INDEX:

Project 1#: Basic HTTPD Web Server Configuration

Project 2#: Project 2: Sudoer User for Limited Privileges

Project 3: Group Management for Shared Files

Project 4#: Static Hostname Configuration

Project 5#: Configuring Secure SSH Access

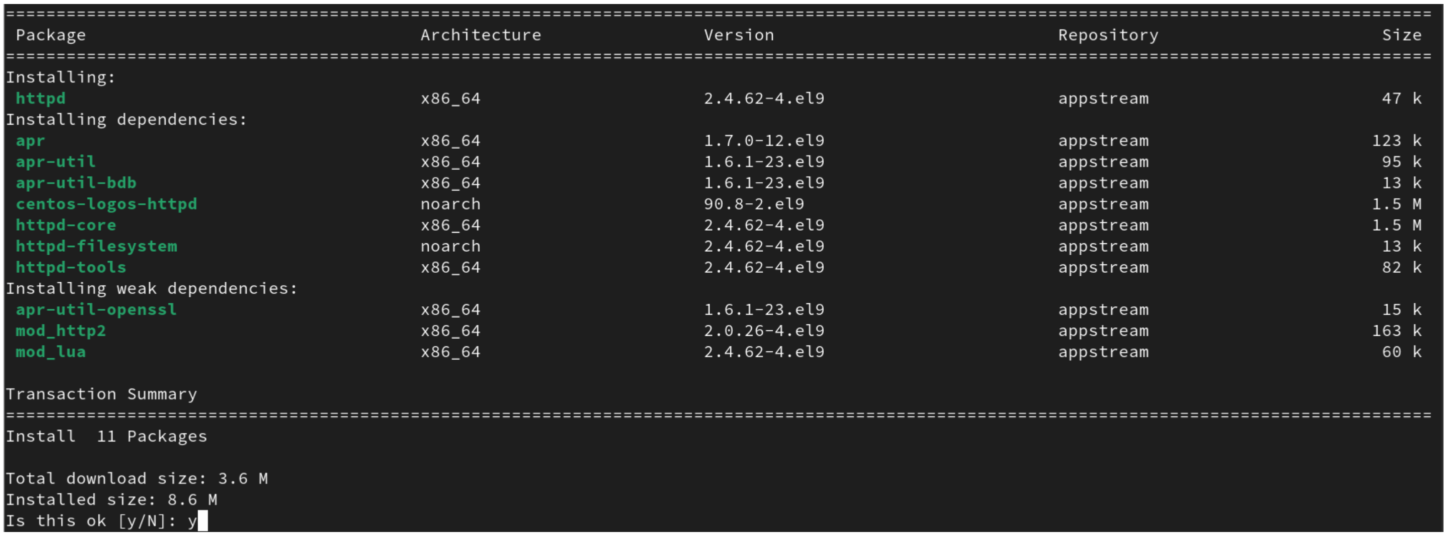
Project 1# : Basic HTTPD Web Server Configuration

Problem Statement:

Organizations need to deploy and secure web servers while ensuring proper access and configuration management.

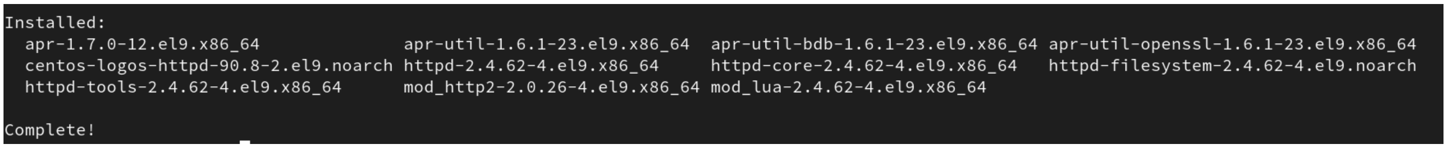
A. Installing Apache web server on server machine

