

Oracle VM VirtualBox Üzerine Debian Linux Kurulumu

İşletim sistemini zaten kurduysanız, bu kısmı atlayabilirsiniz.

- Debian yükleyici görüntüsünü buradan alın:

<https://cdimage.debian.org/debian-cd/current/amd64/iso-cd/>

ISO **netinst** CD, yalnızca temel Debian yükleyici kodunu ve küçük bir çekirdek metin modu programları (Debian'da "standart" olarak bilinir) içeren küçük bir CD görüntüsüdür.

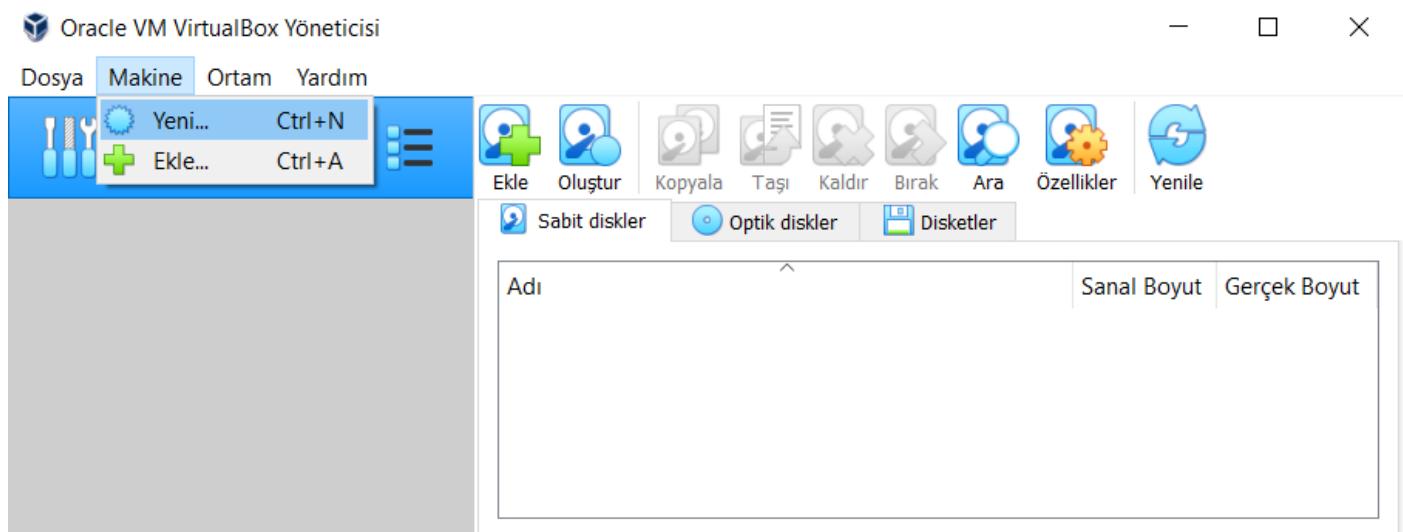
Ben standart **debian-11.2.0-amd64-netinst.iso** seçtim.

| Name | Last modified | Size |
|---|------------------|------|
| Parent Directory | - | |
| SHA256SUMS | 2021-12-18 21:42 | 302 |
| SHA256SUMS.sign | 2021-12-18 21:45 | 833 |
| SHA512SUMS | 2021-12-18 21:42 | 494 |
| SHA512SUMS.sign | 2021-12-18 21:45 | 833 |
| debian-11.2.0-amd64-netinst.iso | 2021-12-18 14:24 | 378M |
| debian-edu-11.2.0-amd64-netinst.iso | 2021-12-18 14:24 | 439M |
| debian-mac-11.2.0-amd64-netinst.iso | 2021-12-18 14:24 | 377M |

Not: Okulda kurulum yapıyorsanız, resmi sgoinfre ([/sgoinfre/goinfre/Perso/your_login](http://sgoinfre/goinfre/Perso/your_login)) içine koyun.

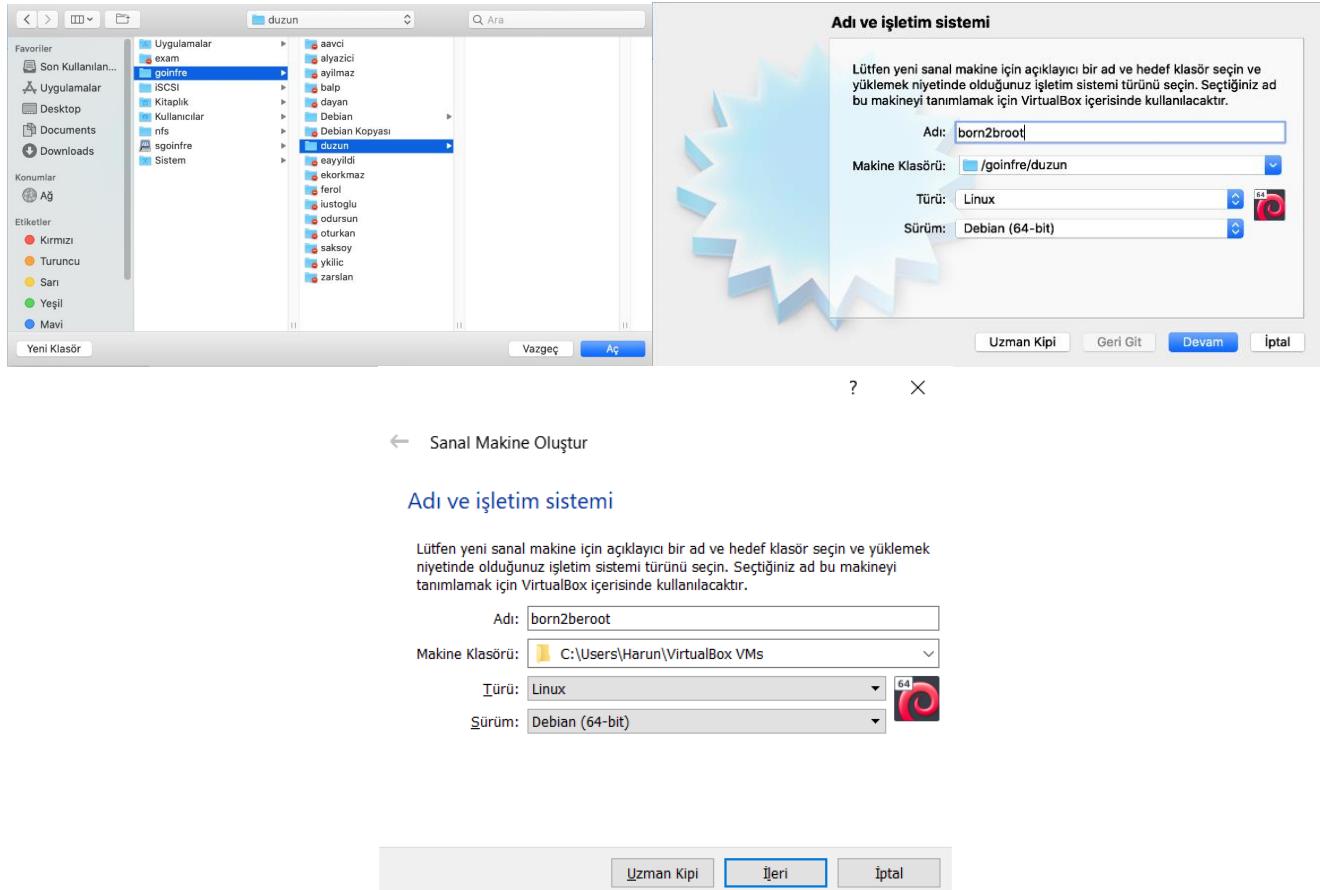
2. Debian 11'i Kurulumu.

“Yeni...”’ı seçin (Ctrl+N)

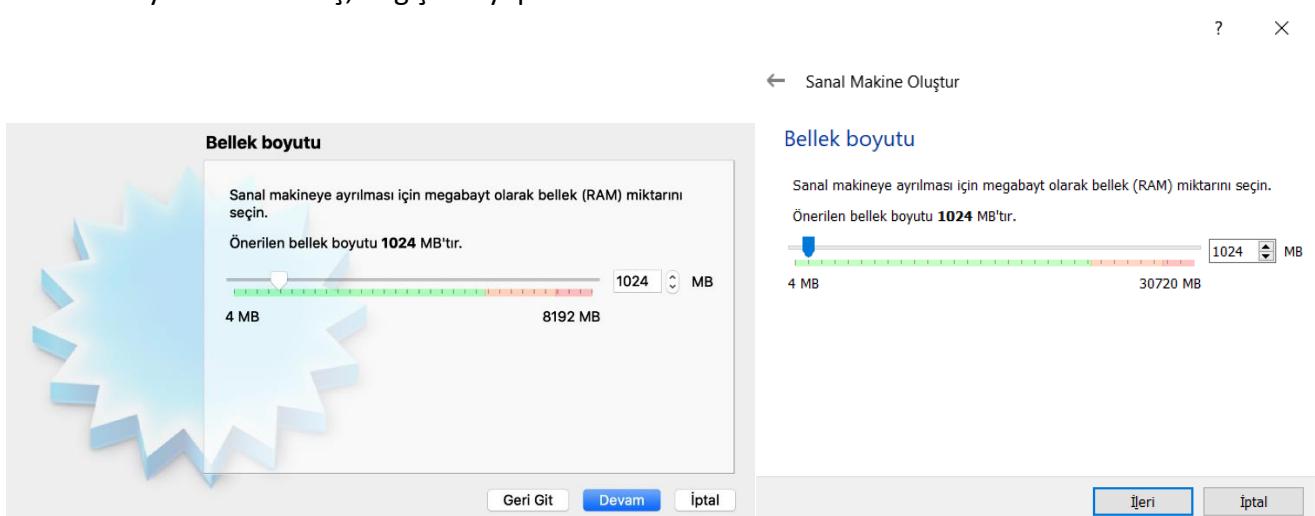


Okul ortamında kurulum yapıyorsanız, Makine Klasörü olarak “**goinfre**” klasörünü seçin. (örneğin: goinfre/duzun) iki farklı resimde durum gösterilmiştir.

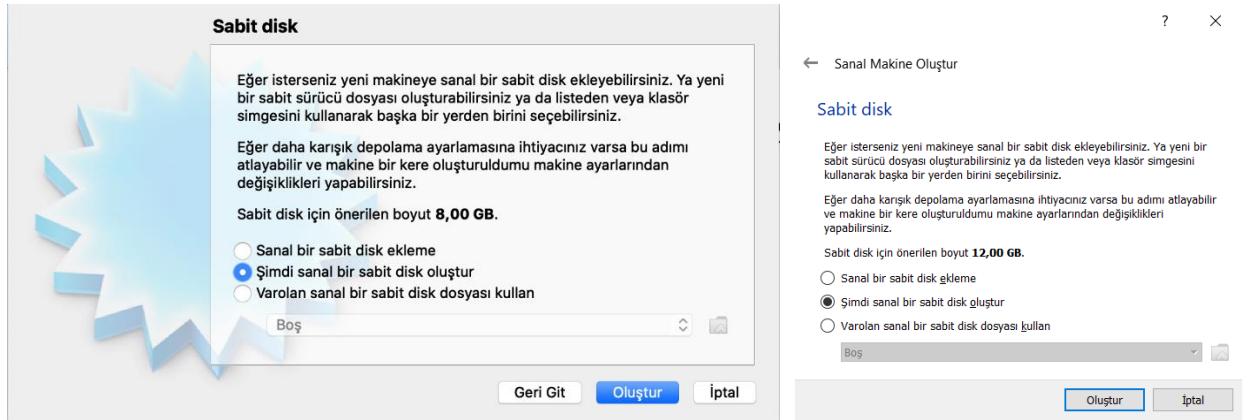
Türü: Linux ve **Sürüm:** Debian (64-bit) seçiyoruz.



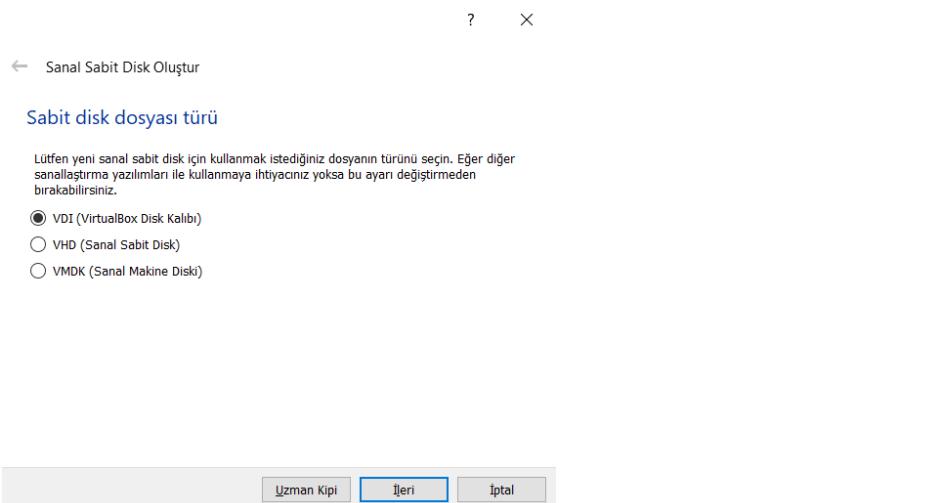
Tümünü varsayılan olarak seç, değişiklik yapmadan “ileri” devam edin.



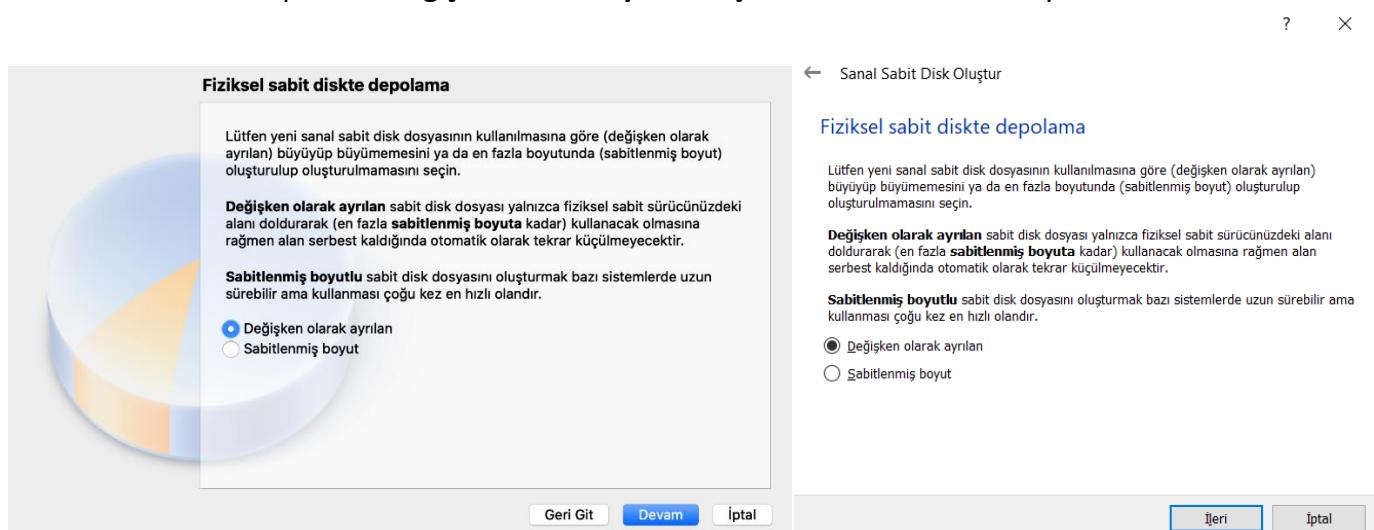
Bu aşamada herhangi bir değişiklik yapmıyoruz. “Şimdi sanal bir sabit disk oluştur” seçimi ile “Oluştur” butonuna basınız.



Sabit disk dosyası türü olarak **VDI (VirtualBox Disk Kalıbı)** seçimi ile “İleri” devam edin.

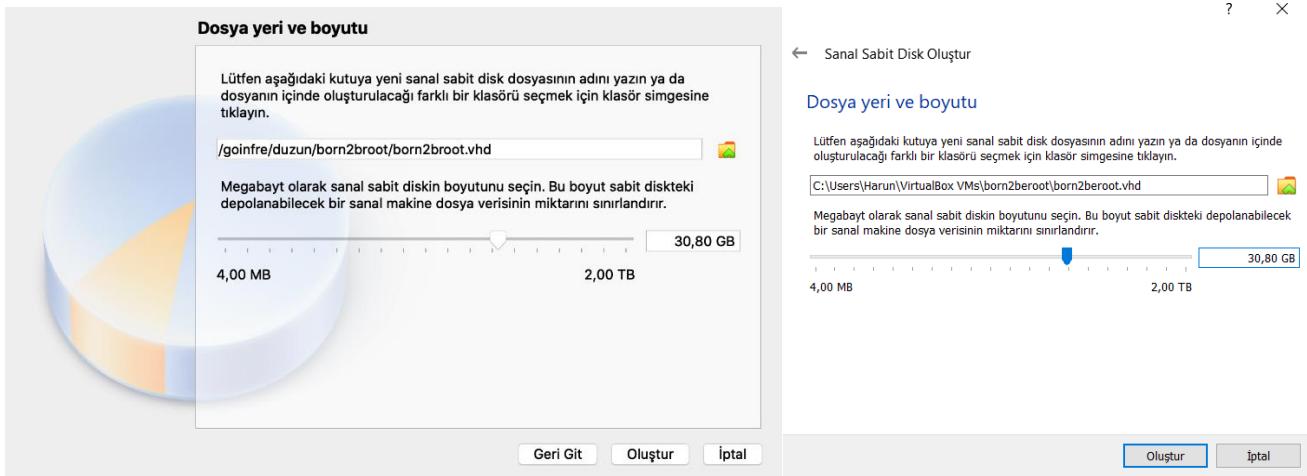


Fiziksel sabit diskte depolama **Değişken olarak ayrılan** seçim ile “İleri” devam ediyoruz.



Dosya yeri ve boyu özeti gösteren ekranda Gigabayt olarak diskimizin toplam boyutunu **30,80 GB** olarak değiştiriyor ve “Oluştur” butonuna basıyoruz. Burada **VDI** dosyanın oluşacağı klasör yeri 42 Okuldaki bilgisayarda farklı gözükecektir.

(Örneğin: goinfre/**duzun**/born2beroot/born2beroot.vdi şeklinde burada ki **duzun** kullanıcı adıdır.)



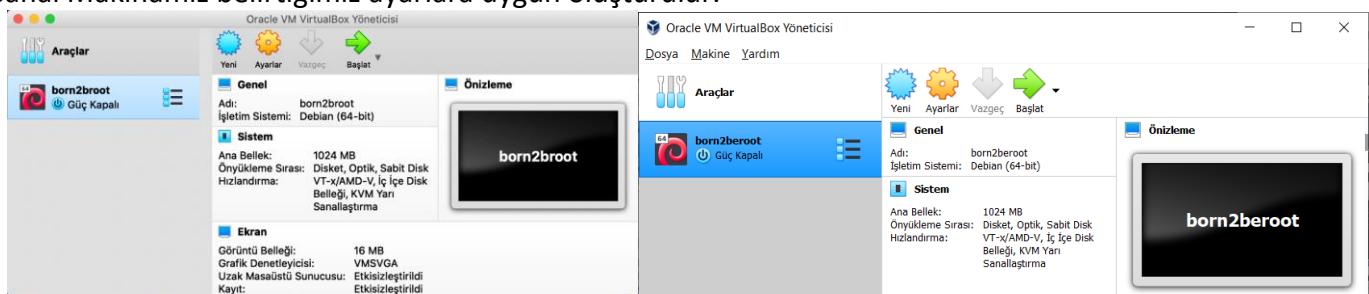
Neden 30,8 GB bir sanal disk alanı belirledik. PDF yönergesinde BONUS kısmındaki aşağıdaki tabloda yer alan disk alanlarının toplamına eşit olacak şekilde bir alan tahsis ediyoruz.

| NAME | MAJ:MIN | RM | SIZE | RO | TYPE | MOUNTPOINT |
|---------------------|---------|----|-------|----|-------|------------|
| sda | 8:0 | 0 | 30.8G | 0 | disk | |
| └─sda1 | 8:1 | 0 | 500M | 0 | part | /boot |
| └─sda2 | 8:2 | 0 | 1K | 0 | part | |
| └─sda5 | 8:5 | 0 | 30.3G | 0 | part | |
| └─sda5_crypt | 254:0 | 0 | 30.3G | 0 | crypt | |
| └─LVMGroup-root | 254:1 | 0 | 10G | 0 | lvm | / |
| └─LVMGroup-swap | 254:2 | 0 | 2.3G | 0 | lvm | [SWAP] |
| └─LVMGroup-home | 254:3 | 0 | 5G | 0 | lvm | /home |
| └─LVMGroup-var | 254:4 | 0 | 3G | 0 | lvm | /var |
| └─LVMGroup-srv | 254:5 | 0 | 3G | 0 | lvm | /srv |
| └─LVMGroup-tmp | 254:6 | 0 | 3G | 0 | lvm | /tmp |
| └─LVMGroup-var--log | 254:7 | 0 | 4G | 0 | lvm | /var/log |
| sr0 | 11:0 | 1 | 1024M | 0 | rom | |

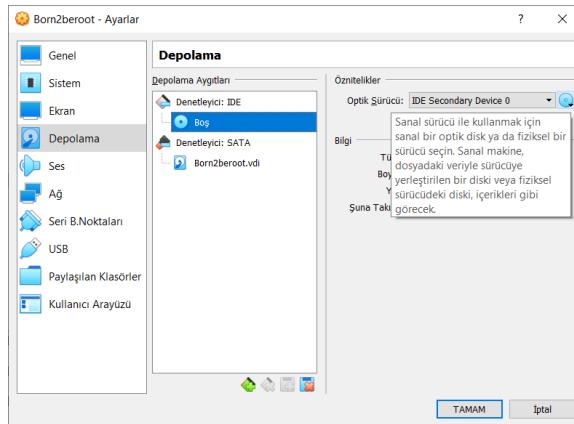
Bilgi Notu: Linux'ta sabit disklerin isimlendirilme mantığı şu şekildedir;

- Sabit diskler eğer IDE kanalından bağlı ise "hda", "hdb" SCSI ve/veya SATA kanalına takılı ise "sda", "sdb" gibi isimler alırlar. Örneğin birinci sabit disk "hda" olarak adlandırılır. İkinci sabit disk "hdb" şeklinde adlandırılır. Buradan da anlaşılacağı üzere a'dan z'ye kadar -eğer mümkünse- 26 adet sabit disk olabilir. Diskleriniz günümüzün popüler HDD kanalı olan SATA'dan bağlı ise bu durumda örneğimizdeki isimler "sda" ve "sdb" olacaktır.
- Sabit disklerin bölümleri ise 1'den 63'e kadar numara ile temsil edilirler. Örneğin birinci sabit diskin birinci bölümü "hda1" veya "sda1" olarak adlandırılır.
- Sistemdeki CD/DVD sürücüler ise "sr0", "sr1" gibi isimler alırlar. Linux'ta her bir sabit disk 63 parçaya bölünebilir.

