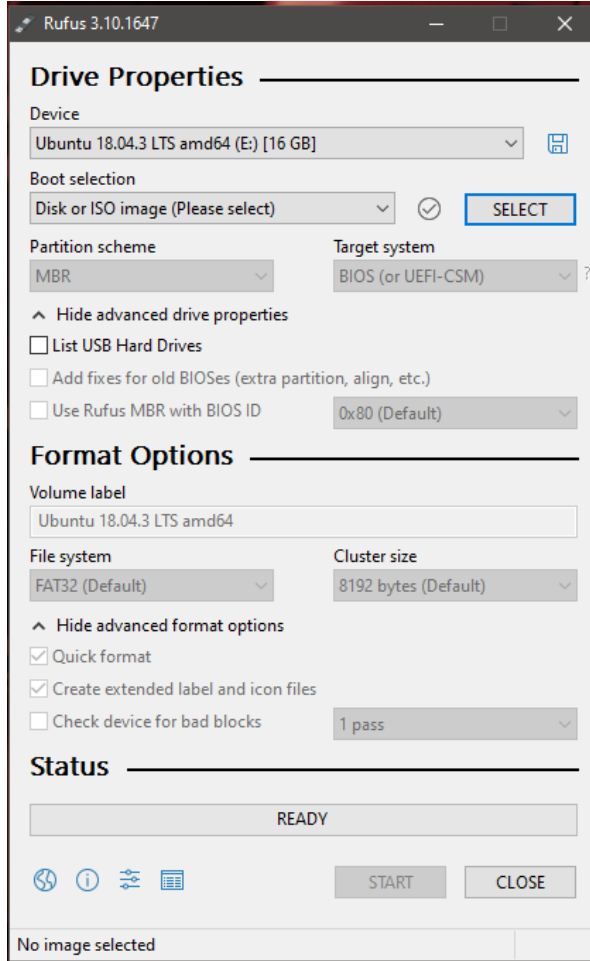


Ubuntu Kurulumu

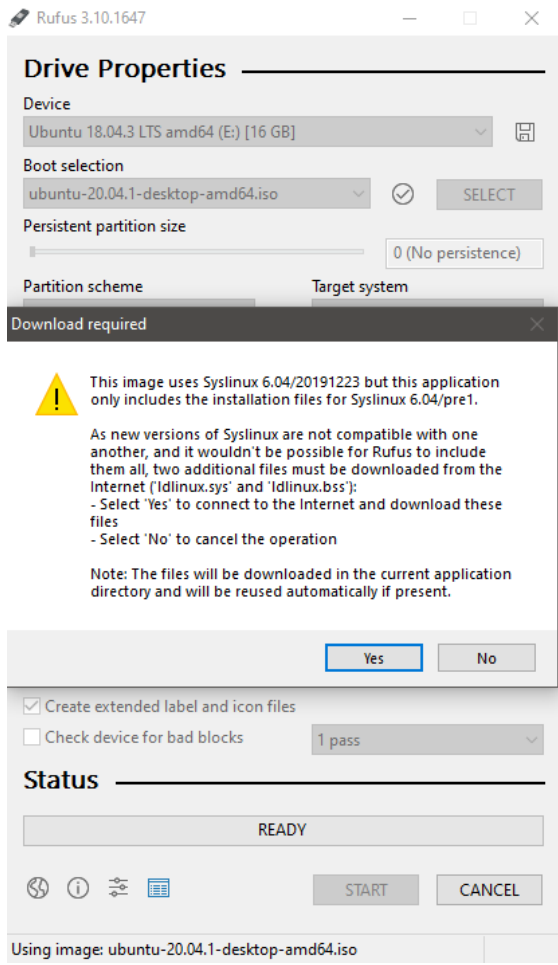
1-) <https://ubuntu.com/download/desktop> adresinden istediğiniz sürümün iso dosyası indirilir. Bu proje için Ubuntu 20.04.1 LTS kullanıldı.

2-) İndirilen iso dosyasını USB'ye yazmak için <https://rufus.ie> adresinden Rufus indirilir.