> sudo yum install httpd



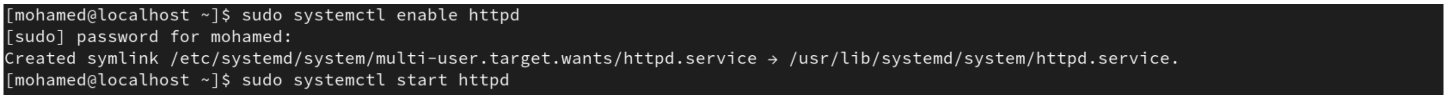
```
================================================================================================================================
 Package                             Architecture             Version                          Repository                 Size
================================================================================================================================
Installing:
 httpd                               x86_64                   2.4.62-4.el9                     appstream                  47 k
Installing dependencies:
 apr                                 x86_64                   1.7.0-12.el9                     appstream                 123 k
 apr-util                            x86_64                   1.6.1-23.el9                     appstream                  95 k
 apr-util-bdb                        x86_64                   1.6.1-23.el9                     appstream                  13 k
 centos-logos-httpd                  noarch                   90.8-2.el9                       appstream                 1.5 M
 httpd-core                          x86_64                   2.4.62-4.el9                     appstream                 1.5 M
 httpd-filesystem                    noarch                   2.4.62-4.el9                     appstream                  13 k
 httpd-tools                         x86_64                   2.4.62-4.el9                     appstream                  82 k
Installing weak dependencies:
 apr-util-openssl                    x86_64                   1.6.1-23.el9                     appstream                  15 k
 mod_http2                           x86_64                   2.0.26-4.el9                     appstream                 163 k
 mod_lua                             x86_64                   2.4.62-4.el9                     appstream                  60 k

Transaction Summary
================================================================================================================================
Install  11 Packages

Total download size: 3.6 M
Installed size: 8.6 M
Is this ok [y/N]: y
```

> installed successfully



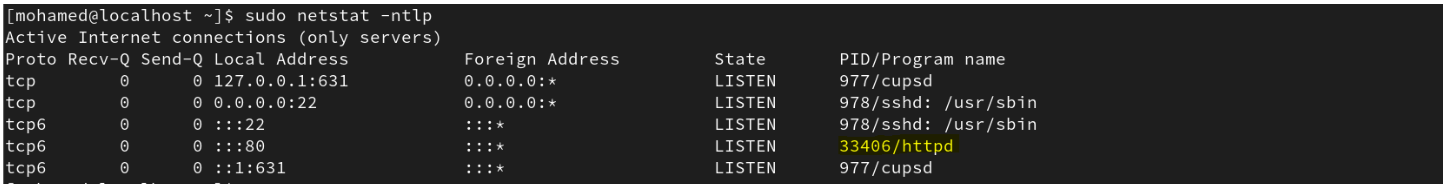
```
Installed:
  apr-1.7.0-12.el9.x86_64              apr-util-1.6.1-23.el9.x86_64   apr-util-bdb-1.6.1-23.el9.x86_64 apr-util-openssl-1.6.1-23.el9.x86_64
  centos-logos-httpd-90.8-2.el9.noarch httpd-2.4.62-4.el9.x86_64      httpd-core-2.4.62-4.el9.x86_64   httpd-filesystem-2.4.62-4.el9.noarch
  httpd-tools-2.4.62-4.el9.x86_64      mod_http2-2.0.26-4.el9.x86_64 mod_lua-2.4.62-4.el9.x86_64

Complete!
```

B. Enabling and starting httpd



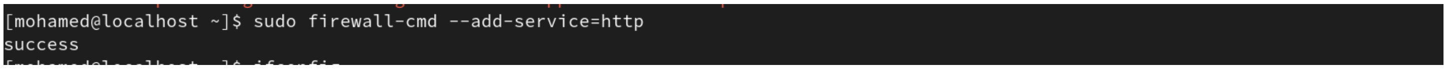
```
[mohamed@localhost ~]$ sudo systemctl enable httpd
[sudo] password for mohamed:
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[mohamed@localhost ~]$ sudo systemctl start httpd
```

C. Verifying that the service is now running



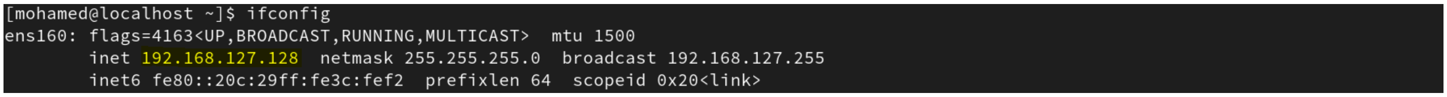
```
[mohamed@localhost ~]$ sudo netstat -ntlp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      977/cupsd
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      978/sshd: /usr/sbin
tcp6       0      0 :::22                   :::*                    LISTEN      978/sshd: /usr/sbin
tcp6       0      0 :::80                   :::*                    LISTEN      33406/httpd
tcp6       0      0 ::1:631                 :::*                    LISTEN      977/cupsd
```

