

ADB Tool Nedir ve Kullanımı

Bu makale şu başlıklardan oluşmaktadır.

- Adb Tool Nedir
- Uygulama 1
 - [!] Uyarı: Genymotion ile Çalışan Android Sistem IP'leri Hk
 - [!] Uyarı: Genymotion ile Çalışan Android Sistemlerin Otomatikmen TcpIp Modunda Bağlanması Hk
- Uygulama 2

Adb Tool Nedir

Adb tool (yani Android Debug Bridge Tool), mobil uygulama geliştiricilerinin kişisel bilgisayar ile android cihaz arasındaki iletişimini kolaylaştıran resmi Android SDK kiti aracıdır. Bu araç ile android uygulama geliştiricileri cihaza apk yollama, apk çekme, cihaz üzerinde komut çalıştırma, cihazdan log çekme gibi işlemler uygularlar. Bu araçla kurulan iletişim çoğunlukla USB kablo üzerinden gerçekleştirilir, fakat ayrıca Wifi bağlantısı üzerinden bağlantı desteklenmektedir.

Bilgisayardan adb tool'u ile android cihaza bağlanmak için (yani apk yollama, apk çekme, komut çalıştırma, log çekme v.b. işlemlerin yapılabileceği kanalı açmak için) android cihazda Developer Options (Geliştirici Seçenekleri) ve Developer Options altında yer alan USB Debugging seçeneği aktif olmalıdır.

adb bağlantısında üç adet unsur vardır. Birincisi bilgisayarda bulunan istemcidir. İstemciden usb kablo aracılığıyla (ya da yapılandırma ile wifi aracılığıyla) komutlar gidecektir. İkincisi hem bilgisayarda hem de telefonda bulunacak adbd daemon'ıdır (servisidir). İstemcideki daemon (servis) telefonda daemon'ın (servisin) komutları kabul etmesi ve çalıştırmasını sağlayacaktır. Üçüncüsü ise sunucudur. Sunucu bilgisayarda (istemciyle bir arada) bulunur ve istemci ile telefonda servis arasında iletişimi yönetir. Dolayısıyla bilgisayarda adb ile bir komut girildiğinde komutlar bilgisayarda istemciden arkaplanda çalışan sunucu process'lerine ve oradan da telefonda daemon'a (servise) gönderilir. Bahsedilen bu üç unsur (istemci, sunucu, ve hem istemci hem de cihazdaki servis) mobil cihazlara, tabletlere, akıllı saatlere ve dahasına erişim için birlikte kullanılır.

Aşağıdaki komutlar

- > adb start-server
- > adb kill-server

ile bilgisayarda (istemcide) adb daemon'ı (servisi) başlatılır veya durdurulur. Bilgisayarda (istemcide) adb daemon'ı (servisi) başladığında girilen adb komutları önce bilgisayarda (istemcide) ayağa kaldırılan adb daemon'ına gider, sonra bu komutlar karşı sistemdeki (cihazdaki) adb daemon'ına gider. Bu şekilde adb yetenekleri karşı cihazlarda uygulanır.

Popüler (sık kullanılan) adb komutları şu şekildedir:

- > adb shell # Android cihazdaki komut satırını getirir.
- > adb push <local> <remote> # Bilgisayardan android cihaza dosya yollar.
- > adb pull <remote> <local> # Android cihazdan bilgisayara dosya çeker.
- > adb logcat # Android cihazın anlık olarak log akışını görüntüler (not: # -b radio ile radio uygulamasının log'ları görüntülenir veya

> adb install <file>

-C parametresi ile log akışı renklendirilmiş şekilde olur).
Bilgisayardan android cihaza .apk uygulaması yükler.

Uygulama 1

(+) Birebir denenmiştir ve başarıyla uygulanmıştır.

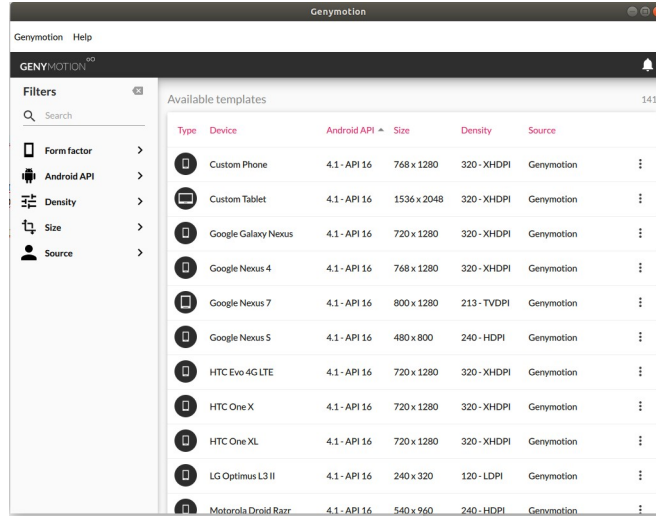
Bu başlık altında genymotion sanallaştırma aracı ile bir android sistem oluşturup ayağa kaldırılacaktır. Ardından bilgisayardan adb tool'u ile sanal android sisteme bağlanıp dosya yükleme, dosya çekme, komut çalıştırma gibi faaliyetler uygulanacaktır.

a) Genymotion Sanallaştırma Aracı Başlatılır

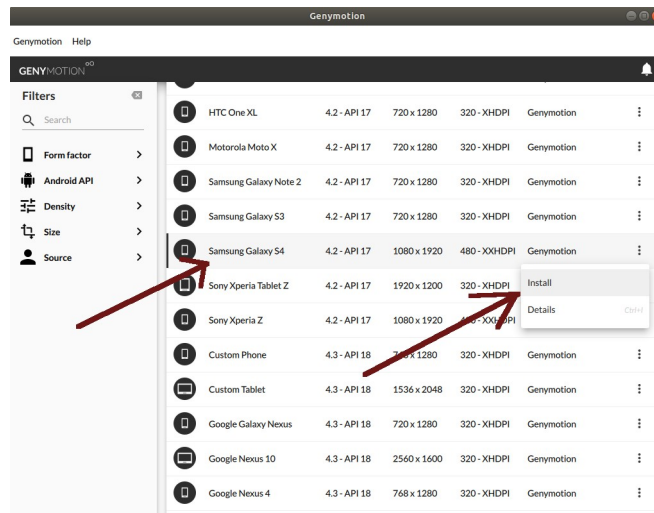
Genymotion sanallaştırma aracını başlatalım.

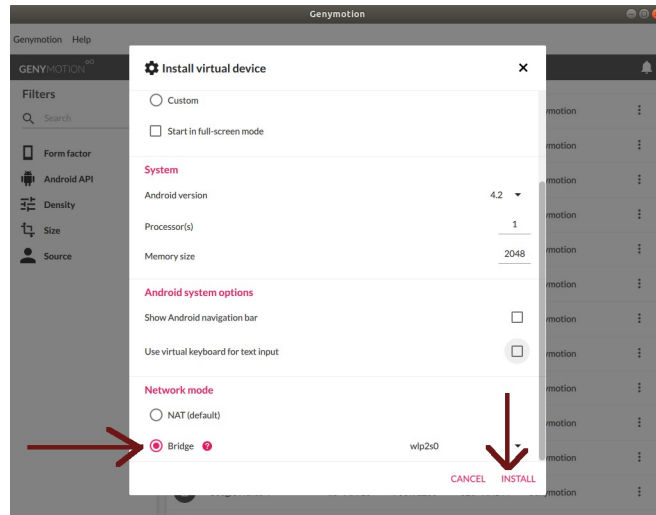
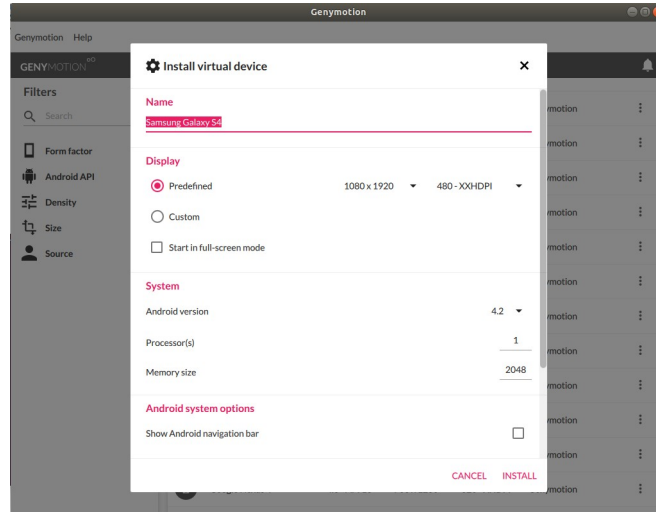
Ubuntu 18.04 LTS Terminal:

> genymotion



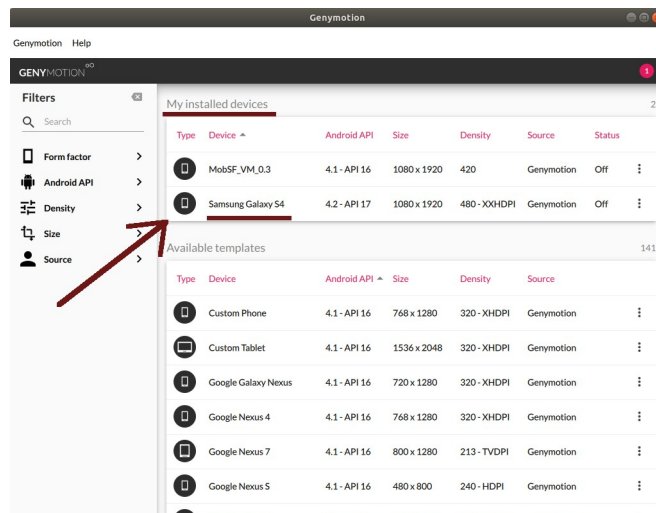
Sıralı android sistem şablonlarından Samsung Galaxy S4 sistemi oluşturalım.