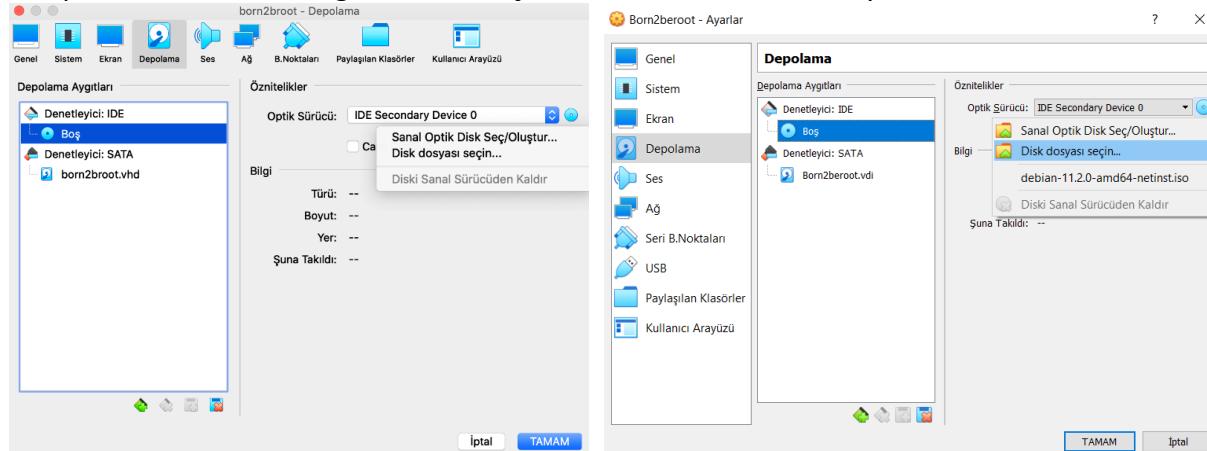
Sanal Makinamız belirttiğimiz ayarlara uygun oluşturulur.



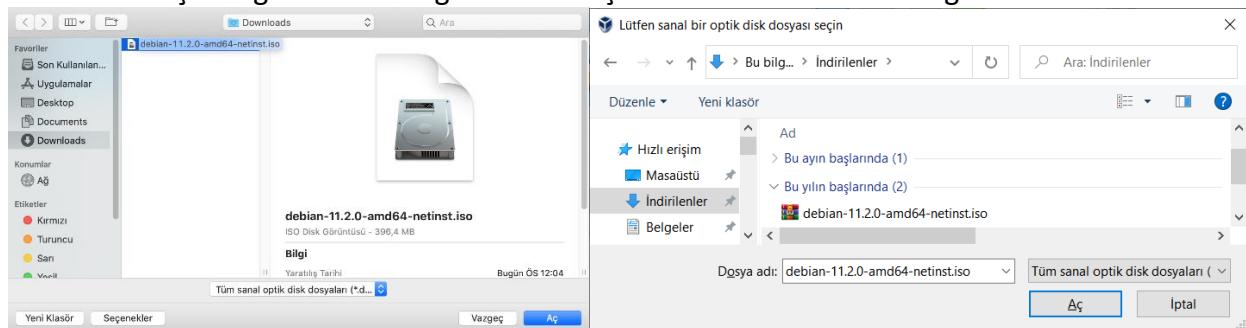
"Ayarlar" -> "Depolama" yi seçin.



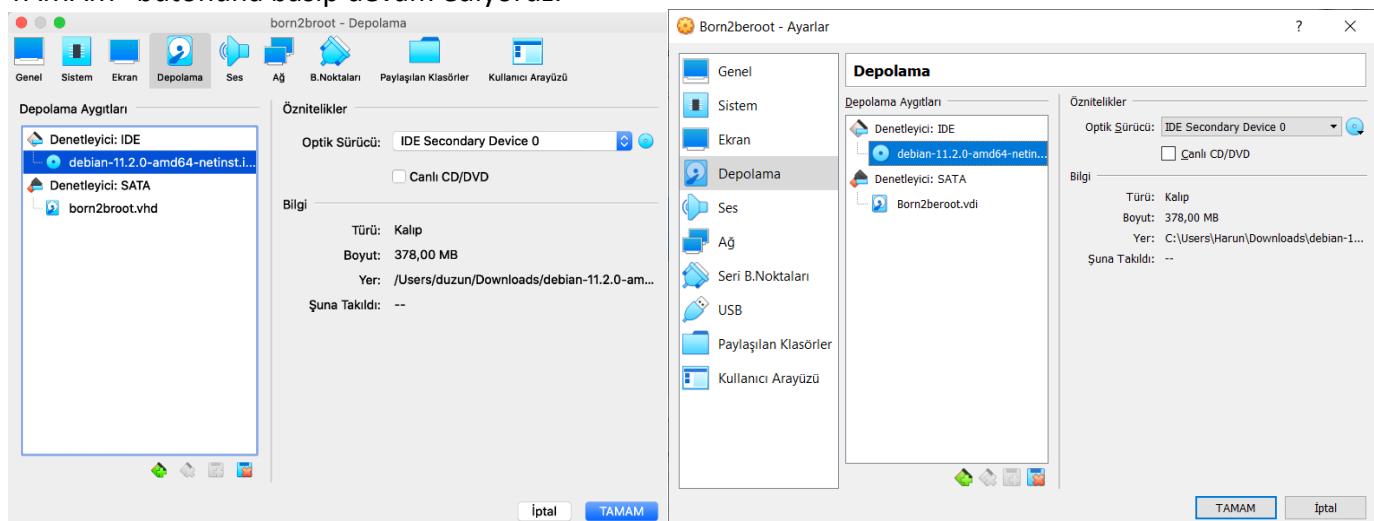
Ardından Optik Sürücünün en sağdaki mavi küçük cd ikonu kutusuna tıklayın.



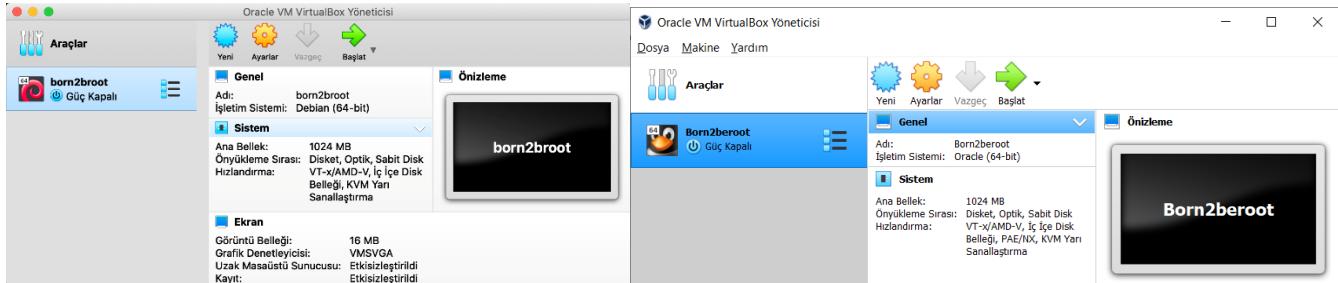
Hâlihazırda indirmiş olduğunuz bir ISO görüntüsü seçin. Mac ortamında ekran görüntüsü farklı olacaktır.



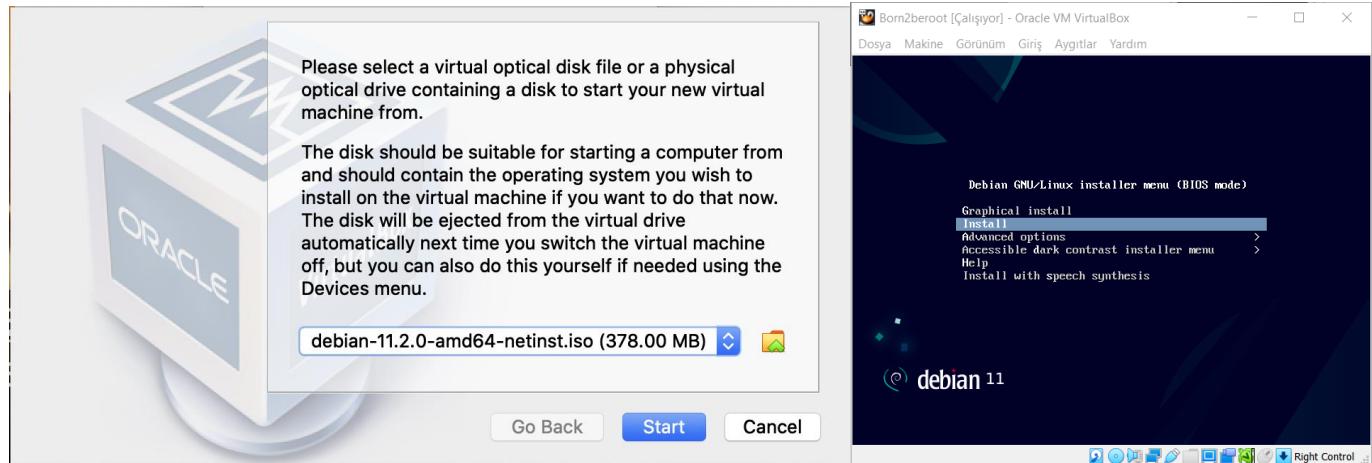
"TAMAM" butonuna basıp devam ediyoruz.



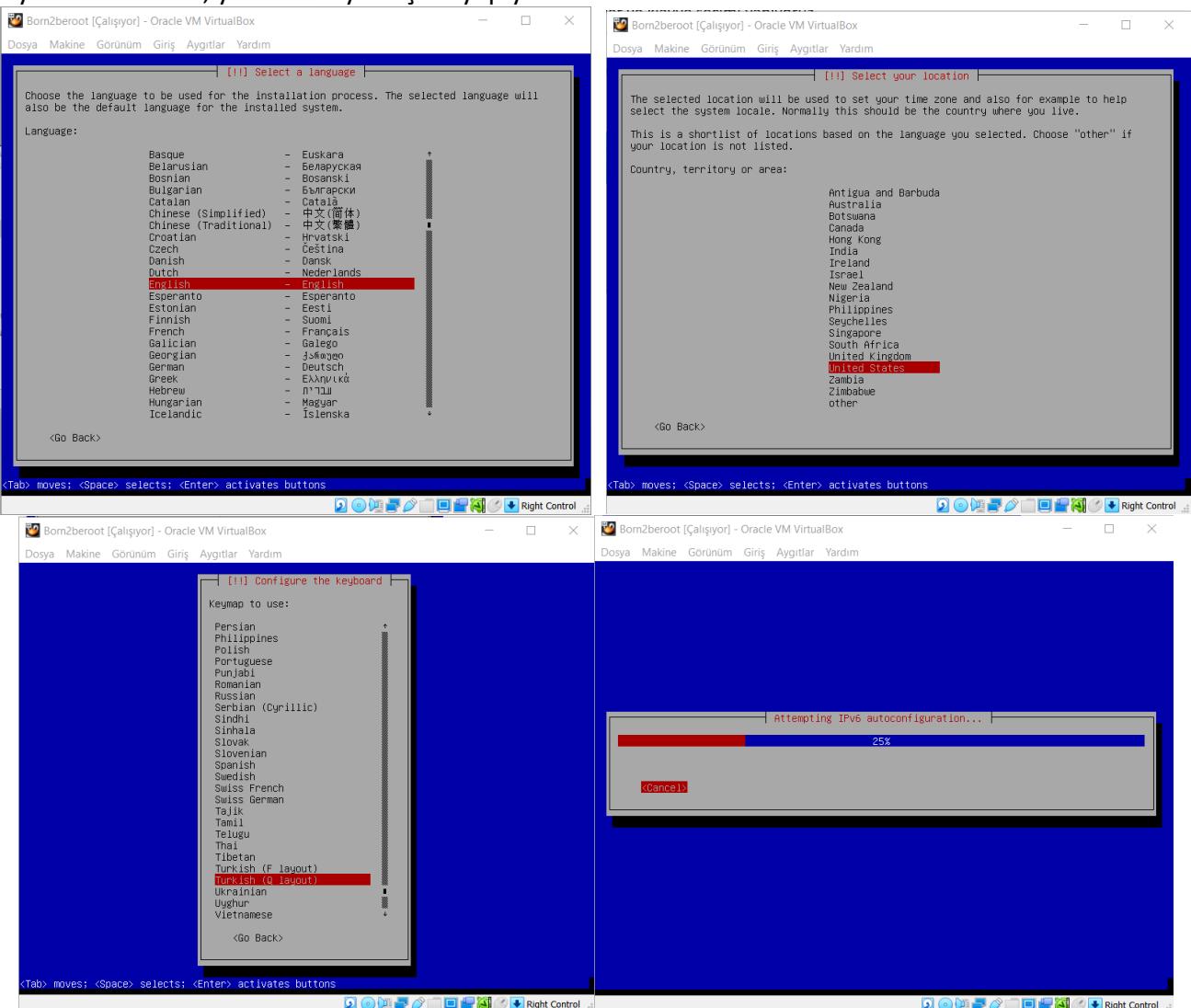
VM'yi başlatmak için "Başlat" yeşil oka tıklayın



"Install" (Yükle) yi seçin.



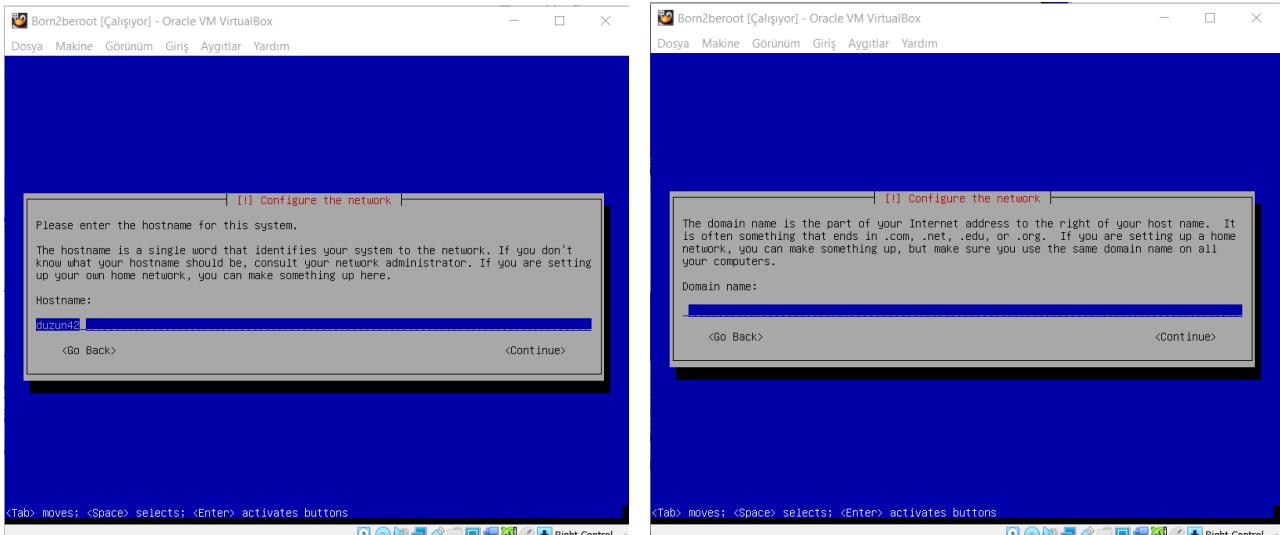
Sırasıyla Yükleme dili, yer ve klavye seçimi yapıyoruz.



Ön yükleme işlemleri gerçekleşir.

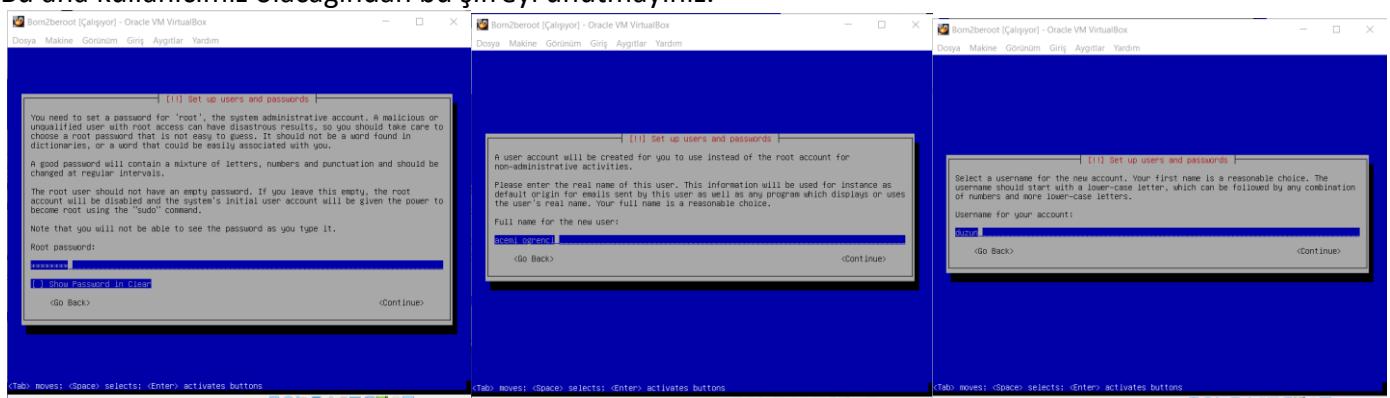
Oturum açma bilgileriniz olarak ana bilgisayar adınızı (Hostname) sonuna 42 yazarak giriniz.

Örnek: duzun kullanıcı adı için duzun42 şeklinde.



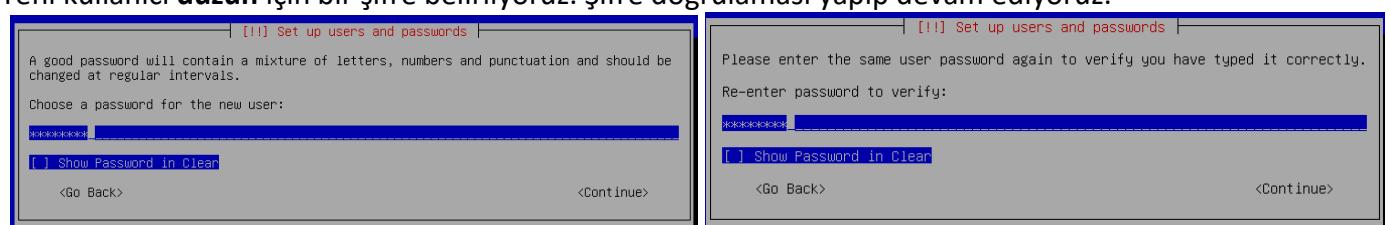
Damain Name alanını varsayılan (boş) olarak bırakıp devam ediyoruz.

root kullanıcıı için bir şifre belirliyoruz. Ardından aynı şifreyi doğrulamak için tekrar girmemizi isteyecek. Bu ana kullanıcımız olacağından bu şifreyi unutmayınız.



Yeni bir kullanıcı için kullanıcının tam adını yazıyoruz. Akabinde kullanıcı adını **duzun** yazıyoruz.

Yeni kullanıcı **duzun** için bir şifre belirliyoruz. Şifre doğrulaması yapıp devam ediyoruz.



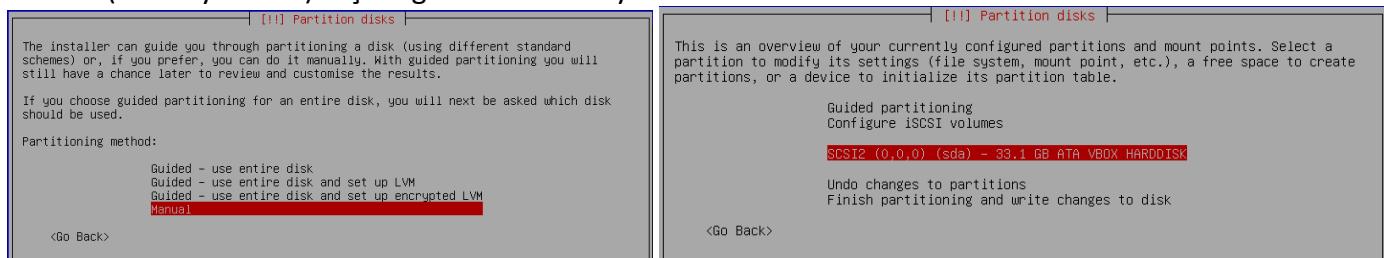
Zaman dilimi için seçim yapıyoruz.



Sanal makinamızda **Linux** işletim sistemimizin disk yapısını **born2beroot** pdf talimatlarına uygun olarak oluşturacağız. Ben burada disk yapısını oluştururken talimatlarda yer alan **BONUS** kısmındaki yapıya uygun olacak ve talimatlarda yer alan aşağıdaki resme uygun bir disk yapısı oluşturacağım.

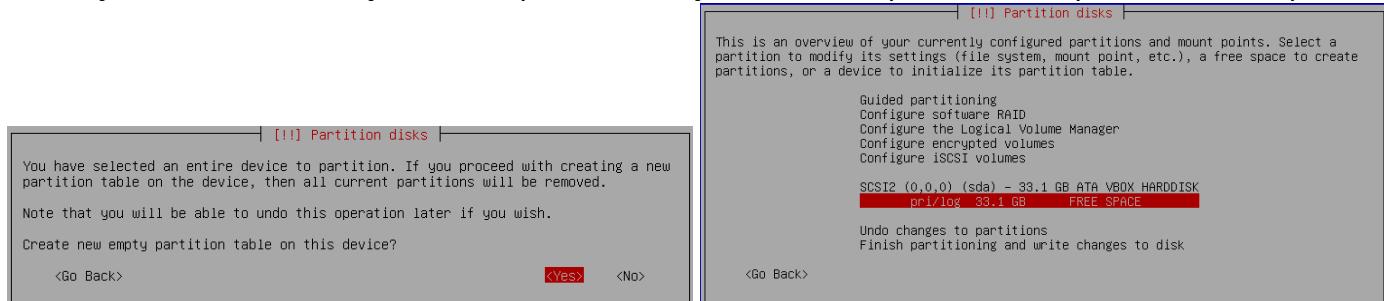
```
# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                  8:0    0 30.8G  0 disk
└─sda1               8:1    0 500M  0 part /boot
└─sda2               8:2    0   1K  0 part
└─sda5               8:5    0 30.3G  0 part
  └─sda5_crypt      254:0   0 30.3G  0 crypt
    ├─LVMGroup-root  254:1   0 10G  0 lvm   /
    ├─LVMGroup-swap  254:2   0 2.3G  0 lvm   [SWAP]
    ├─LVMGroup-home  254:3   0 5G   0 lvm   /home
    ├─LVMGroup-var   254:4   0 3G   0 lvm   /var
    ├─LVMGroup-srv   254:5   0 3G   0 lvm   /srv
    ├─LVMGroup-tmp   254:6   0 3G   0 lvm   /tmp
    └─LVMGroup-var--log 254:7   0 4G   0 lvm   /var/log
sr0                 11:0   1 1024M 0 rom
```

Disk yapılandırma/bölme aşamasındaki işlemleri yukarıdaki tabloya uygun olacak şekilde ayarlamak için Manual (el ile ayarlama) seçeneği ile devam ediyoruz.



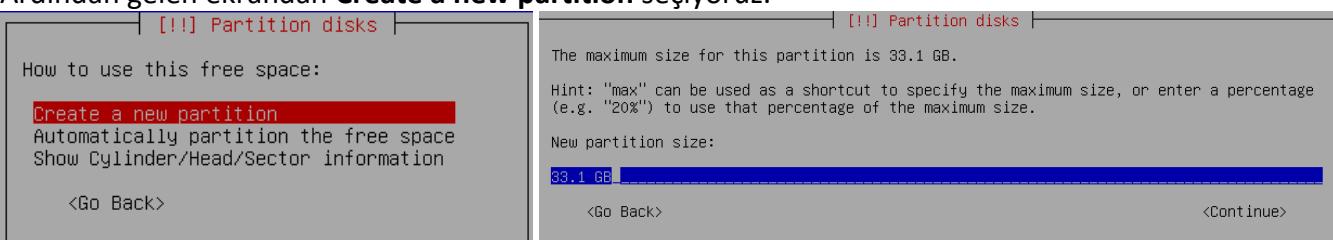
Üzerinde işlem yapacağımız ve daha önce oluşturduğumuz diski seçiyoruz.