D. Permitting the http service through the firewall



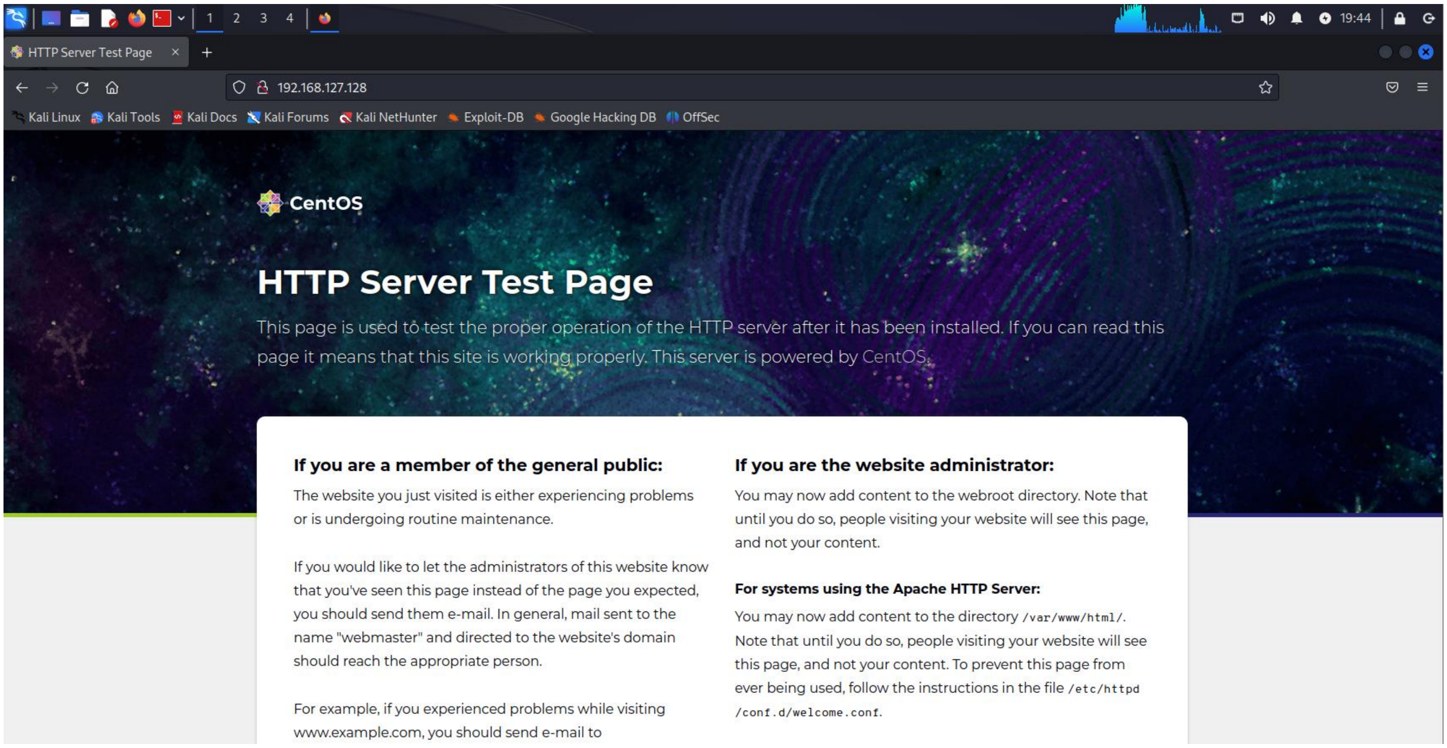
```
[mohamed@localhost ~]$ sudo firewall-cmd --add-service=http
success
```

E. Now we need to know the ip address of the server in order to make a connection from another machine



```
[mohamed@localhost ~]$ ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.127.128  netmask 255.255.255.0  broadcast 192.168.127.255
        inet6 fe80::20c:29ff:fe3c:fef2  prefixlen 64  scopeid 0x20<link>
```

F. Accessing the server from a (Kali) client machine



HTTP Server Test Page

This page is used to test the proper operation of the HTTP server after it has been installed. If you can read this page it means that this site is working properly. This server is powered by CentOS.

If you are a member of the general public:

The website you just visited is either experiencing problems or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to

If you are the website administrator:

You may now add content to the webroot directory. Note that until you do so, people visiting your website will see this page, and not your content.

