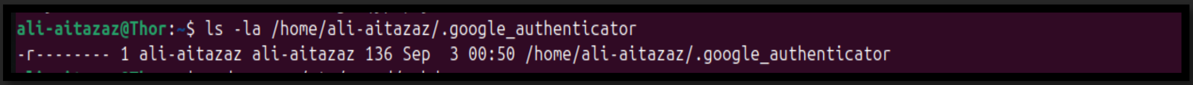
```
ali-aitazaz@Thor:~$ google-authenticator

Do you want authentication tokens to be time-based (y/n) y
Warning: pasting the following URL into your browser exposes the OTP secret to Google:
  https://www.google.com[redacted]
```

A QR code will pop up on the screen, scan it using your phone’s google authenticator and a new account will be created on your app for your Ubuntu user. Open up the new account with Ubuntu user_name created on your authenticator app and you will see codes being generated every 30 seconds, these are your MFA codes. Put the code back into Ubuntu immediately after scanning the QR code and now your MFA is set up, your Ubuntu machine will ask you about some options you might want to set up or not, as default press ‘Y’ for all.

Check to see if the directory google_authenticator is created or not using command

ls –la /home/user_name/.google_authenticator



```
ali-aitazaz@Thor:~$ ls -la /home/ali-aitazaz/.google_authenticator
-r-------- 1 ali-aitazaz ali-aitazaz 136 Sep  3 00:50 /home/ali-aitazaz/.google_authenticator
```

Configure SSH Daemon

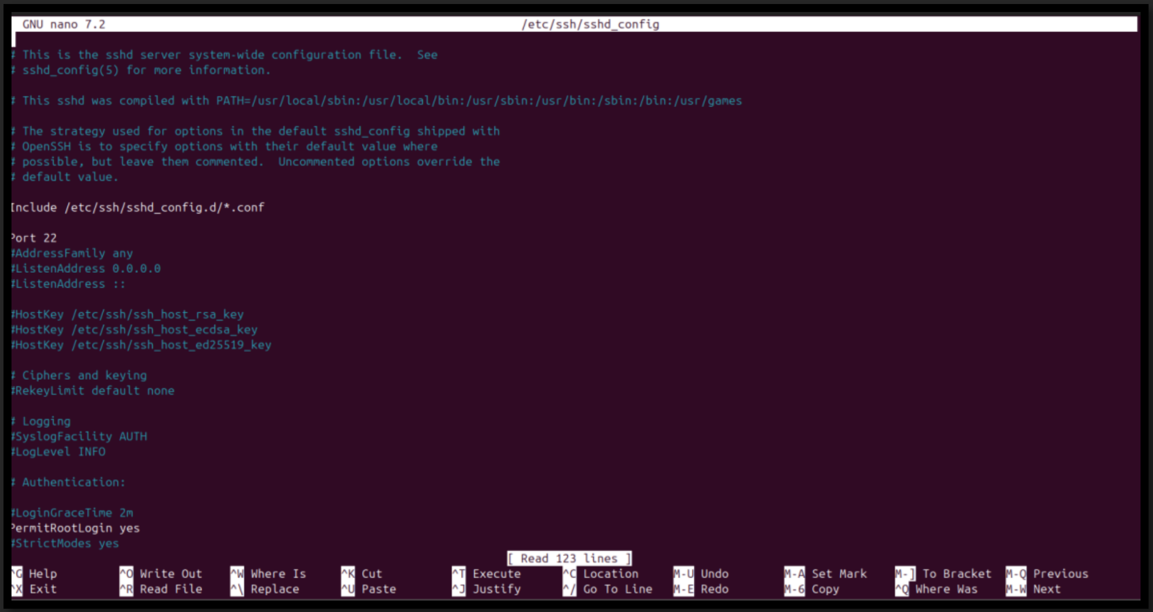
Edit the SSH configuration file using command

sudo nano /etc/ssh/sshd_config

Modify these lines:

> *PermitRootLogin no (or yes if you want to allow root login, but generally it's better to keep it 'no') PasswordAuthentication yes*
> *ChallengeResponseAuthentication yes*
> *UsePAM yes*

Save and exit the file (Ctrl + X, then Y, then Enter).



```
  GNU nano 7.2                                   /etc/ssh/sshd_config
# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
                                       [ Read 123 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark  M-] To Bracket  M-Q Previous
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line M-E Redo      M-6 Copy      ^Q Where Was    M-W Next
```

