Değerlendirme komutları⌨️

1 ◦ Herhangi bir grafik arayüzün kullanımda olmadığını doğrulayın.
Komutu kullanacağız **ls /usr/bin/*session** ve ekran görüntüsündeki ile aynı sonucu vermeli. Farklı bir şey görünüyorsa, bir grafik arayüz kullanılıyordur.

```
gemartin@gemartin42:~$ ls /usr/bin/*session
/usr/bin/dbus-run-session
gemartin@gemartin42:~$
```

2 ◦ UFW hizmetinin kullanımda olduğunu kontrol edin.
sudo ufw status

```
root@gemartin42:/home/gemartin# sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
4242                       ALLOW       Anywhere
80                         ALLOW       Anywhere
4242 (v6)                  ALLOW       Anywhere (v6)
80 (v6)                    ALLOW       Anywhere (v6)

root@gemartin42:/home/gemartin# _
```

sudo service ufw status

```
root@gemartin42:/home/gemartin# sudo service ufw status
● ufw.service - Uncomplicated firewall
     Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
     Active: active (exited) since Thu 2022-11-24 01:19:28 CET; 5min ago
       Docs: man:ufw(8)
    Process: 316 ExecStart=/lib/ufw/ufw-init start quiet (code=exited, status=0/SUCCESS)
   Main PID: 316 (code=exited, status=0/SUCCESS)
        CPU: 41ms

Nov 24 01:19:28 gemartin42 systemd[1]: Finished Uncomplicated firewall.
Warning: journal has been rotated since unit was started, output may be incomplete.
root@gemartin42:/home/gemartin# _
```

3 ◦ SSH hizmetinin kullanımda olduğunu kontrol edin.
sudo service ssh status

```
root@gemartin42:/home/gemartin# sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-11-24 01:19:30 CET; 7min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 552 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 613 (sshd)
      Tasks: 1 (limit: 1127)
     Memory: 3.8M
        CPU: 18ms
     CGroup: /system.slice/ssh.service
             └─613 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Nov 24 01:19:30 gemartin42 systemd[1]: Starting OpenBSD Secure Shell server...
Nov 24 01:19:30 gemartin42 sshd[613]: Server listening on 0.0.0.0 port 4242.
Nov 24 01:19:30 gemartin42 sshd[613]: Server listening on :: port 4242.
Nov 24 01:19:30 gemartin42 systemd[1]: Started OpenBSD Secure Shell server.
root@gemartin42:/home/gemartin#
```

4 ◦ Debian veya Centos işletim sistemini kullandığınızdan emin olun.
uname -vÖuname --kernel-version

```
root@gemartin42:~# uname -v
#1 SMP Debian 5.10.149-2 (2022-10-21)
root@gemartin42:~#
```

5 ◦ Kullanıcınızın "sudo" ve "user42" gruplarında olup olmadığını kontrol edin.
getent group sudo
getent group user42

```
root@gemartin42:~# getent group sudo
sudo:x:27:gemartin
root@gemartin42:~# getent group user42
user42:x:1001:gemartin
root@gemartin42:~# _
```

6 ◦ Yeni bir kullanıcı oluşturun ve oluşturduğumuz parola politikasına uyduğunu gösterin.
sudo adduser name_user ve ilkeye uyan bir parola girin.

```
root@gemartin42:~# sudo adduser newuser
Adding user `newuser' ...
Adding new group `newuser' (1002) ...
Adding new user `newuser' (1001) with group `newuser' ...
Creating home directory `/home/newuser' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for newuser
Enter the new value, or press ENTER for the default
        Full Name []: _
```

7 ◦ "Değerlendirme" adında yeni bir grup oluşturuyoruz.
sudo addgroup evaluating

```
root@gemartin42:~# sudo addgroup evaluating
Adding group `evaluating' (GID 1003) ...
Done.
root@gemartin42:~# _
```

8 ◦ Yeni kullanıcıyı yeni gruba ekliyoruz.
sudo adduser name_user evaluating

```
root@gemartin42:~# sudo adduser newuser evaluating
Adding user `newuser' to group `evaluating' ...
Adding user newuser to group evaluating
Done.
root@gemartin42:~# _
```

Doğru girildiğini doğrulamak için.
getent group evaluating

```
root@gemartin42:~# getent group evaluating
evaluating:x:1003:newuser
root@gemartin42:~#
```

9 ◦ Makinenin ana bilgisayar adının doğru olduğunu kontrol edin login42.
hostname

```
root@gemartin42:~# hostname
gemartin42
root@gemartin42:~#
```

10 ◦ Oturum açma bilgilerinizi değerlendiricininkiyle değiştirmek için ana bilgisayar

adını değiştirin. Bu durumda, onu student42 ile değiştireceğiz.
sudo nano /etc/hostname ve giriş bilgimizi yenisiyle değiştirin.

```
root@gemartin42:/home/gemartin# sudo nano /etc/hostname
```

```
  GNU nano 5.4                                    /etc/hostname *
student42
```

sudo nano /etc/hosts ve giriş bilgimizi yenisiyle değiştirin.

```
root@gemartin42:/home/gemartin# sudo nano /etc/hosts
```

```
  GNU nano 5.4                                      /etc/hosts *
127.0.0.1       localhost
127.0.1.1       student42

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Makineyi yeniden başlatın.

```
root@gemartin42:/home/gemartin# sudo reboot_
```

Tekrar giriş yaptığımızda, host adının nasıl doğru bir şekilde değiştirildiğini görebiliriz.