Yeni boş bir disk bölümü oluşturmak istiyor musunuz şeklindeki soruya **Yes/Evet** dierek devam ediyoruz.

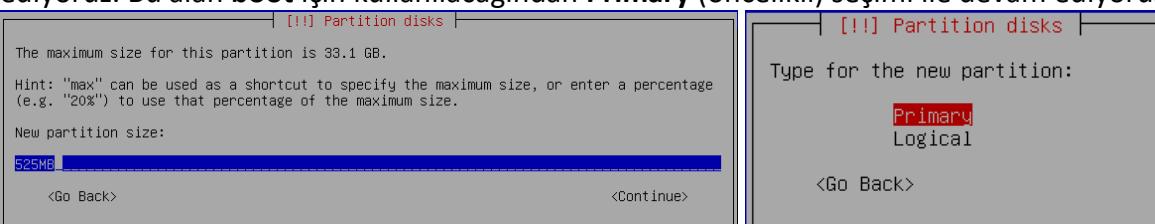


Oluşturulan boş alanı seçiyoruz.

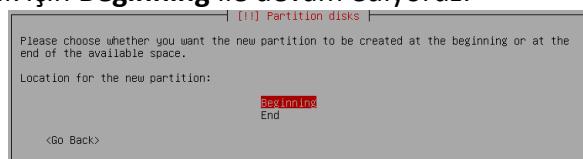
Ardından gelen ekranın **Create a new partition** seçiyoruz.



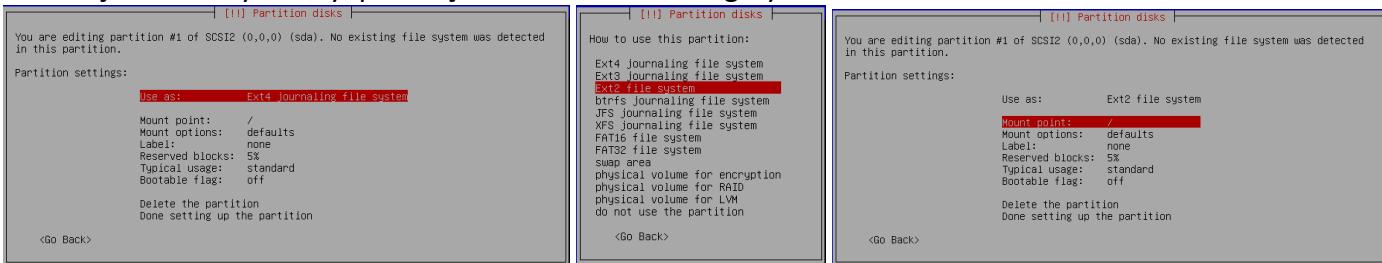
En fazla oluşturabileceğimiz disk alanı için toplam disk alanımız olan **33.1 GB** önermektedir. Biz bu alandaki değeri silip projemize uygun şekilde disk bölmeleri oluşturacağız. İlk olarak alanı silip **525MB** yazıp devam ediyoruz. Bu alan **boot** için kullanılacağından **Primary** (öncelikli) seçimi ile devam ediyoruz.



Yeni bir bölüm daha olusturmak için **Beginning** ile devam ediyoruz.

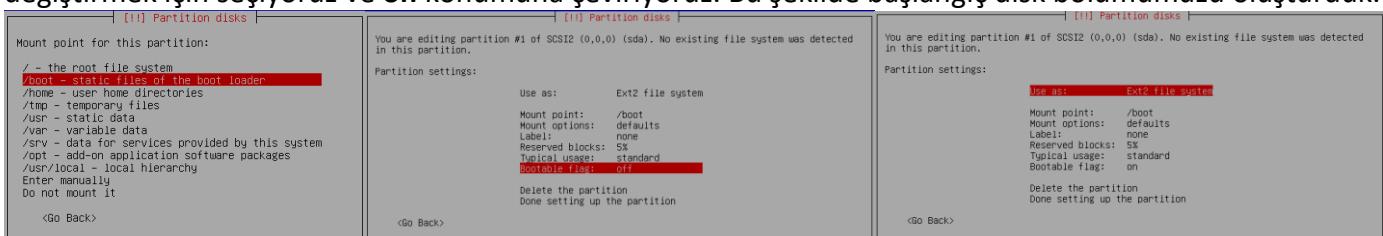


Bölüm için temel ayarları yapmak için Use as bölümüne giriyoruz.



Bu bölümü nasıl kullanacağımızı seçimini yapıyoruz. Bölüm için **Ext2 file system** seçiyoruz. Bağlama noktasını belirlemek için **Mount point** seçerek devam ediyoruz.

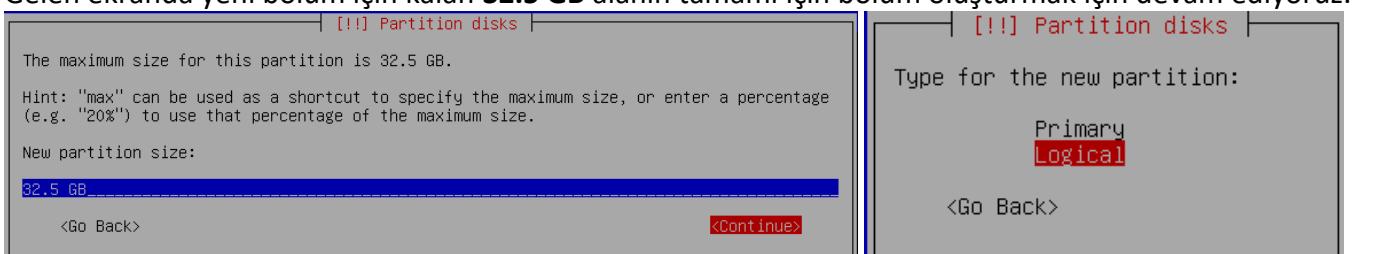
Bu alanı **boot** olarak kullanacağımızdan **/boot** seçerek devam ediyoruz. Yine aynı ekrana döndüğümüzde bu bölümün **boot** yani başlangıç bölümü olduğunu ayarlamak için **Bootable flag** işaretini **off** konumunda ayarını değiştirmek için seçiyoruz ve **on** konumuna çeviriyoruz. Bu şekilde başlangıç disk bölümümüzü oluşturduk.



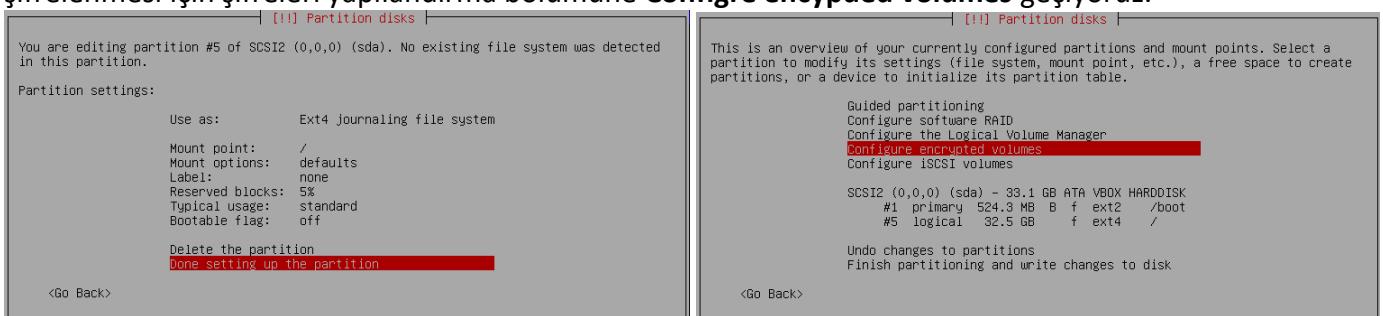
Yeni bir bölüm oluşturmak için **Done setting up the partition** ile dönüşüyoruz. Geriye kalan **32.5 GB** boş alanı seçip devam ediyoruz. Yeni bölüm oluşturma **Create a new partition** seçenekleri ile devam ediyoruz.



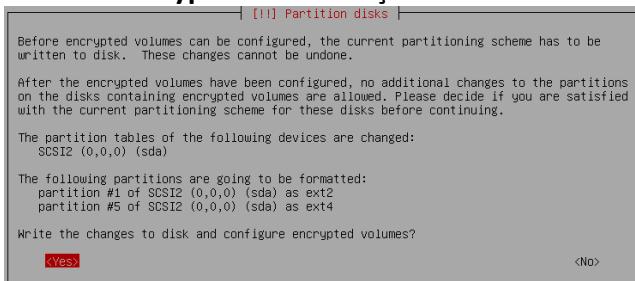
Gelen ekranın yeni bölüm için kalan **32.5 GB** alanın tamamı için bölüm oluşturmak için devam ediyoruz.



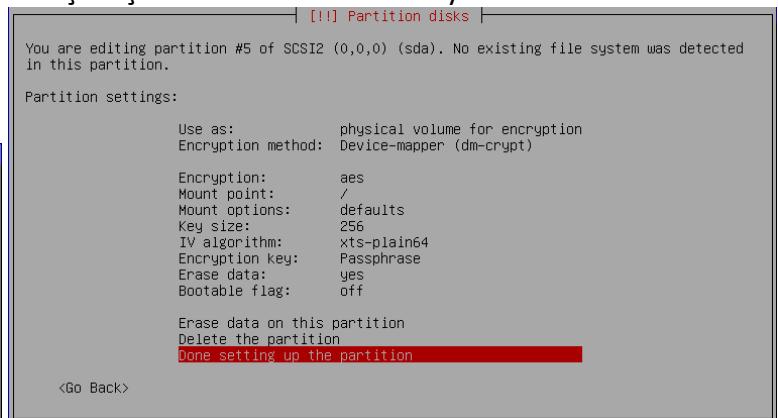
Primary (öncelikli) bölümü daha önce oluşturmuştu bu bölüm için **Logical** (mantıksal) türde ayarlama yapıyoruz. Ardından bölüm oluşturma bölümüne dönüyoruz. Pdf talimatına uygun olarak ayrılmış bölgenin şifrelenmesi için şifreleri yapılandırma bölümüne **Configure encrypted volumes** geçiyoruz.



Bölümlerin şifrelemesini başlatmak için **Yes** ile devam ediyoruz. Gerekli ayarlama yapılır. Gelen ekranda **Cereate encrypted volumes** şifreleme birimleri oluşturma seçeneği ile devam ediyoruz.



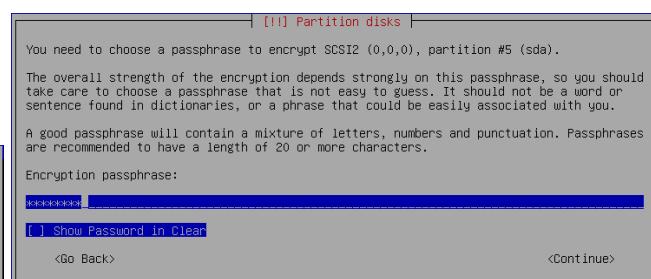
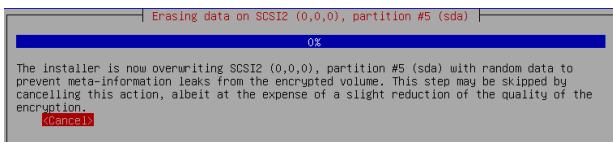
Yeni gelen ekranı şifreleyeceğim birimi seçiyorum. Seçimi ikinci sıradaki birim olan **/dav/sda5** için yapıyorum. Seçim yaparken klavyeden yön tuşları ile üzerine gelip space (boşluk) tuşuna bir kere basıp seçim alanında * işaretini eklendiğinde o bölüm seçilmiş olur. Ardından bölüm ayarları bölümüne döneriz.



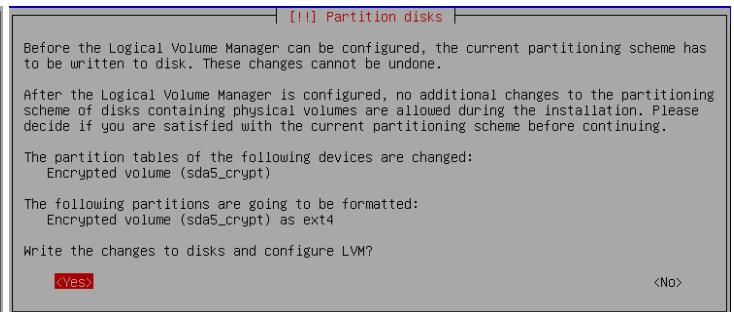
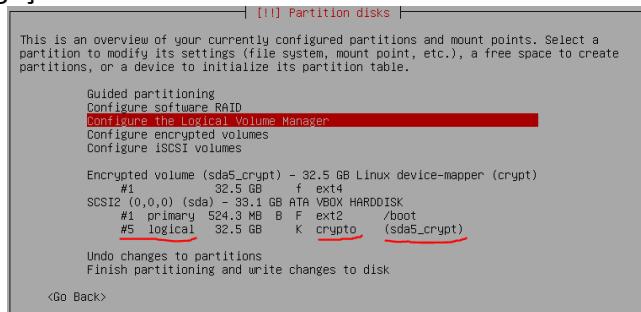
Şifreleme için başka bir bölüm daha ekleme yapmayacağımdan **Finish** ile çıkışıyoruz. Şifrelemeli bölümün oluşturulacak **sda5** deki tüm mevcut bilgilerin silinerek kaybolacağı uyarısına **Yes** diyerek işleme devam ediyoruz.



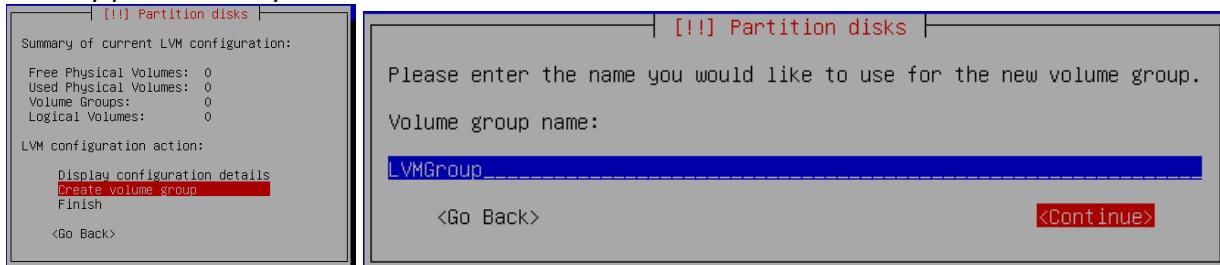
Silme ve rasgele veri yazma işlemi başlar biraz bekleyelim. Ardından güçlü bir şifre grip işleme devam ediyoruz. Şifre tekrar yazıp devam ediyoruz.



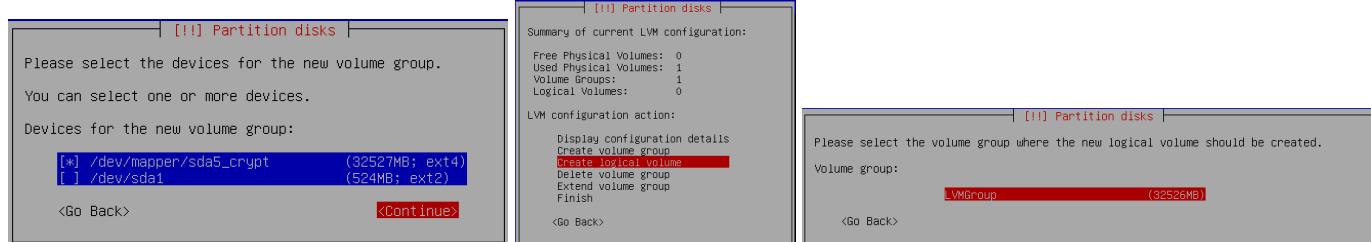
Aşağıdaki resimde de görüldüğü üzere **sda5** disk bölgesi şifrelenmiş şekilde çalışmaya ayarlandı. Şimdi Mantıksal sürücümüzün ayarları yapmaya devam edelim. **Configure the Logical Volume Manager** kısmına geçelim.



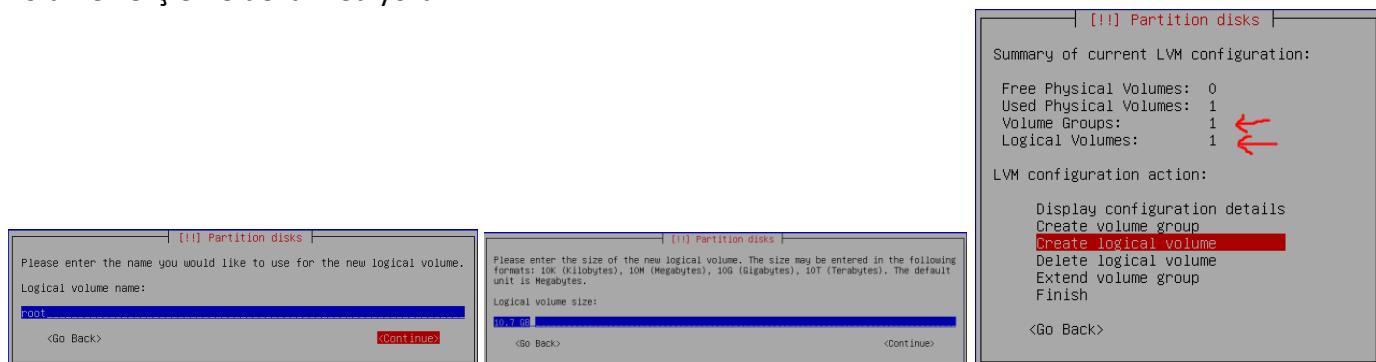
Önce yeni bir grup oluşturmak için **Create volume group** seçeneği ile devam edin. Grup adını **LVMGroup** olarak belirleyip devam ediyoruz.



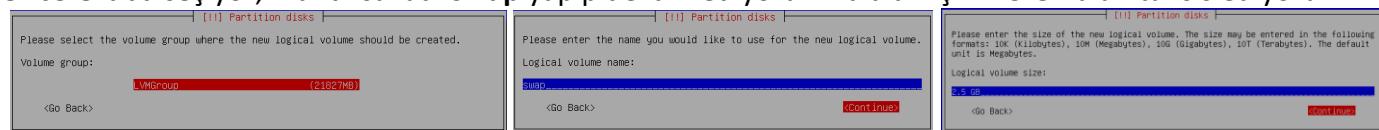
Ardından bu grubun hangi bölümde olacağını seçiyoruz. Oluşturduğumuz şifrelenmiş **sda5** seçimi ile devam ediyorum. Bu gruba bağlı yeni bir mantıksal seviye eklemek için **Create logical volume** seçeneği ile devam edin. **LVMGroup** seçimi ile devam edin.



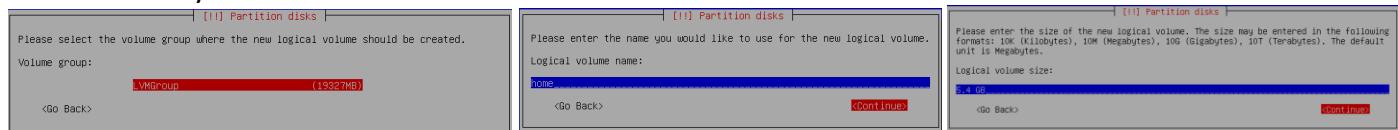
Bölümün adını PDF talimatını uygun olarak ismini **root** olarak belirtiyoruz. Bu alan için **10.7 GB** alan tahsis ediyoruz. Şu ana kadar 1 grup ve 1 mantıksal bölüm oluşturduk. Yeni bir mantıksal bölüm için Create logical volume ile işleme devam ediyoruz.



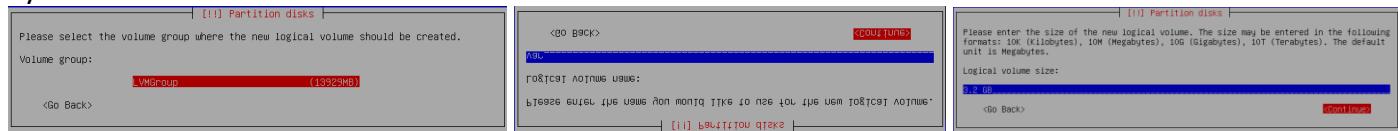
Önce Grubu seçiyor, mantıksal adı **swap** yapıp devam ediyoruz. Bu alan için 2.5 GB alan tahsis ediyoruz.



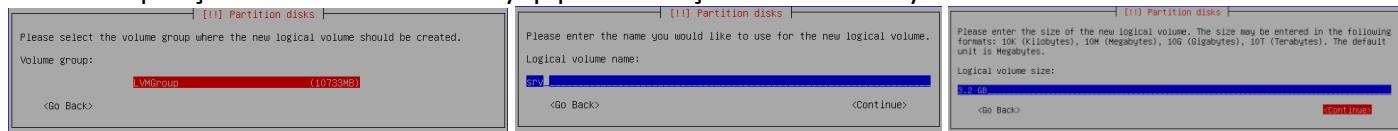
Yeni bölüm oluşturmaya tekrar grip grubu seçikten sonra mantıksal adı **home** yapıp bu alan için **5.4 GB** alan tahsis ediyoruz.



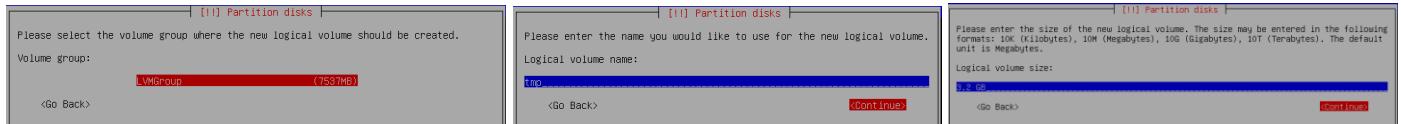
Yeni bir bölüm daha ekliyoruz. Grup seçiminden sonra mantıksal adı **var** yapıp bu alan için **3.2 GB** alan ayıralım.



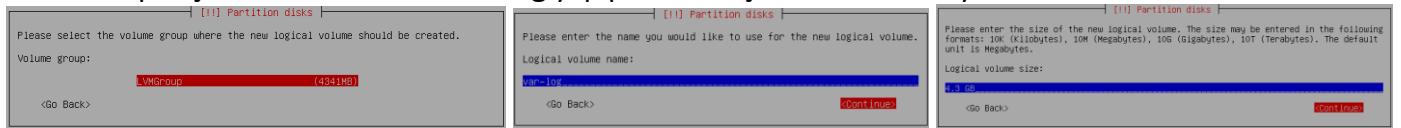
Ekle > Grup seçimi > mantıksal adı **srv** yapıp bu alan için **3.2 GB** alan ayıralım.



