L10
[student@servera]$ #sudo dnf install httpd -y
[student@servera ~]$ #sudo systemctl enable httpd
[student@servera ~]$ #sudo systemctl start httpd
[student@servera ~]$ #sudo vim /etc/httpd/conf.d/www.doesitwork.local.conf
[student@servera ~]$ #sudo mkdir -p/var/www/www.doesitwork.local
[student@servera ~]$ #sudo vim /var/www/www.doesitwork.local/index.html
[student@servera ~]$ #httpd-t
[student@servera ~]$ #sudo systemctl restart httpd
[student@servera ~]$ #sudo firewall-cmd-permanent-add-service=http
[student@servera ~]$ #sudo firewall-cmd --reload
[student@servera ~]$ #ps aux grep httpd
[student@servera ~]$ #pstreep | grep httpd
[student@servera ~]$ #sudo renice 8 $(pgrep httpd)
[student@servera ~]$ #sudo vim /etc/security/limits.conf add at the end student soft priority 10
[student@servera 1$ #ps-u root--soft-comm
[student@servera]$ #psu root-sort commo comm,args
[student@servera 1$ #ps-u root-sort=commo comm,args sed 's/%//g
[student@servera]$ #psu root o args
[student@servera]$ #pseo pid,ppid,cmd, smem,&cpu-sort=-%cpu head maybe -n 1
[student@servera]$ #pseo pid,ppid,cmd,%mem,%cpu-sort-%mem | head n 2 | tail -n 1
[student@servera 1$ #pseo pid,ppid,cmd,etime-sort-etime tail -n 1
[student@servera ~]$ #sudo firewall-cmd --permanent --add-service=ssh
[student@servera ~]$ #sudo firewall-cmd -permanent-add-service=http
[student@servera ~]$ #sudo firewall-cmd --reload
[student@servera ~]$ #sudo vim /etc/firewalld/firewalld.conf and change LogDenied=all
[student@servera ~]$ #sudo firewall-cmd --runtime-to-permanent
[student@servera ~]$ #sudo journalctl -xe and check website http:/serverip:8080
[student@serverb ~]$ # sestatus getenforce
[student@serverb ~]$ # sudo setenforce 1
[student@serverb~]$ # add at the end student soft nofile 10000 next row student hard nofile 10000
[student@serverb ~]$ # add at the end student soft priority 3 next row student hard priority 3

Rgext:

1. Select Rows Containing `fish` grep 'fish' /usr/share/dict/words

2. Select Rows Containing `cat` with 2 Rows Before and 1 After:grep -B 2 -A 1 'cat' /usr/share/dict/words

3. Number of Occurrences of `cat`:grep -o 'cat' /usr/share/dict/words | wc -l

4. Select Rows with `cat` and Show Row Number:grep -n 'cat' /usr/share/dict/words

5. Select Rows with `t`, a Vowel After, and Ending with `sh` grep -E 't[aeiou].*sh$' /usr/share/dict/words

6. Select Specific Words Exactly grep -E '^abominable$|^abominate$|^anomie$|^atomize$' /usr/share/dict/words

7.Count Words with `t`, Vowel After, Ending with `sh` grep -E 't[aeiou].*sh$' /usr/share/dict/words | wc -l

8. Select Words Exactly 14 Characters Long grep -E '^.{14}$' /usr/share/dict/words

9. Select Rows Starting with `bl`, Vowel After, Anything After Vowel** grep -E '^bl[aeiou].*' /usr/share/dict/words

10.Select Rows Containing a 2-Digit Number grep -E '[0-9]{2}' /usr/share/dict/words

11.Begin with Letter Followed by `e`, or Begin with Number**:grep -E '^[A-Za-z]e|^[0-9]' /usr/share/dict/words

12.Find Specific Words in a Single Command grep -E 'Bank|Banking|Flunking|Walking' /usr/share/dict/words

13. Find Files Ending with `password find / -type f -name '*password'

14.Find Files Beginning with `password find / -type f -name 'password*'

15.Find Files Named `password find / -type f -name 'password'

16.Find Files Containing `password grep -r 'password' /

mi2
10. Ensure Apache works with SELinux in enforcing mode (on SERVERB) You can check if SELinux is in enforcing mode: sestatus To enable SELinux enforcement sudo setenforce 1 To make it persist after reboot:
sudo sed -i 's/SELINUX=permissive/SELINUX=enforcing/g' /etc/selinux/config

11. Configure Apache to work on port 99 with SELinux enforcing (on SERVERB)
1. Edit the Apache config file to listen on port 99: sudo nano /etc/httpd/conf/httpd.conf Add or modify Listen 99
2. Allow the port in SELinux sudo semanage port -a -t http_port_t -p tcp 99 3. Restart Apache: sudo systemctl restart httpd

12. Create a new virtual host served from `/web-aplikacija` with SELinux in enforcing mode (on SERVERB)
1. Create the directory sudo mkdir -p /web-aplikacija sudo chown apache:apache /web-aplikacija
2. Configure SELinux permissions sudo semanage fcontext -a -t httpd_sys_content_t "/web-aplikacija(/.*)?" sudo restorecon -Rv /web-aplikacija
3. Create a virtual host configuration in Apache sudo nano /etc/httpd/conf.d/web-aplikacija.conf DocumentRoot "/web-aplikacija" ServerName your.server.name
4. Restart Apache sudo systemctl restart httpd

mi2
:1. **[I9-M, 2 points] On SERVERA, using the `grep` command (or any other command), display the number of lines in the `/usr/share/dict/words` file that begin with the letters "bird".**
    grep -c '^bird' /usr/share/dict/words