```
gemartin@student42:~$ hostname
student42
gemartin@student42:~$
```

11 ◦ Tüm bölümlerin konuda belirtildiği gibi olduğunu kontrol edin.
lsblk

```
gemartin@gemartin42:~$ lsblk
NAME                    MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                       8:0    0   30G  0 disk
├─sda1                    8:1    0  476M  0 part  /boot
├─sda2                    8:2    0    1K  0 part
└─sda5                    8:5    0 29.5G  0 part
  └─sda5_crypt          254:0    0 29.5G  0 crypt
    ├─LVMGroup-root     254:1    0  9.3G  0 lvm   /
    ├─LVMGroup-swap     254:2    0  2.1G  0 lvm   [SWAP]
    ├─LVMGroup-home     254:3    0  4.7G  0 lvm   /home
    ├─LVMGroup-var      254:4    0  2.8G  0 lvm   /var
    ├─LVMGroup-srv      254:5    0  2.8G  0 lvm   /srv
    ├─LVMGroup-tmp      254:6    0  2.8G  0 lvm   /tmp
    └─LVMGroup-var--log 254:7    0  3.7G  0 lvm   /var/log
sr0                      11:0    1 1024M  0 rom
gemartin@gemartin42:~$
```

12 ◦ Sudo'nun kurulu olduğunu kontrol edin.
which sudo

```
gemartin@gemartin42:~$ which sudo
/usr/bin/sudo
gemartin@gemartin42:~$ _
```

Who'nun arandığı yollarda tüm paketler bulunmadığından, which kullanmak aslında iyi bir uygulama değildir. Ancak, basit ve öğrenmesi kolay bir komut olduğu için değerlendirme için daha iyidir. Daha iyi kullanım için aşağıdaki komutu kullanacağız:
dpkg -s sudo

```
gemartin@gemartin42:~$ dpkg -s sudo
Package: sudo
Status: install ok installed
Priority: optional
Section: admin
Installed-Size: 4589
Maintainer: Sudo Maintainers <sudo@packages.debian.org>
Architecture: amd64
Version: 1.9.5p2-3
Replaces: sudo-ldap
Depends: libaudit1 (>= 1:2.2.1), libc6 (>= 2.27), libpam0g (>= 0.99.7.1), libselinux1 (>= 3.1~), zl
b1g (>= 1:1.2.0.2), libpam-modules, lsb-base
Conflicts: sudo-ldap
Conffiles:
 /etc/init.d/sudo 1153f6e6fa7c0e2166779df6ad43f1a8
 /etc/pam.d/sudo 85da64f888739f193fc0fa896680030e
 /etc/sudo.conf cdb3df319152dbf3a1ccab9d5bd01ad0
 /etc/sudo_logsrvd.conf 8f2d34058527c9b8155de178aacff2cd
 /etc/sudoers b1f89c8342752a2a29bc5a3f8fd70437
 /etc/sudoers.d/README 8d3cf36d1713f40a0ddc38e1b21a51b6
Description: Provide limited super user privileges to specific users
 Sudo is a program designed to allow a sysadmin to give limited root
 privileges to users and log root activity.  The basic philosophy is to give
 as few privileges as possible but still allow people to get their work done.
 .
 This version is built with minimal shared library dependencies, use the
 sudo-ldap package instead if you need LDAP support for sudoers.
Homepage: https://www.sudo.ws/
gemartin@gemartin42:~$
```

13 ◦ Yeni kullanıcıyı sudo grubuna ekleyin.
sudo adduser name_user sudo

```
root@gemartin42:/home/gemartin# sudo adduser newuser sudo
Adding user `newuser' to group `sudo' ...
Adding user newuser to group sudo
Done.
root@gemartin42:/home/gemartin# _
```

Grup içinde olup olmadığını kontrol ediyoruz.
getent group sudo

```
root@gemartin42:/home/gemartin# getent group sudo
sudo:x:27:gemartin,newuser
root@gemartin42:/home/gemartin# _
```

14 ◦ Özne tarafından sudo için konulan kuralların uygulanmasını gösterin.
cd /var/log/sudo & nano /etc/sudoers.d/sudo_config

```
root@gemartin42:/var/log/sudo# nano /etc/sudoers.d/sudo_config
```

```
  GNU nano 5.4                          /etc/sudoers.d/sudo_config
Defaults passwd_tries=3
Defaults badpass_message="Clave incorrecta"
Defaults logfile="/var/log/sudo_config"
Defaults log_input, log_output
Defaults iolog_dir="/var/log/sudo"
Defaults requiretty
Defaults secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"
```

15 ◦ **/var/log/sudo/** yolunun var olduğunu ve en az bir dosya içerdiğini gösterin, burada sudo ile kullanılan komutların geçmişini görmeliyiz.

```
root@gemartin42:/var/log/sudo# cd
root@gemartin42:~# cd /var/log/sudo
root@gemartin42:/var/log/sudo# ls
00  seq  sudo_config
root@gemartin42:/var/log/sudo#
```

```
root@gemartin42:/var/log/sudo# cat sudo_config
Nov 24 05:16:28 : root : TTY=tty1 ; PWD=/var/log/sudo ; USER=root ; TSID=00001W
    ; COMMAND=/usr/bin/nano 00
Nov 24 05:17:21 : root : TTY=tty1 ; PWD=/var/log/sudo ; USER=root ; TSID=00001X
    ; COMMAND=/usr/bin/nano hellogithub
root@gemartin42:/var/log/sudo#
```

Sudo ile bir komut çalıştırın ve dosyanın güncellenip güncellenmediğini kontrol edin.