For systems using the Apache HTTP Server:

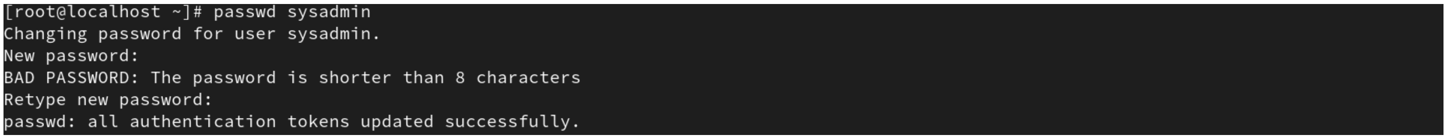
You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

Project 2#: Sudoer User for Limited Privileges

Problem Statement:

Organizations need to grant specific administrative privileges to users without providing full root access, in this scenario our super user should have user management

A. Creating a user (sysadmin) for granting administrative privileges

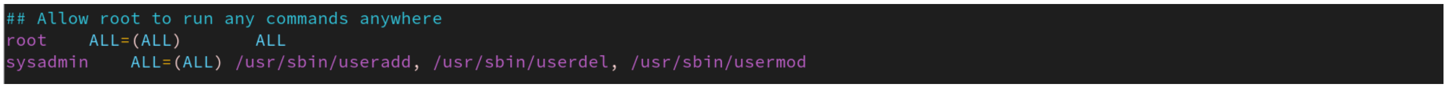


```
[root@localhost ~]# passwd sysadmin
Changing password for user sysadmin.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
```

B. Adding sysadmin in the sudoers file

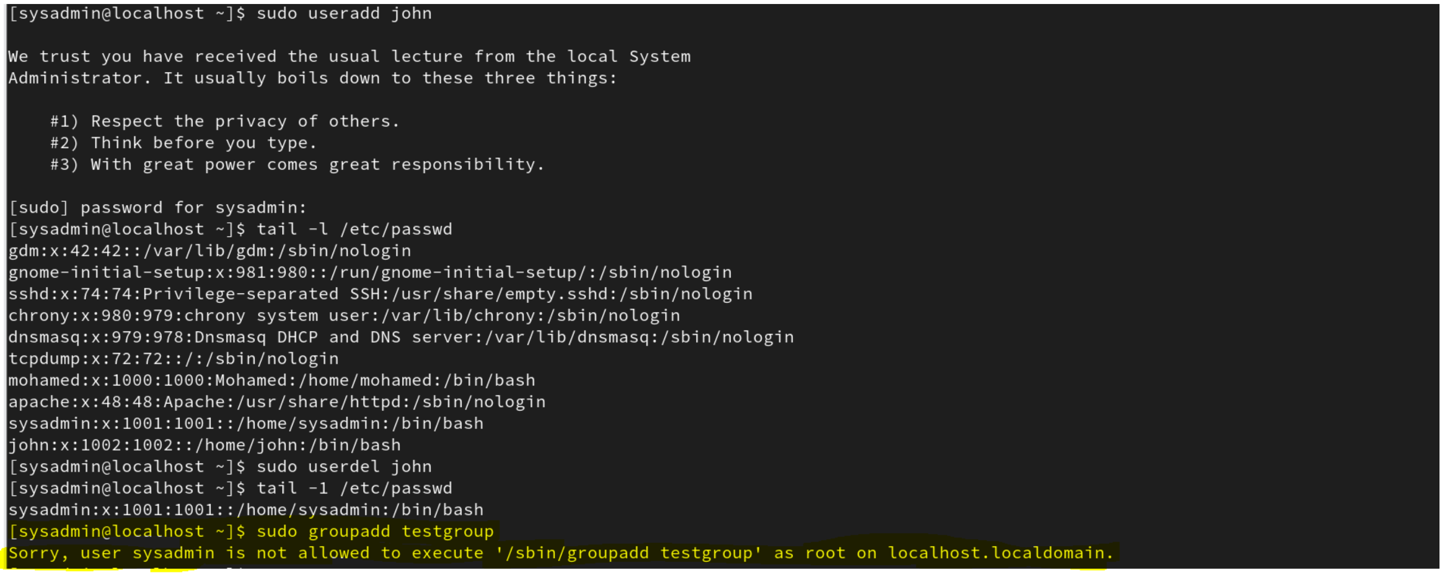
> visudo

> 100G to go the sudoers line



```
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL
sysadmin    ALL=(ALL) /usr/sbin/useradd, /usr/sbin/userdel, /usr/sbin/usermod
```