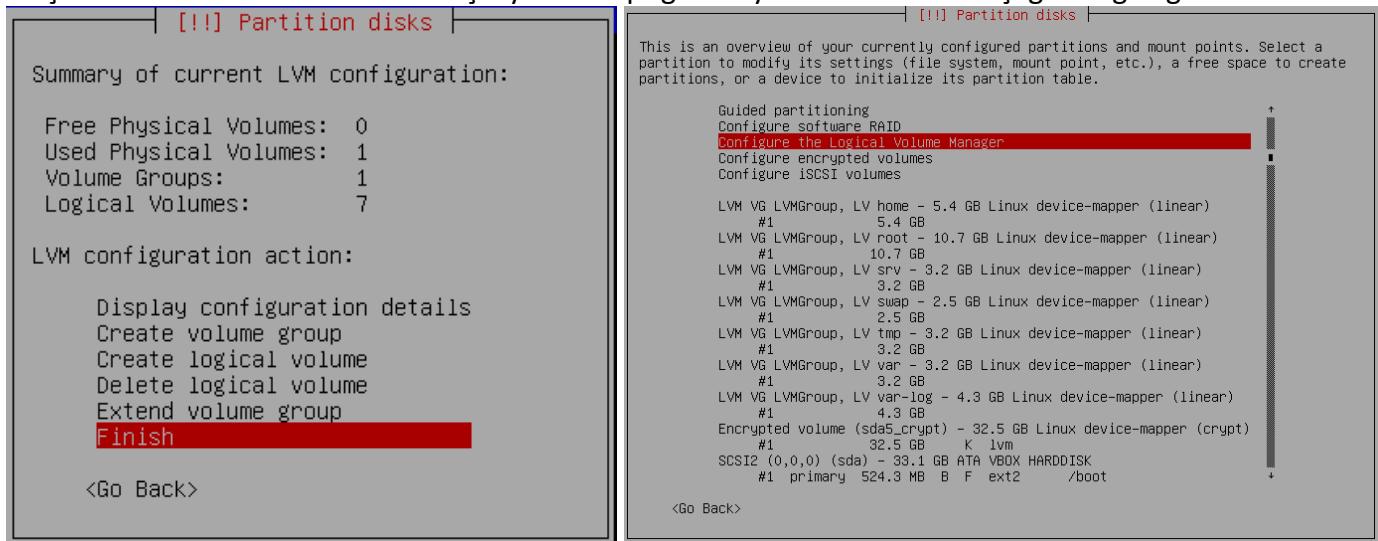
Ekle > Grup seçimi > mantıksal adı **tmp** yapıp bu alan için **3.2 GB** alan ayıralım.



Ekle > Grup seçimi > mantıksal adı **var-log** yapıp bu alan için **4.3 GB** alan ayıralım.



Tüm **LVMGroup** bağlı alanları oluşturduk. Toplamda **1 fiziksel sürücü 1 grup ve 7 de mantıksal sürücü** oluşturduk. **Finish** ile bu bölümden çıkmak istiyoruz. Yaptığımız ayarlamaların listesi aşağıdaki gibi görünecektir.

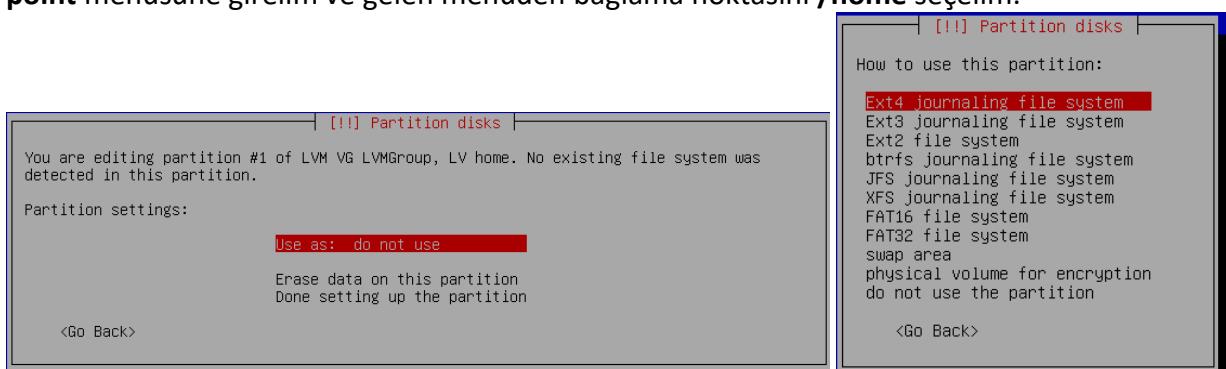


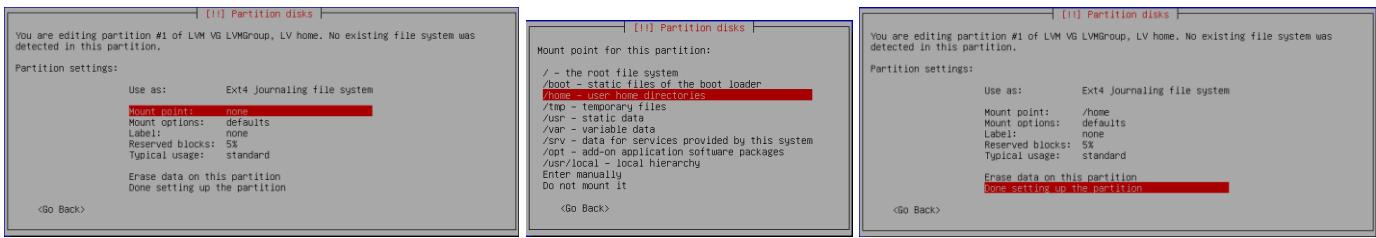
Tahsis edilen diskin kapasite yapısı şematik olarak gösterilmiştir.

| | | | | | |
|-----|------------|---------|------------|------------|--------------------|
| sda | -> 30.8 GB | -> sda1 | -> 525 MB | ~> 500 MB | /boot |
| | | -> sda5 | -> 32.5 GB | -> 10.7 GB | ~> 10 GB /root |
| | | | | -> 2.5 GB | ~> 2.3 GB /swap |
| | | | | -> 5.4 GB | ~> 5.0 GB /home |
| | | | | -> 3.2 GB | ~> 3.0 GB /var |
| | | | | -> 3.2 GB | ~> 3.0 GB /srv |
| | | | | -> 3.2 GB | ~> 3.0 GB /tmp |
| | | | | -> 4.3 GB | ~> 4.0 GB /var-log |

Oluşturduğumuz bölümlerin ayarlarını sırayla yapalım.

home 5.4 GB seçip gelen ekranında **use as:** bölümünü girelim. **Ext4 journaling file system** seçelim. **Mount point** menüsüne girelim ve gelen menüden bağlama noktasını **/home** seçelim.





Ardından partition ayarları menüsüne dönelim.

Sırasıyla işlemlere devam edelim **root 10.7 GB** olan alanı seçelim > dosya tipi **Ext4** > bağlama noktası **/ (root)** seçiyoruz.

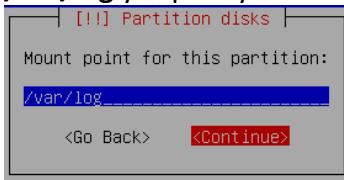
srv 3.2 GB alanı seçelim > dosya tipi **Ext4** > bağlama noktası **/srv** seçiyoruz.

swap 2.5 GB alanı seçelim > dosya tipi **swap area** > bağlama noktası yok.

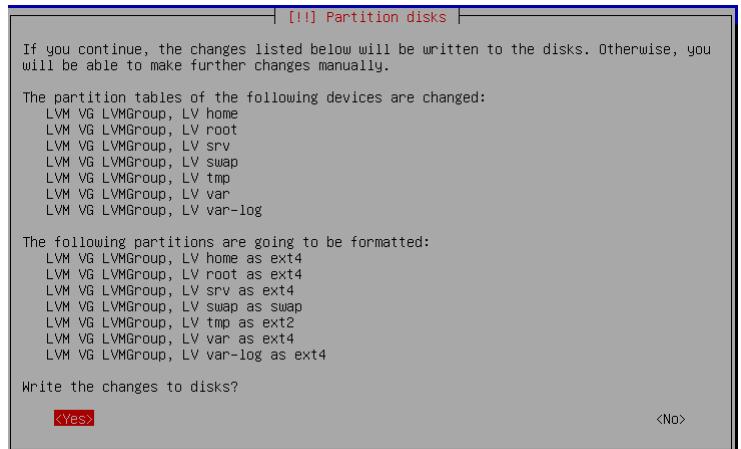
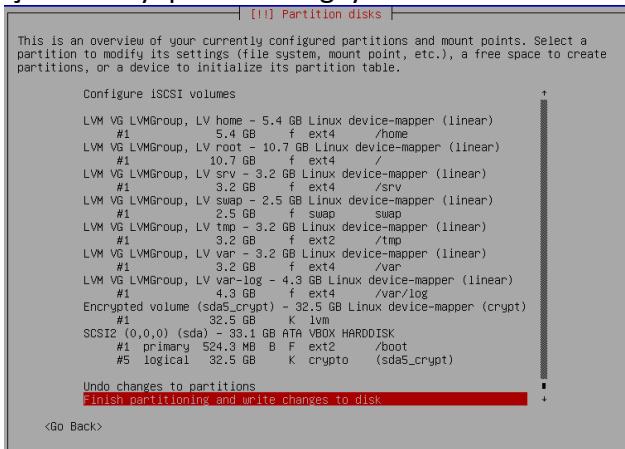
tmp 3.2 GB alanı seçelim > dosya tipi **Ext2** > bağlama noktası **/tmp** seçiyoruz.

var 3.2 GB alanı seçelim > dosya tipi **Ext4** > bağlama noktası **/var** seçiyoruz.

var-log 4.3 GB alanı seçelim > dosya tipi **Ext4** > bağlama noktası için **Enter manually** seçeneğine girip **/var/log** yazıp bu yeni alana bağlıyoruz.

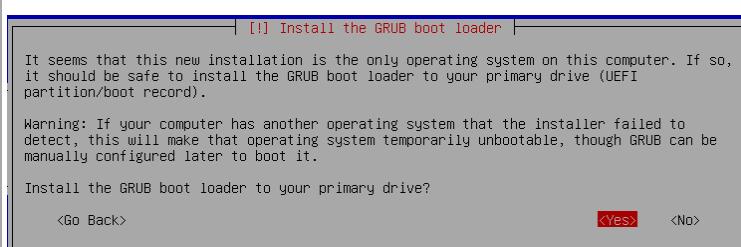
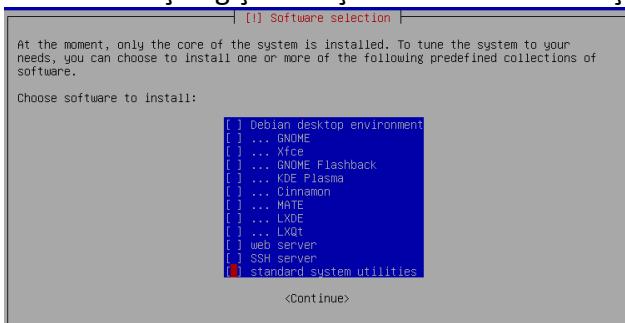


Ayarlama bitirdikten sonra **Finish partitioning and write ...** basalık gelen yeni ekranda yaptığımız ayarlamaların bir özeti gösterecek ve işlemi başlatmak için onayımızı isteyecek **Yes** ile onay veriyor ve işlemlerin yapılmasını sağlıyoruz.

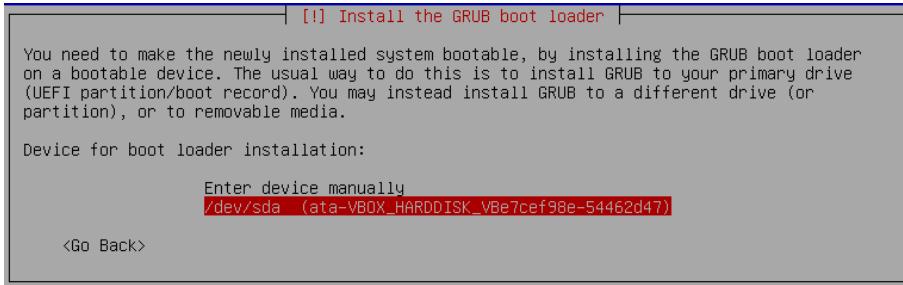


Yeni gelen uyarı ekranını **No** seçip devam ediyoruz. Sonraki adımda **Debian** kütüphanesine hangi ülke sunucuları ile erişeceğimizi belirliyoruz **enter** ardından tekrar **enter** ve **continue** ile devam edelim. Kurulum programı internetten güncellemleri de alarak kurulumu devam eder. Bir sonraki uyarıyı da **No** ile geçiyoruz.

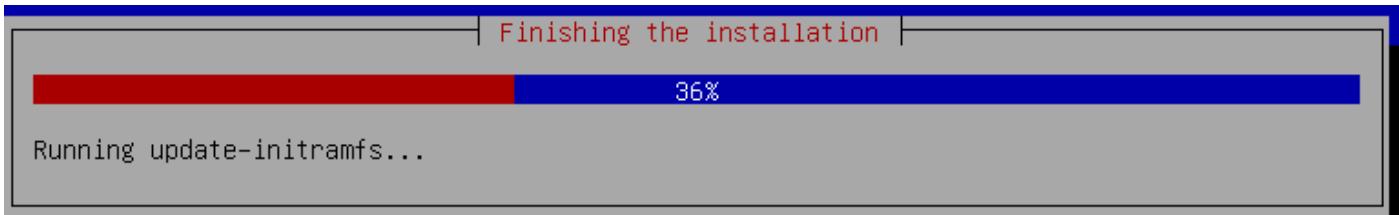
Gelen yeni penceredeki tüm seçimleri kaldırıyoruz. Gerekli kurulumları kendimiz daha sonra yapacağız. Ardından başlangıç **boot** için birincil bölümü seçmek için uyarı ekranında **Yes** seçiyor.



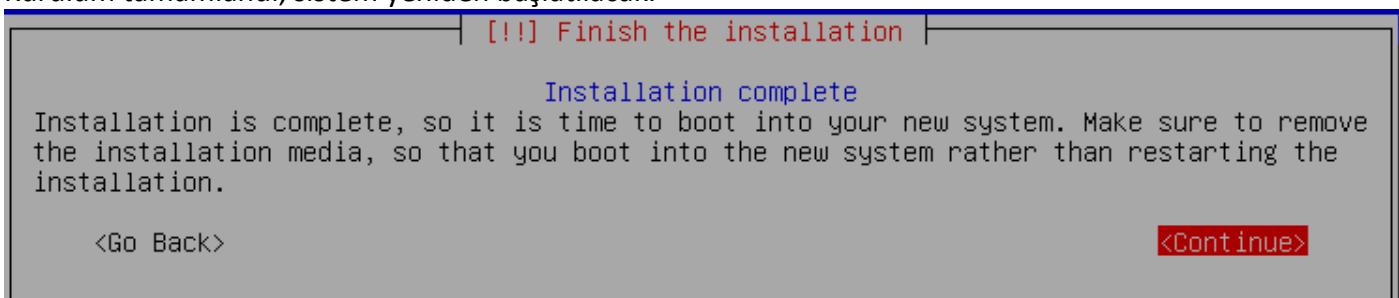
Daha sonra gelen ekranda **/dev/sda** seçimi yapıyoruz.



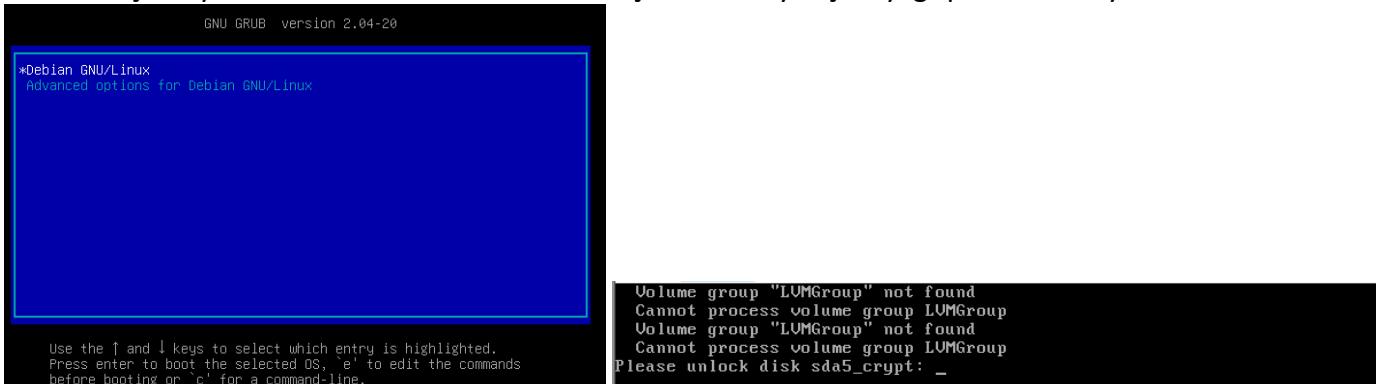
Kurulum tamamlanmak üzere..



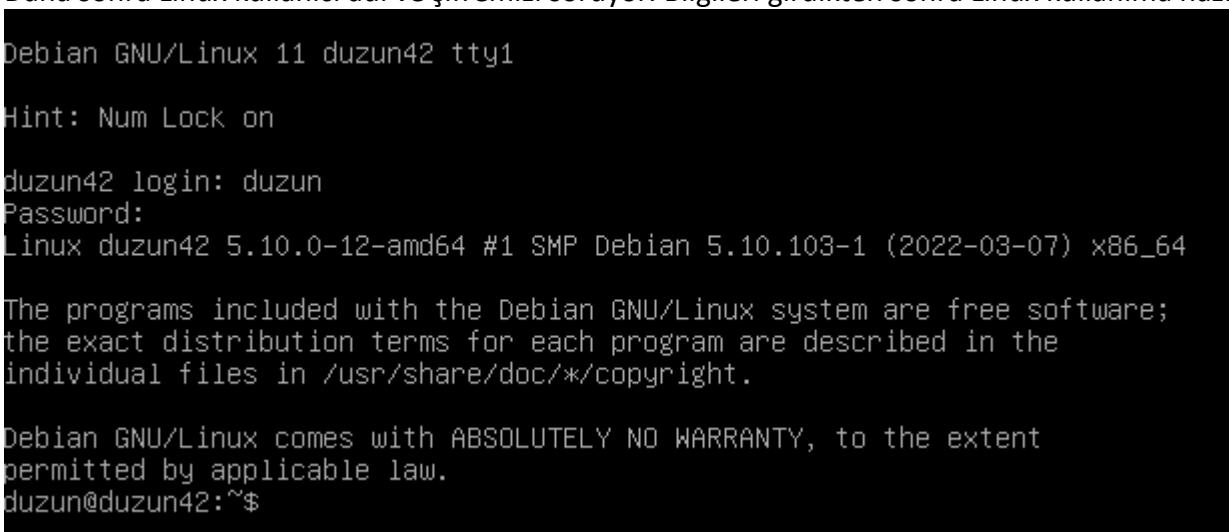
Kurulum tamamlandı, sistem yeniden başlatılacak.



Sistem başlatılıyor... sistem bizden **sda5** diskinin şifresini istiyor şifreyi grip devam ediyoruz.



Daha sonra Linux kullanıcı adı ve şifremizi soruyor. Bilgileri girdikten sonra Linux kullanıma hazır.



KURULUM TAMAMLANDI !!!

Şimdi de born2beroo PDF talimatındaki Bonus da belirtilen disk yapısının bire bir oluşturabilmişiyiz kontrol edelim. Komut satırında **lsblk** yazıp enter yapalım.

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
duzun@duzun42:~$ lsblk
NAME           MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda            8:0    0 30.8G  0 disk
├─sda1          8:1    0  500M  0 part  /boot
├─sda2          8:2    0   1K  0 part
└─sda5          8:5    0 30.3G  0 part
  └─sda5_crypt 254:0   0 30.3G  0 crypt
    ├─LVMGroup-root 254:1   0   10G  0 lvm   /
    ├─LVMGroup-swap 254:2   0   2.3G  0 lvm   [SWAP]
    ├─LVMGroup-home 254:3   0   5G  0 lvm   /home
    ├─LVMGroup-var  254:4   0   3G  0 lvm   /var
    ├─LVMGroup-srv  254:5   0   3G  0 lvm   /srv
    ├─LVMGroup-tmp  254:6   0   3G  0 lvm   /tmp
    └─LVMGroup-var--log 254:7   0   4G  0 lvm   /var/log
sr0           11:0   1 1024M  0 rom
duzun@duzun42:~$
```

Cıktıdan da görüneceği gibi bizden istenen bire bir aynısı elde ettik.

Bundan sonra diğer temel kurulum ve ayarlara geçiyoruz.

Not: Kurulumun bir yedegini alip bundan sonraki işlemlere öyle başlayın. Sorun çıkarsa tüm kurulumu baştan yapmaya gerek kalmaz.

VirtualBox ve Debian Iunix'i bonus ayarlarına (disk yapısına) göre kurduktan sonraki aşamalar.

Adım 1: sudo'yu yükleme

Root ortamına geçmek için : su –

Komutunu çalıştırıp root şifresini giriyorum.

Sudo'yu aşasındaki komutları sırayla uygulayarak yükleyelim (güncelle ve kur):

apt update -y

apt upgrade -y

apt install sudo

```
root@duzun42:~# dpkg -l |grep sudo
ii  sudo                           1.9.5p2-3          amd64      Provide limited super user
ser privileges to specific users
root@duzun42:~# _
```

sudo'nun başarı ile yüklenmediğini test edelim: **dpkg -l | grep sudo**

Bilgi Notu: **dpkg**, **Debian** paket yönetim sisteminin temelini oluşturan yazılımdır. **dpkg . deb** paketlerini yüklemeye, silmeye ve onlar hakkında bilgi toplamaya yarar. **dpkg** düşük seviyeli bir araçtır. **apt** gibi yüksek seviyeli araçlar, paketleri uzak kaynaktan çekmeye ve karışık paket ilişkilerini halletmeye yararlar.

Adım 2: Sudo Grubuna Kullanıcı Ekleme

Sudo grubuna kullanıcı eklemek için bu yapıyı kullanıyoruz: **adduser <username> sudo**

adduser duzun sudo

```
root@duzun42:~# adduser duzun sudo
Adding user `duzun' to group `sudo' ...
Adding user duzun to group sudo
Done.
root@duzun42:~# _
```

Bu çıktında **duzun** kullanıcısı **sudo** grubuna eklendi.

Alternatif kullanım: **usermod -aG sudo <username>**

Örnek: usermod -aG sudo duzun

*Bu komut ile **duzun** kullanıcısı sudo grubuna eklenmiş olur.*

Kullanıcının **sudo** grubuna başarıyla eklenip eklenmediğini doğrulamak için şu komutu çalıştırın:

getent group sudo

```
root@duzun42:~# getent group sudo
sudo:x:27:duzun
root@duzun42:~# _
```

duzun kullanıcısı **sudo** grubuna eklendiğini görüyoruz.

Kullanıcıya su olarak ayrıcalık verin. Bunun için **sudoers** dosyasını açıp aşağıdaki komutu ekleyin.

sudo visudo

Aşağıdaki komut satırını ekleyin

your_username ALL=(ALL) ALL

Benim örneğimde yazım şu şekilde: **duzun ALL=(ALL) ALL**

root ile başlayan satırın altına ekleme yaptım. Farklı bir satır konumunda da yazsanız sorun yapmaz.