```
root@gemartin42:/var/log/sudo# sudo nano hello42world_
```

```
root@gemartin42:/var/log/sudo# cat sudo_config
Nov 24 05:16:28 : root : TTY=tty1 ; PWD=/var/log/sudo ; USER=root ; TSID=00001W
    ; COMMAND=/usr/bin/nano 00
Nov 24 05:17:21 : root : TTY=tty1 ; PWD=/var/log/sudo ; USER=root ; TSID=00001X
    ; COMMAND=/usr/bin/nano hellogithub
Nov 24 05:23:10 : root : TTY=tty1 ; PWD=/var/log/sudo ; USER=root ; TSID=00001Y
    ; COMMAND=/usr/bin/nano hello42world
root@gemartin42:/var/log/sudo# _
```

16 ◦ UFW programının sanal makinede kurulu olduğunu ve düzgün çalıştığını kontrol edin.
dpkg -s ufw

```
root@gemartin42:~# dpkg -s ufw
Package: ufw
Status: install ok installed
Priority: optional
Section: admin
Installed-Size: 837
Maintainer: Jamie Strandboge <jamie@ubuntu.com>
Architecture: all
Version: 0.36-7.1
Depends: iptables, lsb-base (>= 3.0-6), ucf, python3:any, debconf (>= 0.5) | debconf-2.0
Suggests: rsyslog
Conffiles:
 /etc/default/ufw a921dd9d167380b04de4bc911915ea44
 /etc/init.d/ufw 4156943ab8a824fcf4b04cc1362eb230
 /etc/logrotate.d/ufw 12b1fb7cee76fc46f161e1ead1a22ce6
 /etc/rsyslog.d/20-ufw.conf 98e2f72c9c65ca8d6299886b524e80d1
 /etc/ufw/applications.d/ufw-bittorent d9451245a3fb2aa85ed91533ce530f27
 /etc/ufw/applications.d/ufw-chat 73204a7a2819499d7802bc83b7e63ee9
 /etc/ufw/applications.d/ufw-directoryserver 28888bb4f7fa81ea2ca23bb86995df5b
 /etc/ufw/applications.d/ufw-dnsserver 7a2634d40515a5baab2d5b355873e1e6
 /etc/ufw/applications.d/ufw-fileserver d43adc11063000fc3c1a824071382047
 /etc/ufw/applications.d/ufw-loginserver 366b3845c4360ea626f78875a400446b
 /etc/ufw/applications.d/ufw-mailserver 37e7910a1da915bcf60dac1c2d157377
 /etc/ufw/applications.d/ufw-printserver 47e009dc96a9eac7b3f2c2483a889756
 /etc/ufw/applications.d/ufw-proxyserver 6e035b6921d41aeee89c3d5867c593c5
 /etc/ufw/applications.d/ufw-webserver 07a41595f0b2c9865b7220bea998f8cf
 /etc/ufw/sysctl.conf 7723079fc108eda8f57eddab3079c70a
Description: program for managing a Netfilter firewall
 The Uncomplicated FireWall is a front-end for iptables, to make managing a
 Netfilter firewall easier. It provides a command line interface with syntax
 similar to OpenBSD's Packet Filter. It is particularly well-suited as a
 host-based firewall.
Homepage: https://launchpad.net/ufw
root@gemartin42:~# _
```

sudo service ufw status

```
root@gemartin42:~# sudo service ufw status
● ufw.service - Uncomplicated firewall
     Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
     Active: active (exited) since Thu 2022-11-24 03:49:57 CET; 1h 35min ago
       Docs: man:ufw(8)
    Process: 315 ExecStart=/lib/ufw/ufw-init start quiet (code=exited, status=0/SUCCESS)
   Main PID: 315 (code=exited, status=0/SUCCESS)
        CPU: 43ms

Nov 24 03:49:57 gemartin42 systemd[1]: Finished Uncomplicated firewall.
Warning: journal has been rotated since unit was started, output may be incomplete.
root@gemartin42:~# _
```

17 ◦ UFW'deki aktif kuralları listeleyin, eğer bonus kısmı yapılmadıysa, sadece 4242 numaralı bağlantı noktası kuralı görünmelidir.

sudo ufw status numbered

```
root@gemartin42:~# sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 4242                       ALLOW IN    Anywhere
[ 2] 80                         ALLOW IN    Anywhere
[ 3] 4242 (v6)                  ALLOW IN    Anywhere (v6)
[ 4] 80 (v6)                    ALLOW IN    Anywhere (v6)

root@gemartin42:~#
```

18 ◦ 8080 numaralı bağlantı noktası için yeni bir kural oluşturun. Etkin kurallara eklendiğini doğrulayın ve ardından silebilirsiniz.
sudo ufw allow 8080 onu oluşturmak için

```
root@gemartin42:~# sudo ufw allow 8080
Rule added
Rule added (v6)
root@gemartin42:~# _
```

sudo ufw status numbered

```
root@gemartin42:~# sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 4242                       ALLOW IN    Anywhere
[ 2] 80                         ALLOW IN    Anywhere
[ 3] 8080                       ALLOW IN    Anywhere
[ 4] 4242 (v6)                  ALLOW IN    Anywhere (v6)
[ 5] 80 (v6)                    ALLOW IN    Anywhere (v6)
[ 6] 8080 (v6)                  ALLOW IN    Anywhere (v6)

root@gemartin42:~# _
```

Kuralı silmek için komutu kullanmalıyız.
sudo ufw delete num_rule

```
root@gemartin42:~# sudo ufw delete 3
Deleting:
 allow 8080
Proceed with operation (y|n)? y
Rule deleted
root@gemartin42:~# _
```

Silinmiş olduğunu kontrol ediyoruz ve silinmesi gereken bir sonraki kuralın numarasını görüyoruz.
sudo ufw status numbered

```
root@gemartin42:~# sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 4242                       ALLOW IN    Anywhere
[ 2] 80                         ALLOW IN    Anywhere
[ 3] 4242 (v6)                  ALLOW IN    Anywhere (v6)
[ 4] 80 (v6)                    ALLOW IN    Anywhere (v6)
[ 5] 8080 (v6)                  ALLOW IN    Anywhere (v6)

root@gemartin42:~# _
```

Yeni kuralı silin.
sudo ufw delete num = 5

```
root@gemartin42:~# sudo ufw delete 5
Deleting:
 allow 8080
Proceed with operation (y|n)? y
Rule deleted (v6)
root@gemartin42:~#
```

Konu içerisinde sadece gerekli kuralların kaldığını kontrol ediyoruz.