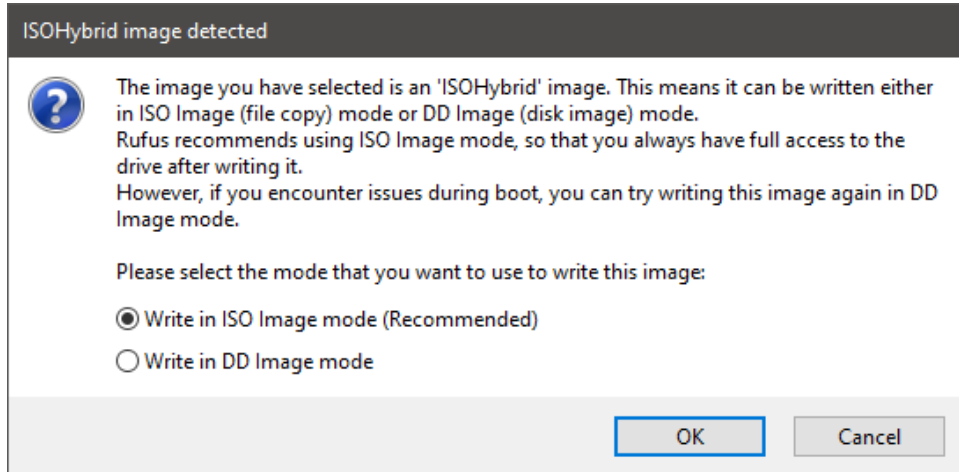
3-) Rufus arayüzünde Device kısmında USB belleğinizin gözüktüğünden ve Boot Selection kısmından "Disk or ISO Image" seçili olduğundan emin olun. Sonra SELECT'e tıklayarak indirdiğiniz ubuntu iso dosyasını seçin.



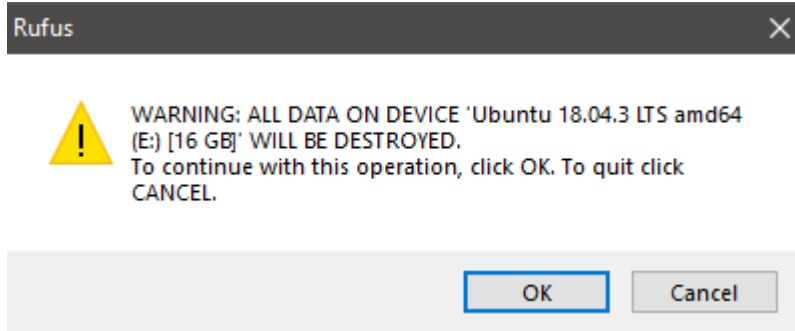
NOT: Aşağıdaki gibi Download Required uyarısı gelirse "Yes" diyerek gereken dosyaları indirin.



NOT 2: Önerilen (Recommended) modu seçip devam edin.

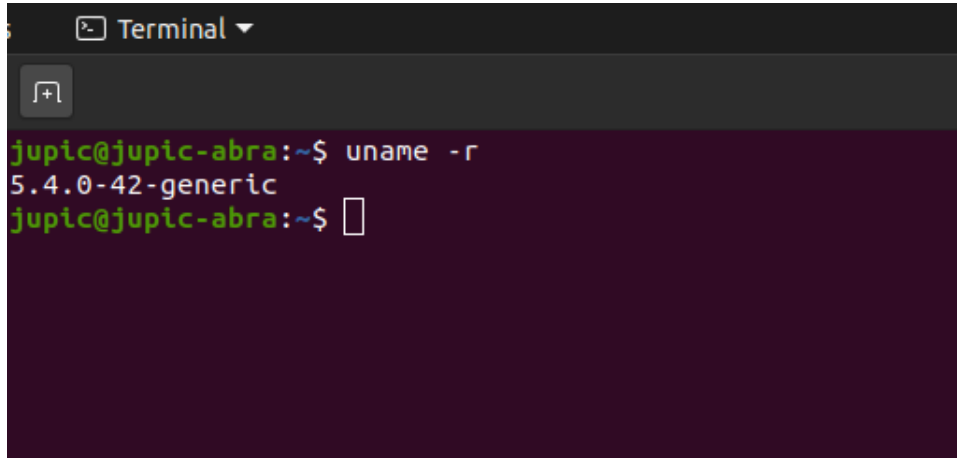


NOT 3: Rufus USB'ye yazmadan önce USB içindeki tüm verilerin silineceğini söyler. USB içinde önemli verileriniz var ise yedeklemeyi unutmayın. Yedekleme yaptıktan sonra OK diyerek USB hazırlığını başlatın.