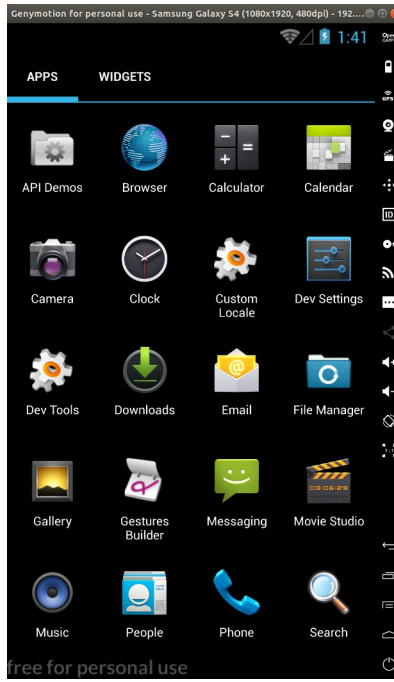
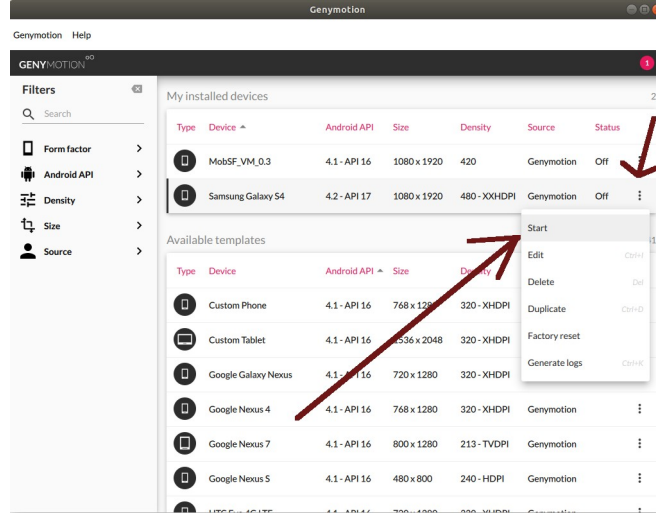


b) Genymotion Sanallaştırma Aracı İçinde Sanal Android Sistem Kurulur

Ardından sanal android sistem kurulacaktır ve “My Android Devices” sekmesi altında sıralanacaktır.



Sanal android sistemi başlatalım.



c) Bilgisayardan Sanal Android Sisteme Bağlanılır

Bilgisayardan adb tool'u ile bu sanal android sisteme bağlanalım.

Ubuntu 18.04 LTS Terminal:

```
> cd /opt/genymobile/genymotion/tools/  
> ./adb  
> ./adb devices  
> ./adb connect IP:portNum
```

Bağlı android sistemleri sıralar
Belirtilen android sisteme bağlanır.

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Help
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb devices
List of devices attached
192.168.57.101:5555    device

hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb connect 192
.168.57.101:5555
already connected to 192.168.57.101:5555
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ █
```

Not:

Sanal android sistem ekran kilit moda geçtiğinde (yani ekranı kapat tuşuna basıp ekranın karanması simulesi gerçekleştiğinde) dahi adb bağlantısı devam etmektedir. Ekran kapalı haldeyken adb devices, push, pull, logcat ve install parametreleri (re-install için telefonda uygulama elle kaldırılmıştır) yeniden denenmiştir ve işlemler olması gerektiği gibi gerçekleşmiştir. Yani ekran kapalıyken adb bağlantısı ve işlemleri halen yürütülebilmektedir.

d) Bilgisayardan Adb Tool'u ile Sanal Android Sistemi Komut Satırı Getirilir

Bilgisayarda adb tool'u ile sanal android sistemin komut satırı alınır. Çeşitli linux komutları sanal android sistem üzerinde çalıştırılır ve çıktıları görüntülenir.

Ubuntu 18.04 LTS Terminal:

```
> ./adb shell
root@android:/ # ls -al
```

Çıktı:

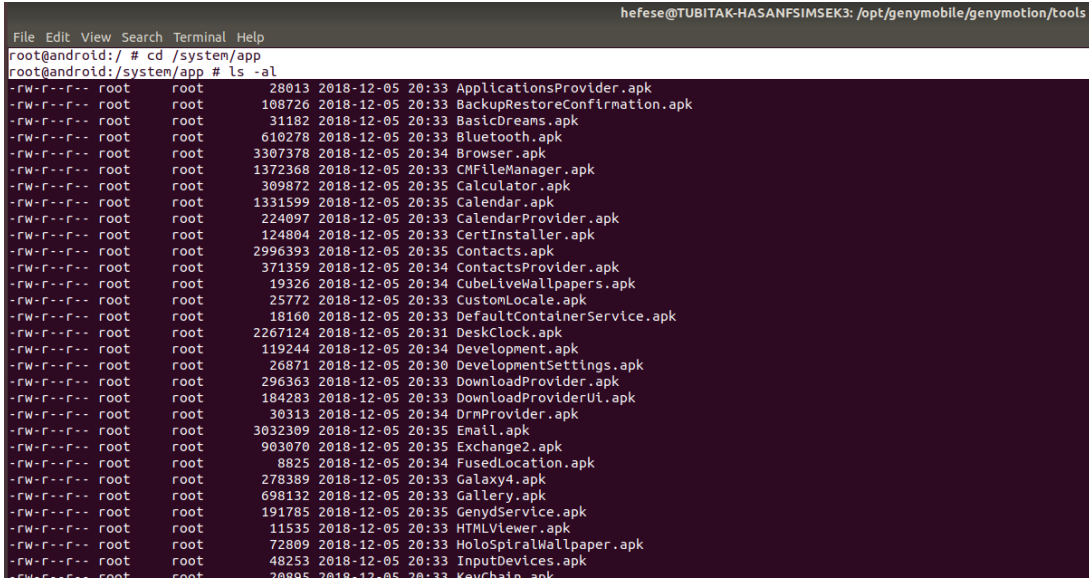
```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Help
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb shell
root@android:/ # ls -al
drwxr-xr-x root    root          2019-10-28 01:41 acct
drwxrwx--- system  cache         1970-01-01 00:00 cache
dr-x----- root    root          2019-10-28 01:41 config
lrwxrwxrwx root    root          2019-10-28 01:41 d -> /sys/kernel/debug
drwxrwx--x system  system       2019-10-28 01:41 data
-rw-r--r-- root    root          151 1970-01-01 00:00 default.prop
drwxr-xr-x root    root          2019-10-28 01:41 dev
lrwxrwxrwx root    root          2019-10-28 01:41 etc -> /system/etc
-rw-r----- root    root          625 1970-01-01 00:00 fstab.vbox86
-rwxr-x--- root    root         244902 1970-01-01 00:00 init
-rwxr-x--- root    root          2487 1970-01-01 00:00 init.goldfish.rc
-rwxr-x--- root    root         16624 1970-01-01 00:00 init.rc
lrwxr-x--- root    root          371 1970-01-01 00:00 init.redis.rc
-rwxr-x--- root    root          1795 1970-01-01 00:00 init.trace.rc
-rwxr-x--- root    root          3947 1970-01-01 00:00 init.usb.rc
-rwxr-x--- root    root          1195 1970-01-01 00:00 init.vbox86.rc
-rwxr-x--- root    root          285 1970-01-01 00:00 init.vbox86p.rc
drwxrwxr-x root    system       2019-10-28 01:41 mnt
dr-xr-xr-x root    root          2019-10-28 01:41 proc
drwx----- root    root          2017-09-19 09:42 root
drwxr-x--- root    root          2019-10-28 01:41/sbin
lrwxrwxrwx root    root          2019-10-28 01:41/sdcard -> /storage/emulat
```

Örneğin android sistemlerde sistem kurulumu ile beraber gelen apk dosyaları /system/app dizini altında, sistem kurulumu sonrası kullanıcı tarafından yüklenen apk dosyaları ise /data/app dizini altında yer alır. Karşı sistemdeki bu dizinler altında yer alan dosyaları görüntüleyelim.

Ubuntu 18.04 LTS Terminal:

```
root@android:/ # cd /system/app # Android sistem kurulumuyla beraber gelen .APK'lar
root@android:/ # ls -al
```

Çıktı:

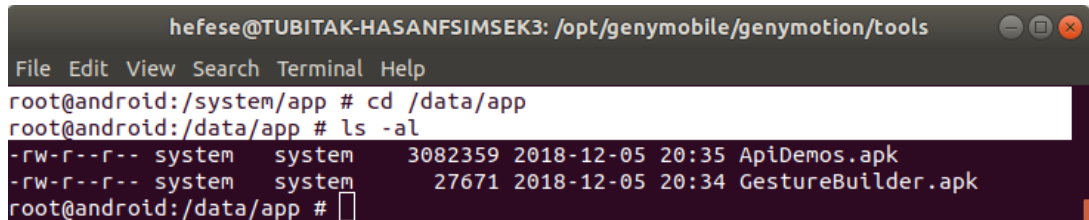


```
File Edit View Search Terminal Help
root@android:/ # cd /system/app
root@android:/system/app # ls -al
-rw-r--r-- root root 28013 2018-12-05 20:33 ApplicationsProvider.apk
-rw-r--r-- root root 108726 2018-12-05 20:33 BackupRestoreConfirmation.apk
-rw-r--r-- root root 31182 2018-12-05 20:33 BasicDreams.apk
-rw-r--r-- root root 610278 2018-12-05 20:33 Bluetooth.apk
-rw-r--r-- root root 3307378 2018-12-05 20:34 Browser.apk
-rw-r--r-- root root 1372368 2018-12-05 20:33 CnFileManager.apk
-rw-r--r-- root root 309872 2018-12-05 20:35 Calculator.apk
-rw-r--r-- root root 1331599 2018-12-05 20:35 Calendar.apk
-rw-r--r-- root root 224097 2018-12-05 20:33 CalendarProvider.apk
-rw-r--r-- root root 124804 2018-12-05 20:33 CertInstaller.apk
-rw-r--r-- root root 2996393 2018-12-05 20:35 Contacts.apk
-rw-r--r-- root root 371359 2018-12-05 20:34 ContactsProvider.apk
-rw-r--r-- root root 19326 2018-12-05 20:34 CubeLiveWallpapers.apk
-rw-r--r-- root root 25772 2018-12-05 20:33 CustomLocale.apk
-rw-r--r-- root root 18160 2018-12-05 20:33 DefaultContainerService.apk
-rw-r--r-- root root 2267124 2018-12-05 20:31 DeskClock.apk
-rw-r--r-- root root 119244 2018-12-05 20:34 Development.apk
-rw-r--r-- root root 26871 2018-12-05 20:30 DevelopmentSettings.apk
-rw-r--r-- root root 296363 2018-12-05 20:33 DownloadProvider.apk
-rw-r--r-- root root 184283 2018-12-05 20:33 DownloadProviderUi.apk
-rw-r--r-- root root 30313 2018-12-05 20:34 DrmProvider.apk
-rw-r--r-- root root 3032309 2018-12-05 20:35 Email.apk
-rw-r--r-- root root 903070 2018-12-05 20:35 Exchange2.apk
-rw-r--r-- root root 8825 2018-12-05 20:34 FusedLocation.apk
-rw-r--r-- root root 278389 2018-12-05 20:33 Galaxy4.apk
-rw-r--r-- root root 698132 2018-12-05 20:33 Gallery.apk
-rw-r--r-- root root 191785 2018-12-05 20:35 GenydService.apk
-rw-r--r-- root root 11535 2018-12-05 20:33 HTMLView.apk
-rw-r--r-- root root 72809 2018-12-05 20:33 HoloSpiralWallpaper.apk
-rw-r--r-- root root 48253 2018-12-05 20:33 InputDevices.apk
-rw-r--r-- root root 28895 2018-12-05 20:33 Keyboard.apk
```

Ubuntu 18.04 LTS Terminal:

```
root@android:/ # cd /data/app # Android sistem kurulumu sonrası yüklenen .APK'lar
root@android:/ # ls -al
```

Çıktı:



```
hefese@TUBITAK-HASANFISIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Help
root@android:/system/app # cd /data/app
root@android:/data/app # ls -al
-rw-r--r-- system system 3082359 2018-12-05 20:35 ApiDemos.apk
-rw-r--r-- system system 27671 2018-12-05 20:34 GestureBuilder.apk
root@android:/data/app #
```

Bu şekilde karşı sistem üzerinde komut çalıştırabilir ve çıktılarını görüntüleyebiliriz.

e) Bilgisayardan Adb Tool'u ile Sanal Android Sisteme Dosya Yollarır

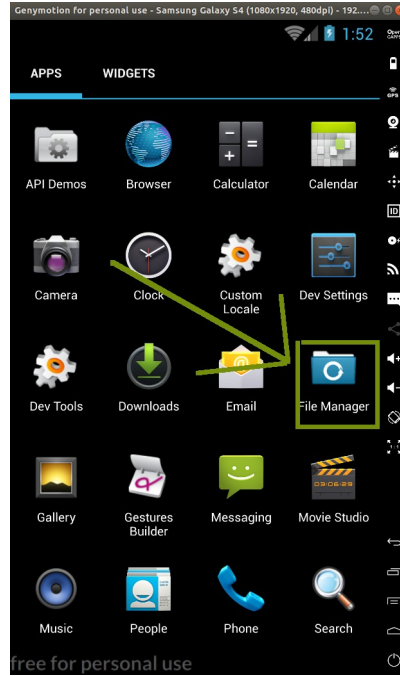
Bilgisayardan adb tool'u ile sanal android sisteme (yazma izni bulunan bir dizinine) dosya yollayalım. Bunun için hedef dizinde yazma izni olması gerekir. Bu nedenle örneğin Download klasörü yazma iznine sahip olduğundan oraya dosya yollayalım.

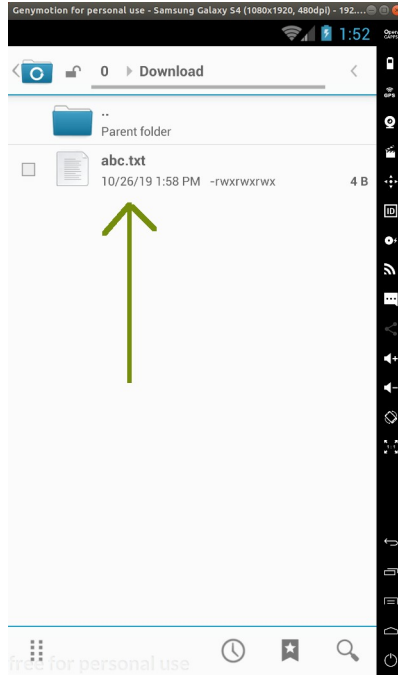
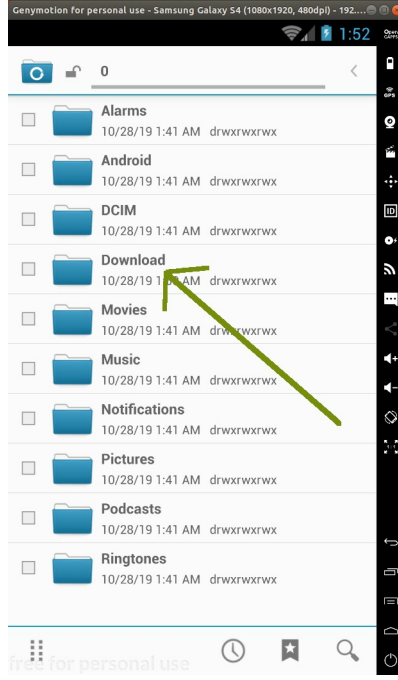
Ubuntu 18.04 LTS Terminal:

```
> ./adb push /home/hefese/abc.txt /mnt/shell/emulated/0/Download/abc.txt  
Çıktı:
```

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools  
File Edit View Search Terminal Help  
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb push /home/  
hefese/abc.txt /mnt/shell/emulated/0/Download/abc.txt  
/home/hefese/abc.txt: 1 file pushed. 0.0 MB/s (4 bytes in 0.017s)  
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$
```

Bilgisayardan push ile android cihaza yollanan dosya android cihazın File Manager->Download dizini altında görülebilir.





f) Bilgisayardan Adb Tool'u ile Sanal Android Sistemden Dosya Çekilir

Bilgisayardan adb tool'u ile sanal android sistemdeki bir dosyayı çekelim. Örneğin sistem kurulumu sonrası yüklenen uygulama apk'larının bulunduğu /data/app dizininden bir uygulamanın dosyasını (.apk'sını) çekelim.

Ubuntu 18.04 LTS Terminal:

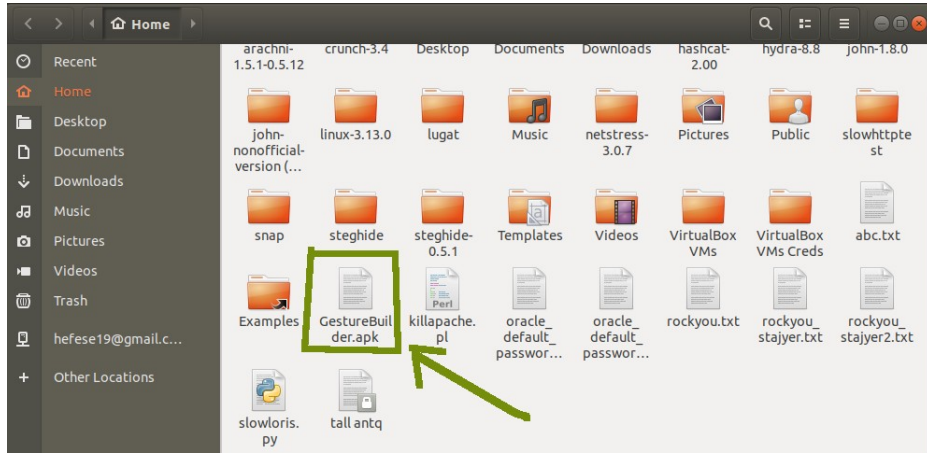
```
> ./adb pull /data/app/GestureBuilder.apk /home/hefese/GestureBuilder.apk
```

(not: Dizin ve apk bilgileri adb shell oturumu ile tespit edilip kullanılmıştır.)

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Help
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb pull /data/
app/GestureBuilder.apk /home/hefese/GestureBuilder.apk
/data/app/GestureBuilder.apk: 1 file pulled. 1.5 MB/s (27671 bytes in 0.018s)
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$
```

(Apk Dosyası Çekilir)



(Apk Dosyası home Dizinine Gelir)

g) Bilgisayardan Adb Tool'u ile Sanal Android Sistemdeki Sistem Anlık Log Akışı İzlenir

Bilgisayardan adb tool'u ile sanal android sistemdeki sistem anlık log akışı izlenir.

Ubuntu 18.04 LTS Terminal:

> ./adb logcat

```
# Android cihazın sistem log akışını anlık olarak
# görüntüler (not: spesifik bir servisin (örn; radio
# 'nun) anlık log kayıtlarını görüntülemek için
# -b radio parametre ve argumanı kullanılır).
```

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Tabs Help
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools x hefese@TUBITAK-HASANF
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb logcat
----- beginning of /dev/log/system
E/baseband-redis( 65): Redis baseband write connect error: Connection refused
----- beginning of /dev/log/main
D/DHCP ( 81): ==== DHCP message:
D/DHCP ( 81): op = BOOTREQUEST (1), htype = 1, hlen = 6, hops = 0
D/DHCP ( 81): xid = 0xb80a0000 secs = 0, flags = 0x8000 optlen = 14
D/DHCP ( 81): ciaddr = 0.0.0.0
D/DHCP ( 81): yiaddr = 0.0.0.0
D/DHCP ( 81): siaddr = 0.0.0.0
D/DHCP ( 81): giaddr = 0.0.0.0
D/DHCP ( 81): chaddr = { 08 00 27 52 ef 1d }
D/DHCP ( 81): sname = ''
D/DHCP ( 81): file = ''
D/DHCP ( 81): op 53 len 1 { 01 } discover
D/DHCP ( 81): op 55 len 4 { 01 03 06 1c }
D/DHCP ( 81): ==== DHCP message:
D/DHCP ( 81): op = BOOTREPLY (2), htype = 1, hlen = 6, hops = 0
D/DHCP ( 81): xid = 0xb80a0000 secs = 0, flags = 0x0000 optlen = 312
D/DHCP ( 81): ciaddr = 192.168.57.101
D/DHCP ( 81): yiaddr = 192.168.57.101
D/DHCP ( 81): siaddr = 0.0.0.0
D/DHCP ( 81): giaddr = 0.0.0.0
D/DHCP ( 81): chaddr = { 08 00 27 52 ef 1d }
D/DHCP ( 81): sname = ''
D/DHCP ( 81): file = ''
D/DHCP ( 81): op 54 len 4 { c0 a8 39 64 }
D/DHCP ( 81): op 53 len 1 { 02 } offer
D/DHCP ( 81): op 51 len 4 { 00 00 04 b0 }
D/DHCP ( 81): op 1 len 4 { ff ff ff 00 }
D/DHCP ( 81): --- dhcp offer (2) ---
D/DHCP ( 81): ip 192.168.57.101 gw 0.0.0.0 prefixLength 24
D/DHCP ( 81): server 192.168.57.100, lease 1200 seconds
D/DHCP ( 81): ==== DHCP message:
D/DHCP ( 81): op = BOOTREQUEST (1), htype = 1, hlen = 6, hops = 0
D/DHCP ( 81): xid = 0xb90a0000 secs = 0, flags = 0x8000 optlen = 26
D/DHCP ( 81): ciaddr = 0.0.0.0
D/DHCP ( 81): yiaddr = 0.0.0.0
D/DHCP ( 81): siaddr = 0.0.0.0
D/DHCP ( 81): giaddr = 0.0.0.0
D/DHCP ( 81): chaddr = { 08 00 27 52 ef 1d }
D/DHCP ( 81): sname = ''
D/DHCP ( 81): file = ''
D/DHCP ( 81): op 53 len 1 { 03 } request
D/DHCP ( 81): op 55 len 4 { 01 03 06 1c }
D/DHCP ( 81): op 50 len 4 { c0 a8 39 65 }
D/DHCP ( 81): op 54 len 4 { c0 a8 39 64 }
D/DHCP ( 81): ==== DHCP message:
D/DHCP ( 81): op = BOOTREPLY (2), htype = 1, hlen = 6, hops = 0
D/DHCP ( 81): xid = 0xb90a0000 secs = 0, flags = 0x0000 optlen = 312
D/DHCP ( 81): ciaddr = 192.168.57.101
D/DHCP ( 81): yiaddr = 192.168.57.101
D/DHCP ( 81): siaddr = 0.0.0.0
```

h) Bilgisayardan Adb Tool'u ile Sanal Android Sisteme Uygulama Yüklendir

Bilgisayardan adb tool'u ile sanal android sisteme uygulama (.apk) yüklemesi yapılır. Bu işlem için (sanal android sistemin eskiliği göz önüne alınarak; çünkü son sürüm apk dosyaları hedef sanal android sistemdeki sdk sürüm eskiliğinden dolayı hata verdi) eski bir .apk dosyası yüklenecektir. Third party apk dosyaları için bkz. <https://www.apkmirror.com/>.