```
root@gemartin42:~# sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 4242                       ALLOW IN    Anywhere
[ 2] 80                         ALLOW IN    Anywhere
[ 3] 4242 (v6)                  ALLOW IN    Anywhere (v6)
[ 4] 80 (v6)                    ALLOW IN    Anywhere (v6)

root@gemartin42:~# _
```

19 ◦ Sanal makinede ssh servisinin kurulu olduğunu, düzgün çalıştığını ve sadece 4242 numaralı portta çalıştığını kontrol ediniz.

which ssh

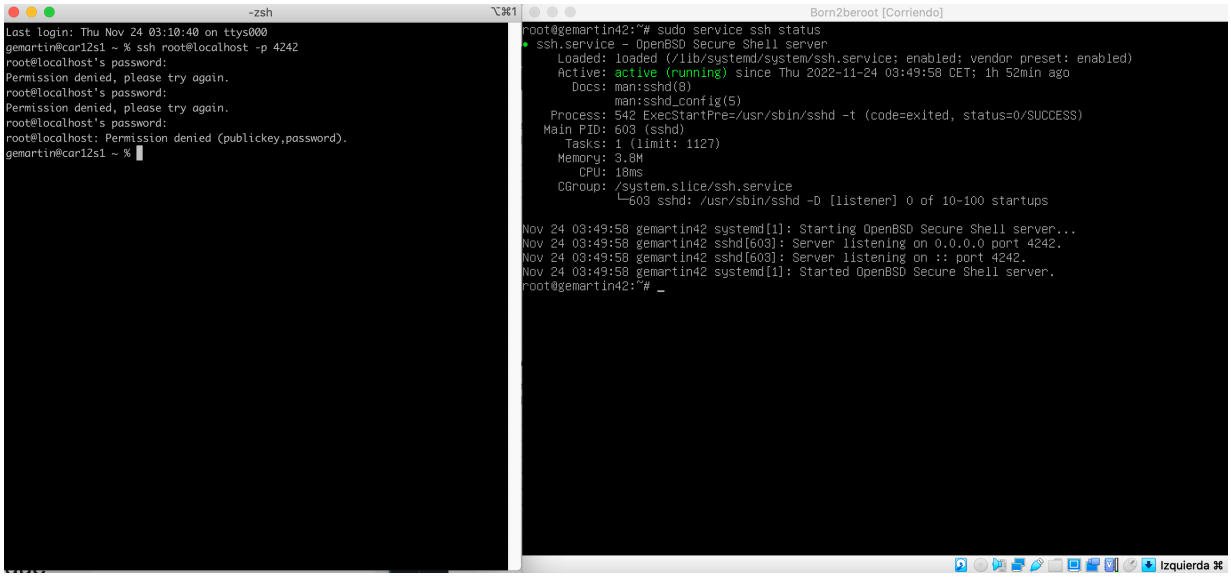
```
root@gemartin42:~# which ssh
/usr/bin/ssh
```

sudo service ssh status

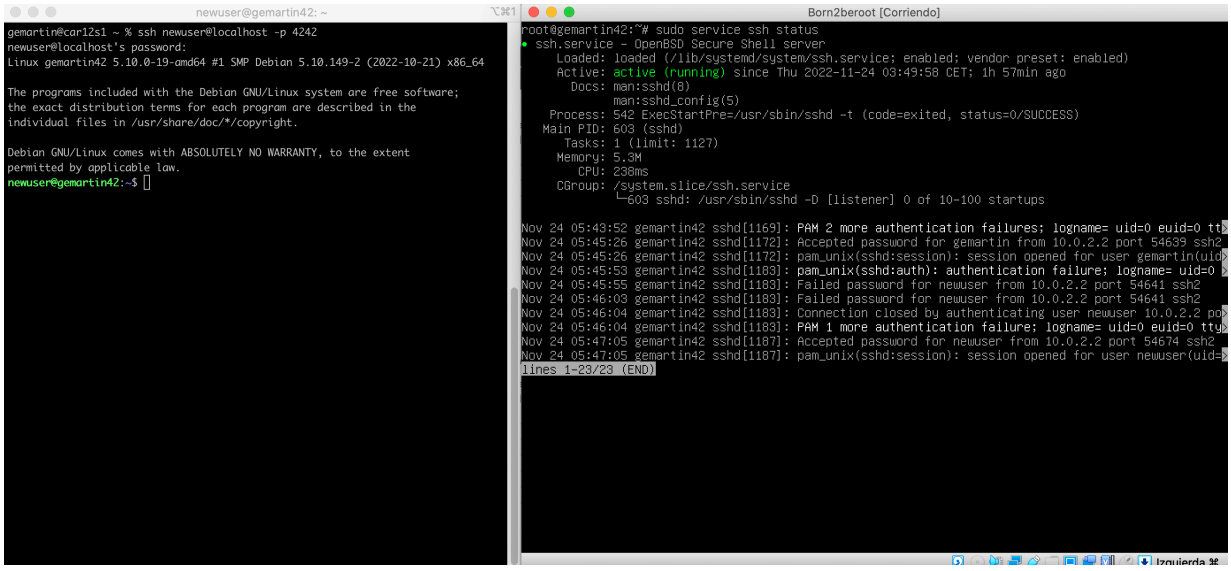
```
root@gemartin42:~# sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-11-24 03:49:58 CET; 1h 48min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 542 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 603 (sshd)
      Tasks: 1 (limit: 1127)
     Memory: 3.8M
        CPU: 18ms
     CGroup: /system.slice/ssh.service
             └─603 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Nov 24 03:49:58 gemartin42 systemd[1]: Starting OpenBSD Secure Shell server...
Nov 24 03:49:58 gemartin42 sshd[603]: Server listening on 0.0.0.0 port 4242.
Nov 24 03:49:58 gemartin42 sshd[603]: Server listening on :: port 4242.
Nov 24 03:49:58 gemartin42 systemd[1]: Started OpenBSD Secure Shell server.
root@gemartin42:~#
```

20 ◦ Yeni oluşturulan kullanıcıyla oturum açmak için ssh'yi kullanın. Kök kullanıcıyla ssh kullanamayacağınızdan emin olun.

Root kullanıcısı ile ssh üzerinden bağlanmaya çalışıyoruz fakat iznimiz yok.



Komutu kullanarak yeni kullanıcı ile ssh üzerinden bağlanıyoruz.
ssh newuser@localhost -p 4242



21 ◦ Komut dosyasının çalışma süresini 10 dakikadan 1'e değiştirin.
Crontab dosyasını değiştirmek için aşağıdaki komutu çalıştırıyoruz.
sudo crontab -u root -e