Configure PAM for SSH

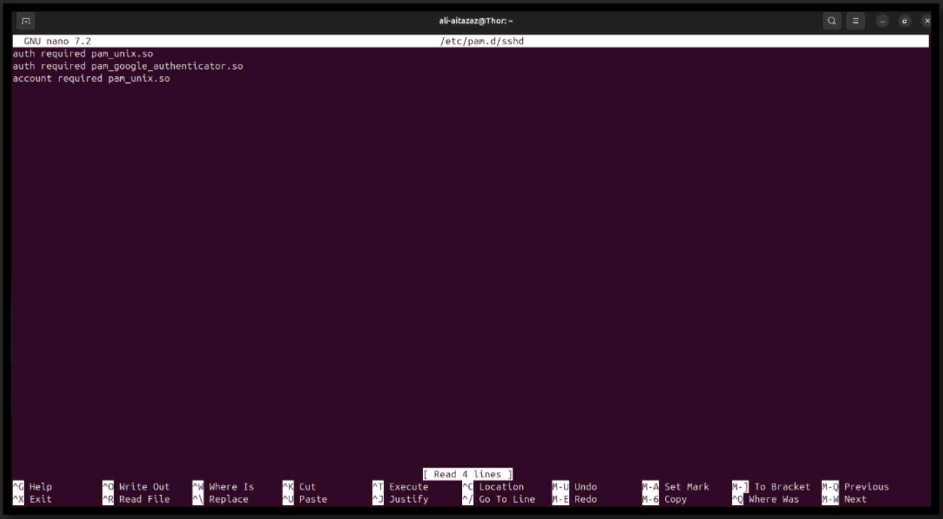
Edit the PAM SSH configuration file using command

sudo nano /etc/pam.d/sshd

Add the following line at the end of the file:

auth required pam_google_authenticator.so

Save **and exit the file** (Ctrl + X, then Y, then Enter).

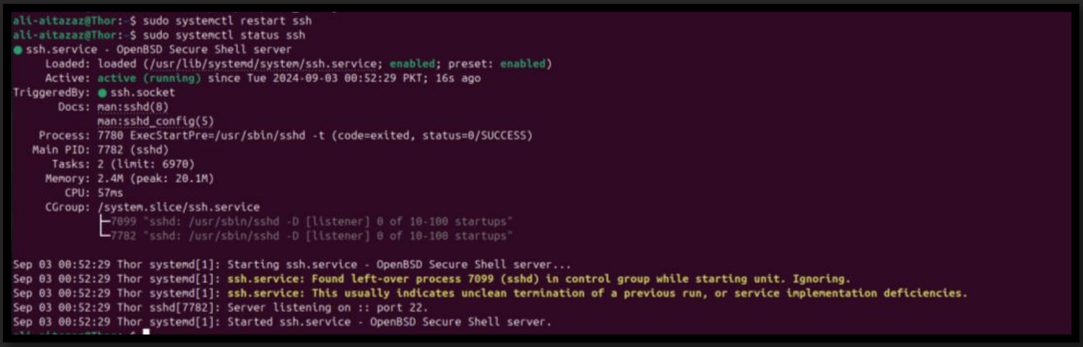


```
  GNU nano 7.2                                   /etc/pam.d/sshd
auth required pam_unix.so
auth required pam_google_authenticator.so
account required pam_unix.so

                                       [ Read 4 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark  M-] To Bracket  M-Q Previous
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line M-E Redo      M-6 Copy      ^Q Where Was    M-W Next
```

Restart the ssh service and check its status using command

sudo systemctl restart ssh
sudo systemctl status ssh



```
ali-aitazaz@Thor:~$ sudo systemctl restart ssh
ali-aitazaz@Thor:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-09-03 00:52:29 PKT; 16s ago
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 7780 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 7782 (sshd)
      Tasks: 2 (limit: 6970)
     Memory: 2.4M (peak: 20.1M)
        CPU: 57ms
     CGroup: /system.slice/ssh.service
             ├─7099 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
             └─7782 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Sep 03 00:52:29 Thor systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Sep 03 00:52:29 Thor systemd[1]: ssh.service: Found left-over process 7099 (sshd) in control group while starting unit. Ignoring.
Sep 03 00:52:29 Thor systemd[1]: ssh.service: This usually indicates unclean termination of a previous run, or service implementation deficiencies.
Sep 03 00:52:29 Thor sshd[7782]: Server listening on :: port 22.
Sep 03 00:52:29 Thor systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
```

Test SSH Connection

Open a terminal on another device (or use PowerShell if you are testing from Windows).