2. **[I9-M, 2 points] On SERVERA, using the `grep` command (or any other command), display all lines that contain the text "food", including the line number.**
grep -n 'food' /usr/share/dict/words

3. **[I9-M, 2 points] On SERVERA, using the `grep` command (or any other command), display all words that are exactly 12 characters long.**
grep -E '^[a-zA-Z]{12}$' /usr/share/dict/words

4. **[I9-Ž, 2 points] On SERVERA, using the `find` command (or any other command), display all files in the filesystem that end with ".config".**
find / -name '*.config'

5. **[I9-M, 2 points] On SERVERA, display the status of the SSH daemon (SSHD).**
systemctl status sshd

6. **[I9-M, 2 points] On SERVERA, display and explain the status of the `mysql-server` process.**
systemctl status mysql Explain the output based on whether the process is running, stopped, or in any other state.

7. **[I9-M, 2 points] On SERVERA, set the priority of the `mysql-server` process to the lowest possible value.**
renice 19 -p $(pgrep mysql)

8. **[I9-Ž, 1 point] On SERVERA, set the maximum number of open files for the user "ishod3_remote" to 12,000.**
- Edit `/etc/security/limits.conf` and add the following line: ishod3_remote soft nofile 12000

9. **[I9-M, 2 points] On SERVERB, configure everything necessary to ensure the Apache web server works with the firewall enabled.**
firewall-cmd --permanent --add-service=http
firewall-cmd --reload

10. **[I9-M, 2 points] On SERVERB, configure everything necessary to ensure the Apache web server works with SELinux in "Enforcing" mode. Set SELinux to start in "Permissive" mode on system startup.**
- Enable SELinux in enforcing mode temporarily:
setenforce 1

- To set SELinux to permissive mode on startup, edit `/etc/selinux/config` and set:
SELINUX=permissive
11. **[I9-M, 2 points] On SERVERB, configure everything necessary to ensure the Apache web server works on port 99 with SELinux in "Enforcing" mode.**
- Allow Apache to listen on port 99:semanage port -a -t http_port_t -p tcp 99

12. **[I9-Ž, 1 point] On SERVERB, create a new Virtual Host within Apache web server to serve from the `/web-aplikacija` directory. Prove that the Virtual Host works with SELinux in "Enforcing" mode.**
- Configure a Virtual Host in Apache with a DocumentRoot set to `/web-aplikacija` and verify with curl http://localhost

MI2

14. Set SELinux to Enforcing mode on SERVERB and prove it.

1. Check the current SELinux mode sudo sestatus

2. If it’s not already in **Enforcing** mode, set SELinux to Enforcing sudo setenforce 1

3. To ensure SELinux starts in **Enforcing** mode after a reboot, edit the `/etc/selinux/config` file
sudo nano /etc/selinux/config Find the line `SELINUX=` and change it to SELINUX=enforcing

4. Verify it by running `sestatus` again to confirm SELinux is now in **Enforcing** mode.

15. Serve a website using Apache from the `/webapp` directory on port 85.

2. Create the `/webapp` directory and add your website content:
   sudo mkdir -p /webapp
   echo "Welcome to the website!" | sudo tee /webapp/index.html

3. Configure Apache to serve from `/webapp` on port 85:
   - Edit the Apache configuration file, usually located at `/etc/httpd/conf/httpd.conf`.
   - Add or modify a VirtualHost block for port 85:
     ```bash
     <VirtualHost *:85>
        DocumentRoot "/webapp"
        <Directory "/webapp">
            Require all granted
        </Directory>
     </VirtualHost>

### 16. Ensure the web server works with the firewall enabled.

1. Allow traffic on port 85 through the firewall:
   sudo firewall-cmd --add-port=85/tcp --permanent
   sudo firewall-cmd --reload

2. Verify the firewall rule sudo firewall-cmd --list-all

### 17. Set the priority of the web server process to 12.

1. Find the Apache (httpd) process ID pgrep httpd

2. Set the process priority using `renice` sudo renice 12 -p <PID>

### 18. Set the maximum number of open files for the user "student" to 10,000.

1. Edit `/etc/security/limits.conf`  student soft nofile 10000 student hard nofile 10000

### 19. Set the default priority for all processes run by the "student" user to 3.

1. Add the default priority setting to `/etc/security/limits.conf  student soft priority 3 student hard priority 3
     ```

I6

sudo dnf -y install httpd, sudo systemctl enable --now httpd, sudo systemctl status httpd
sudo dnf -y install nginx, sudo systemctl enable --now nginx

sudo vim /etc/httpd/conf.d/www.apache.local.conf

```
<VirtualHost *.80>
        ServerName www.apache.local
        DocumentRoot /var/www/www.apache.local
</VirtualHost>
```

sudo mkdir -p /var/www/www.apache.local
sudo vim /var/www/www.apache.local/index.html insert www.apache.local
sudo mkdir -p /var/log/httpd/
sudo httpd -t syntax check
sudo vim /etc/host/ 172.25.250.11 primer-ispita 172.25.250.10 www.apache.local
sudo systemctl restart httpd
curl www.apache.local
sudo vim /etc/nginx/conf.d/www.nginx.local.conf

```
server {
        server name www.nginx.local;
        root                    /var/www/www,nginx.local;
}
```

sudo mkdir -p /var/www/www.nginx.local
sudo vim /var/www/www.ngix.local/index.html insert www.nginx.local
sudo -R o+r /var/www/
sudo nginx -t
sudo vim /etc/host insert 172.25.250.11 www.nginx.local web.nginx.local
sudo systemctl restart nginx