```
root@gemartin42:/home/gemartin# sudo crontab -u root -e
```

İlk parametreyi değiştiriyoruz, 10 yerine 1 olarak değiştiriyoruz.

```
  GNU nano 5.4                          /tmp/crontab.xTOGMU/crontab
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command

*/1 * * * * sh /home/gemartin/monitoring.sh
```

22 ◦ Translate English: Son olarak, komut dosyasını değiştirmeden sunucu başladığında komut dosyasının çalışmasını durdurun.
sudo /etc/init.d/cron stop

```
root@gemartin42:/home/gemartin# sudo /etc/init.d/cron stop
Stopping cron (via systemctl): cron.service.
root@gemartin42:/home/gemartin# _
```

Şunları yapmanız gerekir:
sudo /etc/init.d/cron start

```
root@gemartin42:/home/gemartin# sudo /etc/init.d/cron start
Starting cron (via systemctl): cron.service.
root@gemartin42:/home/gemartin# _
```

Bonus

Bellek ayrımı 2 puan
Sitenin oluşturulması 2 puan
Ekstra görev 1 puan

Test Cihazı🆗

Hiçbir şeyi unutmadığınızı kontrol edin! Kurulum ve yapılandırmanın başarıyla

gerçekleştirildiğini kontrol etmek için kendi test cihazı.
BURADA

```
------------------------------------------------------------

               TEST CREATED BY: GEMARTIN

------------------------------------------------------------
Graphical enviroment
[OK]

Disk partitions
[OK]
[OK]
[OK]
[OK]

SSH
[OK]
[OK]

UFW
[OK]
[OK]

Hostname
[OK]

Password policy
1.[OK]  minlen
2.[OK]  uppercase
3.[OK]  lowercase
4.[OK]  digit
5.[OK]  consecutive char
6.[OK]  difok
7.[OK]  enforce for root
8.[OK]  reject username
9.[OK]  passwd expire days
10.[OK]  days allowed before the modification
11.[OK]  warning message
12.[OK]  folder /var/log/sudo exist
```

