Project Report

Student Name: Aashu **UID:** 24MCA20202

Branch: MCA(General) **Section/Group:** MCA 3(B)

Semester: 1st **Subject Code:** 24CAP-607

Subject Name: Linux Administration

Title of the Project: User Authentication System

Introduction:

In Linux, user authentication plays a vital role in system security. It ensures that only authorized users have access to the system, protects sensitive data, and maintains the integrity of user sessions. This project report provides an overview of the essential components and mechanisms behind user authentication in Linux, including user management, password policies, access control, and additional security features.

- **Background**: Explain the need for secure user authentication in Linux systems, especially in multi-user environments where access control and security are critical. In a modern computing environment, Linux is widely used for servers, development environments, and multi-user systems due to its stability, flexibility, and strong security features. However, as Linux systems grow to accommodate multiple users and critical applications, ensuring a secure and manageable user authentication system becomes essential.
- **Objectives:**
  1. To set up a system that authenticates users securely.
  2. To control user access to files and system services.
  3. To enhance security with Two-Factor Authentication (2FA).
- **Scope**: This project is limited to implementing authentication techniques on a Linux server, focusing on user management and access control features.

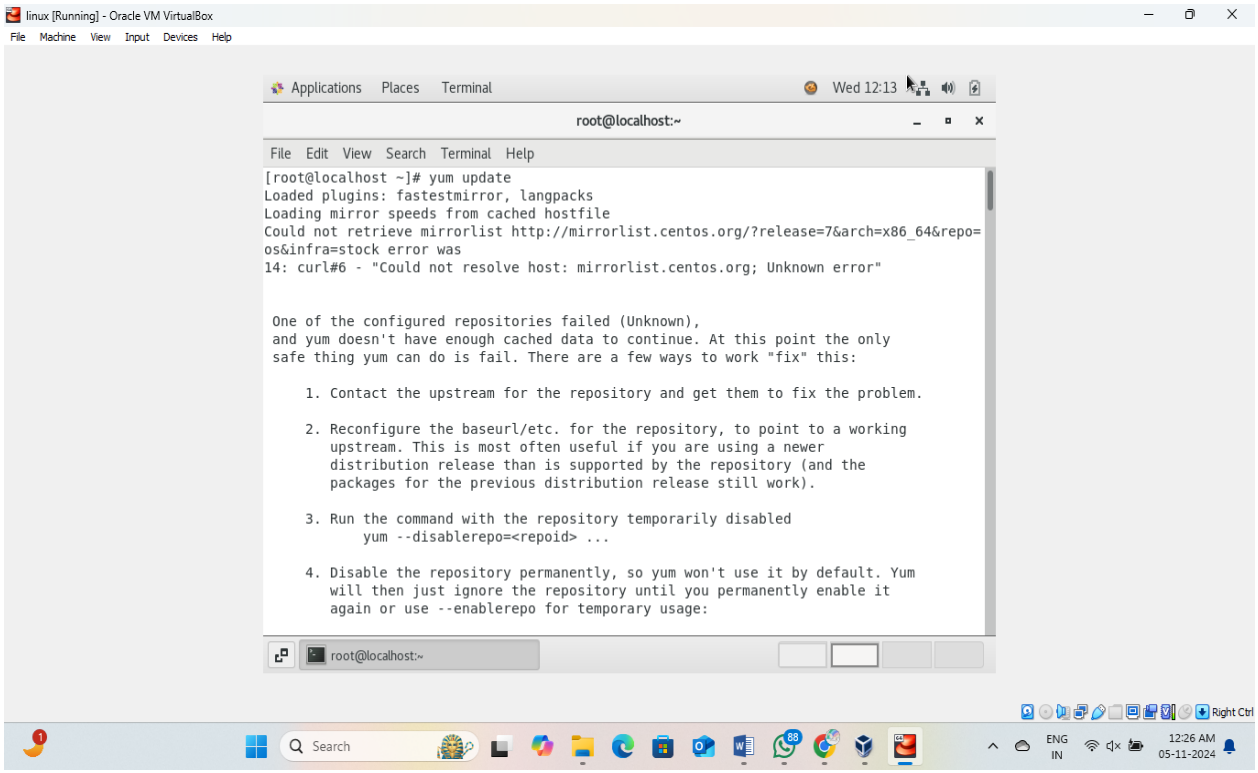
System Requirements

- **Software Requirements:**
  - Linux OS (e.g., Ubuntu, CentOS)
  - OpenSSL for encryption (optional for secure storage)
  - PAM (Pluggable Authentication Modules) for handling authentication
  - Google Authenticator for 2FA
- **Hardware Requirements**: A server or virtual machine capable of running a Linux distribution

System Architecture

- **User Management**: Using commands like useradd, usermod, passwd, and groupadd to create and manage user accounts and groups.
- **Access Control Lists (ACLs)**: To assign specific permissions to files and directories using commands like chmod, chown, and setfacl.
- **Role-Based Access Control**: Creating groups for various roles (e.g., Admin, User) and assigning permissions accordingly.
- **2FA Implementation**: Using PAM to integrate Two-Factor Authentication with the google-authenticator module.