C. Logging into sysadmin and trying their new privileges



```
[sysadmin@localhost ~]$ sudo useradd john

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for sysadmin:
[sysadmin@localhost ~]$ tail -l /etc/passwd
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:981:980::/run/gnome-initial-setup/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/sbin/nologin
chrony:x:980:979:chrony system user:/var/lib/chrony:/sbin/nologin
dnsmasq:x:979:978:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
mohamed:x:1000:1000:Mohamed:/home/mohamed:/bin/bash
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
sysadmin:x:1001:1001::/home/sysadmin:/bin/bash
john:x:1002:1002::/home/john:/bin/bash
[sysadmin@localhost ~]$ sudo userdel john
[sysadmin@localhost ~]$ tail -1 /etc/passwd
sysadmin:x:1001:1001::/home/sysadmin:/bin/bash
[sysadmin@localhost ~]$ sudo groupadd testgroup
Sorry, user sysadmin is not allowed to execute '/sbin/groupadd testgroup' as root on localhost.localdomain.
```

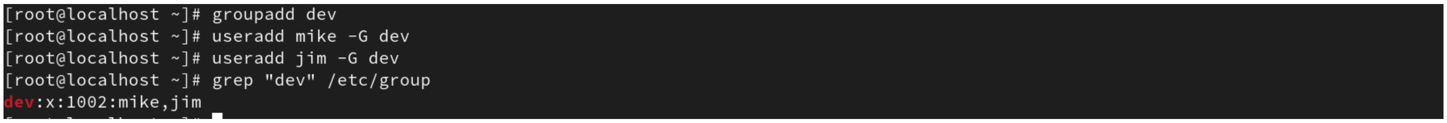
> sysadmin tried to add a group but he couldn’t due to lack of privileges

Project 3#: Group Management for Shared Files

Problem Statement:

Teams need shared access to specific files and directories with appropriate permissions

A. Creating a group and users for development team

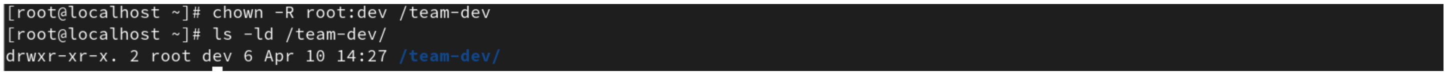


```
[root@localhost ~]# groupadd dev
[root@localhost ~]# useradd mike -G dev
[root@localhost ~]# useradd jim -G dev
[root@localhost ~]# grep "dev" /etc/group
dev:x:1002:mike,jim
```

B. Creating a shared directory for the development team

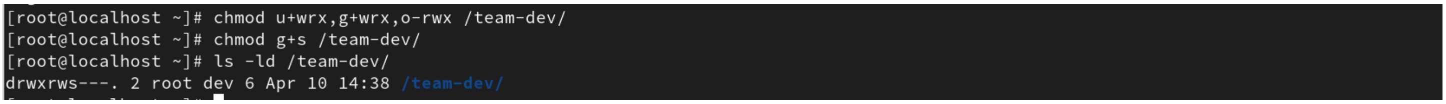
> sudo mkdir team-dev

C. Changing directory ownership to the dev team



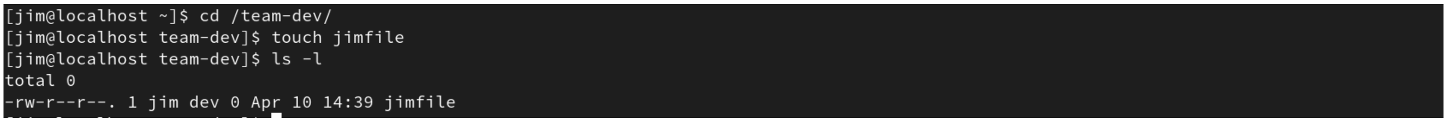
```
[root@localhost ~]# chown -R root:dev /team-dev
[root@localhost ~]# ls -ld /team-dev/
drwxr-xr-x. 2 root dev 6 Apr 10 14:27 /team-dev/
```