```
# User privilege specification
root    ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:
```

Eklemeden sonra dosyayı **^O** (kCtrl + O) ile kaydedin **^Q** (Ctrl + X) ile çıkış yapın.

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
duzun   ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:
```

Değişikliklerin etkili olması için oturumu yeniden başlatıyoruz: **reboot**

```
Debian GNU/Linux 11 duzun42 tty1

duzun42 login: duzun
Password:
```

Sudo gücünü devreye almak için komutu çalıştırın:

sudo -v

```
duzun@duzun42:~$ sudo -v

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 duzun:
duzun@duzun42:~$ _
```

Adım 3: Kök (root) - Ayrıcalıklı Komutları Çalıştırma

Buradan itibaren, **sudo** ön eki aracılığıyla **root** (kök) ayrıcalıklı komutları çalıştırabiliriz.

(**login** **duzun** komutu ile **root** kullanıcısından **duzun** kullanıcısına geçiyorum)

Örneğin Linux güncellemelerini yapmayı deneyelim: **sudo apt-get update -y**

```
duzun@duzun42:~$ sudo apt-get update -y
Hit:1 http://deb.debian.org/debian bullseye InRelease
Hit:2 http://deb.debian.org/debian bullseye-updates InRelease
Hit:3 http://security.debian.org/debian-security bullseye-security InRelease
Reading package lists... Done
duzun@duzun42:~$
```

sudo apt-get upgrade

```
duzun@duzun42:~$ sudo apt-get upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
duzun@duzun42:~$
```

Komut çalıştı ancak en son sürümü yüklediğimden herhangi bir güncelleme gerçekleşmedi.

Adım 4: Çok işimize yarayacak kurulum araçlarını kuralım

git'i yükleme

Aşağıdaki komutları sırayla çalıştıralım.

sudo apt update -y

sudo apt upgrade -y

sudo apt install git -y

Kurulum tamamlanınca git sürümünü kontrol edelim.

git --version

wget'i yükleme (**wget**, **web depolarından dosya indirmek için ücretsiz ve açık kaynaklı bir araçtır.**)

wget yüklemek için aşağıdaki komutu çalıştırınız.

sudo apt-get install wget

```

duzun@duzun42:~$ sudo apt-get install wget
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  wget
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 964 kB of archives.
After this operation, 3,559 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bullseye/main amd64 wget amd64 1.21-1+deb11u1 [964 kB]
Fetched 964 kB in 8s (124 kB/s)
Selecting previously unselected package wget.
(Reading database ... 27291 files and directories currently installed.)
Preparing to unpack .../wget_1.21-1+deb11u1_amd64.deb ...
Unpacking wget (1.21-1+deb11u1) ...
Setting up wget (1.21-1+deb11u1) ...

```

vim yükleme (vim çok kullanışlı bir metin düzenleyici editördür.)

Yüklemek için aşağıdaki komutu çalıştırınız.

sudo apt-get install vim -y

oh my zsh kurulumu (Bu kullanımı çok daha kolay bir kabuk (shell) sağlıyor. git ve diğer programlar için sunduğu özel bilgi ekranları ile terminal kullanımını daha konforlu hale getirmektedir.)

Yüklemek ve kontrol için aşağıdaki komutları sırasıyla çalıştırınız.

sudo apt install zsh -y

zsh –version

tools dosyasını yükleyelim.

zsh -c "\$(wget https://raw.githubusercontent.com/robbyrussell/oh-my-zsh/master/tools/install.sh -O -)"

Kurulum esnasında **default shell** (varsayılan arayüz) olarak **zsh’ı** kullanmayı istediğimizi sordduğunda

Yes/y seçerek devam edelim ki sistem aktif kullanıcımızı bu yüz ile başlatsın. Ardında sudo yetki şifresini girmemiz istiyor, şifreyi girip devam ediyoruz.

```

HTTP request sent, awaiting response... 200 OK
Length: 16963 (17K) [text/plain]
Saving to: 'STDOUT'

    100%[=====] 16.57K --.-KB/s   in 0.002s

2022-04-05 14:51:48 (6.98 MB/s) - written to stdout [16963/16963]

Cloning 'Oh My Zsh'...
remote: Enumerating objects: 1297, done.
remote: Counting objects: 100% (1297/1297), done.
remote: Compressing objects: 100% (1253/1253), done.
remote: Total 1297 (delta 26), reused 1219 (delta 25), pack-reused 0
Receiving objects: 100% (1297/1297), 1.06 MiB | 2.01 MiB/s, done.
Resolving deltas: 100% (26/26), done.
From https://github.com/ohmyzsh/ohmyzsh
 * [new branch]      master    -> origin/master
Branch 'master' set up to track remote branch 'master' from 'origin'.
Already on 'master'

Looking for an existing zsh config...
Using the Oh My Zsh template file and adding it to '~/.zshrc'.

Time to change your default shell to zsh:
Do you want to change your default shell to zsh? [Y/n] y
Changing your shell to /usr/bin/zsh...
[sudo] password for duzun:
....is now installed!

```

Before you scream Oh My Zsh! look over the `~/.zshrc` file to select plugins, themes, and options.

- Follow us on Twitter: <https://twitter.com/ohmyzsh>
- Join our Discord community: <https://discord.gg/ohmyzsh>
- Get stickers, t-shirts, coffee mugs and more: <https://shop.planetargon.com/collections/oh-my-zsh>

Kurulumu tamamladık. Komut istem imleç yapısının değiştiğini gözlemleyebilirsiniz.

Eğer zsh tools yüklemey ve kurulum sırasında NO/n ile devam ederseniz zsh devreye almak için aşağıdakia adımları uygulayınız.

Çalıştırmak için **zsh** yazıp enter basıyorum.

zsh’yi ilk açığınızda aşağıdaki menü ile karşılaşacaksınız.

Bu menünü de (1) seçerseniz, tuş atamalarını ve bir sürü başka şeyi yapılandırmamıza izin veren bir menüye yönlendirileceksiniz. Ancak (2) seçerseniz, önerilen varsayılan ayarlarla profil yapılandırmamızı kendisi oluşturacaktır. (2) seçmenizi öneririm.

```

This is the Z Shell configuration function for new users,
zsh-newuser-install.
You are seeing this message because you have no zsh startup files
(the files .zshenv, .zprofile, .zshrc, .zlogin in the directory
"). This function can help you with a few settings that should
make your use of the shell easier.

You can:
(q) Quit and do nothing. The function will be run again next time.
(o) Exit, creating the file `~/.zshrc` containing just a comment.
    That will prevent this function being run again.
(1) Continue to the main menu.
(2) Populate your `~/.zshrc` with the configuration recommended
    by the system administrator and exit (you will need to edit
    the file by hand, if so desired).

-- Type one of the keys in parentheses --

```

Seçim 2 yapıldıktan sonra **Shell** kullanımız renklendi...

```
--- Type one of the keys in parentheses --- 2
/home/duzun/.zshrc:15: scalar parameter HISTFILE created globally in function zsh-newuser-install
duzun@duzun42 ~ %
```

Zsh tools yüklediğimize göre popüler bir ara yüz seçmek için ayar dosyasını aşağıdaki komut ile açın.

vim **~/.zshrc**

Aşağıda açılan dosya üzerinde 11 satırındaki ZSH_THEME="robbyrussell" tema ismini ZSH_THEME="bira" olarak değiştirmiyorum.

```
1 # If you come from bash you might have to change your $PATH.
2 # export PATH=$HOME/bin:/usr/local/bin:$PATH
3
4 # Path to your oh-my-zsh installation.
5 export ZSH="$HOME/.oh-my-zsh"
6
7 # Set name of the theme to load --- if set to "random", it will
8 # load a random theme each time oh-my-zsh is loaded, in which case,
9 # to know which specific one was loaded, run: echo $RANDOM_THEME
10 # See https://github.com/ohmyzsh/ohmyzsh/wiki/Themes
11 ZSH_THEME="robbyrussell"
12
13 # Set list of themes to pick from when loading at random
14 # Setting this variable when ZSH_THEME=random will cause zsh to load
15 # a theme from this variable instead of looking in $ZSH/themes/
16 # If set to an empty array, this variable will have no effect.
17 # ZSH_THEME_RANDOM_CANDIDATES=( "robbyrussell" "agnoster" )
18
19 # Uncomment the following line to use case-sensitive completion.
20 # CASE_SENSITIVE="true"
```

```
1 # If you come from bash you might have to change your $PATH.
2 # export PATH=$HOME/bin:/usr/local/bin:$PATH
3
4 # Path to your oh-my-zsh installation.
5 export ZSH="$HOME/.oh-my-zsh"
6
7 # Set name of the theme to load --- if set to "random", it will
8 # load a random theme each time oh-my-zsh is loaded, in which case,
9 # to know which specific one was loaded, run: echo $RANDOM_THEME
10 # See https://github.com/ohmyzsh/ohmyzsh/wiki/Themes
11 ZSH_THEME="bira"
12
13 # Set list of themes to pick from when loading at random
14 # Setting this variable when ZSH_THEME=random will cause zsh to load
15 # a theme from this variable instead of looking in $ZSH/themes/
16 # If set to an empty array, this variable will have no effect.
17 # ZSH_THEME_RANDOM_CANDIDATES=( "robbyrussell" "agnoster" )
18
19 # Uncomment the following line to use case-sensitive completion.
20 # CASE_SENSITIVE="true"
```

bira temasının aktif olması için **zsh** komutunu yeniden çalıştırıyorum. (diğer tüm temalar için **github** bakınız)

```
duzun@duzun42 ~ %
$ komutu çalıştırdıktan sonra - seklindeki imlecimiz duzun@duzun42 ~
seklinde daha okunaklı bir görsel haline dönüştü.
```

Adım 5: SSH'i Yükleme ve Yapılandırma

Openssh kurulumu için aşağıdaki komutu kullanıyoruz.

sudo apt install openssh-server -y

yada

sudo apt install ssh -y

Openssh-server'in başarıyla yüklenip yüklenmediğini doğrulayın.

dpkg -l | grep ssh

Aşağıdaki çıktıyı inceleyin...

```
duzun@duzun42 ~ % dpkg -l | grep ssh
ii libssh2-1:amd64          1.9.0-2           amd64      SSH2 client-side libr
ary
ii openssh-client           1:8.4p1-5          amd64      secure shell (SSH) cl
ient, for secure access to remote machines
ii openssh-server            1:8.4p1-5          amd64      secure shell (SSH) se
rver, for secure access from remote machines
ii openssh-sftp-server       1:8.4p1-5          amd64      secure shell (SSH) sf
tp server module, for SFTP access from remote machines
duzun@duzun42 ~ %
```

SSH sunucu durumunu kontrol edin.

sudo systemctl status ssh

Aşağıdaki çıktıyı inceleyiniz.

```
duzun@duzun42 ~ % sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-03-24 12:02:57 CDT; 2min 7s ago
       Docs: man:sshd(8)
              man:sshd_config(5)
   Process: 4147 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 4148 (sshd)
    Tasks: 1 (limit: 1128)
   Memory: 1.1M
      CPU: 19ms
     CGroup: /system.slice/ssh.service
             └─4148 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Mar 24 12:02:57 duzun42 systemd[1]: Starting OpenBSD Secure Shell server...
Mar 24 12:02:57 duzun42 sshd[4148]: Server listening on 0.0.0.0 port 22.
Mar 24 12:02:57 duzun42 sshd[4148]: Server listening on :: port 22.
Mar 24 12:02:57 duzun42 systemd[1]: Started OpenBSD Secure Shell server.
duzun@duzun42 ~ %
```

SSH hizmetini yeniden başlatın
sudo service ssh restart

Varsayılan bağlantı noktasını (22) yi 4242 olarak değiştirme
Aşağıdaki komut ile config dosyasını açıp dosyayı düzenleyin.
sudo vim /etc/ssh/sshd_config

```
# OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp #
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.
# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin
# 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 prohibit-password
#StrictModes yes
#MaxAuthTries 6
```

Bağlantı Noktası 4242'yi kullanarak SSH'yi bağlanmak için **#Port 22** satırı değiştirin (# olmadan):

Eski hali

#Port 22

Yeni hali

Port 4242

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

Port 4242

```
#AddressFamily any
```

Kimlik doğrulama mekanizmasından bağımsız olarak SSH girişini **kök** olarak devre dışı bırakmak için aşağıdaki satırı değiştirin.

Eski hali

#PermitRootLogin prohibit-password

Yeni hali

PermitRootLogin no

Düzenlemeleri aşağıdaki görselde gösterilmiştir.

```
Include /etc/ssh/sshd_config.d/*.conf
#Port 4242
#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 no
#StrictModes yes
```

Aşağıdaki komut ile SSH durumunu kontrol edin.

sudo service ssh status

SSH'nin durumunu şu alternatif komutu kullanarak da kontrol edebiliriz.

sudo systemctl status ssh

Resimde de görüldüğü gibi servis çalışmaktadır.

```
duzun@duzun42 ~ % systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
  Active: active (running) since Thu 2022-03-24 12:07:54 CDT; 40min ago
    Docs: man:sshd(8)
           man:sshd_config(5)
  Process: 4180 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 4181 (sshd)
   Tasks: 1 (limit: 1128)
  Memory: 1.1M
    CPU: 20ms
   CGroup: /system.slice/ssh.service
           └─4181 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
duzun@duzun42 ~ % _
```

Bağlantı noktası ayarlarının doğru olup olmadığını kontrol edin. Doğru sonucu alamazsanız sistemi yeniden başlatın tekrar kontrol edin. Ayar dosyasındaki port numarasını aşağıdaki komut ile kontrol edebilirsiniz.

```
sudo grep Port /etc/ssh/sshd_config
```

```
duzun@duzun42 ~ % sudo grep Port /etc/ssh/sshd_config
Port 4242
#GatewayPorts no
duzun@duzun42 ~ %
```

SSH hizmetini yeniden başlatın.

```
sudo service ssh restart
```

Adım 6: Güvenlik yazılımı UFW'yi Yükleme ve Yapılandırma

ufw'yi aşağıdaki komut aracılığıyla yükleyin

```
sudo apt install ufw -y
```

ufw'nin başarıyla yüklenip yüklenmediğini doğrulayın aşağıdaki komut ile doğrulayın.

```
dpkg -l | grep ufw
```

```
duzun@duzun42 ~ % dpkg -l | grep ufw
ii  ufw                      0.36-7.1          all      program for managing
a Netfilter firewall
duzun@duzun42 ~ %
```

Güvenlik Duvarını aşağıdaki komut ile etkinleştirin.

```
sudo ufw enable
```

```
duzun@duzun42 ~ % sudo ufw enable
Firewall is active and enabled on system startup
duzun@duzun42 ~ % _
```

```
duzun@duzun42 ~
$ sudo ufw enable
Firewall is active and enabled on system startup
duzun@duzun42 ~
$ sudo systemctl status ufw
● ufw.service - Uncomplicated firewall
  Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
  Active: inactive (dead)
    Docs: man:ufw(8)
duzun@duzun42 ~
$
```

Bu şekilde bir uyarı alırsanız **ufw** aktif edilememiştir. Statusune baktığımızda da inactive (aktif değil) durumunda gözükmektedir. Sistemi yeniden başlatmak için **sudo reboot** komutunu çalıştırınız. Sistem yeniden açıldıktan sonra aşağıdaki komut ile tekrar **ufw** aktif mi kontrol edelim.

```
sudo systemctl status ufw
```

Güvenlik duvarı **ufw** çalışmaya başlamış.

```
duzun@duzun42 ~
$ sudo systemctl status ufw
● ufw.service - Uncomplicated firewall
  Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
  Active: active (exited) since Tue 2022-04-05 15:46:58 EDT; 1min 52s ago
    Docs: man:ufw(8)
  Process: 283 ExecStart=/lib/ufw/ufw-init start quiet (code=exited, status=0/SUCCESS)
 Main PID: 283 (code=exited, status=0/SUCCESS)
    CPU: 117ms

Apr 05 15:46:58 duzun42 systemd[1]: Finished Uncomplicated firewall.
Warning: Journal has been rotated since unit was started, output may be incomplete.
duzun@duzun42 ~
$ _
```

Kuralları yapılandırın. SSH'ye izin ver

sudo ufw allow ssh

```
duzun@duzun42 ~ % sudo ufw allow ssh
Rule added
Rule added (v6)
duzun@duzun42 ~ %
```

Aşağıdaki komut ile Bağlantı Noktası **4242**'yi kullanarak gelen bağlantılarla izin verin.

sudo ufw allow 4242

```
duzun@duzun42 ~ % sudo ufw allow 4242
Rule added
Rule added (v6)
duzun@duzun42 ~ %
```

Yeni kuralı silin: (Bu, Born2beroot'unuzu savundığınız zaman için)

sudo ufw status numbered

sudo ufw delete (silinecek kural numarası, örneğin 1 yada 3)

```
duzun@duzun42 ~ % sudo ufw status numbered
Status: active

      To          Action    From
      --          -----   ---
[ 1] 22/tcp      ALLOW IN  Anywhere
[ 2] 4242       ALLOW IN  Anywhere
[ 3] 22/tcp (v6) ALLOW IN  Anywhere (v6)
[ 4] 4242 (v6)  ALLOW IN  Anywhere (v6)

duzun@duzun42 ~ %
```

Bir numaralı kuralı silmek için (22 port kurallarını silin)

sudo ufw delete 1

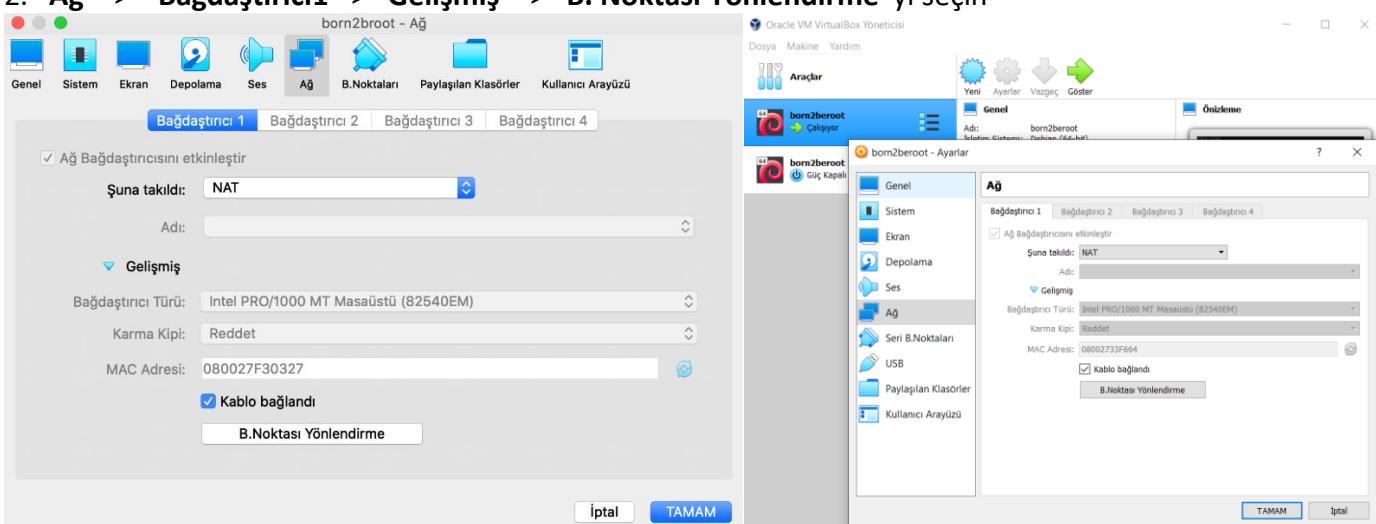
```
duzun@duzun42 ~ % sudo ufw delete 1
Deleting:
allow 22/tcp
Proceed with operation (y|n)? y
Rule deleted
```

Adım 7: Sunucuya SSH ile Bağlanma

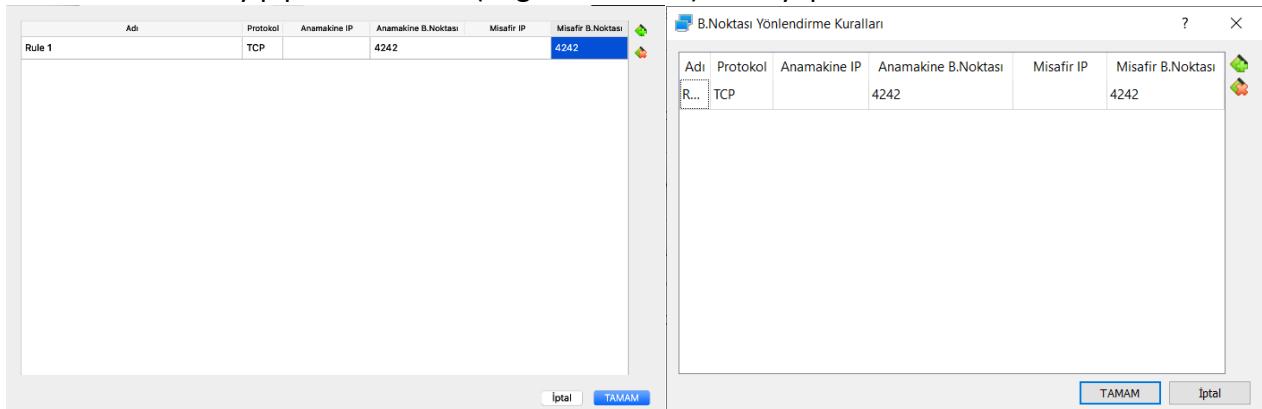
VirtualBox için ileri kural ekle

1. **VirtualBox'a gidin -> VM -> Ayarlar'i seçin**

2. "Ağ" -> "Bağdaştırıcı1"-> "Gelişmiş" -> "B. Noktası Yönlendirme"yi seçin



Yeni bir kural ekleyip port numarası (bağlantı noktası) 4242 yapın.



SSH sunucusunu yeniden başlatın (VM makinenize gidin)

sudo systemctl restart ssh

ssh durumunu kontrol edin:

sudo service sshd status

yada

sudo systemctl status ssh

iTerm2 veya Terminal'den ana bilgisayar tarafından aşağıda gösterildiği gibi girin:

ssh <username>@127.0.0.1 -p 4242

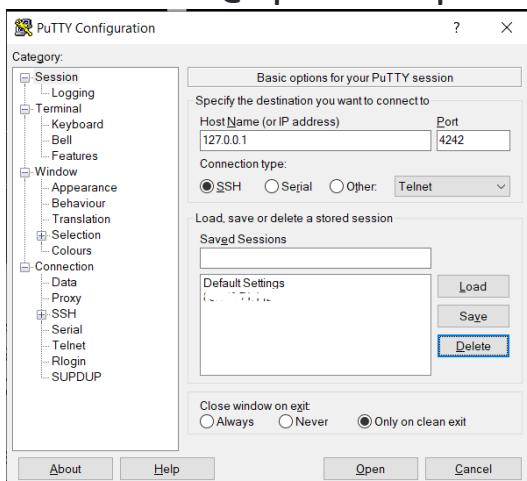
```
duzun@duzun42:~$ Last login: Thu Apr  7 13:42:17 on ttys008
duzun@1m35s07 ~
$ ssh duzun@127.0.0.1 -p 4242
duzun@127.0.0.1's password:
Linux duzun42 5.10.0-13-amd64 #1 SMP Debian 5.10.106-1 (2022-03-17) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Apr  7 08:50:54 2022 from 10.0.2.2
duzun@duzun42 ~
$
```

Yada PuTTY ve benzeri programlar aracılığı Port 4242'yi kullanarak sanal makinenize bağlanabilirsiniz.

ssh <username>@<ip-address> -p 4242



```
duzun@duzun42:~$ login as: duzun
duzun@127.0.0.1's password:
Linux duzun42 5.10.0-13-amd64 #1 SMP Debian 5.10.106-1 (2022-03-17) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Apr  5 16:03:29 2022 from 10.0.2.2
duzun@duzun42 ~
$
```

SSH oturumunu istediğiniz zaman aşağıdaki komut ile sonlandırın

logout

Alternatif olarak, SSH oturumunu exit ile de yapabilirsiniz.

exit

Adım 8: Ana bilgisayar adını değiştir (!!!Bu, savunma yaparken size sorulacak!!!)