Implementation



linux [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# useradd
Usage: useradd [options] LOGIN
       useradd -D
       useradd -D [options]

Options:
  -b, --base-dir BASE_DIR       base directory for the home directory of the
                                new account
  -c, --comment COMMENT         GECOS field of the new account
  -d, --home-dir HOME_DIR       home directory of the new account
  -D, --defaults                print or change default useradd configuration
  -e, --expiredate EXPIRE_DATE  expiration date of the new account
  -f, --inactive INACTIVE       password inactivity period of the new account
  -g, --gid GROUP               name or ID of the primary group of the new
                                account
  -G, --groups GROUPS           list of supplementary groups of the new
                                account
  -h, --help                    display this help message and exit
  -k, --skel SKEL_DIR           use this alternative skeleton directory
  -K, --key KEY=VALUE           override /etc/login.defs defaults
  -l, --no-log-init             do not add the user to the lastlog and
                                faillog databases
  -m, --create-home             create the user's home directory
  -M, --no-create-home          do not create the user's home directory
  -N, --no-user-group           do not create a group with the same name as
                                the user
```

linux [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# /etc/login.defs
bash: /etc/login.defs: Permission denied
[root@localhost ~]# chage
Usage: chage [options] LOGIN

Options:
  -d, --lastday LAST_DAY        set date of last password change to LAST_DAY
  -E, --expiredate EXPIRE_DATE  set account expiration date to EXPIRE_DATE
  -h, --help                    display this help message and exit
  -I, --inactive INACTIVE       set password inactive after expiration
                                to INACTIVE
  -l, --list                    show account aging information
  -m, --mindays MIN_DAYS        set minimum number of days before password
                                change to MIN_DAYS
  -M, --maxdays MAX_DAYS        set maximum number of days before password
                                change to MAX_DAYS
  -R, --root CHROOT_DIR         directory to chroot into
  -W, --warndays WARN_DAYS      set expiration warning days to WARN_DAYS

[root@localhost ~]# #access control using ACLs
[root@localhost ~]# chmod
chmod: missing operand
Try 'chmod --help' for more information.
[root@localhost ~]# chmod
chmod: missing operand
```

linux [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# # authentication security
[root@localhost ~]# /etc/ssh/sshd_config
bash: /etc/ssh/sshd_config: Permission denied
[root@localhost ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Passphrases do not match.  Try again.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Saving key "/root/.ssh/id_rsa" failed: passphrase is too short (minimum five characters
)
[root@localhost ~]# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Max kernel policy version:      31
[root@localhost ~]# #configure policies
[root@localhost ~]# semanage
```

linux [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# journalctl
-- Logs begin at Wed 2024-09-25 03:50:55 +07, end at Wed 2024-09-25 12:00:01 +07. --
Sep 25 03:50:55 localhost.localdomain systemd-journal[89]: Runtime journal is using 8.0
Sep 25 03:50:55 localhost.localdomain kernel: Initializing cgroup subsys cpuset
Sep 25 03:50:55 localhost.localdomain kernel: Initializing cgroup subsys cpu
Sep 25 03:50:55 localhost.localdomain kernel: Initializing cgroup subsys cpuacct
Sep 25 03:50:55 localhost.localdomain kernel: Linux version 3.10.0-1160.71.1.el7.x86_64
Sep 25 03:50:55 localhost.localdomain kernel: Command line: BOOT_IMAGE=/vmlinuz-3.10.0-
Sep 25 03:50:55 localhost.localdomain kernel: [Firmware Bug]: TSC doesn't count with P0
Sep 25 03:50:55 localhost.localdomain kernel: e820: BIOS-provided physical RAM map:
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x0000000000000000-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x000000000009fc00-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x00000000000f0000-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x0000000000100000-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x000000007fff0000-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x00000000fec00000-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x00000000fee00000-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: BIOS-e820: [mem 0x00000000fffc0000-0x0000
Sep 25 03:50:55 localhost.localdomain kernel: NX (Execute Disable) protection: active
Sep 25 03:50:55 localhost.localdomain kernel: SMBIOS 2.5 present.
Sep 25 03:50:55 localhost.localdomain kernel: DMI: innotek GmbH VirtualBox/VirtualBox,
Sep 25 03:50:55 localhost.localdomain kernel: Hypervisor detected: KVM
Sep 25 03:50:55 localhost.localdomain kernel: e820: update [mem 0x00000000-0x00000fff]
Sep 25 03:50:55 localhost.localdomain kernel: e820: remove [mem 0x000a0000-0x000fffff]
Sep 25 03:50:55 localhost.localdomain kernel: e820: last_pfn = 0x7fff0 max_arch_pfn = 0
```