D. Changing the privelleges that outsiders from the group cant access or modify



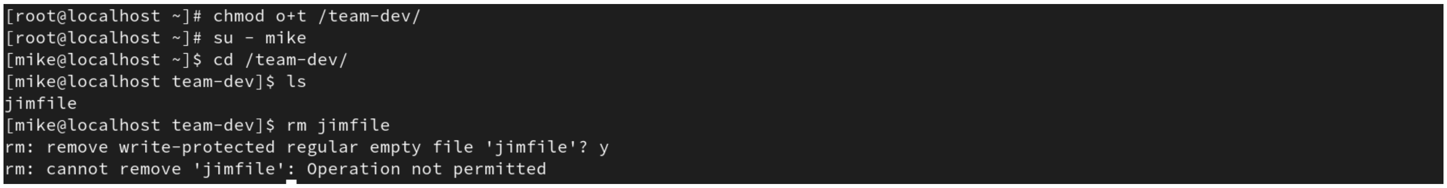
```
[root@localhost ~]# chmod u+wrx,g+wrx,o-rwx /team-dev/
[root@localhost ~]# chmod g+s /team-dev/
[root@localhost ~]# ls -ld /team-dev/
drwxrws---. 2 root dev 6 Apr 10 14:38 /team-dev/
```

E. Now group members (development team) can access the directory and create files



```
[jim@localhost ~]$ cd /team-dev/
[jim@localhost team-dev]$ touch jimfile
[jim@localhost team-dev]$ ls -l
total 0
-rw-r--r--. 1 jim dev 0 Apr 10 14:39 jimfile
```

F. Adding stickybit to the directory so that only file owners can remove their files



```
[root@localhost ~]# chmod o+t /team-dev/
[root@localhost ~]# su - mike
[mike@localhost ~]$ cd /team-dev/
[mike@localhost team-dev]$ ls
jimfile
[mike@localhost team-dev]$ rm jimfile
rm: remove write-protected regular empty file 'jimfile'? y
rm: cannot remove 'jimfile': Operation not permitted
```

Project 4# : Static Hostname Configuration

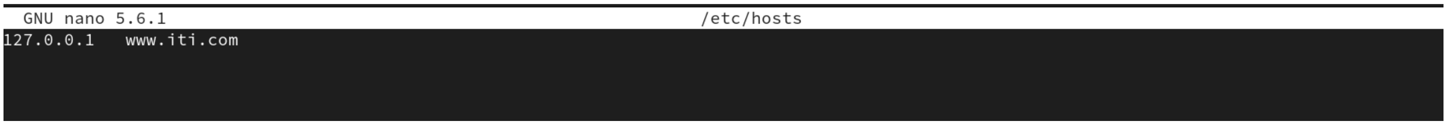
Problem Statement:

Systems need consistent identification across network reboots and restarts.

A. Setting the hostname via command

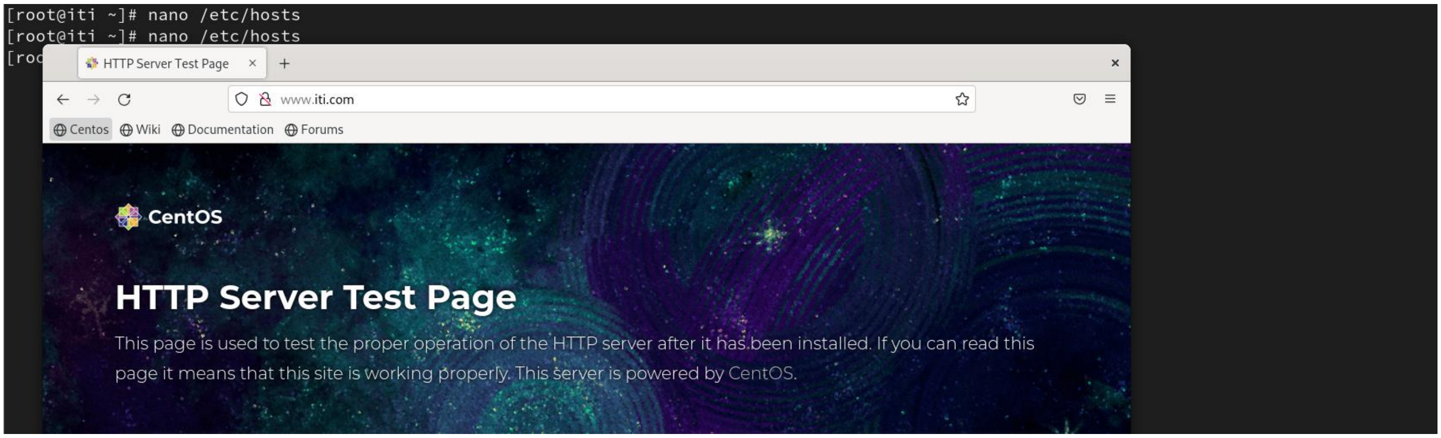
> sudo hostnamectl set-hostname iti

B. Adding the hostname to hosts file for the server machine



```
  GNU nano 5.6.1                                    /etc/hosts
127.0.0.1    www.iti.com
```