Kernel Update Nasıl Yapılır?

1-) Terminali açıp “uname -r” komutu ile şuan kullandığınız kernel sürümünü görüntüleyin. Bende 5.4.0 sürümü kullanılıyor.

A screenshot of a terminal window. The title bar says 'Terminal'. The prompt is 'jupic@jupic-abra:~\$'. The command 'uname -r' has been entered, and the output '5.4.0-42-generic' is displayed. The prompt is now 'jupic@jupic-abra:~\$' followed by a cursor.

2-) <https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.9.2/> adresinden ubuntu için derlenmiş .deb dosyalarını indirin.

NOT: Bu dosyaları kernel.org üzerinden indirip kendi bilgisayarınızda da derleyerek .deb dosyalarını oluşturabilirsiniz.

3-) Terminalden “cd Downloads” klasörüne geçin. (Dosyaları indirdiğiniz klasör.)

4-) “sudo dpkg -i *.deb” komutu ile downloads klasöründeki tüm .deb dosyalarını yükleyin.

5-) “sudo reboot” ile bilgisayarı yeniden başlatın.

6-) “uname -r” komutu ile tekrardan kernel sürümünü görüntüleyerek kernel’in güncellemesini kontrol edin. (5.4.0 ‘dan 5.9.2 sürümüne güncellendi.)

```
jupic@jupic-abra: ~  
jupic@jupic-abra:~$ uname -r  
5.9.2-050902-lowlatency  
jupic@jupic-abra:~$
```

Eğer kendi cihazınızda derleyerek deb dosyalarını oluşturmak istiyorsanız:

- 1-) "cd /indirme-yapmak-istediğiniz-dosya-yolu" komutu ile indirme yapacağınız klasöre gidin.
- 2-) "wget <https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.9.2.tar.xz>" komutu ile linux-5.9.2 sürümlü kernel tar dosyasını indirin. Tar dosyası linux işletim sistemlerinde kullanılan zip veya rar dosyası gibi sıkıştırılmış dosya formatıdır.
- 3-) "unxz -v linux-5.9.2.tar.xz" ile indirdiğiniz tar dosyasını çıkartın.

```
jupic@jupic-abra: /media/jupic/D/ubuntu-kernel$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.9.2.tar.xz  
--2020-10-29 22:31:56-- https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.9.2.tar.xz  
Resolving cdn.kernel.org (cdn.kernel.org)... 151.101.241.176, 2a04:4e42:1b::432  
Connecting to cdn.kernel.org (cdn.kernel.org)[151.101.241.176]:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 115519708 (110M) [application/x-xz]  
Saving to: 'linux-5.9.2.tar.xz'  
  
linux-5.9.2.tar.xz 100%[=====]  
2020-10-29 22:33:27 (1,22 MB/s) - 'linux-5.9.2.tar.xz' saved [115519708/115519708]  
  
jupic@jupic-abra: /media/jupic/D/ubuntu-kernel$ unxz -v linux-5.9.2.tar.xz  
linux-5.9.2.tar.xz (1/1)  
100 % 110,2 MiB / 964,3 MiB = 0,114 55 MiB/s 0:17  
jupic@jupic-abra: /media/jupic/D/ubuntu-kernel$
```

4-) “wget <https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.9.2.tar.sign>” komutu ile imza dosyasını indirin.

```
jupic@jupic-abra:/media/jupic/D/ubuntu-kernel$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.9.2.tar.sign
--2020-10-29 22:35:21-- https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.9.2.tar.sign
Resolving cdn.kernel.org (cdn.kernel.org)... 151.101.113.176, 2a04:4e42:39::432
Connecting to cdn.kernel.org (cdn.kernel.org)[151.101.113.176]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 987 [text/plain]
Saving to: 'linux-5.9.2.tar.sign'

linux-5.9.2.tar.sign                                100%[=====]
2020-10-29 22:35:22 (9,78 MB/s) - 'linux-5.9.2.tar.sign' saved [987/987]

jupic@jupic-abra:/media/jupic/D/ubuntu-kernel$
```