Ana bilgisayar adını değiştirin

1. Sunucunuza giriş yapın
2. **sudo -s** ya da **su -** kullanarak bir kök kullanıcı olun
3. Mevcut ana bilgisayar adını kontrol edin

Hostnamectl

```
duzun@duzun42 ~ % hostnamectl
  Static hostname: duzun42
    Icon name: computer-vm
      Chassis: vm
    Machine ID: 0b971396935a40a492b3649cdf88d707
      Boot ID: 75fa733b56f84aa69423ccbcb13e1835
  Virtualization: oracle
  Operating System: Debian GNU/Linux 11 (bullseye)
        Kernel: Linux 5.10.0-12-amd64
      Architecture: x86-64
duzun@duzun42 ~ % _
```

4. Ana bilgisayar adını sunucu1 olarak ayarlamak için şunu çalıştırın:

5. **hostnamectl set-hostname server1**

```
root@duzun42:/home/duzun# su -
root@duzun42:~# hostnamectl set-hostname server1
root@duzun42:~# hostnamectl
  Static hostname: server1
    Icon name: computer-vm
      Chassis: vm
    Machine ID: 0b971396935a40a492b3649cdf88d707
      Boot ID: 75fa733b56f84aa69423ccbcb13e1835
  Virtualization: oracle
  Operating System: Debian GNU/Linux 11 (bullseye)
        Kernel: Linux 5.10.0-12-amd64
      Architecture: x86-64
root@duzun42:~#
```

6. /etc/hosts dosyasını düzenleyin ve girişleri güncelleyin

vim /etc/hosts

```
127.0.0.1      localhost
127.0.1.1      duzun42
```

Değiştirip aşağıdaki gibi yapıyoruz.

```
127.0.0.1      localhost
127.0.1.1      server1
```

7. Yeniden başlatın ve değişikliği kontrol edin.

reboot

8. **hostnamectl** Komutu tekrar çalıştırarak doğrulayın.

```
duzun@server1 ~
$ su -
Password:
root@server1:~# hostnamectl
  Static hostname: server1
    Icon name: computer-vm
      Chassis: vm
    Machine ID: ee266cfdae4241fdb45eb459315f295f
      Boot ID: 17cb0148c39945a88aea2d21e44a88b0
  Virtualization: oracle
  Operating System: Debian GNU/Linux 11 (bullseye)
        Kernel: Linux 5.10.0-12-amd64
      Architecture: x86-64
root@server1:~# +
```

Adım 9: sudo'yu yapılandırma

sudo kullanırken yapılan her işlem (tüm girdi ve çıktılar) kayıt altında tutulmalıdır. Kayıtların tutulduğu log dosyası **/var/log/sudo/** klasörüne kaydedilmelidir. Bunu gerçekleştirmek için ilgili klasörü oluşturalım.

sudo mkdir /var/log/sudo

```
duzun@duzun42 ~ % cd /var/log/sudo  
cd: no such file or directory: /var/log/sudo  
duzun@duzun42 ~ % sudo mkdir /var/log/sudo  
duzun@duzun42 ~ % cd /var/log/sudo  
duzun@duzun42 /var/log/sudo % _
```

Yukarıdaki resimde görüleceği üzere **/var/log/sudo** dizinine erişmeye çalıştığımızda böyle bir klasör olmadığını görüyoruz. Ardından **mkdir** komutu ile ilgili klasörü oluşturunca içine erişebiliyoruz.

sudo aracılığımız ile sistemi yapılandıralım. Yâda önce bir yapılandırma dosyası oluşturalım. Bunun için aşağıdaki komut yapısını kullanacağınız: **sudo vim /etc/sudoers.d/<filename>**

Ancak ben mevcut ayar dosyası üzerinden işlemleri yapacağım.

sudo vim /etc/sudoers

yada

sudo visudo

Komutuyla ayar dosyasını açıyorum.

Tüm sudo girişlerini ve çıkışlarını **/var/log/sudo/** yoluna arşivlemek için aşağıdaki satırları ekliyorum:

```
Defaults    log_input,log_output  
Defaults    logfile="/var/log/sudo/sudo.log"  
# This file MUST be edited with the 'visudo' command as root.  
#  
# Please consider adding local content in /etc/sudoers.d/ instead of  
# directly modifying this file.  
#  
# See the man page for details on how to write a sudoers file.  
#  
Defaults    log_input,log_output  
Defaults    logfile="/var/log/sudo/sudo.log"  
Defaults    env_reset  
Defaults    mail_badpass  
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"  
  
# Host alias specification  
  
# User alias specification  
  
# Cmnd alias specification  
  
# User privilege specification  
root    ALL=(ALL:ALL) ALL  
duzun   ALL=(ALL:ALL) ALL  
# Allow members of group sudo to execute any command  
xsudo   ALL=(ALL:ALL) ALL  
  
# See sudoers(5) for more information on "@include" directives:  
@includedir /etc/sudoers.d
```

Dosya eklemeleri yazdıktan sonra **esc** tuşuna basıp **:w** komutu ile kayıt etme esnasında **readonly** uyarısı alırsanız **:w!** Komutu ile dosyayı kaydediniz. **:q** ile vim editörünü kapatabilirsiniz.

Güvenlik sebepleriyle TTY modu aktif hale getirilmelidir

TTY'yi istemek için aşağıdaki komut satırını ekliyoruz:

Defaults requiretty

TTY Nedir ?

Günümüzde işletim sisteminde terminal olarak işlev görebilen tüm device'lara erişim için kullanılan alt sisteme **TTY** denilmektedir. Yani **TTY** artık fiziksel bir cihaz yerine gerçek veya sanal terminal olarak düşünülmeli.

Linux'ta tty'yi nasıl açarım?

Bir seferde, Ctrl + Alt + F1'den Ctrl + Alt + F6'ya kadar, tam ekran TTY konsollarını açar ve Ctrl + Alt + F7 sizin grafiksel masaüstü ortamınıza döndürür. Daha eski bir Linux dağıtımını çalıştırıyorsanız, sisteminizin davranışını bu olabilir.

Bu, Manjaro, Ubuntu ve Fedora'nın mevcut sürümlerinde test edildi ve hepsi şu şekilde davrandı:

Ctrl + Alt + F1 : TTY 1'i açar yada Sizi grafik masaüstü ortamı oturum açma ekranına döndürür.

Ctrl + Alt + F2 : TTY 2'yi açar yada Sizi grafik masaüstü ortamına döndürür.

Ctrl + Alt + F3 : TTY 3'ü açar.

Ctrl + Alt + F4 : TTY 4'ü açar.

Ctrl + Alt + F5 : TTY 5'i açar.

Ctrl + Alt + F6 : TTY 6'yi açar.

Apple bilgisayarda kısa yol tuşları fn+Option+F1...F6 gibi

Bu tam ekran konsollara erişim, yalnızca komut satırı kullanan kişilere yalnızca Linux kurulumları sağlar ve birçok Linux sunucusu bu şekilde yapılandırılmıştır - birden çok konsola sahip olabilir.

Ctrl + Alt + F1 tuşlarına basmak sizi TTY 1'e yada grafiksel masaüstü oturumunuzun oturum açma istemine döndürür.

Herhangi bir grafik masaüstü ortamına sahip bir Linux makinesinde çalışığınız ve oturumunuzun donmasına neden olan bir şey oldu mu? Artık durumu düzeltmeyi deneyebilmek için TTY konsol oturumlarından birine geçebilirsiniz.

Pts ve tty arasındaki fark nedir?

tty normal bir terminal cihazıdır (örneğin sunucunuzdaki konsol).

pts, sanal/taklit (uzak) terminal bağımlısıdır (bir xterm veya **ssh** bağlantısı).

Kim Hangi Terminali TTY kullanıyor?

who komutu ile bunu görebiliriz.

Ayrıca şu an çalışmaktadır olduğumuz terminalin TTY öğrenmek için **tty** komutunu kullanabiliriz.

Örnek Çıktayı inceleyiniz. (PuTTY programı ile uzaktan bağlantı kurduğum bağlantı terminal adı **pts/0** olarak görünüyor)

```
duzun@duzun42 ~
$ who
duzun    tty1          2022-04-05 17:53
duzun    pts/0          2022-04-05 17:53 (10.0.2.2)
duzun    tty2          2022-04-05 17:54
duzun@duzun42 ~
$ tty
/dev/tty2
duzun@duzun42 ~
$ _
```

Yine güvenlik sebebiyle, **sudo** tarafından kullanılan dizinler sınırlanmalıdır.

Örnek olarak: **/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin**

Aşağıdaki komutu ekliyoruz (eğer böyle bir komut varsa ilgili komutu düzenliyoruz.)

Defaults secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

sudo ile yetkilendirme 3 yanlış parola denemesi ile sınırlandırılmalıdır.

Yanlış bir parola olması durumunda **sudo** kullanarak kimlik doğrulamasını 3 denemeye (varsayılan 3'tür) sınırlamak için dosyaya aşağıdaki satırı ekleyin.

Defaults passwd_tries=3

sudo kullanırken yanlış şifre sebebiyle bir hata meydana gelirse seçtiğiniz özel bir mesaj gösterilmelidir. Yanlış şifre durumunda özel bir hata mesajı eklemek için aşağıdaki komutu ekleyin:

Defaults badpass_message="<custom-error-message>"

Örneğin:

Defaults badpass_message="Hatalı Şifre Girdiniz! Lütfen Tekrar Giriniz."

Yaptığımız ayarlamalar aşağıdaki görüntüde toplu olarak görülmektedir

```
# See the man page for details on how to write a sudoers file.
#
Defaults        log_input,log_output
Defaults        logfile="/var/log/sudo/sudo.log"
Defaults        requiretty
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"
Defaults        passwd_tries=3
Defaults        badpass_message="Hatalı Şifre Girdiniz! Lütfen Tekrar Giriniz."
#
# Host alias specification
#
# User alias specification
#
# Cmnd alias specification
#
# User privilege specification
root    ALL=(ALL:ALL) ALL
duzun  ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL
#
# See sudoers(5) for more information on "@include" directives:
```

Adım 10: Kullanıcı Yönetimi

Adım 1: Güçlü Bir Parola Politikası Ayarlama

Şifre Yaşı

Parola yaşı ilkesini **sudo vim /etc/login.defs** aracılığıyla yapılandıralım. **login.defs** dosyasını açalım.

sudo vim /etc/login.defs

```
# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be used.
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#      PASS_WARN_AGE    Number of days warning given before a password expires.
#
PASS_MAX_DAYS  99999
PASS_MIN_DAYS  0
PASS_WARN_AGE  7
#
# Min/max values for automatic uid selection in useradd
```

Parolayı her 30 günde bir sona erecek şekilde ayarlamak için aşağıdaki satırı değiştirin (160.satır)
İlk hali

PASS_MAX_DAYS 99999

Son hali

PASS_MAX_DAYS 30

Şifre değişiklikleri arasındaki minimum gün sayısını 2 gün olarak ayarlamak için aşağıdaki satırı değiştirin.(

161. Satır)

İlk hali

PASS_MIN_DAYS 0

Son hali

PASS_MIN_DAYS 2

Kullanıcıya parola süresi dolmadan 7 gün önce (varsayılanı 7'dir) bir uyarı mesajı göndermek için, aşağıdaki satırı olduğu gibi tutun. (162. Satır)

İlk ve son hali

PASS_WARN_AGE 7

Ayarlamaları yaptıktan sonra dosyayı kaydedin. Resimde son durum gözükmektedir.

```
158 #      PASS_WARN_AGE   Number of days warning given before a password expires.
159 #
160 PASS_MAX_DAYS  30
161 PASS_MIN_DAYS  2
162 PASS_WARN_AGE  7
163
```

Yeni oluşturulacak kullanıcılar bu kurallara tabi olacak, eski kullanıcıların ayarlarını aşağıdaki komutları uygulayarak düzeltin.

Bu parametreleri mevcut kullanıcılara uygulayın (benim çalışmamda **root** ve **duzun** kullanıcıları için ayrı ayrı komutları uyguladım):

PASS_MIN_DAYS 2:

```
$ sudo chage --mindays 2 <user>
```

PASS_MAX_DAYS 30:

```
$ sudo chage --maxday 30 <user>
```

PASS_WARN_AGE 7:

```
$ sudo chage --warndays 7 <user>
```

Şifre politikasını kontrol edin:

```
$ chage -l <user>
```

Aşağıdaki çıktıyı kontrol ediniz.

```
duzun@duzun42 ~
└─$ sudo chage --mindays 2 duzun
duzun@duzun42 ~
└─$ sudo chage --maxday 30 duzun
duzun@duzun42 ~
└─$ sudo chage --warndays 7 duzun
duzun@duzun42 ~
└─$ sudo chage --mindays 2 root
duzun@duzun42 ~
└─$ sudo chage --warndays 7 root
duzun@duzun42 ~
└─$ sudo chage --maxday 30 root
duzun@duzun42 ~
└─$ sudo chage -l duzun
Last password change : Mar 25, 2022
Password expires      : Apr 24, 2022
Password inactive     : never
Account expires        : never
Minimum number of days between password change : 2
Maximum number of days between password change : 30
Number of days of warning before password expires : 7
duzun@duzun42 ~
└─$ sudo chage -l root
Last password change : Mar 26, 2022
Password expires      : Apr 25, 2022
Password inactive     : never
Account expires        : never
Minimum number of days between password change : 2
Maximum number of days between password change : 30
Number of days of warning before password expires : 7
duzun@duzun42 ~
└─$
```

Sifrenin Gizlilik Gücü Ayarlama

İkinci olarak, parola gücüyle ilgili ilkeleri ayarlamak için **libpam-pwquality** paketini kurun.

```
sudo apt install libpam-pwquality -y
```

libpam-pwquality'nin başarıyla yüklenip yüklenmediğini doğrulayın.

```
dpkg -l | grep libpam-pwquality
```

```
duzun@duzun42 ~ % dpkg -l | grep libpam-pwquality
ii  libpam-pwquality:amd64          1.4.4-1                               amd64      PAM module to check p
password strength
duzun@duzun42 ~ % _
```

Kurulum gerçekleşmiş sorun yok.

Parola gücünü ayarlamak **sudo vim /etc/pam.d/common-password** ile ayar dosyasını açalım.

Aşağıda görüntüsü yer alan 25. satır aracılığı ile özellikleri yapılandırın:

```
...
25 password    requisite          pam_pwquality.so retry=3
...
24 # here are the per-package modules (the "Primary" block)
25 password    requisite          pam_pwquality.so retry=3
26 password    [success=1 default=ignore]      pam_unix.so obscure use_authtok try_first_pass y
   encrypt
27 # here's the fallback if no module succeeds
28
29 # here's the fallback if no module succeeds
```

Şimdi 25. ve 26. Satırlarda değişiklikler yaparak şifre özelliklerini belirleyelim.

26. Satırı değiştirerek tam olarak aşağıdaki gibi düzenleyin:

```
password    [success=2 default=ignore] pam_unix.so obscure sha512
  credit=-1 maxrepeat=3 usercheck=1 difok=7 enforce_for_root minlen=10
26 password    [success=1 default=ignore]      pam_unix.so obscure sha512
27 # here's the fallback if no module succeeds
```

Şifrenin en az bir büyük harf içermesini zorunlu kılmak için:

ucredit=-1

Şifrenin en az bir küçük harf içermesi zorunlu kılmak için:

lcredit=-1

Şifrenin en az bir sayısal karakter içermesini zorunlu kılmak için:

dcredit=-1

En fazla 3 ardışık aynı karakter ayarlamak için:

maxrepeat=3

Bir biçimde <kullanıcı adı> içeriyorsa parolayı reddetmek için:

usercheck=1

Yeni şifrede gerekli değişiklik sayısını eski şifreden 7'ye ayarlamak için:

difok=7

Tüm bu şifre politikasını **root** kullanıcısı üzerinde uygulamak için :

enforce_for_root

Şifre minimum uzunluğunu 10 karakter olarak ayarlamak için:

minlen=10

Son olarak ayarladığımız satır, aşağıdaki gibi görünmelidir:

```
password    requisite          pam_pwquality.so retry=3 dccredit=-1 difok=7 enforce_for_root lcredit=-1
maxrepeat=3 ucredit=-1 usercheck=1 minlen=10
```

```
24 # here are the per-package modules (the "Primary" block)
25 password    requisite          pam_pwquality.so retry=3 ucredit=-1 dccredit=-1 l
   credit=-1 maxrepeat=3 usercheck=1 difok=7 enforce_for_root minlen=10
26 password    [success=1 default=ignore]      pam_unix.so obscure sha512
27 # here's the fallback if no module succeeds
```

Değişikliklerin etkinleşmesi için sistemi yeniden başlatın:

sudo reboot

Yeni Bir Kullanıcı Oluşturma

sudo adduser <username> aracılığıyla yeni kullanıcı oluşturun.

sudo adduser kullanici42

Linux Kullanıcı Var Mı / Oluşturuldu Mu?

Kullanıcının aracılığıyla başarıyla oluşturulup oluşturulmadığını aşağıdaki komut ile doğrulayın:

sudo getent passwd kullanici42

yada

id <kullanıcı adı>

id kullanici42

Yeni oluşturulan kullanıcının parola süre sonu bilgilerini aşağıdaki komut aracılığıyla ile doğrulayın.
(Kullanıcılarda şifre kurallarının çalışıp çalışmadığını kontrol edin)
sudo chage -l kullanıcı42

Not: ilerde grup oluşturmayı öğrenince **user42** grubu oluşturup oluşturduğumuz **duzun** 'u (root kullanıcısı dışında oluşturduğunuz kullanıcı) bu gruba ekleyelim. Savunmada kontrol edilecek.

Aşağıdaki resimdeki akışı inceleyiniz.

```
duzun@duzun42 ~ % sudo adduser kullanıcı42
Adding user `kullanıcı42' ...
Adding new group `kullanıcı42' (1001) ...
Adding new user `kullanıcı42' (1001) with group `kullanıcı42' ...
Creating home directory `/home/kullanıcı42' ...
Copying files from `/etc/skel' ...
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
passwd: password updated successfully
Changing the user information for kullanıcı42
Enter the new value, or press ENTER for the default
    Full Name []: test Kullanıcı
    Room Number []: 42
    Work Phone []: 374 220 55 55
    Home Phone []: 0532 354 50 13
    Other []: 202
Is the information correct? [Y/n] y
duzun@duzun42 ~ % sudo getent passwd kullanıcı42
kullanıcı42:x:1001:1001:test kullanıcı,42,374 220 55 55,0532 354 50 13,202:/home/kullanıcı42:/bin/bash
duzun@duzun42 ~ % sudo chage -l kullanıcı42
Last password change : Mar 24, 2022
Password expires       : Apr 23, 2022
Password inactive      : never
Account expires        : never
Minimum number of days between password change : 2
Maximum number of days between password change : 30
Number of days of warning before password expires: 7
duzun@duzun42 ~ % _
```

Aktif kullanıcı hesabının hangi grplara ait olduğunu kontrol edin:

groups

```
duzun@duzun42 ~ % groups
duzun cdrom floppy sudo audio dip video plugdev netdev bluetooth
duzun@duzun42 ~ % _
```

Herhangi bir kullanıcının hangi grplara üye olduğunu öğrenmek için

groups <kullanıcı adı>

groups user42 gibi

```
[duzun@duzun42 ~]
$ groups duzun
duzun : duzun cdrom floppy sudo audio dip video plugdev netdev bluetooth
[duzun@duzun42 ~]
$ groups user42
user42 : user42 sudo
[duzun@duzun42 ~]
$ _
```

Debian 10'da bir kullanıcıyı silmek için **deluser** komutunu kullanmanız gereklidir.

sudo deluser <username>

Bir kullanıcıyı ana dizini ile kaldırmak için **deluser** komutunu **--remove-home** parametresiyle çalıştırın.

sudo deluser --remove-home <username>

Bir kullanıcıyla ilişkili tüm dosyaları silmek için **--remove-all-files** parametresini kullanın.

sudo deluser --remove-all-files <username>

Linux şifreler diğer sistemlere oldukça benzer biçimde yönetir. Ancak, şifre yönetiminde iki önemli dizini bilmek önemlidir.

Bunlardan ilki sistemdeki bütün kullanıcıları bulabileceğiniz **/etc/passwd** dizinidir. Ek olarak kullanıcıların sistemin hangi grubuna ait olduklarını gösterir. Birçok sayıda kullanıcı görmek mümkündür ancak sadece birkaç sistem kullanıcıları oluklarından dolayı giriş yapabilecektir.

Tüm Kullanıcı listesini görmek için

Şu basit komutu kullanabilir

compgen -u

yada

Eğer **/etc/passwd** dizininin içeriği görmek isterseniz aşağıdaki komutla yapabilirsiniz:

sudo cat /etc/passwd

kullanıcıları ayıklamak için komutu şu şekilde çalıştırın.

cut -d ":" -f 1 /etc/passwd

Şifrelerle alakalı olan ikinci Linux dizini ise **/etc/shadow** dizinidir. **Shadow** dosyasında kullanıcıların şifrelenmiş parolalarını ve SSH anahtarına sahip olup olmadıklarını görebilirsiniz.

sudo cat /etc/shadow

Linux Şifre Değiştirme Nasıl Yapılır?

Linux'da şifrenizi değiştirmek aşağıdaki komutu girmeniz gerekiyor:

Kullanıcı belirtmediğimiz ve **sudo** yetkisi ile işlem yaptığımız için **root** kullanıcısının şifresi değiştirilecektir.

sudo passwd

sudo olmadan komutu yazarsak mevcut kullanıcının şifresinin değiştirileceğini ve bizden önce mevcut şifreyi girmemi ister ve ardından yeni şifreyi 2 defa girmemizi ister.

passwd

Ekran çıktısını inceleyin.

```
duzun@duzun42 ~
$ sudo passwd
New password:
Retype new password:
passwd: password updated successfully
duzun@duzun42 ~
$ passwd
Changing password for duzun.
Current password:
New password:
Retype new password:
passwd: password updated successfully
duzun@duzun42 ~
$
```

Başka Bir Kullanıcı İçin Linux Şifre Değiştirme Nasıl Yapılır?

Bunu yapmak içinse **kök (root)** kullanıcıyı olarak giriş yapmış olmanız veya **sudo** yetkisi kullanarak ve kullanıcı adını **passwd** komutuna eklemeniz gereklidir.

root yetkisi ile

passwd kullanıcı_adi

yada sudo ile

sudo passwd kullanıcı_adi

Yeni Bir Grup Oluşturma

sudo addgroup <grup> aracılığıyla **ofis** isimli yeni bir grup oluşturun:

sudo addgroup ofis

```
duzun@duzun42 ~ % sudo addgroup ofis
Adding group `ofis' (GID 1002) ...
Done.
duzun@duzun42 ~ % _
```

sudo adduser <username> <group> aracılığıyla ofis grubuna kullanıcı ekleyin.

sudo adduser kullanıcı42 ofis (alternatif yöntem: **sudo usermod -aG ofis kullanıcı42**)

```
duzun@duzun42 ~ % sudo adduser kullanıcı42 ofis
Adding user `kullanıcı42' to group `ofis' ...
Adding user kullanıcı42 to group ofis
Done.
duzun@duzun42 ~ %
```

kullanıcı42 ofis grubuna eklendi.

kullanıcı42 kullanıcısının **ofis** grubuna başarıyla eklenip eklenmediğini doğrulayın. Bunun için hangi gruplara üye olduğunu aşağıdaki komut ile öğrenebiliriz.