```
Crontab
[OK]
```

Monitoring.sh

sort = alfabetik sırlama.
uniq = tekrar eden satırları ayırma.
$1,$2... = Mer $1 haba $2 dün $3 ya $4 sütunları tutuyor diyebiliriz.
free -m = Ram miktarını Mebibayt cinsinde gösterir. Böyle yapmamızın sebebi scripte kullanımlar üzerinden yüzde hesabı vs. yapmamız için.
grep '^/dev/' = ^ ekinden sonra gelen kelime ile başlayan yerleri alır.
grep -v '/boot$' = -v eki çıkarılacak kelimeyi gösterir
awk '{ft += $2} END {print ft}' = ft bir değişken olarak düşünüle bilir ft içerisinde $2 de yer alan verileri ekler ve ft yi ekrana yazdırır.
cut -c 9- | xargs | awk '{printf("%.1f%%") = "cut -c 9-" Bir karakteri veya bir karakter diziyi silmek için kullanılır. "xargs" Fonksiyon olarak öncesinden kullanılan çıktıyı bir sonraki komuta iletir. "printf("%.1f%%")" Float değer tipinde "."dan sonra 1 karakter alıp sonuna "%" ekler.

---

arc=$(uname -a)
--> Mevcut işletim sisteminin mimarisi ve kernel versiyonunu gösterir.
pcpu=$(grep "physical id" /proc/cpuinfo | sort | uniq | wc -l)
--> Fiziksel işlemci sayısını verir.
vcpu=$(grep "^processor" /proc/cpuinfo | wc -l)
--> Sanal işlemci sayısını verir.
fram=$(free -m | awk '$1 == "Mem:" {print $2}')
--> Sunucunun erişilebilir RAM miktarını verir.
uram=$(free -m | awk '$1 == "Mem:" {print $3}')
--> Kullanılan RAM miktarını verir.
pram=$(free | awk '$1 == "Mem:" {printf("%.2f"), $3/$2*100}')
--> printf("%.2f") virgülden sonra alınacak 2 değeri yollar $3/$2*100 ise yüzde olarak kullanımı verir

fdisk=$(df -Bg | grep '^/dev/' | grep -v '/boot$' | awk '{ft += $2} END {print ft}') --> Sunucunun erişilebilir depolama miktarını verir.
udisk=$(df -Bm | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} END {print ut}') --> Sunucunun kullanılan depolama alanını verir.
pdisk=$(df -Bm | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} {ft+= $2} END {printf("%d"), ut/ft*100}') --> "kullanılan depolama alanı / erişilebilir alan * 100" bize yüzde kullanımını verir.
cpul=$(top -bn1 | grep '^%Cpu' | cut -c 9- | xargs | awk '{printf("%.1f%%"), $1 + $3}') --> Yüzde olarak işlemci kullanım oranını verir.
lb=$(who -b | awk '$1 == "sistem" {print $3 " " $4}') --> Son yeniden başlatma tarihi ve saatini verir.
lvmt=$(lsblk | grep "lvm" | wc -l) --> LVM ile yapılandırılmış disklerin bilgisini verir.
lvmu=$(if [$lvmt -eq 0]; then echo hayır; else echo evet; fi) --> Sistemde LVM'nin aktif olup olmadığı bilgisini verir. NOT: Diğer öğelerin sorunsuz çalışması için net-tools paketini yüklemeniz gerekli.
ctcp=$(cat /proc/net/sockstat{,6} | awk '$1 == "TCP:" {print $3}') --> Mevcut aktif bağlantı sayısını verir.
ulog=$(users | wc -w) --> Sunucuyu kullanan kullanıcı sayısını verir.
ip=$(hostname -I) --> Sunucu IP Adresini verir.
mac=$(ip link show | awk '$1 == "link/ether" {print $2}') --> Sunucu MAC Adresini verir.
cmds=$(journalctl _COMM=sudo | grep COMMAND | wc -l) --> Sudo ile çalışıtırlmış komut sayısını verir. (Eğer sudoya giriş yapılmış ise diğer kullanıcıların sudo kullanım sayısıyla birlikte yazar öyle değilse içinde bulunulan kullanıcının kullandığı sudo command kadar bilgi verir.)
---

wall "   #Mimari ve Kernel Versiyonu: $arc

#Fiziksel İşlemci Sayısı: $pcpu
#Sanal İşlemci Sayısı: $vcpu
#RAM Miktar ve Kullanımı: $uram/${fram}MB ($pram%)
#Disk Miktarı ve Kullanımı: $udisk/${fdisk}Gb ($pdisk%)
#CPU Kullanım Oranı: $cpul
#Son Yeniden Başlama Tarihi: $lb
#LVM Kullanım Durumu: $lvmu
#Aktif Bağlantı Sayısı: $ctcp ESTABLISHED
#Sunucuyu Kullanan K. Sayısı: $ulog
#IP ve MAC Adresi: IP $ip ($mac)
#Kullanılmış Sudo Sayısı: $cmds cmd" # Diğer termianallerde gözükmesi için bunu ekliyoruz.

NOT : chmod +x monitoring.sh ile sh dosyamıza yetki tanımı yapmayı unutmayalım.

KURULUM

NOT: "No commands-Komut yok" gibi hatalarla karşılaşırsanız "sudo" parametresiyle yeniden deneyin, büyük ihtimal ile sorun çözülecektir.

SDA'lar yapılandırıldıktan sonra,
1-) "su" --> Yazıp root hesabına giriş yapılır (Root parolası soracak sistemi kurarken belirlemiş olduğumuz ilk parola)
2-) "apt install sudo" --> Yazıp sudo paketini yüklüyoruz (Sudo ile kullanıcılara root ayrıcalıkları tanınır)
3-) "adduser kullanıcı_adı sudo" --> Belirtilen kullanıcıyı sudo grubuna ekler. NOT: "getent group sudo" ile kullanıcı gruba eklenmiş mi diye kontrol edin!
4-) Sistemi "reboot" komutu ile yeniden başlatıyoruz.

5-) Gruba eklenen kullanıcıya login olduktan sonra "sudo -v" ile tekrar kontrol sağlayalım.
6-) "sudo apt update" --> Yazıp paket listelerini

güncelleyelim.