Third party apk uygulama sitesinden Simple Note adlı bir uygulamanın en eski sürümü (1.3.4) indirilir ve bilgisayardan adb tool'u ile sanal android sisteme yüklenir.

Ubuntu 18.04 LTS Terminal:

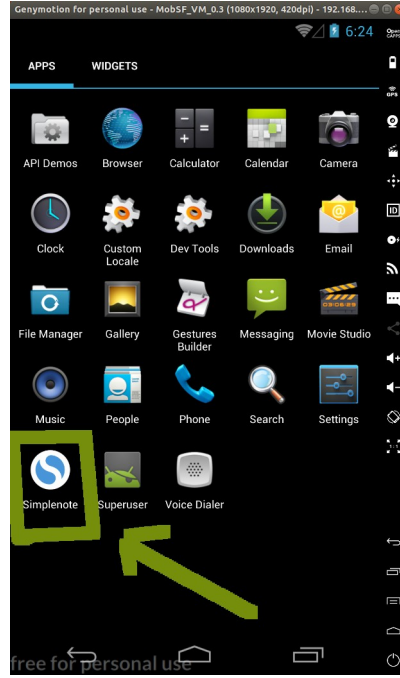
```
> ./adb install "/home/hefese/Desktop/com.automattic.simplenote_1.3.4-44_minAPI15(nod pi).apk"
```

(Not: Bu SimpleNote isimli apk dosyası <https://www.apkmirror.com/apk/automattic-inc/simplenote/simplenote-1-3-4-release/> adresinden en eski sürümü seçilmek suretiyle indirilmiştir. ~/Downloads/Mobil Sızma Testi Gereçler/ dizini altında mevcuttur.)

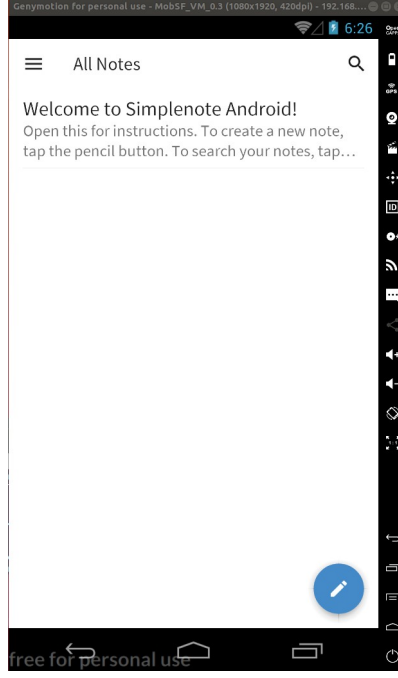
Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Tabs Help
hefese@TUBITAK-HASANFSIMSEK3: /opt/g... x hefese@TUBITAK-HASANFSIMSEK3: /opt/g... x
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb install "/home/hefese/Desktop/com.automattic.simplenote_1.3.4-44_minAPI15(nodpi).apk"
Performing Push Install
/home/hefese/Desktop/com.automattic.si...d. 70.4 MB/s (4578077 bytes in 0.062s)
pkg: /data/local/tmp/com.automattic.simplenote_1.3.4-44_minAPI15(nodpi).apk
Success
rm failed for -f, No such file or directory
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$
```

Uygulamanın sanal android sisteme yüklenip yüklenmediğini test ettiğimizde ana menüde uygulama görünecektir.



Uygulamaya girdiğimizde uygulama sorunsuz açılacaktır.



[!] Uyarı: Genymotion ile Çalışan Android Sistem IP'leri Hk.

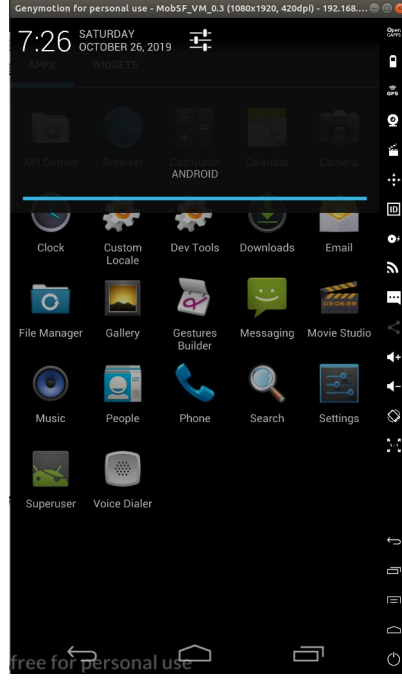
Genymotion çalışmak için Virtualbox yönetim sistemi motorundan yararlandığından android sistem ayağa kaldırıldığında bir, genymotion player tarafından android sisteme ethernet arayüzü tanımlanmakta, bir de virtualbox yönetim sistemi tarafından android sisteme ethernet arayüzü tanımlanmaktadır. Bu iki arayüz iki adet ayrı ip adresi sunmaktadır. İki ayrı ip adres aynı android sistemi ifade etmektedir.

Not:

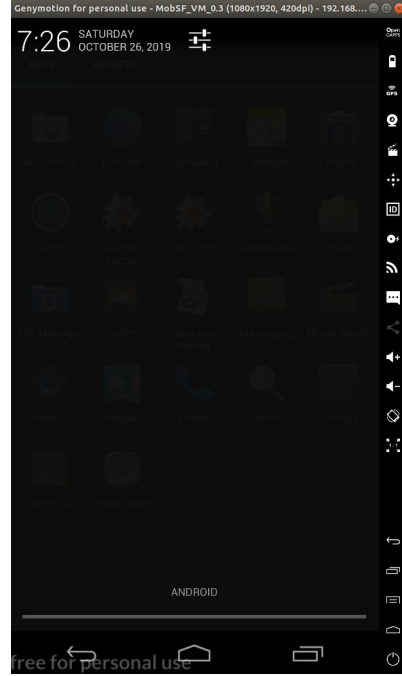
Tıpkı Ubuntu ana makinanda wifi ve ethernet ile iki adet ayrı arayüz ve bu arayüzlerde iki adet ayrı ip var olması gibi. Bu iki ip adresiyle de Ubuntu ana makinana ulaşılabilmekte.

Yani android sanal sisteme iki adet ethernet arayüzü tanımlanmaktadır. Birisi genymotion player tarafından, diğeri virtualbox tarafından tanımlanmaktadır.

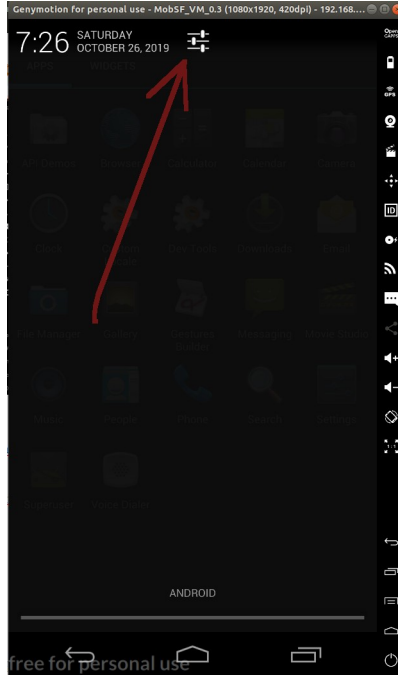
Şimdi Genymotion emulatrörünün android sanal makinasına verdiği ip adresini öğrenelim. Bunun için android sanal makinanın en yukarisından fare imleci ile basılı tutup aşağı indir yapılır. Daha sonra ayarlar çarkına tıklanır. Daha sonra Wifi seçeneğine tıklanır. Son olarak ise bağlı olunan wifi ağına tıklanarak ağ bilgileri ekrana getirilir.



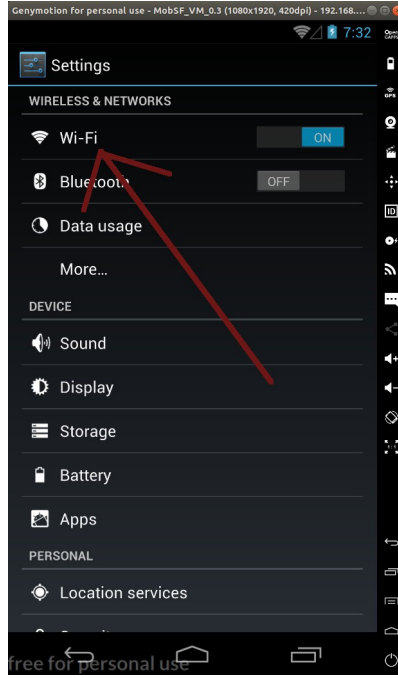
(Üst menü aşağı çekilir (1))



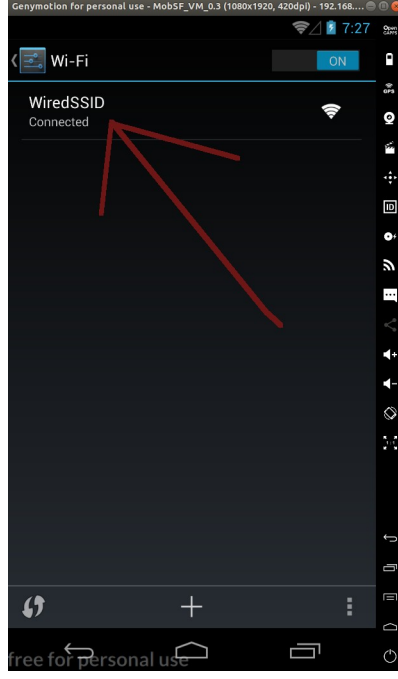
(Üst menü aşağı çekilir (2))



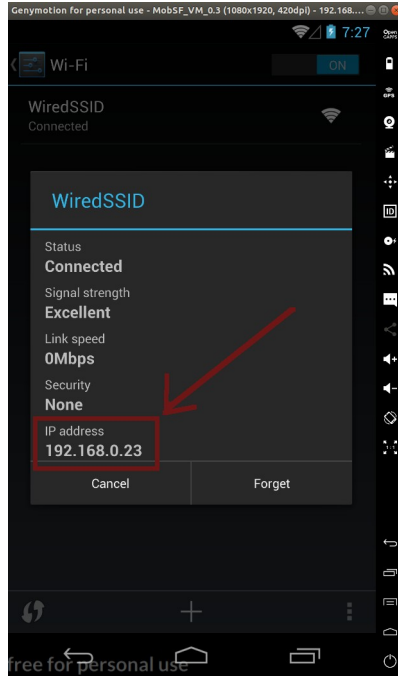
(Ayarlar arkına basılır)



(Wifi seeneđine gidilir)



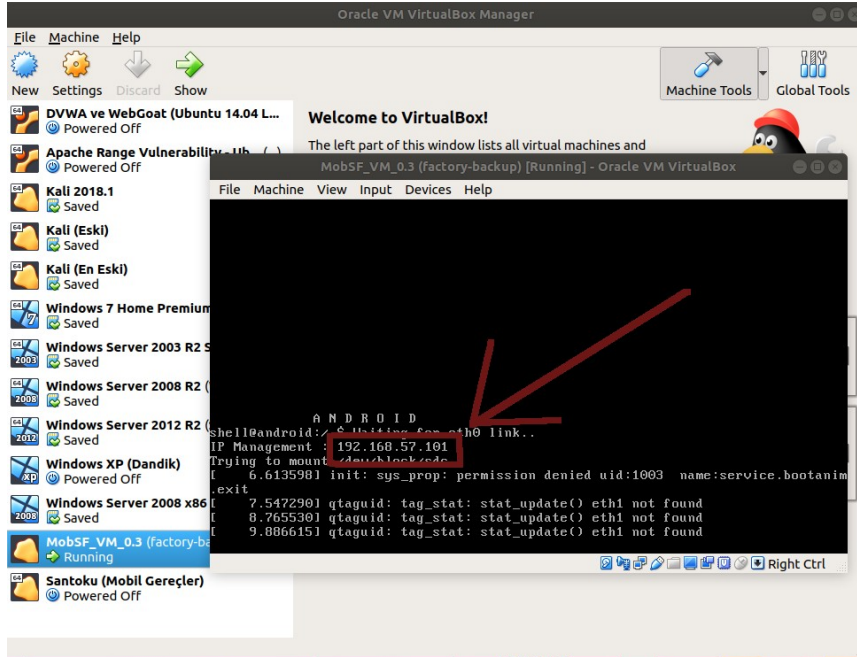
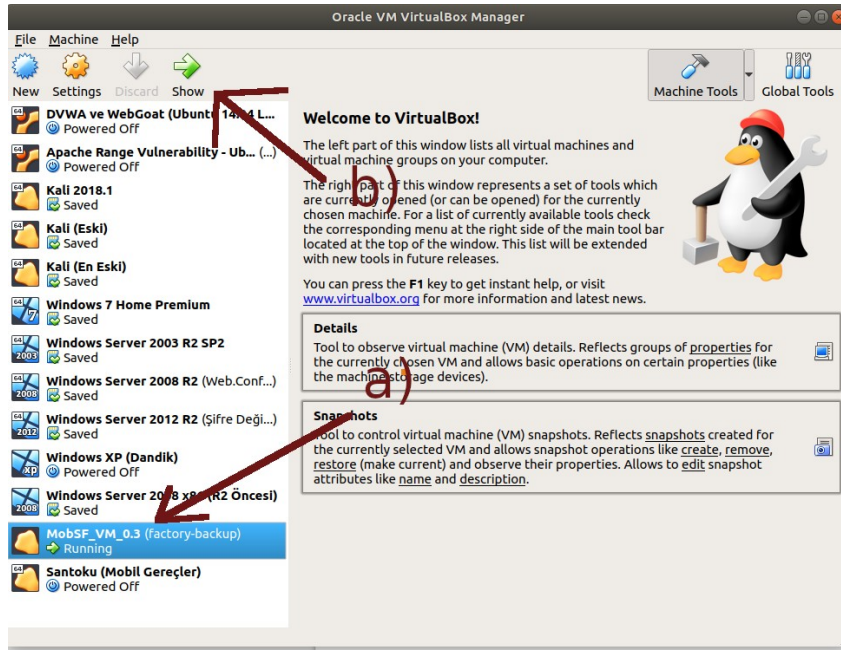
(Baęlı olunan aęa tıklanır)



(Baęlı olunan aę bilgileri ekrana gelir)

Böylece genymotion emulatörünün android sanal makinaya verdiği IP elde edilir. Şimdi genymotion'ın üzerinde çalıştığı virtualbox sanallaştırma yönetim motorunun android sanal makinasına verdiği ip adresini öğrenelim.

Virtualbox açılır. Genymotion'da çalışmakta olan makina virtualbox arayüzünde sıralı haldedir ve seçilip Show denir.



Böylece virtualbox yönetim sisteminin android sanal makinaya verdiği IP elde edilir. Bu iki IP aynı makınayı işaret etmektedir. Örneğin nmap ile port/servis taraması bu iki ip adresine yapıldığında aynı çıktı bizi karşılamaktadır:

Ubuntu 18.04 LTS Terminal:

```
> nmap -sV 192.168.0.23
```

Çıktı:

Starting Nmap 7.60 (<https://nmap.org>) at 2019-10-26 16:02 +03
Nmap scan report for **192.168.0.23**
Host is up (0.00053s latency).
Not shown: 998 closed ports

PORT STATE SERVICE VERSION
5555/tcp open freeciv?
24800/tcp open unknown
MAC Address: **08:00:27:78:11:E5** (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at <https://nmap.org/submit/> .
Nmap done: 1 IP address (1 host up) scanned in 156.07 seconds

Ubuntu 18.04 LTS Terminal:

```
> nmap -sV 192.168.57.101
```

Çıktı:

Starting Nmap 7.60 (<https://nmap.org>) at 2019-10-26 15:54 +03
Nmap scan report for **192.168.57.101**
Host is up (0.00034s latency).
Not shown: 998 closed ports

PORT STATE SERVICE VERSION
5555/tcp open freeciv?
24800/tcp open unknown
MAC Address: **08:00:27:52:EF:1D** (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at <https://nmap.org/submit/> .
Nmap done: 1 IP address (1 host up) scanned in 157.11 seconds

Çıktılarda ayrıca ip adres & mac adres eşlemelerine bakıldığında iki çıktudaki mac adreslerinin farklılığından hareketle iki adet network kartı tanımlaması yapıldığı açık bir şekilde görülebilir.

Adb tool'u sisteme bağlı cihazlar listelemesinde android sistemin bu iki ayrı network kartı tanımlamasından virtualbox yönetim sistemi network kartı ile (IP'si ile) cihazı sıralamaktadır. (Not: Anlaşıldığı kadarıyla low level olan virtualbox olduğu için adb tool'u virtualbox'ın ethernet arayüzü bilgisini esas almaktadır).

Ubuntu 18.04 LTS Terminal:

```
> adb devices
```

Çıktı:

List of devices attached

```
192.168.57.101:5555 device
```

[!] Uyarı: Genymotion ile Çalışan Android Sistemlerin Otomatikmen TcpIp Modunda Bağlanması Hk

Genymotion'da çalıştırılan / ayağa kaldırılan sanal android sistemler usb debugging özelliği açık halde gelmektedir. "adb devices" ile listelendiklerinde genymotion'ın üstü kapalı olarak yaptığı bir ayar gereği tcpip modunda bağlantıya geçmektedirler. Normalde her adb bağlantısı önce usb modunda kurulur. Fakat android sdk geliştiricilerinin usb kablo gerekliliğini ortadan kaldırmak için sunduğu wifi ağı üzerinden bağlantıyı sürdürme çözümü daha kullanışlı olduğundan genymotion otomatikmen bağlantıları tcpip moduna çekmektedir. Bu nedenle usb debugging bağlantıları ve izinleri en başta kurulduğundan wifi ağı üzerinde bağlantı sürdürülürken usb debugging özelliği açık olsun veya olmasın adb tool'u ile cihazlarda işlemler yürütülebilmektedir.

Örneğin genymotion'daki sanal bir android sistemde USB Debugging kapatıldığında ve hatta Developer Options (Geliştirici Seçenekleri) tümüyle kapalı hale getirildiğinde adb tool'u ile bağlantı halen sağlanabilmektedir.

Ubuntu 18.04 LTS Terminal:

```
// Sanal Android Sisteminde Developer Options ve Developer Options->USB
// Debugging Açıkken
> ./adb push /home/hefese/test.html /sdcard/Download/test.html
```

Çıktı:

```
/home/hefese/test.html: 1 file pushed. 0.0 MB/s (4 bytes in 0.000s)
```

Ubuntu 18.04 LTS Terminal:

```
// Sanal Android Sisteminde Developer Options ve Developer Options->USB
// Debugging Kapalıyken
> ./adb push /home/hefese/test.html /sdcard/Download/test2.html
```

Çıktı:

```
/home/hefese/test2.html: 1 file pushed. 0.0 MB/s (4 bytes in 0.000s)
```

İkisinde de test.html hedef sanal android sistemin /sdcard/Download dizinine upload'lanabilmektedir. Dolayısıyla bağlantı usb debugging kapalıyken yine de wifi üzerinden sağlanabilmektedir.

Uygulama 2

(+) Birebir denenmiştir ve başarıyla uygulanmıştır.

Bu başlık altında adb tool'u ile bu sefer fiziksel / gerçek bir android mobil telefona bağlanıp dosya yükleme, dosya çekme, komut çalıştırma gibi faaliyetler uygulanacaktır.

Gereksinimler

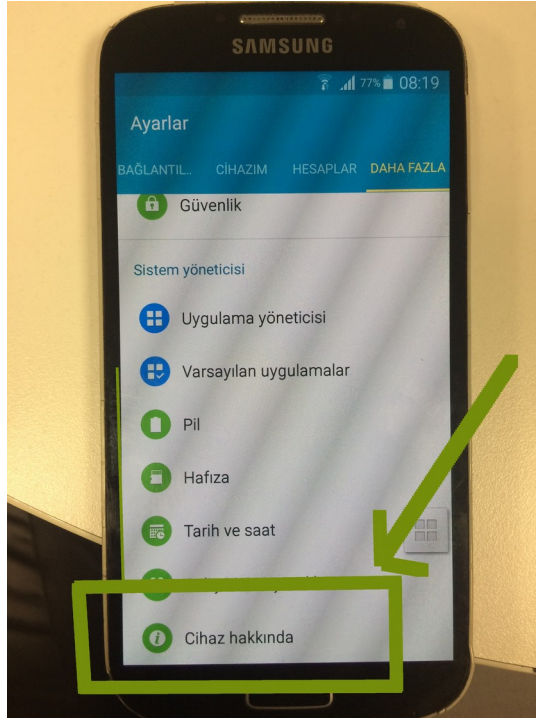
Samsung Galaxy S4
Data Cable

// Fiziksel / Gerçek Android Mobil Telefon
// PC, mobil cihazı görebilsin diye şarj
// yerine şarj & data kablo gerekli

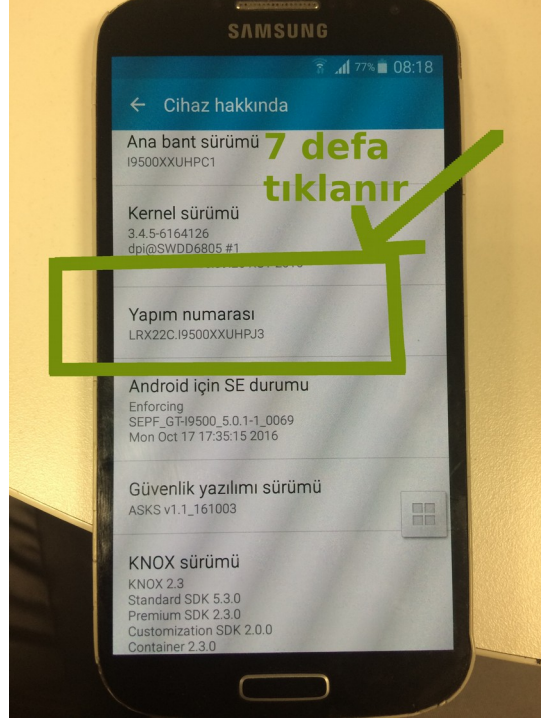
Öncelikle bilgisayardan adb tool'u ile Samsung Galaxy S4 android telefona bağlanabilmek için Samsung Galaxy S4 android telefonunda developer options ve usb debugging seçenekleri açık olmalıdır.

A. Usb Debugging Özelliğini Açma

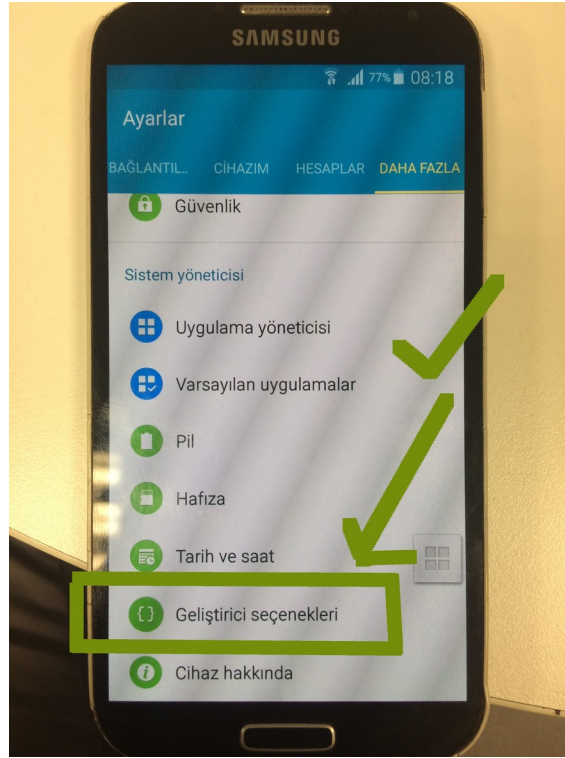
i) Cihazda "Ayarlar" açılır ve "Hakkında" seçeneğine gidilir.



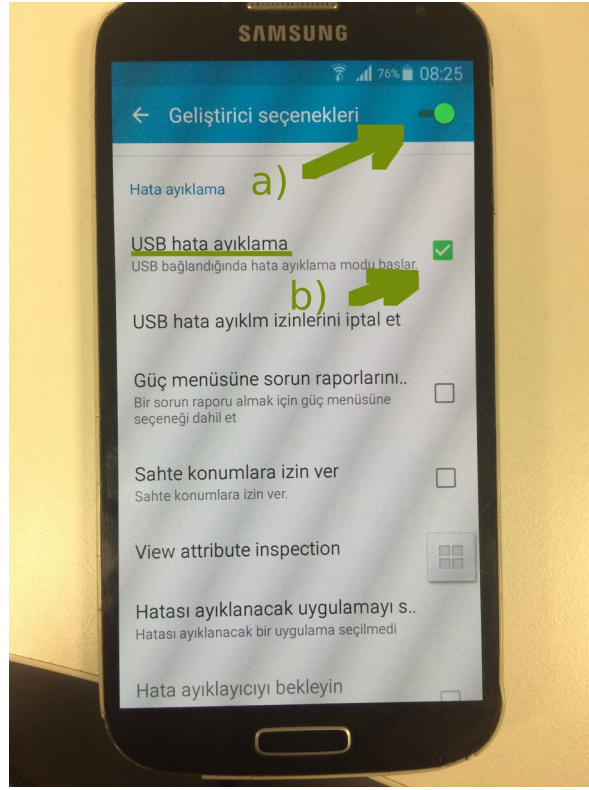
ii) "Build Number" (Yapım Numarası) seçeneğine 7 defa basılır (bu şekilde cihazda "Geliştirici Seçenekleri" ayarı açılacaktır).



iii) "Ayarlar"a geri dönülür ve yeni gelecek "Geliştirici Seçenekleri" seçeneğine gidilir (Not: android pie sürümü sonrasında "Geliştirici Seçenekleri" seçeneği Ayarlar->Sistem altında görünürken, eski android sürümlerde "Hakkında"nın hemen altında görünür).



iv) “Geliştirici Seçenekleri” içerisinde USB Debugging (diğer adıyla Android Debugging) girdisine tick atılır.



Bu şekilde samsung galaxy s4 telefonunda developer options ve usb debugging açılmış olur.

v) Adb tool’u ile mobil telefona bağlanmak için usb charge & data kablosu takılır.

vi) Adb tool’u cihaz listeleme komutu çalıştırılır ve usb’yle bağlanmış mobil cihaz tanımsız olarak listelenir.

Ubuntu 18.04 LTS Terminal:

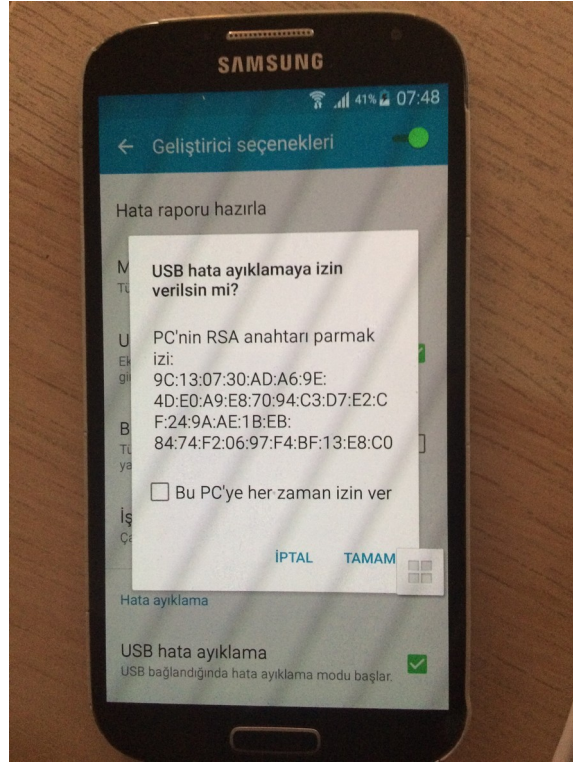
> adb devices

Adb sunucusunu üstü kapalı olarak
başlatır ve bağlı mobil cihazları listeler

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Help
hefese@TUBITAK-HASANFSIMSEK3:/opt/genymobile/genymotion/tools$ ./adb devices
* daemon not running; starting now at tcp:5037
* daemon started successfully
List of devices attached
4d00a9524a056173    unauthorized
hefese@TUBITAK-HASANFSIMSEK3:/opt/genymobile/genymotion/tools$
```

vi) “adb devices” ile bağılı cihazları listeleme komutu çalıştırıldığında mobil telefon ekranına usb debugging için izin isteyen bir popup gelir:



vii) Popup’a Tamam dendiğinde ve “adb devices” komutu tekrar çalıştırıldığında cihaz tanımlı olarak listelenir.

Ubuntu 18.04 LTS Terminal:

> adb devices

Adb sunucusunu üstü kapalı olarak
başlatır ve bağılı mobil cihazları listeler

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Help
hefese@TUBITAK-HASANFSIMSEK3:/opt/genymobile/genymotion/tools$ ./adb devices
List of devices attached
4d00a9524a056173      device
hefese@TUBITAK-HASANFSIMSEK3:/opt/genymobile/genymotion/tools$
```

viii) Böylece adb tool’u mobil cihaz üzerinde kullanıma hazırdır.

B. Adb Tool'u ile Fiziksel / Gerçek Mobil Cihaz Üzerinde İşlemler Yürütme

a) Bilgisayardan Fiziksel / Gerçek Android Sisteme Bağlanılır

Bilgisayardan adb tool'u ile fiziksel / gerçek android sisteme bağlanalım. Samsung Galaxy S4 telefonu usb data kablosu ile bilgisayara bağlayalım ve adb devices komutu ile bağlı cihazları listeleyip cihazlardan usb debugging bağlantı izni talep edelim.

Ubuntu 18.04 LTS Terminal:

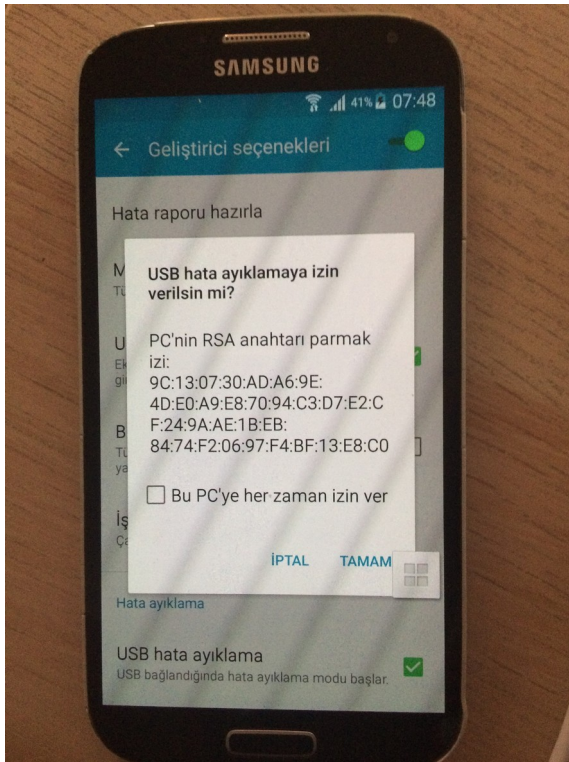
```
> cd /opt/genymobile/genymotion/tools/  
> ./adb  
> ./adb devices
```

Bağlı android sistemleri sıralar.

Çıktı:

List of devices attached

4d00a9524a056173 **unauthorized**



Tamam diyerek tekrar “adb devices” dediğimizde bilgisayar cihazı tanır duruma gelecektir.

Ubuntu 18.04 LTS Terminal:

```
> ./adb devices
```

Bağlı android sistemleri sıralar

Çıktı:

List of devices attached

4d00a9524a056173 **device**

Bu şekilde bilgisayara eğer bir mobil cihaz bağlıysa “adb devices” ile bağlantı otomatik sağlanacaktır.

Ayrıntı:

Eğer bilgisayara birden fazla mobil cihaz bağlanmışsa adb parametrelerini kullanırken ekstradan -s parametresi kullanılmalıdır ve listelenen bağlı cihazların birinin seri numarası arguman olarak verilmelidir. Bu sayede bağlı cihazlardan belirtileni üzerinde faaliyetler yürütülebilir.

Örneğin;

Fiziksel / gerçek android cihaz bağlıyken ve genymotion’daki sanal android sistem açık bırakıldığında;

Ubuntu 18.04 LTS Terminal:

```
> ./adb devices
```

Çıktı:

```
List of devices attached
```

```
4d00a9524a056173    device
192.168.57.101:5555    device
```

şeklinde listelenen cihazlardan ilk cihazda işlemler yürütmek için -s parametresi kullanılır:

Ubuntu 18.04 LTS Terminal:

```
> ./adb -s 4d00a9524a056173 shell           # Samsung Galaxy S4 mobil
                                           # cihazda shell oturumu alınır
```

Çıktı:

```
shell@ja3g:/$
```

DipNot:

Eğer bilgisayara usb kablo ile bağlanılan cihazlara wifi üzerinden erişim istenmekteyse adb tool’u parametreleri aracılığıyla bunu yapmak mümkündür ve bu durumda adb -e seçeneği kullanılır. Fakat birden fazla cihaza wifi üzerinden erişmek gerekirse mobil cihazların farklı tcpip port numarasından dinlenir hale gelmesi ayarı yapılır ve adb connect ile her birine ayrı ayrı bağlanılabilir. Ayrıntılı bilgi için bkz. <https://stackoverflow.com/questions/43973838/how-to-connect-multiple-android-devices-with-adb-over-wifi>

Not:

Fiziksel / gerçek android sistem ekran kilit moda geçtiğinde (yani ekranı kapat tuşuna basıp ekran karardığında) dahi adb bağlantısı devam etmektedir. Ekran kapalı haldeyken adb devices, push, pull, logcat ve install parametreleri (re-install için telefonda uygulama elle kaldırılmıştır) yeniden denenmiştir ve işlemler olması gerektiği gibi gerçekleşmiştir. Yani ekran kapalıyken adb bağlantısı ve işlemleri halen yürütülebilmektedir.

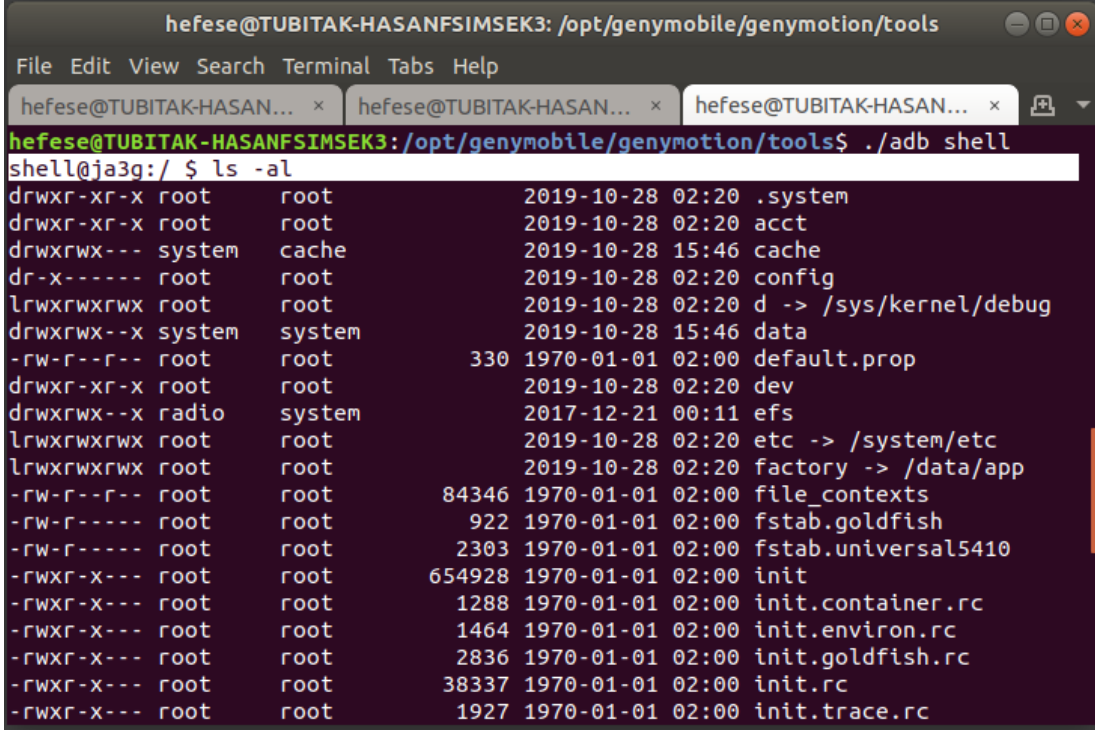
b) Bilgisayardan Adb Tool'u ile Fiziksel / Gerçek Android Sistemi Komut Satırı Getirilir

Bilgisayarda adb tool'u ile fiziksel / gerçek android sistemin komut satırı alınır. Çeşitli linux komutları fiziksel / gerçek android sistem üzerinde çalıştırılır ve çıktıları görüntülenir.

Ubuntu 18.04 LTS Terminal:

```
> ./adb shell  
root@android:/ # ls -al
```

Çıktı:



```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools  
File Edit View Search Terminal Tabs Help  
hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x  
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb shell  
shell@jja3q:/ $ ls -al  
drwxr-xr-x root root 2019-10-28 02:20 .system  
drwxr-xr-x root root 2019-10-28 02:20 acct  
drwxrwx--- system cache 2019-10-28 15:46 cache  
dr-x----- root root 2019-10-28 02:20 config  
lrwxrwxrwx root root 2019-10-28 02:20 d -> /sys/kernel/debug  
drwxrwx--x system system 2019-10-28 15:46 data  
-rw-r--r-- root root 330 1970-01-01 02:00 default.prop  
drwxr-xr-x root root 2019-10-28 02:20 dev  
drwxrwx--x radio system 2017-12-21 00:11 efs  
lrwxrwxrwx root root 2019-10-28 02:20 etc -> /system/etc  
lrwxrwxrwx root root 2019-10-28 02:20 factory -> /data/app  
-rw-r--r-- root root 84346 1970-01-01 02:00 file_contexts  
-rw-r----- root root 922 1970-01-01 02:00 fstab.goldfish  
-rw-r----- root root 2303 1970-01-01 02:00 fstab.universal5410  
-rwxr-x--- root root 654928 1970-01-01 02:00 init  
-rwxr-x--- root root 1288 1970-01-01 02:00 init.container.rc  
-rwxr-x--- root root 1464 1970-01-01 02:00 init.enviro.rc  
-rwxr-x--- root root 2836 1970-01-01 02:00 init.goldfish.rc  
-rwxr-x--- root root 38337 1970-01-01 02:00 init.rc  
-rwxr-x--- root root 1927 1970-01-01 02:00 init.trace.rc
```

Örneğin android sistemlerde sistem kurulumu ile beraber gelen apk dosyaları /system/app dizini altında, sistem kurulumu sonrası kullanıcı tarafından yüklenen apk dosyaları ise /data/app dizini altında yer alır. Karşı sistemdeki bu dizinler altında yer alan dosyaları görüntüleyelim.

Ubuntu 18.04 LTS Terminal:

```
root@android:/ # cd /system/app # Android sistem kurulumuyla beraber gelen .APK'lar  
root@android:/ # ls -al
```

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Tabs Help
hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x
shell@ja3g:/ $ cd /system/app
shell@ja3g:/system/app $ ls -al
drwxr-xr-x root root 2016-11-28 09:14 AASAservice
drwxr-xr-x root root 2015-08-05 00:47 ANTPlusPlugins
drwxr-xr-x root root 2016-11-28 09:14 ANTPlusTest
drwxr-xr-x root root 2016-11-28 09:14 AccessControl
drwxr-xr-x root root 2016-11-28 09:13 AccuweatherPhone2013_J_L
MR
drwxr-xr-x root root 2016-11-28 09:14 AirMotionTryActually
drwxr-xr-x root root 2016-11-28 09:14 AllShareCastPlayer
drwxr-xr-x root root 2016-11-28 09:14 AllshareFileShare
drwxr-xr-x root root 2016-11-28 09:14 AllshareFileShareClient
drwxr-xr-x root root 2016-11-28 09:14 AllshareFileShareServer
drwxr-xr-x root root 2016-11-28 09:14 AllshareMediaServer
drwxr-xr-x root root 2016-11-28 09:14 AllshareMediaShare
drwxr-xr-x root root 2016-11-28 09:14 AntHalService
drwxr-xr-x root root 2016-11-28 09:15 AssistantMenu2
drwxr-xr-x root root 2016-11-28 09:14 BCSERVICE
drwxr-xr-x root root 2016-11-28 09:14 BadgeProvider
drwxr-xr-x root root 2016-11-28 09:14 BandService
drwxr-xr-x root root 2016-11-28 09:14 BasicDreams
drwxr-xr-x root root 2016-11-28 09:14 BeaconManager
```

Ubuntu 18.04 LTS Terminal:

```
root@android:/ # cd /data/app # Android sistem kurulumu sonrası yüklenen .APK'lar
root@android:/ # ls -al
```