Groups <kullanıcı adı>

Groups kullanıcı42

Alternatif olarak aşağıdaki komuttan da faydalabilirsiniz.

id kullanıcı42

```
duzun@duzun42 ~ %
$ sudo usermod -aG ofis kullanıcı42
duzun@duzun42 ~ %
$ groups kullanıcı42
kullanıcı42 : kullanıcı42 ofis
duzun@duzun42 ~ %
$ id kullanıcı42
uid=1002(kullanıcı42) gid=1003(kullanıcı42) groups=1003(kullanıcı42),1002(ofis)
duzun@duzun42 ~ %
$
```

Crontab yapılandırması

Bir **crontab** dosyası, **cron** arka plan programı için aşağıdaki basitleştirilmiş şekilde yöneticiler içerir: "bu komutu şu anda, bu tarihte çalıştır"

1. **netstat** araçlarını yükleyin.
sudo apt update -y
sudo apt install -y net-tools
2. **monitoring.sh** dosyasını **/usr/local/bin/** dizinine yerleştirin

```
#!/bin/bash
arc=$(uname -a)
pcpu=$(grep "physical id" /proc/cpuinfo | sort | uniq | wc -l)
vcpu=$(grep "^processor" /proc/cpuinfo | wc -l)
fram=$(free -m | grep Mem: | awk '{print $2}')
uram=$(free -m | grep Mem: | awk '{print $3}')
pram=$(free -m | grep Mem: | awk '{printf("%.2f"), $3/$2*100}')
fdisk=$(df -Bg | grep '^/dev/' | grep -v '/boot$' | awk '{ft += $2} END {print ft}')
udisk=$(df -Bm | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} END {print ut}')
pdisk=$(df -Bm | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} {ft+= $2} END {printf("%d"), ut/ft*100}')
cpul=$(top -bn1 | grep '^%Cpu' | cut -c 9- | xargs | awk '{printf("%.1f%%"), $1 + $3}')
lb=$(who -b | awk '$1 == "system" {print $3 " " $4}')
lvmt=$(lsblk -o TYPE | grep "lvm" | wc -l)
lvmu=$(if [ $lvmt -eq 0 ]; then echo no; else echo yes; fi)
# net-tools araçları gereklidir:
ctcp=$(cat /proc/net/tcp | wc -l | awk '{print $1-1}' | tr " ")
ulog=$(users | wc -w)
ip=$(hostname -I)
mac=$(ip link show | awk '$1 == "link/ether" {print $2}')
# Journalctl çalışırabilir cunku komut dosyası sudo cron'dan yürütülür.
cmds=$(journalctl _COMM=sudo | grep COMMAND | wc -l)
wall " #Architecture: $arc
#CPU physical: $pcpu
#vCPU: $vcpu
#Memory Usage: $uram/${fram}MB ($pram%)
#Disk Usage: $udisk/${fdisk}Gb ($pdisk%)
#CPU load: $cpul
#Last boot: $lb
#LVM use: $lvmu
#Connexions TCP : $ctcp ESTABLISHED
#User log: $ulog
#Network: IP $ip ($mac)
#Sudo: $cmds cmd"
```

3. Komut dosyasının sudo parolası olmadan yürütüleceği kuralı ekleyin:

sudoers dosyasını açın:

sudo vim /etc/sudoers

yada

sudo visudo

Bu satırı ekleyin:

your_username ALL=(ALL) NOPASSWD: /usr/local/bin/monitoring.sh

düzenleme aşağıdaki resimdeki gibi görünecek.

```
# Defaults      log_input,log_output
Defaults      logfile="/var/log/sudo/sudo.log"
Defaults      requiretty
Defaults      env_reset
Defaults      mail_badpass
Defaults      secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/snap/bin"
Defaults      passwd_tries=3
Defaults      badpass_message="Hatalı Şifre Girdiniz! Lütfen Tekrar Giriniz."
# Host alias specification
# User alias specification
# Cmnd alias specification
# User privilege specification
root    ALL=(ALL:ALL) ALL
duzun  ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
ssudo   ALL=(ALL:ALL) ALL
duzun  ALL=(ALL) NOPASSWD: /usr/local/bin/monitoring.sh
# See sudoers(5) for more information on "@include" directives:
@includedir /etc/sudoers.d
```

4. Yeniden Başlatın.

sudo reboot

5. Komut dosyasını şu şekilde yürütün:

sudo bash /usr/local/bin/monitoring.sh

6. Aşağıdaki komut crontab'ı açın ve cron'u kök olarak ayarlayan kuralı ekleyin.

sudo crontab -u root -e (root yetkisi ile crontab ayarı ben bunu kullandım)

sudo crontab -e (kullanıcı bazlı crontab ayarı)

```
# 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
```

Bir kabuk betığını her 10 dakikada bir çalışacak şekilde programlamak için aşağıdaki satırı değiştirin.

İlk hali

m h dom mon dow command

Son hali

***/10 * * * * bash /usr/local/bin/monitoring.sh**

m h dom mon dow command

***/10 * * * * bash /usr/local/bin/monitoring.sh**

crontab ayrıntıları, bir komutun nasıl formüle edileceği:

Bir **crontab** komutu, tek bir satırla temsil edilir. \ Bir komutu birden çok satırda yazmak için kullanamazsınız. Karma (#) işaretleri, bu satırda herhangi bir şeyin **cron** tarafından yok sayıldığı anlamına gelen bir yorumu temsil eder. Baştaki boşluk ve boş satırlar yok sayılır.

(%) Komutunuzda yüzde işaretini kullanırken çok dikkatli olun. Kaçmadıkları sürece \% yeni satırlara dönüştürülürler ve kaçmayan ilk satırdan sonraki her şey % değişkenindeki komutunuza ilettilir.

Savunmada sorulacak **crontab** servisini durdurma ve yeniden başlatma:

Durdurmak için:

sudo service cron stop

Başlatmak için:

sudo service cron start

crontab dosyaları için iki biçim vardır:

Kullanıcı crontab'ları

Example of job definition:

```
# ----- minute (0 - 59)
# | ----- hour (0 - 23)
# | | ----- day of month (1 - 31)
# | | | ----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | ----- day of week (0 - 6) (Sunday=0 or 7)
# | | | |
# * * * * * command to be executed
```

Sistem genelinde /etc/crontab ve /etc/cron.d parçalar

Example of job definition:

```
# ----- minute (0 - 59)
# | ----- hour (0 - 23)
# | | ----- day of month (1 - 31)
# | | | ----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | ----- day of week (0 - 6) (Sunday=0 or 7)
# | | | |
# * * * * * user-name command to be executed
```

İkincisinin bir kullanıcı adı gerektirdiğine dikkat edin. Komut, adlandırılmış kullanıcı olarak çalıştırılacaktır.

Satırın ilk 5 alanı, komutun çalıştırılması gereken zaman(ları) temsil eder. Zaman belirtiminde sayıları veya uygun olduğunda gün/ay adlarını kullanabilirsiniz.

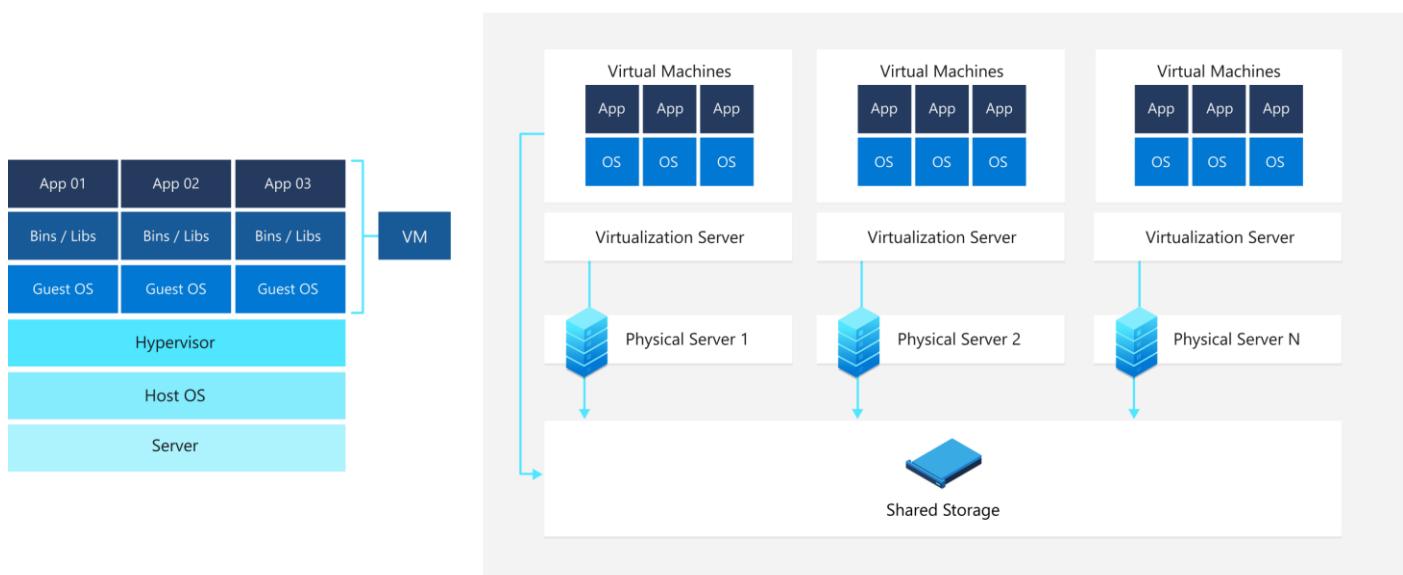
- Alanlar boşluk veya sekmelerle ayrılır.
- Bir listeyi belirtmek için virgül (,) kullanılır, örneğin 1,4,6,8, yani 1,4,6,8'de çalıştırılır.
- Aralıklar bir tire (-) ile belirtilir ve listelerle birleştirilebilir, örneğin 1-3,9-12 yani 1 ile 3 ve ardından 9 ile 12 arasında anlamına gelir.
- Karakter /bir adımı tanıtmak için kullanılabilir, örneğin 2/5'ten başlamak, ardından her 5'te bir (2,7,12,17,22...). Sonunu sarmazlar.
- Bir alandaki yıldız işaretü (*), o alan için tüm aralığı belirtir (örneğin 0-59, dakika alanı için).
- Aralıklar ve adımlar birleştirilebilir, örneğin */2 ilgili alan için minimumdan başlamayı ve ardından her 2'de bir, örneğin dakikalar için 0 (0,2...58), aylar için 1 (1,3 ... 11) vb.

Sanal Makine nasıl çalışır?

Sanal Makineler, donanımdan ayrılmış katmanda bir bilgisayarın sanalörneğini çalıştırırlar. Bu işlem sanallaştırma (virtualization) teknigi sayesinde mümkündür. Makinenin kaynaklarının sanallaştırılma işlemi, VM'leri oluşturan, barındıran ve çalıştırılan hipervizör (Hypervisor) adı verilen bir yazılım tarafından gerçekleştirilir.

Sanal makineler: bilgisayarlar içindeki sanal bilgisayarlar

Genellikle yalnızca VM olarak kısaltan bir sanal makine, dizüstü bilgisayar, akıllı telefon veya sunucu gibi diğer fizikal bilgisayardan farklı değildir. Dosyalarınızı depolamak için bir CPU, bellek ve disklere sahiptir ve gerekirse internete bağlanabilir. Bilgisayarınızı oluşturan parçalar (donanım) fiziks ve somut olduğundan, VM'ler genellikle fiziks sunucular içinde yalnızca kod olarak bulunan sanal bilgisayarlar veya yazılım tanımlı bilgisayarlar olarak düşünülebilir.



Bir sanal makine nasıl çalışır?

Sanallaştırma, fiziks bir konak bilgisayardan (örneğin kendi kişisel bilgisayarınız) ve/veya uzak bir sunucudan (örneğin bir bulut sağlayıcısının veri merkezindeki bir sunucu) “ödünç alınan” ayrılmış miktarlarda CPU, bellek ve depolama ile bir bilgisayarın yazılım tabanlı veya “sanal” bir sürümünü oluşturma işlemidir. Sanal makine, gerçek bir bilgisayar gibi hareket eden bir bilgisayar dosyasıdır. Bu dosyaya görüntü adı verilir. Birçok kişinin iş bilgisayarında yaygın olduğu gibi, bir pencerede, genellikle farklı bir işletim sistemini çalıştırmak için veya kullanıcının tüm bilgisayar deneyimi olarak işlev görmesini sağlamak üzere ayrı bir bilgi işlem ortamı olarak çalışabilir. Sanal makine, sistemin geri kalanından bölümlenmiştir, yani bir VM içindeki yazılım, ana bilgisayarın birincil işletim sistemine müdahale etmez.

VM'ler ne amaçla kullanılır?

Aşağıda, sanal makinelerin kullanıldığı birkaç örnek verilmiştir:

- Uygulamaları bulutta derleme ve dağıtma.
- Beta sürümleri dahil olmak üzere yeni bir işletim sistemi (OS) deneme.
- Geliştiricilerin geliştirme-test senaryolarını çalıştırmasını kolaylaştırmak ve hızlandırmak için yeni bir ortam oluşturma.
- Mevcut işletim sisteminizi yedekleme.
- Virüs bulaşmış verilere erişme veya eski bir işletim sisteminin yükleyerek eski bir uygulamayı çalıştırma.
- Yazılım veya uygulamaları, başlangıcta amaçlanmadıkları işletim sistemlerinde çalışma.

VM'leri kullanmanın avantajları nelerdir?

Sanal makineler esneklikleri ve taşınabilirlikleri nedeniyle birçok avantaj sağlar, örneğin:

- Maliyet tasarrufu:** Tek bir altyapı parçasından birden çok sanal ortam çalışırmak, fiziksel altyapı ayak izinizi önemli ölçüde azaltabileceğiniz anlamına gelir. Bu, neredeyse aynı sayıda sunucuyu koruma ihtiyacını azaltıp bakım maliyetlerinden ve elektrikten tasarruf sağlayarak karınızı artırır.
- Çeviklik ve hız:** Bir VM'yi kurmak nispeten kolay ve hızlıdır. Ayrıca geliştiricileriniz için tamamen yeni bir ortam sağlamaktan çok daha basittir. Sanallaştırma, geliştirme ve test senaryolarını çok daha hızlı çalışma sürecini kolaylaştırır.
- Azaltılmış kapalı kalma süresi:** VM'ler son derece taşınabilirdir ve farklı bir makinede bir hiper yöneticiden diğerine taşınması kolaydır. Bu, konağın beklenmedik bir şekilde kapanması durumunda yedekleme için ideal bir çözüm oldukları anlamına gelir.
- Ölçeklenebilirlik:** Sanal makineler, birden fazla VM arasında iş yükünü dağıtmak için daha fazla fiziksel veya sanal sunucu ekleyerek uygulamalarınızı daha kolay ölçeklendirmenize olanak tanır. Bunun sonucunda, uygulamalarınızın kullanılabilirliğini ve performansını artırabilirsiniz.
- Güvenlik avantajları:** Sanal makineler birden çok işletim sisteminde çalıştığından, bir VM'de konuk işletim sisteminin kullanılması güvenliği şüpheli uygulamaları çalıştırmanıza ve ana bilgisayar işletim sisteminizi korumanıza olanak sağlar. VM'ler ayrıca daha iyi güvenlik araştırması sağlar, genellikle bilgisayar virüslerini güvenli bir şekilde incelemek ve konakları riske atmaktan kaçınmak için virüsleri izole etmek amacıyla kullanılır.

CentOS ve Debian Arasındaki Temel Farklar

| parametreler | CentOS | Debian |
|----------------------|--|--|
| Toplum | Red Hat topluluğu tarafından desteklenmektedir. | Debian bireyleri tarafından desteklenir. |
| Pazar Varlığı | CentOS, kullanıcı dostu yapısı nedeniyle büyük bir pazar sahiptir. | Debian, terminal son kullanımı nedeniyle pazar varlığından yoksundur. |
| Mimari Destek | CentOS, çoklu mimari desteği ile gelmiyor. | Debian, diğer dağıtımlara kıyasla çoklu mimari destegine sahiptir. |
| Yayın Dönüşü | Yeni güncellemeler ve yükseltmeler genellikle zaman alır ve bu nedenle kararlı hale gelir. | 2 yıllık bir sürüm döngüsüne sahiptir, bu nedenle hataları düzeltmesi için yeterli zaman sağlar. |
| Sürüm | Eski sürümü yükseltmek yerine yeni bir CentOS sürümü yüklemek daha iyidir. Bu görev zahmetlidir. | Debian, bir kararlı sürümünden diğerine kolayca yükseltebilir. |

| | | |
|-----------------------------|---|--|
| Kullanıcı arayüzü | CentOS'un karmaşık bir GUI'si var. | Debian, kullanıcı dostu uygulamalar ve GUI ile birlikte gelir. |
| Paketleme yöneticisi | CentOS, paket yöneticisi olarak YUM'u kullanır. | Debian, paket yöneticisi olarak apt-get'i kullanır. |
| Ambalaj numarası | CentOS'un sınırlı paketleri vardır. | Debian, varsayılan deposunda çok sayıda pakete sahiptir. |

Apt ve Aptitude nedir? ve Aralarındaki Gerçek Fark Nedir?

Aptitude ve apt-get , paket yönetimini idare eden popüler araçlardan ikisidir. Her ikisi de, kurulum, kaldırma, arama vb. dahil olmak üzere paketlerdeki her türlü etkinliği gerçekleştirebilir.

Apt nedir

Apt veya Advanced Packaging Tool, yazılım yükleme ve kaldırma işlemlerini incelikle işleyen ücretsiz ve açık kaynaklı bir yazılımdır. Başlangıçta Debian'in .deb paketleri için tasarlanmıştır ancak RPM Paket Yöneticisi ile uyumlu hale getirilmiştir.

(RPM = RedHat Package Manager yani Redhat Paket yöneticisi anlamına gelmektedir.)

Apt , GUI'siz tüm komut satırıdır. Komut satırından yüklenecek paketin adını belirterek her çağrılığında, bu paketi, '/etc/apt/sources.list' de belirtilen kaynakların yapılandırılmış listesinde ve paketin bağımlılık listesiyle bulur ve sıralar ve mevcut paketle birlikte bunları otomatik olarak yükler, böylece kullanıcının bağımlılıkları yükleme konusunda endişelenmemesine izin verir.

Kullanıcının çeşitli konfigürasyonları kolayca kontrol etmesini sağlayan son derece esnektir: paketleri aramak için herhangi bir yeni kaynak eklemek, uygun şekilde sabitlemek, yani sistem yükseltme sırasında mevcut olmayan herhangi bir paketi işaretlemek ve böylece mevcut versiyonunun son sürümü kurulu olması "aklılı" yükseltme yani en önemli paketleri yükseltme ve en önemsiz paketleri bırakma.

Aptitude Nedir?

Aptitude, işlevselligi bir kullanıcı arabirimini ekleyen, böylece bir kullanıcının etkileşimli olarak bir paket aramasına ve yüklemesine veya kaldırmasına izin veren gelişmiş paketleme aracının ön ucudur. İlk olarak Debain için oluşturulan Aptitude, işlevselligini RPM tabanlı dağıtımlara da genişletiyor.

Kullanıcı arayüzü, GUI'lerde yaygın olarak görülen çeşitli öğeleri ekleyen ncurses kütüphanesine dayanmaktadır. En önemlilerinden biri, apt-get komut satırı argümanlarının çoğunu taklit edebilmesidir.

Genel olarak, Aptitude , düşük düzeyli ayrıntıları özetleyen ve hem metin tabanlı etkileşimli UI modunda hem de komut satırı etkileşimli olmayan modda çalışabilen daha üst düzey bir paket yöneticisidir.

APT ve Aptitude Arasındaki Farklar Nelerdir?

Aptitude'un üst düzey bir paket yöneticisi olmasının yanı sıra, APT diğer üst düzey paket yöneticileri tarafından kullanılabilen alt düzey bir paket yöneticisi olmakla birlikte, bu iki paket yöneticisini ayıran diğer önemli özellikler şunlardır:

Aptitude işlevselligi apt- get'den daha genişdir ve apt-get ve apt-mark ve apt-cache gibi diğer varyantlarının işlevlerini bütünlüğe getirir.

Apt-get tüm paket kurulumunu, yükseltme, sistem yükseltme, boşaltma paketi, bağımlılıkları çözme vb. İşlemlerini gerçekleştirirken, Aptitude apt-mark ve apt-cache işlevlerini içeren, yani apt-cache işlevlerini içeren bir çok şeyi işler. kurulu paketlerin listesi, otomatik veya manuel olarak yüklenecek bir paketi işaretleme, bir paketi yukarı derecelendirme için kullanılamaz hale getirme vb.

Apt-get kullanıcı arayüzünden yoksun olsa da, Aptitude'un salt metin ve etkileşimli bir kullanıcı arayüzü vardır

Apt-get'in daha düşük seviye bir paket yönetici olması sadece komut satırı ile sınırlıdır, Aptitude ise daha yüksek seviye bir araç olmak için gerekli komutları girerek komut satırı işlemi seçeneğinin yanı sıra varsayılan bir salt metin etkileşimli arayüze sahiptir.

Aptitude, apt-get'den daha iyi bir paket yönetimine sahiptir

Paketler için kurulum, kaldırma ve çakışma çözümlemesini içeren birçok durumda, Aptitude apt-get yerine değerini kanıtlar. Bazı durumlar şunları içerir:

1. Yüklü herhangi bir paketi kaldırırken, Aptitude kullanılmayan paketleri otomatik olarak silerken , apt-get kullanıcının ek '**—auto-remove**' seçeneğini ekleyerek veya '**apt-get autoremove**' belirterek bunu açıkça belirtmesini '**apt-get autoremove**'

2. Belirli eylemin neden engellendiğini veya belirli bir eylemin neden ya da neden yapılmaması gerektiğini araştırmak için Aptitude 'why' ve 'neden olmasın' komutları sunar.

SELinux ve APPArmor arasındaki fark?

"Bu güvenlik sistemleri, uygulamaları birbirinden yalıtmak için araçlar sağlar ve bir uygulamanın güvenliği ihlal edildiğinde bir saldırganı sistemin geri kalanından yalıtır.

SELinux kural kümeleri inanılmaz derecede karmaşıktır ancak bu karmaşıklıkla süreçlerin nasıl izole edildiği üzerinde daha fazla kontrol sahip olursunuz. Bu ilkelerin oluşturulması otomatikleştirilebilir. Bu güvenlik sistemine karşı bir grep, bağımsız olarak doğrulanmanın çok zor olmasıdır.

AppArmor (ve SMACK) çok basittir. Profiller insanlar tarafından elle yazılabilir veya kullanılarak oluşturulabilir aa-logprof. AppArmor, bağımsız olarak doğrulanabilmesi için sistemi daha şeffaf hale getiren yola dayalı kontrol kullanır."