5-) “gpg --verify linux-5.9.2.tar.sign” komutu ile indirdiğiniz tar dosyasının imzasını kontrol edin. Eğer “No Public Key” yazıyorsa “RSA key” ‘i kopyalayın.

```
jupic@jupic-abra:/media/jupic/D/ubuntu-kernel$ gpg --verify linux-5.9.2.tar.sign
gpg: assuming signed data in 'linux-5.9.2.tar'
gpg: Signature made Prş 29 Eki 2020 12:15:22 +03
gpg:                using RSA key 647F28654894E3BD457199BE38DBDC86092693E
gpg: Can't check signature: No public key
```

6-) “gpg --keyserver keyserver.ubuntu.com --recv-keys /sizin-RSA-key-numaranız” kodu ile keyserver’den sizin RSA key’inize karşılık gelen sign dosyasını bulup import edin.

```
jupic@jupic-abra:/media/jupic/D/ubuntu-kernel$ gpg --keyserver keyserver.ubuntu.com --recv-keys 647F28654894E3BD457199BE38DBDC86092693E
gpg: key 38DBDC86092693E: public key "Greg Kroah-Hartman <gregkh@linuxfoundation.org>" imported
gpg: Total number processed: 1
gpg:                imported: 1
```

7-) Tekrar “gpg --verify linux-5.9.2.tar.sign” komutunu çalıştırdığınızda “Primary Key Fingerprint” gösteren bir output almanız gerek.

```
jupic@jupic-abra:/media/jupic/D/ubuntu-kernel$ gpg --verify linux-5.9.2.tar.sign
gpg: assuming signed data in 'linux-5.9.2.tar'
gpg: Signature made Prş 29 Eki 2020 12:15:22 +03
gpg:                using RSA key 647F28654894E3BD457199BE38DBDC86092693E
gpg: Good signature from "Greg Kroah-Hartman <gregkh@linuxfoundation.org>" [unknown]
gpg:                aka "Greg Kroah-Hartman <gregkh@kernel.org>" [unknown]
gpg:                aka "Greg Kroah-Hartman (Linux kernel stable release signing key) <greg@kroah.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:       There is no indication that the signature belongs to the owner.
Primary key fingerprint: 647F 2865 4894 E3BD 4571 99BE 38DB BDC8 6092 693E
jupic@jupic-abra:/media/jupic/D/ubuntu-kernel$
```

8-) Kernel build alınmadan önce Linux kernel özelliklerinin konfigüre edilmesi gerekir. Ayrıca işletim sisteminin hangi kernel modüllerinin (sürücülerinin) yükleneceğini de belirtmek gerekir. Bu işte yeni insanlar için mevcut konfigürasyonların kopyalanması bu aşamayı kolaylaştırır.

“cd linux-5.9.2” ile tardan çıkarttığınız klasöre geçip “cp -v /boot/config-\$(uname -r) .config” komutu ile mevcut config dosyanızı kopyalayın. Çıktı aşağıdaki gibi olmalıdır.

```
jupic@jupic-abra: ~/linux-5.9.2
jupic@jupic-abra:~$ cd linux-5.9.2
jupic@jupic-abra:~/linux-5.9.2$ cp -v /boot/config-$(uname -r) .config
'/boot/config-5.9.2-050902-lowlatency' -> '.config'
jupic@jupic-abra:~/linux-5.9.2$
```

9-) “sudo apt-get install build-essential libncurses-dev bison flex libssl-dev libelf-dev” komutu ile gerekli paketleri yükleyin.

```
jupic@jupic-abra:~/linux-5.9.2$ sudo apt-get install build-essential libncurses-dev bison flex libssl-dev libelf-dev
```