> testing accessing the website via hostname

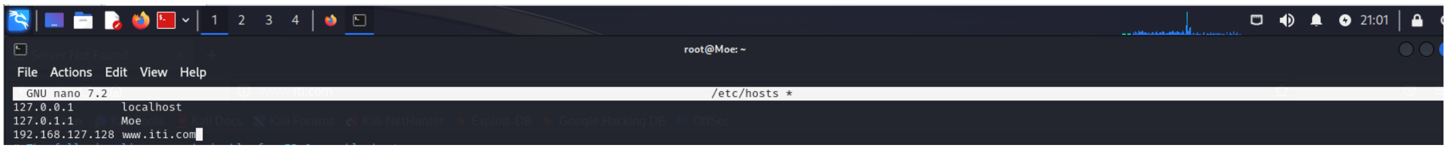


```
[root@iti ~]# nano /etc/hosts
[root@iti ~]# nano /etc/hosts
[roc
```

HTTP Server Test Page

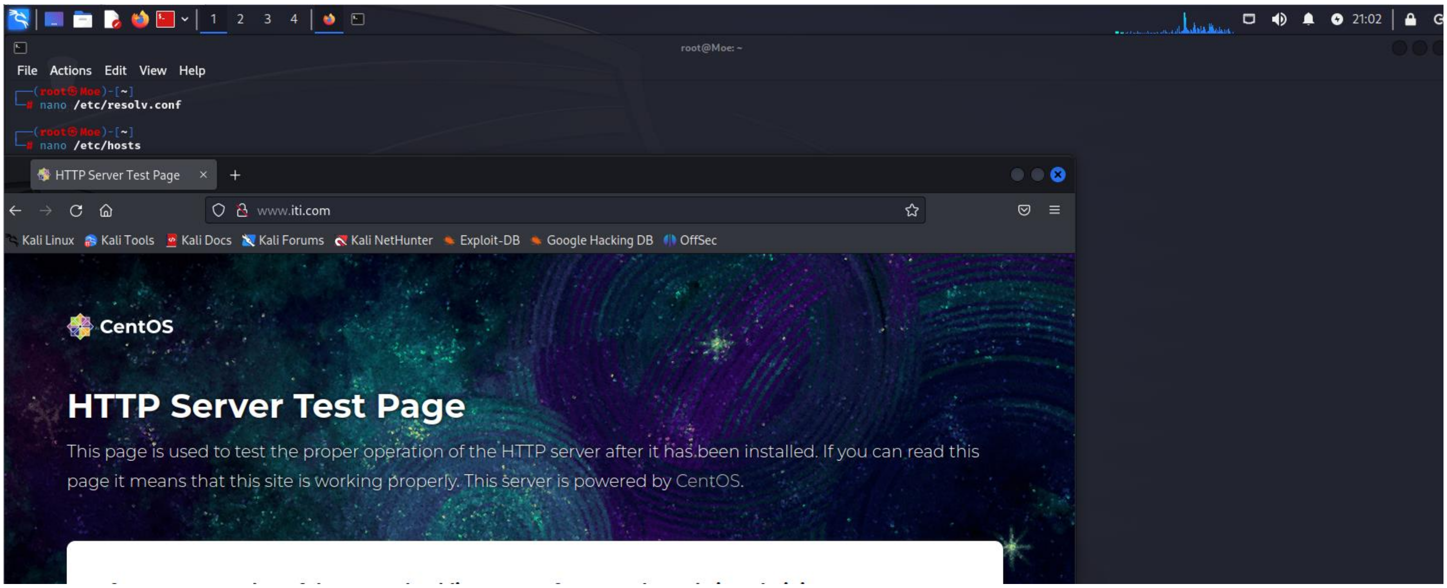
This page is used to test the proper operation of the HTTP server after it has been installed. If you can read this page it means that this site is working properly. This server is powered by CentOS.

C. Adding the hostname to hosts file for the client machine



```
  GNU nano 7.2                                      /etc/hosts *
127.0.0.1       localhost
127.0.1.1       Moe
192.168.127.128 www.iti.com
```

> testing accessing the website via hostname



```
┌──(root㉿Moe)-[~]
└─# nano /etc/resolv.conf

┌──(root㉿Moe)-[~]
└─# nano /etc/hosts
```

HTTP Server Test Page

This page is used to test the proper operation of the HTTP server after it has been installed. If you can read this page it means that this site is working properly. This server is powered by CentOS.