BONUS KURULUMLARI

Debian Lighttpd (Web sunucusu) Nasıl Kurulur

lighttpd, doğrudan **Debian** üzerindeki resmi paket kaynaklarından kurulabilir, bu nedenle **Apache** gibi herhangi bir üçüncü taraf deposuna ihtiyacımız yok. Bu web sunucusunu kurmak için verilen tek komutu çalıştırın.

sudo apt install lighttpd -y

Kurulum tamamlandıktan sonra **lighttpd** sunucusunun çalışıp çalışmadığını kontrol edelim.

sudo systemctl status lighttpd

```
duzun@duzun42 ~
$ sudo systemctl status lighttpd
● lighttpd.service - Lighttpd Daemon
   Loaded: loaded (/lib/systemd/system/lighttpd.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2022-04-07 11:32:47 EDT; 3min 44s ago
     Process: 2500 ExecStartPre=/usr/sbin/lighttpd -tt -f /etc/lighttpd/lighttpd.conf (code=exited, ▶
   Main PID: 2505 (lighttpd)
      Tasks: 1 (limit: 1128)
     Memory: 1.0M
        CPU: 172ms
       CGroup: /system.slice/lighttpd.service
               └─2505 /usr/sbin/lighttpd -D -f /etc/lighttpd/lighttpd.conf

Apr 07 11:32:47 duzun42 systemd[1]: Starting Lighttpd Daemon...
Apr 07 11:32:47 duzun42 systemd[1]: Started Lighttpd Daemon.
duzun@duzun42 ~
$ _
```

Sunucu aktif ve çalışıyor.

Sunucumuzun http portu olan **80**. Portu **ufw** güvenlik duvarından izin verelim

sudo ufw allow 80

Kontrol etmek için

sudo ufw status

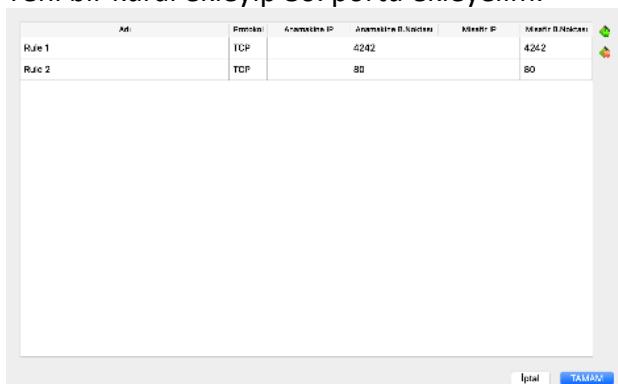
Cıktıyı inceleyin.

```
duzun@duzun42 ~
$ sudo ufw allow 80
Rule added
Rule added (v6)
duzun@duzun42 ~
$ sudo ufw status
Status: active

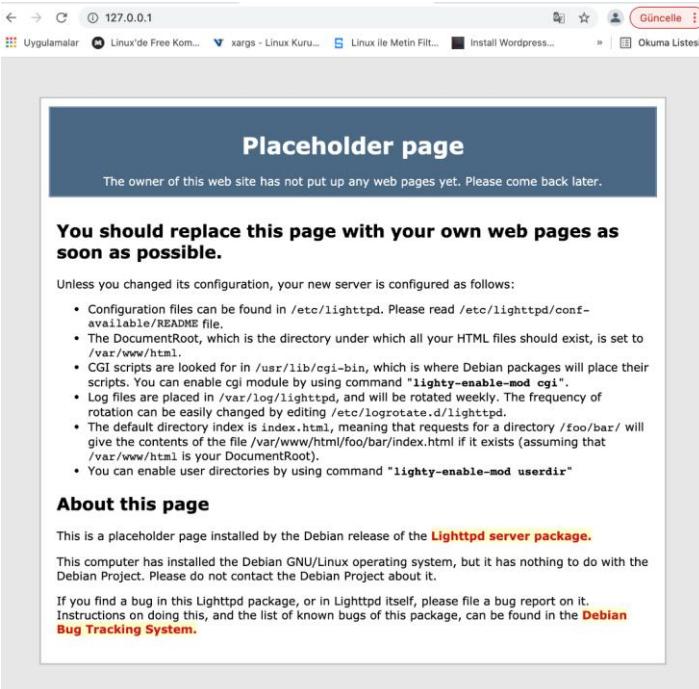
To                         Action      From
--                         --          --
4242                       ALLOW      Anywhere
80                        ALLOW      Anywhere
4242 (v6)                   ALLOW      Anywhere (v6)
80 (v6)                     ALLOW      Anywhere (v6)

duzun@duzun42 ~
```

Şimdide **VirtualBox** üzerinden **80**. portu yönlendirilelim. Sırasıyla **Ayarlar/Ağ/B.noktası Yönlendirme** Yeni bir kural ekleyip **80**. portu ekleyelim.



Şimdi tarayıcımızdan **localhost 127.0.0.1** adresine girelim. İlk kurulum sayfası aşağıdaki gibi açılıyorsa her şey yolundadır. **Lighttpd** web sunucumuz kuruldu ve çalışıyor. Şimdi kendi web sayfamızı oluşturup **/var/www/html** dizini içerisine yerleştirip yayına sunalım.



WordPress Kurulumu

WordPress'i Sıfırdan Yükleme

WordPress'i Debian bulut sunucunuza kurmak için **WordPress** organizasyonunun önerdiği temel "5 dakikalık kurulum" ile birlikte takip edeceksiniz. Ancak, bu **WordPress** kendi özel bulut sunucunuza kurulacağından, birkaç ek sunucu tarafı yazılım uygulaması yüklemeniz gerekecektir. Ama temelde süreç şu şekilde işliyor:

- **PHP'yi sunucu tarafı betik dili olarak kurun.** Bu, WordPress'in yazıldığı dildir, bu nedenle bir gereklilikdir. PHP, WordPress kodunu yorumlayacak ve eksiksiz sayfaları tarayıcıda sunacak ve ayrıca veritabanı bağlantısını kolaylaşdıracaktır.
- **MariaDB'yi yükleyin.** WordPress içerik yönetim sistemi, içeriğinizi depolamak için bir veritabanı kullanır. Bu makalede, içeriğinizin depolanmasını ve alınmasını gerçekleştirecek ilişkisel veritabanı yazılımı olarak MariaDB'yi kuracaksınız.
- **WordPress çekirdek dosyaları.** WordPress çekirdek dosyalarını WordPress.org'dan indirmeniz gerekecek. Bunlar temel olarak WordPress uygulamasını çalıştırınca düz metin program dosyalarıdır.
- **Veritabanı bağlantısı.** Son olarak, çekirdek dosyalardan birini düzenleyerek çekirdek dosyaları veritabanına bağlamanız gerekecektir.

Tüm bu adımları aşağıda ayrıntılı olarak ele alacağız.

MariaDB'yi yükleyin

MariaDB'yi WordPress tarafından kullanılacak veritabanı depolama yazılımı olarak nasıl kuracağınızı öğrenmek için aşağıdakileri takip edin.

sudo apt install mariadb-server -y

Bu noktada, yeni MariaDB kurulumunuzun güvenliğini sağlamak için aşağıdaki komutu çalıştırmanız önerilir:
sudo mysql_secure_installation

Bu, 1. Adımda oluşturduğunuz **root** parolayı girmenizi isteyecektir. Şifreyi girip devam edin. Sonraki soket uyarısını da **N** seçeneği ile devam edelim. **Root** parolasını değiştirmek isteyip istemediğini soran soru yada **N** ile devam edelim, sonraki tüm sorular için varsayılanları kabul etmek için **Y** düğmesine basabilirsiniz.

```

duzun@duzun42 ~
$ sudo mysql_secure_installation
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none): [REDACTED]

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none): [REDACTED]

duzun@duzun42 ~
$ sudo mysql_secure_installation
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none): [REDACTED]

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
[REDACTED]

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] n
[REDACTED]

production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
duzun@duzun42 ~
$ [REDACTED]

```

MySQL'i Test Etme

Nasıl yüklediğinize bakılmaksızın, MySQL otomatik olarak çalışmaya başlamış olmalıdır. Bunu test etmek için durumunu kontrol edin. Aşağıdaki resimdeki gibi bir çıktı göreceksiniz.

`sudo systemctl status mysql.service`

```

duzun@duzun42 ~
$ sudo systemctl status mysql.service
● mysql.service - MariaDB 10.5.15 database server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-04-08 07:13:52 EDT; 1min 59s ago
     Docs: man:mysqld(8)
           https://mariadb.com/kb/en/library/systemd/
Process: 954 ExecStartPre=/usr/bin/mysql_install_privilege --user=mysql --socket=/var/run/mysqld/mysqld.sock
          -c systemctl unmask-environment _SYSTEMD_STARTPOSITION=28
Process: 955 ExecStartPre=/bin/sh -c /usr/bin/mysqld_safe --user=mysql &> /var/log/mysqld.log
          -c systemctl unmask-environment _SYSTEMD_STARTPOSITION=28
Process: 654 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/SUCCESS)
Main PID: 634 (mysqld)
  Status: "Taking your SQL requests now..."
   Tasks: 9 (limit: 1120)
  Memory: 102.1M
    CPU: 29ms
   CGroups: /system.slice/mysql.service
             └─ 634 /usr/sbin/mysqld

Apr 08 07:13:52 duzun42 mariadb[934]: Version: '10.5.15-MariaDB-0+deb11u1' socket: '/run/mysqld/mysqld.sock'
Apr 08 07:13:52 duzun42 systemd[1]: Started MariaDB 10.5.15 database server.
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[657]: Upgrading MySQL tables if necessary.
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[650]: Looking for 'mysql' as: /usr/bin/mysql
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[650]: Looking for 'mysqldump' as: /usr/bin/mysqldump
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[650]: This installation of MariaDB is already up-to-date.
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[650]: There is no need to run mysql_upgrade again.
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[650]: You can use --force if you still want to run.
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[655]: Checking for insecure root accounts.
Apr 08 07:13:52 duzun42 /etc/mysql/debian-start[655]: Triggering myisam-recover for all MyISAM tables.

duzun@duzun42 ~
$ [REDACTED]

```

MySQL çalışmıyorsa, aşağıdaki komut ile başlatılabilirsiniz.

`sudo systemctl start mysql`

PHP'yi yükleyin

Şimdi PHP betik dilini kurmanın zamanı geldi. Bu, **WordPress'in** sisteminizde düzgün çalışması için kullandığınız dildir. Sunucunuzda hala komut satırındayken, devam edin ve PHP'yi ve gerekli modülleri kurmak için bu komutu çalıştırın:

```
sudo apt install php7.4-fpm php7.4-common php7.4-mysql php7.4-xml php7.4-xmlrpc php7.4-curl  
php7.4-gd php7.4-imagick php7.4-cli php7.4-mbstring php7.4-mbstring php7.4-opcache php7.4-readline  
php7.4-soap php7.4-json php7.4-zip -y
```

Şimdi aşağıdaki komutu kullanarak PHP sürümünü kontrol edebilirsiniz:

```
php -v
```

```
[duzun@duzun42 ~]  
$ php -v  
PHP 7.4.28 (cli) (built: Feb 17 2022 16:17:19) ( NTS )  
Copyright (c) The PHP Group  
Zend Engine v3.4.0, Copyright (c) Zend Technologies  
    with Zend OPcache v7.4.28, Copyright (c), by Zend Technologies  
[duzun@duzun42 ~]  
$
```

PHP-FPM'yi Lighttpd ile Çalışacak Şekilde Yapılandırın

Ardından, PHP-FPM'yi Lighttpd ile çalışacak şekilde yapılandırmanız gerekecek. Bunu yapmak için www.conf dosyasını düzenleyin:

```
sudo vim /etc/php/7.4/fpm/pool.d/www.conf
```

Aşağıdaki satırı bulun:

```
listen = /run/php/php7.4-fpm.sock
```

Bu satırı aşağıdaki gibi değiştirin:

```
listen = 127.0.0.1:9000
```

Dosyayı kaydedip kapatın, ardından 15-fastcgi-php.conf dosyasını düzenleyin:

```
sudo vim /etc/lighttpd/conf-available/15-fastcgi-php.conf
```

Aşağıdaki satırları bulun:

```
"bin-path" => "/usr/bin/php-cgi",  
"socket" => "/var/run/lighttpd/php.socket",
```

Ve şu satırlar ile değiştirin:

```
"host" => "127.0.0.1",  
"port" => "9000",
```

Dosyayı kaydedip kapatın, ardından aşağıdaki komutla gerekli modülleri etkinleştirin:

```
sudo lighty-enable-mod fastcgi
```

```
sudo lighty-enable-mod fastcgi-php
```

```
[duzun@duzun42 /var/www/html]  
$ sudo lighty-enable-mod fastcgi  
perl: warning: Setting locale failed.  
perl: warning: Please check that your locale settings:  
    LANGUAGE = (unset),  
    LC_ALL = (unset),  
    LC_CTYPE = "UTF-8",  
    LC_TERMINAL = "iTerm2",  
    LANG = "en_US.UTF-8"  
are supported and installed on your system.  
perl: warning: Falling back to a fallback locale ("en_US.UTF-8").  
already enabled  
Run "service lighttpd force-reload" to enable changes  
[duzun@duzun42 /var/www/html]  
$ sudo lighty-enable-mod fastcgi-php  
perl: warning: Setting locale failed.  
perl: warning: Please check that your locale settings:  
    LANGUAGE = (unset),  
    LC_ALL = (unset),  
    LC_CTYPE = "UTF-8",  
    LC_TERMINAL = "iTerm2",  
    LANG = "en_US.UTF-8"  
are supported and installed on your system.  
perl: warning: Falling back to a fallback locale ("en_US.UTF-8").  
Enabling Fastcgi-php: ok  
Run "service lighttpd force-reload" to enable changes  
[duzun@duzun42 /var/www/html]
```

Ardından, değişiklikleri uygulamak için **Lighttpd** ve PHP-FPM hizmetini yeniden başlatın:

```
sudo systemctl restart lighttpd
```

```
sudo systemctl restart php7.4-fpm
```

Aşağıdaki **php** kodlarının olduğu **phpinfo.php** dosyası oluşturup **/var/www/html** dizini içerisine yerleştirin ve tarayıcıdan **127.0.0.1/phpinfo.php** adresine gidiniz. Php nin düzgün çalışıp çalışmadığını kontrol edelim.

```
<?php  
phpinfo();  
phpinfo(INFO_MODULES);  
?>
```

WordPress için bir Veritabanı Oluşturun

Ardından, aşağıdaki komutla MariaDB'ye giriş yapın:

Mysql

Giriş yaptıktan sonra, aşağıdaki komutla bir veritabanı ve kullanıcı oluşturun:

```
CREATE DATABASE wordpress; (wordpress adında bir database oluşturur.)
```

```
GRANT ALL PRIVILEGES on wordpress.* TO 'admin'@'localhost' IDENTIFIED BY 'Kullanici42';
```

```
duzun@duzun42 ~  
└─$ sudo mysql  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 31  
Server version: 10.5.15-MariaDB-0+deb11u1 Debian 11  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement  
. .  
  
MariaDB [(none)]> CREATE DATABASE wordpress;  
Query OK, 1 row affected (0.001 sec)
```

Kendi çalışmaya göre database adı, kullanıcı adı ve şifreyi belirleyiniz.

Ardından, ayrıcalıkları temizleyin ve MariaDB'den çıkışın:

```
MariaDB [(none)]> GRANT ALL PRIVILEGES on wordpress.* TO 'admin'@'localhost'  
IDENTIFIED BY 'Kullanici42';  
Query OK, 0 rows affected (0.002 sec)  
  
MariaDB [(none)]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.001 sec)  
  
MariaDB [(none)]> EXIT;  
Bye  
duzun@duzun42 ~  
└─$
```

WordPress Kurulumunu

Yükleme işleminin bu son aşamasında, WordPress kaynak dosyalarının yeni bir kopyasını indirecek, bunları veritabanınıza bağlayacak ve yepyeni bir WordPress sitesi oluşturmaya başlayacaksınız.

Bu eğitim için işletim sistemimizle birlikte gelen varsayılan dizini kullanacağımız: **/var/www/html/**
cd komutuyla varsayılan web köküne geçin:

```
cd /var/www/html
```

En son WordPress kaynak paketini aşağıdaki komutla indirebilirsiniz:

```
sudo wget http://wordpress.org/latest.tar.gz
```

Az önce sıkıştırılmış bir dosya çok benzeyen bir "arşiv" dosyası indirdiniz, bu, birden çok dosyayı sıkıştıran tek bir dosyadır. Böyle **tar** dosyasını aşağıdaki komutu kullanarak onu "arşivden çıkarabilirsiniz" :

```
sudo tar -xvf latest.tar.gz
```

Bu, içindeki tüm WordPress kaynak dosyalarının bulunduğu bir "wordpress/" dizini oluşturur. Ancak bu dosyaları dizinde istiyoruz, böylece WordPress dosyalarını bir dizine geri taşımak **/var/www/html/** için aşağıdaki komutu kullanabilir ve yeni "wordpress/" dizinini etkin bir şekilde boşaltabilirsiniz:

Wordpress dizinindeki dosyaları bulunduğuuz **/var/www/html/** dizinine taşımak için

```
sudo mv wordpress/* ./
```

Wordpress dizini silmek için

```
sudo rm -rf wordpress
```

Ayrıca varsayılan Debian Apache dizin dosyasını da silmek isteyeceksiniz:

```
sudo rm index.html
```

Şimdi Wordpress kurulumunu başlatmak için tarayıcıdan **127.0.0.1** adresine girelim ve ekran talimatlarına göre kurulumu yapalım.

Aşağıda veritabanı bağlantı detaylarınızı girmelisiniz. Bunlardan emin değilseniz, sunucunuzla iletişime geçin.

| | | |
|----------------------------|-------------|---|
| Veri tabanı ismi | wordpress | WordPress ile kullanmak istediğiniz veritabanının adı. |
| Kullanıcı adı | admin | Veritabanı kullanıcı adınız. |
| Parola | Kullanici42 | Veritabanı şifreniz. |
| Veritabanı Ana Bilgisayarı | localhost | localhost'a yaramazsa, bu bilgiyi web sunucusundan alabilmek gereklidir. |
| Tablo öneki | wp | Tek bir veritabanında birden fazla WordPress kurulumu çalıştırma istiyorsanız bunu değiştirebilirsiniz. |

Hadi gidelim! **Göndermek**

wp-config.php Dosyaya yazılamıyor Uyarısı alırsınız.

Dosyayı manuel olarak oluşturabilir **wp-config.php** ve verilen metni içine yapıştırıp kaydediyoruz.

sudo vim wp-config.php

Dosyayı kaydettikten sonra kuruluma devam ediyoruz.

Hoş geldin

Ünlü beş dakikalık WordPress kurulum sürecine hoş geldiniz! Sadece aşağıdaki bilgileri doldurun ve dünyadan en genişletilebilir ve güçlü kişisel yaymcılık platformunu kullanma yolunda olacaksınız.

Bilgi gereklisi

Lütfen aşağıdaki bilgileri girin. Endişelenmeyin, bu ayarları daha sonra istediğiniz zaman değiştirebilirsiniz.

| | |
|--------------------------|---|
| Site Başlığı | duzun 42 Ekol Test Sitesi |
| Kullanıcı adı | admin |
| Parola | Kullanici42 Güçlü |
| E-posta adresiniz | duzun@42.fr |
| Arama Motor Görünürülüğü | <input type="checkbox"/> Arama motorlarının bu siteyi dizine eklemesini engelleinyin Bu isteği yerine getirmek arama motorlarının görevidir. |

WordPress'i yükleyin

Kullanıcı Adı veya Eposta Adresi
admin

Parola
••••••••••••••••

Beni Hatırla **Giriş yapmak**

Şifreni mi unuttun?
← duzun 42 Ekol Test Sitesi'ne gidin

Başarı!

WordPress kuruldu. Teşekkürler ve keyfini çıkarın!

Kullanıcı adı yönetici

Parola Seçtiğiniz şifre.

Giriş yapmak

Web Sitemizin Admin Panel Görüntüsü

The screenshot shows the WordPress Admin Dashboard. At the top, there's a banner for 'Welcome to WordPress!' with the version number '5.9'. Below the banner, there are three main sections: 'Author rich content with blocks and patterns', 'Customize your entire site with block themes', and 'Switch up your site's look & feel with Styles'. Each section includes a brief description and a link to learn more or edit styles. At the bottom of the dashboard, there are two status bars: 'Site Health Status' and 'Quick Draft'.

Web Sitemizin Görünümü

The screenshot shows the website homepage. The title 'duzun 42 Ekol Test Sitesi' is at the top left, and 'Sample Page' is at the top right. The main content area features a black background with a white dashed arc and a colorful hummingbird in flight.

Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

April 8, 2022

Lighttpd web sunucusu üzerine wordpress kurulumunu tamamlamış olduk.

Yeni bir FTP kullanıcısı oluşturmak için aşağıdaki komutu kullanın:

sudo adduser ftpuser

Kullanıcıyı izin verilen FTP kullanıcıları listesine ekleyin:

echo "ftpuser" | sudo tee -a /etc/vsftpd.userlist

Tee Komutu ve Kullanımı

Tee komutu ile Standart girdiyi verilen dosyalara ve ayrıca standart çıktıya kopyalar yani komutların çıktılarını hem görüntüleyip hem de log dosyalarına yazdırabiliriz. Böyle bir komuta ihtiyaç olmasının sebebi komutların çıktılarını hem log dosyasına yazılıp hem de ekranda görüntülenemiyor olmasıdır.

Örneğin aşağıdaki gibi bir kullanımdan sonra ekranda görüntüyü göremeyiz.

ls > stdout.log

Standard Output (Stdout) olarak nitelendirdiğimiz verileri tee komutunu kullanarak aşağıdaki gibi görebiliriz.

ls | tee stdout.log

Tee komutuyla yapabileceğimiz diğer önemli işlem ise aynı anda birden fazla log dosyasına yazabilmektir. Bu önemlidir çünkü scriptlerimiz veya komutlarımızı içerisinde birden fazla farklı yerle etkileşime girebiliriz. Bu yüzden aynı sonucu hem örnek olarak abc.log dosyasına hem de efg.log dosyasına yazmamız gerekebilir. Bu yüzden aşağıdaki ifadeyi kullanabiliriz.

ls | tee dosya1 dosya2 dosya3

tee [seçenek]... [dosya]...

seçenekler:

- **-a –append** :dosyaların üzerine yazmaz sonuna ekler.
- **-i –ignore-interrupts** : Kesme sinyallerini yok sayar.
- **–help** :Yardım iletisini görüntüler ve çıkar.
- **–version** : Sürüm bilgilerini görüntüler ve çıkar.

Başlangıç olarak vsftpd'yi etkinleştirerek Terminal'de aşağıdaki komutu çalıştırın:

sudo systemctl enable vsftpd

FTP Erişimini Test Edin

FTP sunucusunu yalnızca "ftpuser" kullanıcısına erişime izin verecek şekilde yapılandırdık. Şimdi test etmek için bir FTP istemcisine ihtiyacınız olacak. Burada FTP istemcisi olarak Filezilla'yi kullanabiliriz ancak VirtualBox üzerinden sorunlar yaşamamız mümkündür. Savunmamızı yaparken terminale ftp programını kurarak yapmak yeterli olacaktır.

Ftp'yi aşağıdaki komut ile kurun.

sudo apt-get install ftp -y

Şimdi FTP sunucumuza ftp ile bağlanalım. (local sunucu ip adresimiz ile bağlanıyoruz)

ftp 127.0.0.1

ftp kullanıcı adı ardından şifreyi giriyoruz. Bilgileri düzgün girdiğimizde sunucuya bağlanmış oluyoruz.

Aşağıdaki görseli inceleyin.

```
[duzun@duzun42 ~]$ ftp 127.0.0.1
Connected to 127.0.0.1.
220 (vsFTPD 3.0.3)
Name (127.0.0.1:duzun): ftpuser
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

SON.