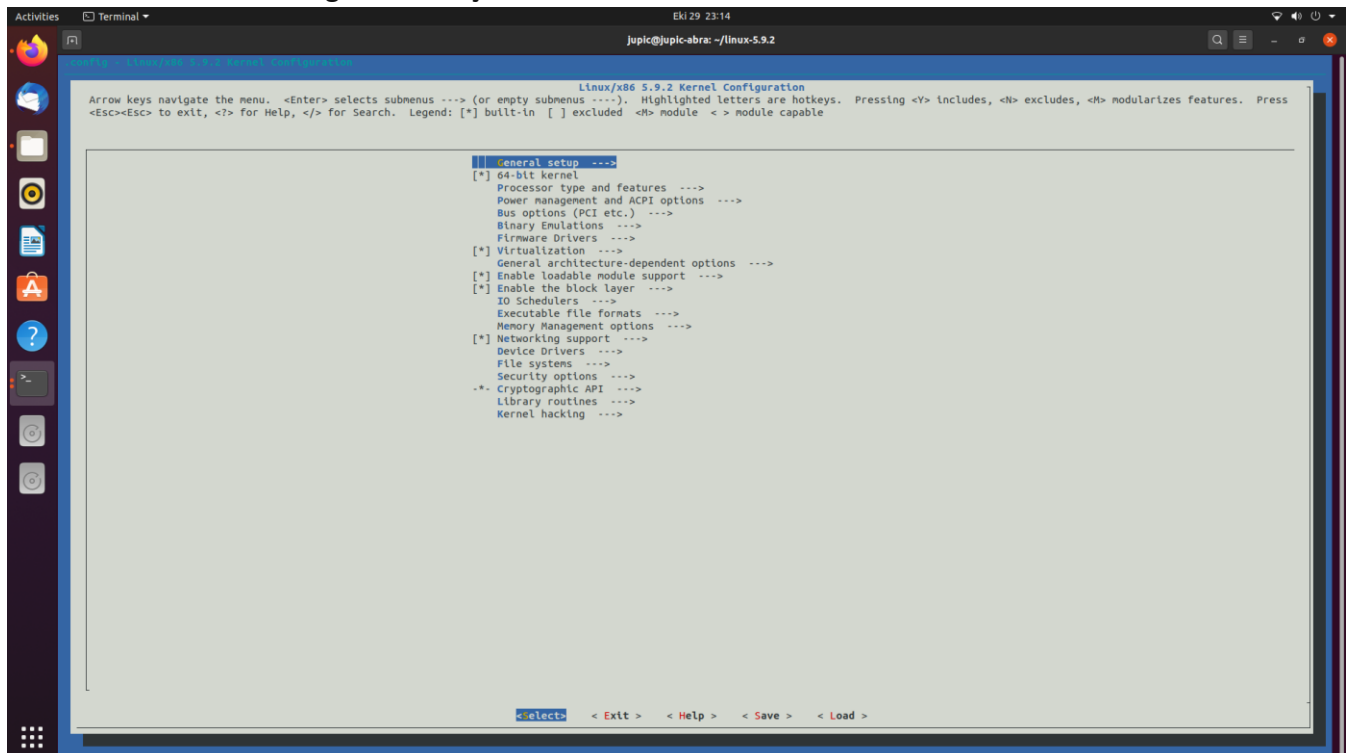
10-) Tecrübeli insanlar için:

“make menuconfig” -> Metin tabanlı bios benzeri bir arayüze sahip konfigürasyon aracı

“make xconfig” -> X Windows (Qt) tabanlı konfigürasyon aracı

“make gconfig” -> X Windows (Gtk) tabanlı konfigürasyon aracı

Örnek “make menuconfig” komutu çıktısı:



11-) “make” komutu ile kernel image dosyasının oluşturulmasını başlatın. “make -j /sizin-cpunuzun-çekirdek-sayısı” komutu ile bu işlemi hızlandırabilirsiniz.

```
jupic@jupic-abra:~/linux-5.9.2$ make -j 8
HOSTCC scripts/kconfig/conf.o
HOSTLD scripts/kconfig/conf
scripts/kconfig/conf --synconfig Kconfig
WRAP arch/x86/include/generated/uapi/asm/bpf_perf_event.h
WRAP arch/x86/include/generated/uapi/asm/errno.h
WRAP arch/x86/include/generated/uapi/asm/fcntl.h
WRAP arch/x86/include/generated/uapi/asm/ioctl.h
WRAP arch/x86/include/generated/uapi/asm/ioctls.h
WRAP arch/x86/include/generated/uapi/asm/ipcbuf.h
WRAP arch/x86/include/generated/uapi/asm/param.h
```

NOT: Eğer görseldeki gibi çıktı olarak pahole paketi ile ilgili bir hata alıyorsanız:

```
GEN modules.builtin
BTF: .tmp_vmlinux.btf: pahole (pahole) is not available
Failed to generate BTF for vmlinux
Try to disable CONFIG_DEBUG_INFO_BTF
make: *** [Makefile:1162: vmlinux] Error 1
```