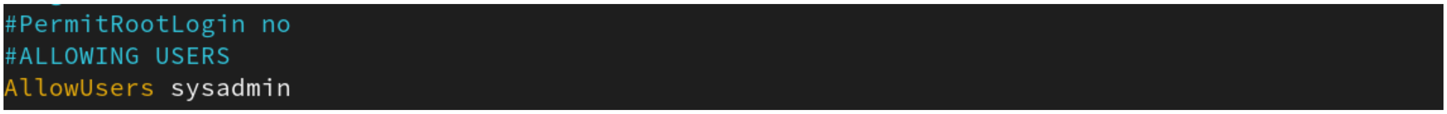
Project 5# : Configuring Secure SSH Access

Problem Statement:

In production environments, SSH access must be tightly controlled to prevent unauthorized access while maintaining efficiency for legitimate users

A. Managing ssh access only for the user sysadmin

> sudo vi /etc/ssh/sshd_config

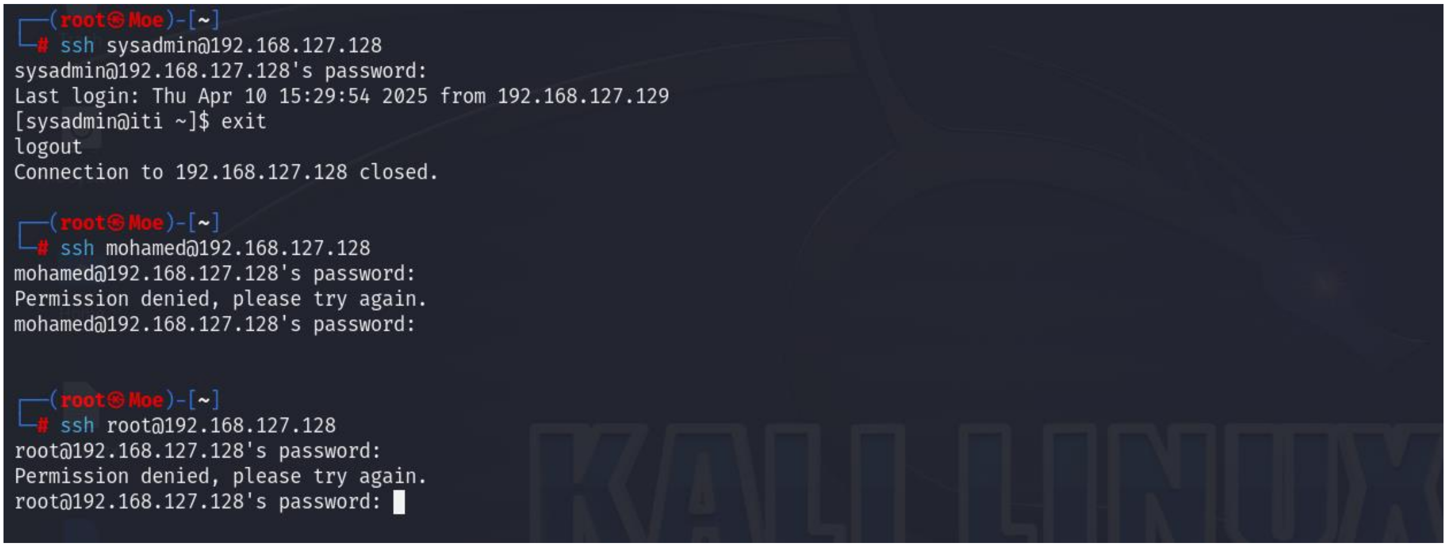


```
#PermitRootLogin no
#ALLOWING USERS
AllowUsers sysadmin
```

B. Restarting the ssh servic

> sudo systemctl restart sshd

C. Testing the ssh service from the client machine



```
┌──(root㉿Moe)-[~]
└─# ssh sysadmin@192.168.127.128
sysadmin@192.168.127.128's password:
Last login: Thu Apr 10 15:29:54 2025 from 192.168.127.129
[sysadmin@iti ~]$ exit
logout
Connection to 192.168.127.128 closed.

┌──(root㉿Moe)-[~]
└─# ssh mohamed@192.168.127.128
mohamed@192.168.127.128's password:
Permission denied, please try again.
mohamed@192.168.127.128's password:


┌──(root㉿Moe)-[~]
└─# ssh root@192.168.127.128
root@192.168.127.128's password:
Permission denied, please try again.
root@192.168.127.128's password: █
```