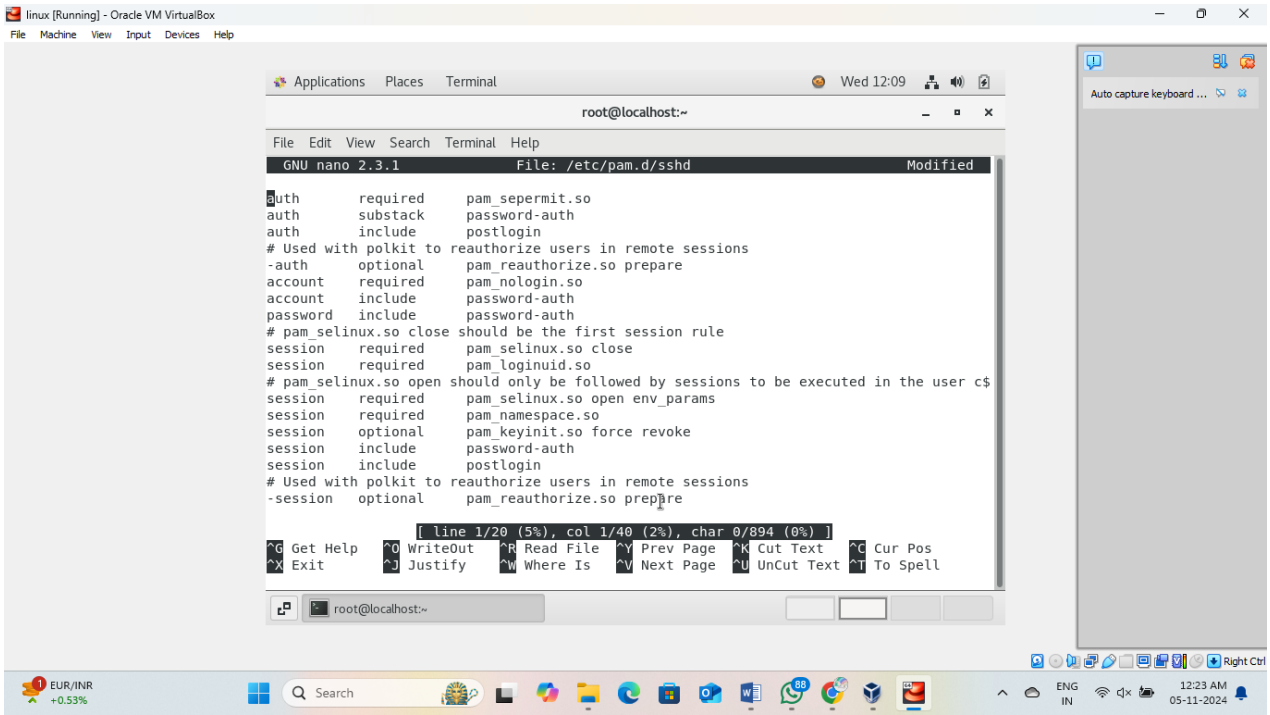
```
File  Edit  View  Search  Terminal  Help

            yum-config-manager --disable <repoid>
        or
            subscription-manager repos --disable=<repoid>

    5. Configure the failing repository to be skipped, if it is unavailable.
       Note that yum will try to contact the repo. when it runs most commands,
       so will have to try and fail each time (and thus. yum will be be much
       slower). If it is a very temporary problem though, this is often a nice
       compromise:

            yum-config-manager --save --setopt=<repoid>.skip_if_unavailable=true

Cannot find a valid baseurl for repo: base/7/x86_64
[root@localhost ~]# google-authenticator
bash: google-authenticator: command not found...
[root@localhost ~]# sudo nano /etc/pam.d/sshd
```



Conclusion

Summarize the project's achievements, highlighting the benefits of implementing a secure, multi-layered authentication system. The **User Authentication System in Linux** project successfully demonstrates how to create a secure and manageable environment for multi-user systems. By implementing structured user management, access controls, and Two-Factor Authentication (2FA), the system ensures that only authorized users can access critical files and services. This project has achieved its primary goal of enhancing Linux security through

robust authentication measures, and it highlights the importance of layered security in safeguarding sensitive information.

This project underscores the critical role of secure user authentication in a Linux environment. By combining traditional password-based authentication with advanced techniques like ACLs and 2FA, the system provides a solid foundation for managing user access securely. As security threats continue to evolve, maintaining a multi-layered, adaptable approach to user authentication will be essential for protecting Linux-based systems.

References

List all resources, manuals, and documentation used to implement and configure the Linux user authentication system, including the official Linux and Google Authenticator documentation.

Appendices

- **Appendix A**: Sample configuration files (e.g., PAM configuration for 2FA, ACL settings).
- **Appendix B**: Screenshots of the system in operation.