“sudo apt install dwarves” komutu ile dwarves yükleyin. Tekrar “make -j 8” komutunu çalıştırın.

```
jupic@jupic-abra:~/linux-5.9.2$ sudo apt install dwarves
[sudo] password for jupic:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libfprint-2-tod1
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  dwarves
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded.
Need to get 228 kB of archives.
After this operation, 2.023 kB of additional disk space will be used.
Get:1 http://tr.archive.ubuntu.com/ubuntu focal/universe amd64 dwarves amd64 1.15-2 [228 kB]
Fetched 228 kB in 1s (304 kB/s)
Selecting previously unselected package dwarves.
(Reading database ... 252297 files and directories currently installed.)
Preparing to unpack .../dwarves_1.15-2_amd64.deb ...
Unpacking dwarves (1.15-2) ...
Setting up dwarves (1.15-2) ...
Processing triggers for man-db (2.9.1-1) ...
jupic@jupic-abra:~/linux-5.9.2$ make -j 8
```

Eğer dwarves bu sorunu çözmez ve aşağıdaki gibi bir hata alırsanız dwarves ve pahole’nin 1.17 sürümünü bu linkten indirin:

http://archive.ubuntu.com/ubuntu/pool/universe/d/dwarves-dfsg/dwarves_1.17-1_amd64.deb

```
GEN      modules.builtin
BTF: .tmp_vmlinux.btf: pahole version v1.15 is too old, need at least v1.16
Failed to generate BTF for vmlinux
Try to disable CONFIG_DEBUG_INFO_BTF
make: *** [Makefile:1162: vmlinux] Error 1
```

Daha sonra “cd Downloads” komutuyla indirdiğiniz .deb uzantılı dosyanın olduğu klasöre gidin ve “sudo dpkg -i dwarves_1.17-1_amd64.deb” komutu ile indirdiğiniz deb dosyasının kurulumunu yapın.

```
jupic@jupic-abra:~$ cd Downloads
jupic@jupic-abra:~/Downloads$ sudo dpkg -i dwarves_1.17-1_amd64.deb
```

Kurulum bittikten sonra tekrar “make -j 8” komutu ile kernel image dosyasını oluşturulmasını başlatın.

```
jupic@jupic-abra:~/Downloads$ cd ..
jupic@jupic-abra:~$ cd linux-5.9.2
jupic@jupic-abra:~/linux-5.9.2$ make -j 8
DESCEND objtool
DESCEND bpf/resolve_btfids
CALL    scripts/atomic/check-atomics.sh
CALL    scripts/checksyscalls.sh
CHK     include/generated/compile.h
```

NOT: make komutu ile kurulum yaparken derleyici belirli noktalarda kayıt noktası (checkpoint) alır ve kaldığı yerden devam eder. Eğer make komutunun çıktısı “Kernel: image is ready” değil ise tekrardan make komutunu çalıştırarak aşağıdaki gibi çıktı aldığınızdan emin olun.

```
LD [M]  sound/x86/snd-hdmi-lpe-audio.ko
LD [M]  sound/xen/snd_xen_front.ko
jupic@jupic-abra:~/linux-5.9.2$ make -j 8
DESCEND objtool
DESCEND bpf/resolve_btfids
CALL    scripts/atomic/check-atomics.sh
CALL    scripts/checksyscalls.sh
CHK     include/generated/compile.h
CHK     kernel/kheaders_data.tar.xz
Kernel: arch/x86/boot/bzImage is ready (#4)
jupic@jupic-abra:~/linux-5.9.2$
```

12-) “sudo make modules_install” komutu ile modüllerin kurulumuna başlayın. Çıktı olarak son komut DEPMOD ise her şey yolunda demektir. Depmod modules.dep dosyasını ve map dosyalarını oluşturmayı sağlayan programdır.


```
INSTALL sound/usb/line6/snd-usb-podhd.ko
INSTALL sound/usb/line6/snd-usb-toneport.ko
INSTALL sound/usb/line6/snd-usb-variak.ko
INSTALL sound/usb/misc/snd-ua101.ko
INSTALL sound/usb/snd-usb-audio.ko
INSTALL sound/usb/snd-usbmidi-lib.ko
INSTALL sound/usb/usx2y/snd-usb-us122l.ko
INSTALL sound/usb/usx2y/snd-usb-usx2y.ko
INSTALL sound/x86/snd-hdmi-lpe-audio.ko
INSTALL sound/xen/snd_xen_front.ko
DEPMOD 5.9.2
```

```
jupic@jupic-abra:~/linux-5.9.2$
```