7-) "sudo visudo -f /etc/sudoers.d/blabla" --> Sudo dosyalarını "visudo" ile açmak daha güvenli bir yapı oluşturur, belirtilen dizine blabla adında yeni bir dosya yarattık ve aşağıda ki kodları ekleyeceğiz.

@-) Defaults passwd_tries=3

// Sudo ile max şifre deneme sayısı 3 yapılır (Standart olarak da 3'tür)

@-) Defaults badpass_message="bla bla bla bla bla bla"

// Hatalı şifre denemelerinden sonra gösterilecek hata mesajınız

@-) Defaults logfile="/var/log/sudo/blabla"

// Tüm kullanılan sudo komutlarını belirtilen dosyanın içerisinde saklar

@-) Defaults log_input,log_output

// Giriş ve çıkışların loglarını tutmak için kullanılır.

@-) Defaults iolog_dir="/var/log/sudo"

// Belirtilen dizine log_input ve log_output olaylarını arşivler.

@-) Defaults requiretty

// TTY modunu zorunlu kılar.

@-) Defaults secure_path="pdf'ye bak"

// Sudo tarafından kullanılan dizinleri sınırlandırmak için.

8-) Sistemi "reboot" komutu ile yeniden başlatıyoruz.

9-) "sudo apt install openssh-server" --> Openssh-server paketini kurar (Openssh dışarıdan gelen güvenli olmayan bir ağ üzerinden güvenli bir kanal sağlar)

10-) "sudo nano /etc/ssh/sshd_config" --> SSH Yapılandırmaların olduğu kısım.

@-) "#Port 22" olan kısmı tırnaklar olmadan "Port 4242" olarak değiştirin

@-) "#PermitRootLogin prohibit-password" kısmını da "PermitRootLogin no" şeklinde değiştirin. NOT: Tırnak işaretleri ve # işaretlerini koymadan yapacaksınız.

11-) Sistemi "reboot" komutu ile yeniden başlatıyoruz.

12-) "sudo service ssh status" --> Bu komutu yazarak ssh durumunu görüntüleye bilirsiniz.

13-) "sudo apt install ufw" --> Açılımı "uncomplicated firewall" yani diyor ki karmaşık olmayan güvenlik duvarı imiş (: o paketi kurarsınız.

14-) "sudo ufw enable" --> Güvenlik duvarımızı

aktif ediyoruz.

15-) "sudo ufw allow 4242" --> 4242 portu için güvenlik duvarımıza yeni kural ekleyip buna izin veriyoruz.

16-) "sudo ufw status" --> güvenlik duvarımızın durumunu kontrol ediyoruz.

17-) Sistemi "poweroff" komutu ile kapatıyoruz.

BURASI ÖNEMLİ: SSH BAĞLANTISI ATABİLMEK İÇİN VM ÜZERİNDEN PORT YÖNLENDİRMESİ YAPMAMIZ GEREKMEKTE BUNU

-Kurduğumuz sanal sunucuya VM arayüzünde sağ tıklayıp ayarlar kısmından ağ kısmına gelinir daha sonra gelişmiş menüsü altında B.noktası yönelndirme bölümüne tıklanır

-Açılan yeni pencerenin sağ tarafında yeşil renkte yeni bağlantı kuralı butonu bulunur, 1 kere tıklayalım. Daha sonra

-ADI Kısmına : SSH

-PROTOKOL Kısmına : TCP

-ANAMAKİNE IP Kısmına : 127.0.0.1

-ANAMAKİNE B.N Kısmına : 4242

-MİSAFİR IP Kısmına : 10.0.2.15

-MİSAFİR B.N Kısmına : 4242

Şeklinde düzenleme yapıp aşağıdan tamama basabilirsiniz.

18-) "sudo nano /etc/login.defs" --> Nano ile dosyayı açarız ve içinde ararız (NOT: TERMINAL UZERINDEN ARTIK SSH BAĞLANTISI ATABILDIGIMIZ İÇİN SANAL SUNUCUNUN KENDİSİNDE YIPRANMAYIN) :)

@-) PASS_MAX_DAYS 99999 --> PASS_MAX_DAYS 30 //Parola her 30 günde süresinin dolacağını belirttik

@-) PASS_MIN_DAYS 0 --> PASS_MIN_DAYS 2 // Şifreyi değiştikten sonra en az 2 gün sonra tekrar değiştirile bilir hale geleceğini belirttik.

@-) PASS_WARN_AGE 7 //Kullanıcı şifresinin bitmesine 7 gün kala bir uyarı ile karşılaşsın diye belirttik.

19-) "sudo apt install libpam-pwquality" --> Şifrelerin güvenliğini arttırmak için libpam-pwquality paketini kuruyoruz.

20-) "sudo nano /etc/pam.d/common-password" --> common-password belgesi açılır ve içerisinde aratılır: "password requisite pam_pwquality.so retry=3"

-retry=3'den sonra birer boşluk bırakılarak şunlar eklenir:

@-) minlen=10 // Minimum uzunluk 10

@-) ucredit=-1 dcredit=-1 // En az bir büyük ve bir sayısal karakter içermesi zorunluğu
@-) maxrepeat=3 // En fazla art arda 3 karakter kullanıla bilir.
@-) reject_username // Kullanıcı adını içermemelidir.
@-) difok=7 // Eski şifrenin içermediği 7 farklı karakter şartı
@-) enforce_for_root // Kuralları root için de geçerli kıldık
@@-) SON HALİ: password requisite pam_pwquality.so retry=3 minlen=10 ucredit=-1 dcredit=-1 maxrepeat=3 reject_username difok=7 enforce_for_root

---

ŞİMDİ YENİ YAPILANDIRMALARIMIZI TEST ETMEK İÇİN ROOT VE KURULUM SIRASINDA ADIMIZA AÇTIĞIMIZ DİĞER KULLANICININ ŞİFRELERİNİ DEĞİŞTİRECEĞİZ

21-) "sudo addgroup user42" --> "user42" adında yeni grup oluşturduk.
22-) "sudo adduser kullanıcı_adı user42" --> user42 grubuna belirlenen kullanıcıyı ekledik.
23-) "sudo passwd kullanıcı_adı" --> şifrelerimizi değişelim.

24-) "sudo crontab -u root -e" --> Crontab zamanlı dosya çalıştırma sistemi olarak düşünüle bilir. En alt satır "*/10 * * * * sh /path/to/script" değiştir ya da ekle.