Before you execute the command make sure to add windows feature required to run ssh service,

Open Settings
Type optional Feature
Click on View Features
Type OpenSSH Server and add feature

Once the feature has installed on your system, this will allow to establish connection with SSH service.

Run Command Prompt as Administrator or PowerShell as Administrator and type:

Start the SSH Service:

In the PowerShell window, enter the following command to start the SSH service:

Start-Service sshd

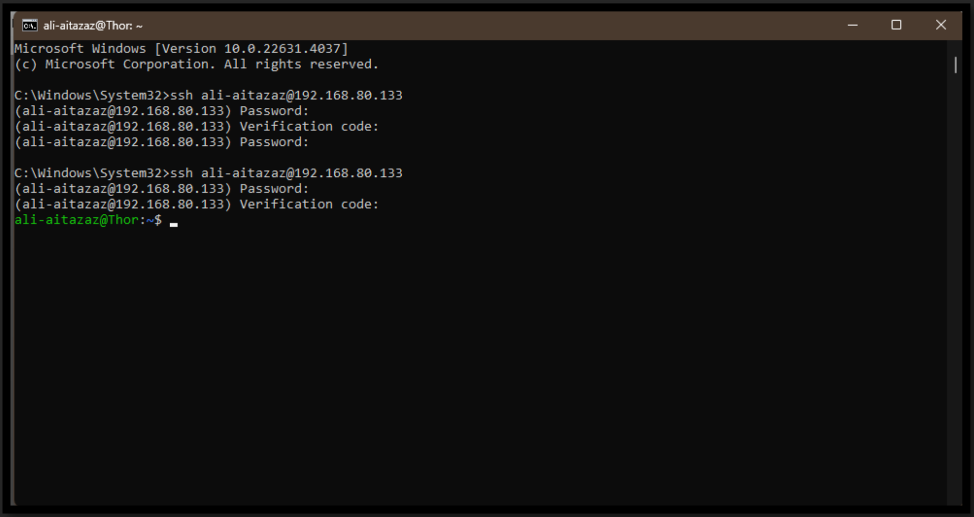
Run the following command to connect:

ssh your_ubuntu_username@your_ubuntu_ip

Hit 'Enter' and you will be prompted to enter your Ubuntu password for SSH connection, once you enter the password it will prompt you for the MFA you have set-up earlier, enter the code generated from the Google Authenticator app.

You are now connected to the Ubuntu through your Windows Command Prompt, everything you enter will be executed in your Ubuntu Machine.

You can terminate the connection anytime by simply entering 'exit'.



```
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>ssh ali-aitazaz@192.168.80.133
(ali-aitazaz@192.168.80.133) Password:
(ali-aitazaz@192.168.80.133) Verification code:
(ali-aitazaz@192.168.80.133) Password:

C:\Windows\System32>ssh ali-aitazaz@192.168.80.133
(ali-aitazaz@192.168.80.133) Password:
(ali-aitazaz@192.168.80.133) Verification code:
ali-aitazaz@Thor:~$
```

About the Company

At Digital Empowerment, we envision a future where youth are equipped with the skills, knowledge, and mindset needed to thrive in a rapidly evolving world. Our mission is to bridge the digital divide, foster leadership development, and enhance academic growth, empowering young minds to realize their full potential.

To achieve this, we offer comprehensive virtual internships across various domains, providing students with invaluable hands-on experience and practical skills essential for success. Furthermore, we are committed to assisting exceptional students in securing positions at prestigious companies, helping to launch their careers and build a brighter future for themselves and future generations.

Digital Empowerment Pakistan

Author’s Note

This work reflects the journey of navigating the complexities of multi-factor authentication (MFA) implementation and SSH configuration in a real-world scenario. Over the course of several hours, we encountered and overcame numerous challenges, ranging from basic SSH setup issues to the intricacies of Google Authenticator integration. The perseverance displayed in troubleshooting, testing, and finally achieving success is a testament to the importance of patience and determination in the field of cybersecurity.

This experience not only reinforced key technical skills but also highlighted the critical role of attention to detail and persistence. For anyone undertaking similar tasks, remember that even when the path is fraught with setbacks, each challenge is an opportunity to learn and grow. Keep pushing forward, and you'll eventually reach your goal.