13-) "sudo make install" komutu ile kernel kurulumunu başlatın.

```
jupic@jupic-abra:~/linux-5.9.2$ sudo make install
sh ./arch/x86/boot/install.sh 5.9.2 arch/x86/boot/bzImage \
    System.map "/boot"
run-parts: executing /etc/kernel/postinst.d/apt-auto-removal 5.9.2 /boot/vmlinuz-5.9.2
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 5.9.2 /boot/vmlinuz-5.9.2
update-initramfs: Generating /boot/initrd.img-5.9.2
W: Possible missing firmware /lib/firmware/rtl_nic/rtl8125b-2.fw for module r8169
I: The initramfs will attempt to resume from /dev/sda5
I: (UUID=7e89c43e-2fae-4f28-85d4-15945587917f)
I: Set the RESUME variable to override this.
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 5.9.2 /boot/vmlinuz-5.9.2
run-parts: executing /etc/kernel/postinst.d/update-notifier 5.9.2 /boot/vmlinuz-5.9.2
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 5.9.2 /boot/vmlinuz-5.9.2
Sourcing file '/etc/default/grub'
Sourcing file '/etc/default/grub.d/init-select.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-5.9.2-050902-lowlatency
Found initrd image: /boot/initrd.img-5.9.2-050902-lowlatency
Found linux image: /boot/vmlinuz-5.9.2-050902-generic
Found initrd image: /boot/initrd.img-5.9.2-050902-generic
Found linux image: /boot/vmlinuz-5.9.2
Found initrd image: /boot/initrd.img-5.9.2
Found linux image: /boot/vmlinuz-5.4.0-42-generic
Found initrd image: /boot/initrd.img-5.4.0-42-generic
Found linux image: /boot/vmlinuz-5.0.0-23-generic
Found initrd image: /boot/initrd.img-5.0.0-23-generic
Found Windows Boot Manager on /dev/nvme0n1p1/EFI/Microsoft/Boot/bootmgfw.efi
Adding boot menu entry for UEFI Firmware Settings
done
jupic@jupic-abra:~/linux-5.9.2$
```

Bu komut /boot klasöründeki 3 dosyayı yükleyip kernel grub config dosyasını güncelleyecektir:

initramfs-5.9.2.img

System.map-5.9.2

vmlinuz-5.9.2

14-) İsteğe bağlı: “sudo make install” komutu ile grub config dosyası da güncellenmiş olacaktır ama isteğe bağlı tekrardan güncellemek isterseniz komutları sırasıyla girin:

```
sudo update-initramfs -c -k 5.6.9
```

```
sudo update-grub
```

```
jupic@jupic-abra:~/linux-5.9.2$ sudo update-initramfs -c -k 5.9.2
update-initramfs: Generating /boot/initrd.img-5.9.2
W: Possible missing firmware /lib/firmware/rtl_nic/rtl8125b-2.fw for module r8169
I: The initramfs will attempt to resume from /dev/sda5
I: (UUID=7e89c43e-2fae-4f28-85d4-15945587917f)
I: Set the RESUME variable to override this.
jupic@jupic-abra:~/linux-5.9.2$ sudo update-grub
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/init-select.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-5.9.2-050902-lowlatency
Found initrd image: /boot/initrd.img-5.9.2-050902-lowlatency
Found linux image: /boot/vmlinuz-5.9.2-050902-generic
Found initrd image: /boot/initrd.img-5.9.2-050902-generic
Found linux image: /boot/vmlinuz-5.9.2
Found initrd image: /boot/initrd.img-5.9.2
Found linux image: /boot/vmlinuz-5.4.0-42-generic
Found initrd image: /boot/initrd.img-5.4.0-42-generic
Found linux image: /boot/vmlinuz-5.0.0-23-generic
Found initrd image: /boot/initrd.img-5.0.0-23-generic
Found Windows Boot Manager on /dev/nvme0n1p1@EFI/Microsoft/Boot/bootmgfw.efi
Adding boot menu entry for UEFI Firmware Settings
done
jupic@jupic-abra:~/linux-5.9.2$
```

15-) Son olarak “sudo reboot” komutu ile bilgisayarı yeniden başlatın. Terminali açıp “uname -r” komutu ile kernel sürümünüzü kontrol edin.

```
jupic@jupic-abra: ~
jupic@jupic-abra:~$ uname -r
5.9.2-050902-lowlatency
jupic@jupic-abra:~$
```