BONUS

sudo apt install lighttpd --> NGNIX ve APACHE Harici alternatif bir http sunucusudur.
sudo ufw allow 80 --> 80 Portu HTTP için standart port

sudo apt install mariadb-server --> Mariadb ilişkisel veritabanıdır. MySQL Databse Serverin kaynak kodundan türemiştir.
sudo mysql_secure_installation --> Varsayılan ve güvenli olmayan ayarları yapılandırmak için...
@-)İlk adımda parola soracak enter diyip atlanır.
@-)Switch to unix seçeneği = N
@-)Change the root password = N
@-)Remove anonymous users = Y
@-)Disallow root login remotely = Y
@-)Remove test database and access to it = Y
@-)Reload privilege tables now = Y
sudo mariadb --> Mariadb consoluna girilir.
CREATE DATABASE veritabanı_adı --> Yeni bir veritabanı oluşturulur.
GRANT ALL ON bonus.* TO 'bonuskullanıcı'@'localhost' IDENTIFIED BY 'bonussifre' WITH GRANT OPTION; --> Burda şunları yaptık...
-Yeni bir veri tabanı kullanıcısı oluşturduk ve bunun local ağda olacağını belirttik ve bu kullanımıza bir şifre belirledik NOT: Burada ki kullanıcıyı sistemde ki kullanıcılar ile karıştırmayın!
-Yeni oluşturduğumuz bu kullanıcıya yeni oluşturduğumuz veri tabanına tam yetki vermesini söylüyoruz.
FLUSH PRIVILEGES; --> Diyerek yaptığımız değişiklikleri okutuyoruz
exit --> Mariadb konsolundan çıkış yapıyoruz.
mariadb -u veritabanı_kullanıcısı -p --> Veri tabanına oluşturduğumuz kullanıcıdan bağlanmayı deneyeceğiz. Bunu yazdıktan sonra bizden şifre isteyecek şifre belirlemiş olduğumuz veritabanı_kullanıcısı_şifresi.
SHOW DATABASES; --> Veri tabanlarını gösteriri. 2 Tane veri tabanı olması lazım biri oluşturduğumuz diğeri information_schema
exit --> Çıkış!
sudo apt install php-cgi php-mysql --> ORTAK AĞ GEÇİDİ ARAYÜZÜ -
-php-cgi (common gateway interface) web serverinin apache, lighttpd vs. harici uygulamalarla PHP, PYTHON, PERL dillerinin yorumlayıcılarıyla iletişim kurmasını sağlayan web teknolojisidir.
-Wordpress PHP ile geliştirildiği için bu eklentileri kuruyoruz.
sudo apt install wget --> Bu eklentiyi HTTP HTTPS VE FTP üzerinden indirme işlemleri almak için kuruyoruz. Bir sonraki işlemde Wordpress'in kendi sitesinden verileri çekeceğiz.
sudo wget http://wordpress.org/latest.tar.gz -P /var/www/html --> Belirtilen dizine (yoksa kendisi oluşturur) wordpress'in en son çıkan panelini indiriyoruz.
sudo tar -xzvf /var/www/html/latest.tar.gz --> İndirdiğimiz tar.gz uzantılı klasörden dosyaları çıkarıyoruz.

sudo rm /var/www/html/latest.tar.gz --> Dosyaları çıkardıktan sonra indirdiğimiz arşiv dosyasını siliyoruz.
sudo cp -r /var/www/html/wordpress/* /var/www/html --> Arşivden çıkan klasörün ismi wordpress bu komut ile wordpress dosyasının içinde ki tüm verileri belirtilen yola kopyalıyoruz.
sudo rm -rf /var/www/html/wordpress
--> Klasörün içeriğini kopyaladıktan sonra klasörle de işimiz kalmadığı için klasörü de siliyoruz.
sudo cp /var/www/html/wp-config-sample.php /var/www/html/wp-config.php --> wp-config-sample dosyasını wp-config adıyla kopyalıyoruz.
sudo nano /var/www/html/wp-config.php
--> kopyalanan dosyayı nano ile açıyoruz ve şu kısımları editliyoruz...
-define('DB_NAME', 'veritabanı_adı');^M
-define('DB_USER', 'veritabanı_kullanıcısı');^M
-define('DB_PASSWORD', 'veritabanı_kullanıcısı_şifresi');^M
@@sudo lighty-enable-mod fastcgi---------
@@sudo lighty-enable-mod fastcgi-php------> fastcgi modülünü ve php yapılandırmasını şu şekilde etkinleştirin
@sudo service lighttpd force-reload------> lighttpd sunucusunu yeniden başlatıyoruz
sudo apt install vsftpd --> FTP sunucusu için gerekli paket.
sudo ufw allow 21 --> Güvenlik duvarından 21 portu için TCP UDP iznini veriyoruz.
sudo nano /etc/vsftpd.conf -->
@-)#write_enable=YES // # kaldırılır ftp komutlarını yazmak için
@-)user_sub_token=$USER
@-)local_root=/home/$USER/ftp
@-)userlist_enable=YES
@-)userlist_file=/etc/vsftpd.userlist
@-)userlist_deny=NO
sudo mkdir /home/kullanıcı_adı/ftp --> Kullanıcımızın bulunduğu dizine ftp klasörü açtık.
sudo mkdir /home/kullanıcı_adı/ftp/files -->Files klasörü açtık
sudo chown nobody:nogroup /home/<username>/ftp --> ftp klasörünün sahipliğini ve grubunu hiç kimse olarak ayarlıyoruz.
sudo chmod a-w /home/<username>/ftp --> Tüm kullanıcılara yazma iznini verdik.
sudo nano /etc/vsftpd.userlist --> CTRL + O Yapıp sonrasında CTRL + X Yapıp kaydedip çıkıyoruz. Amacı izin verilen kullanıcılar ile bağlanmamızı sağlar.
echo kullanıcı_adı | sudo tee -a /etc/vsftpd.userlist --> İle kullanıcımızı userlist içerisine ekliyoruz.