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Tabs Help
hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x
shell@ja3g:/system/app/AASAservice $ cd /data/app
shell@ja3g:/data/app $ ls -al
opendir failed, Permission denied
255|shell@ja3g:/data/app $
```

Kullanıcı tarafından yüklenen apk'lar (/data/app/) listelenememiştir. Çünkü bağlı mobil cihaz (Samsung Galaxy S4) root'lu değildir. Bu nedenle izin engeline takılı kalınacaktır. Hedef sistem (mobil cihaz) root 'lanınca bu v.b. kullanıcı kişisel verileri görüntülenebilecektir.

Sonuç olarak shell oturumu ile karşı sistem üzerinde komut çalıştırabilir ve çıktılarını görüntüleyebiliriz.

c) Bilgisayardan Adb Tool'u ile Fiziksel / Gerçek Android Sisteme Dosya Yolları

Bilgisayardan adb tool'u ile fiziksel / gerçek android sisteme (yazma izni bulunan bir dizinine) dosya yollayalım. Bunun için hedef dizinde yazma izni olması gerekir. Bu nedenle örneğin Download klasörü yazma iznine sahip olduğundan oraya dosya yollayalım.

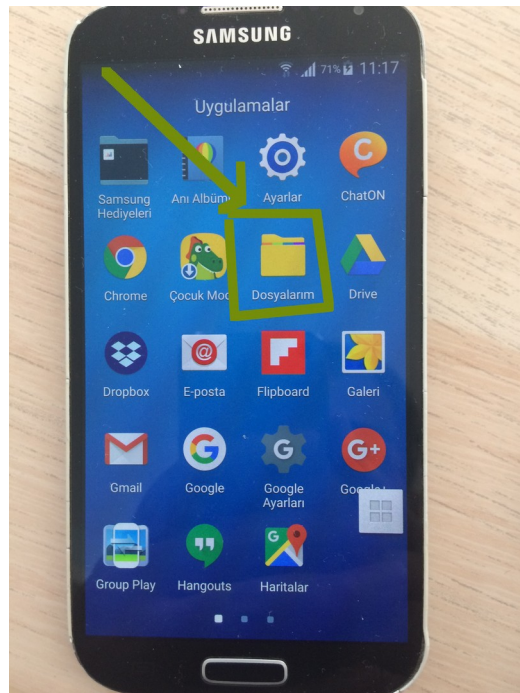
Ubuntu 18.04 LTS Terminal:

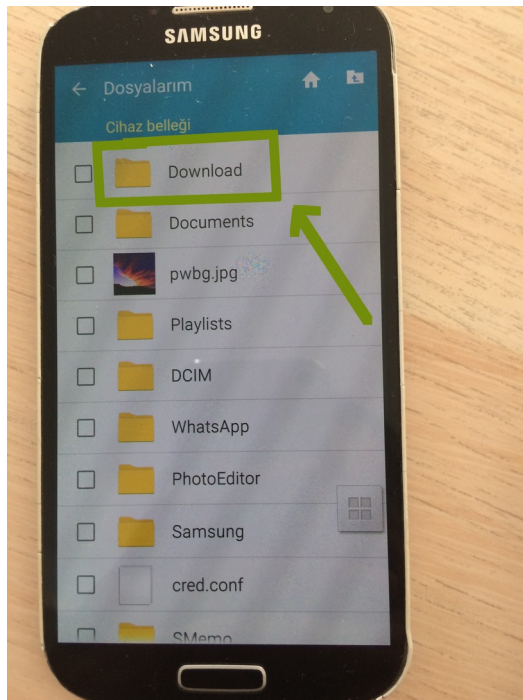
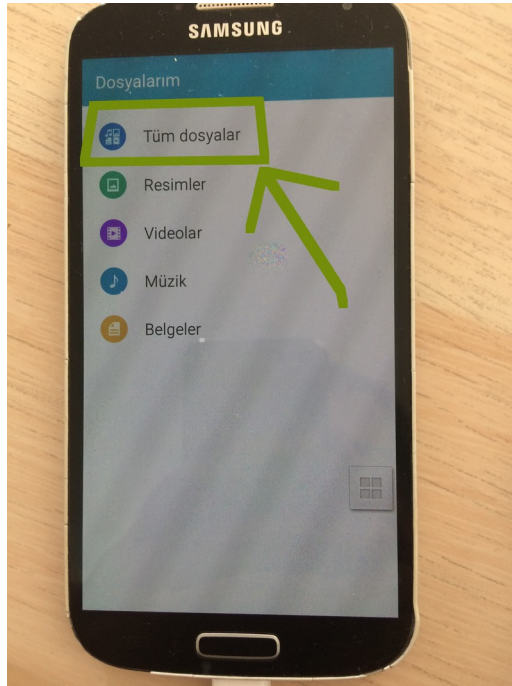
```
> ./adb push /home/hefese/abc.txt /mnt/shell/emulated/0/Download/abc.txt
```

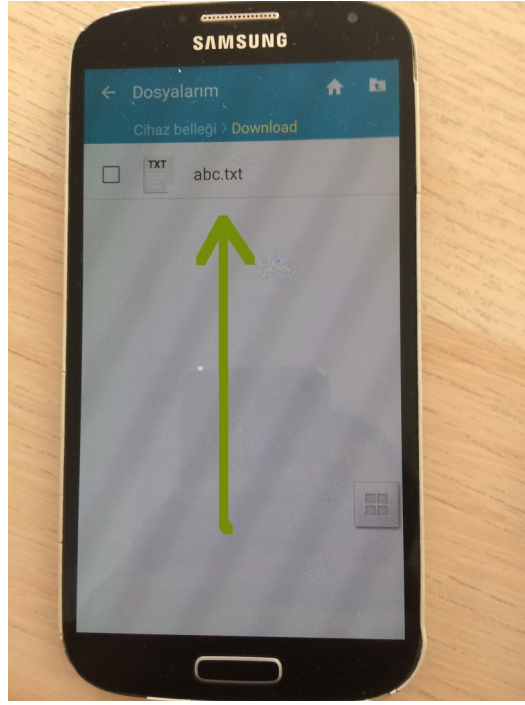
Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Tabs Help
hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb push /home
/hefese/abc.txt /mnt/shell/emulated/0/Download/abc.txt
/home/hefese/abc.txt: 1 file pushed. 0.0 MB/s (4 bytes in 0.121s)
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$
```

Bilgisayardan push ile android cihaza yollanan dosya android cihazın Files->All Files->Download dizini altında görülebilir.







d) Bilgisayardan Adb Tool'u ile Fiziksel / Gerçek Android Sistemden Dosya Çekilir

Bilgisayardan adb tool'u ile fiziksel / gerçek android sistemdeki bir dosyayı çekelim. Örneğin sistem kurulumu ile beraber gelen uygulama apk'larının bulunduğu /system/app dizininden bir uygulamanın dosyasını (.apk'sını) çekelim.

Ubuntu 18.04 LTS Terminal:

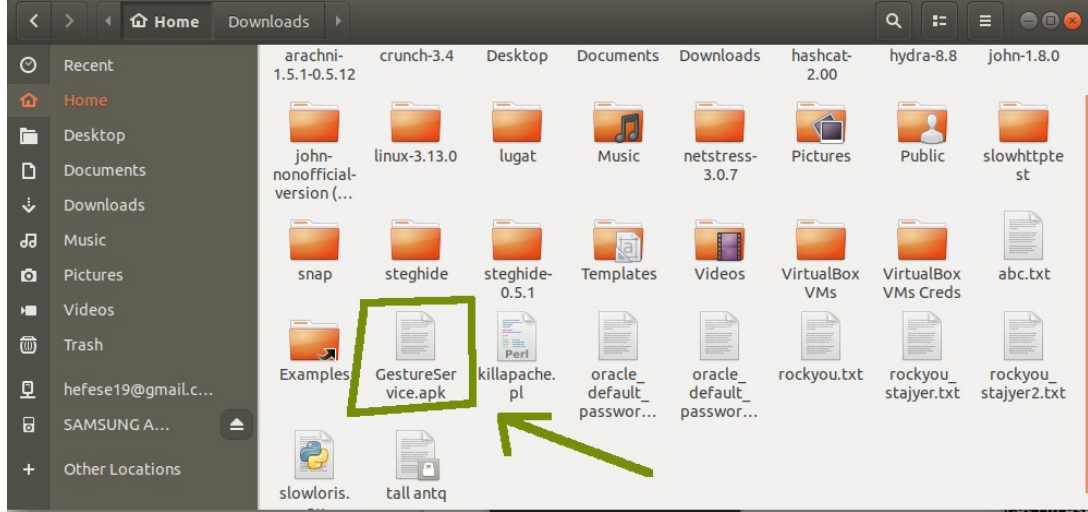
```
> ./adb pull /system/app/GestureService/GestureService.apk  
/home/hefese/GestureService.apk
```

(not: Dizin ve apk bilgileri adb shell oturumu ile tespit edilip kullanılmıştır.)

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools  
File Edit View Search Terminal Tabs Help  
hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x hefese@TUBITAK-HASAN... x  
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb pull /system/app/GestureService/GestureService.apk /home/hefese/GestureService.apk  
/system/app/GestureService/GestureService.apk: 1 file pulled. 0.3 MB/s (15742 bytes in 0.045s)  
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$
```

(Apk Dosyası Çekilir)



(Apk Dosyası home Dizinine Gelir)

e) Bilgisayardan Adb Tool'u ile Fiziksel / Gerçek Android Sistemdeki Sistem Anlık Log Akışı İzlenir

Bilgisayardan adb tool'u ile fiziksel / gerçek android sistemdeki sistem anlık log akışı izlenir.

Ubuntu 18.04 LTS Terminal:

```
> ./adb logcat
```

```
# Android cihazın sistem log akışını anlık olarak  
# görüntüler (not: spesifik bir servisin (örn; radio  
# 'nun) anlık log kayıtlarını görüntülemek için  
# -b radio parametre ve argumanı kullanılır).
```

Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Tabs Help
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/geny... x hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/geny... x hefese@TUBITAK-HASANFSIMSEK3: /opt/genym...
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb logcat
----- beginning of main
W/BeaconManager(26075): BLEHelper.LeScanCallback - receive packet
D/QuickConnect[3.0.7253268][33] (25955): BLEHelper.getBTAddress - FLAG:26
D/GpsLocationProvider(3165): receive broadcast intent, action: android.intent.action.BATTERY_CHANGED
D/KeyguardUpdateMonitor(3401): received broadcast android.intent.action.BATTERY_CHANGED
D/KeyguardUpdateMonitor(3401): handleBatteryUpdate
I/LockboxService(4144): Completed task: true
D/GpsLocationProvider(3165): receive broadcast intent, action: android.intent.action.BATTERY_CHANGED
D/KeyguardUpdateMonitor(3401): received broadcast android.intent.action.BATTERY_CHANGED
D/KeyguardUpdateMonitor(3401): handleBatteryUpdate
D/ScanRecord(26325): parseFromBytes
D/ScanRecord(26325): first manudata for manu ID
D/ScanRecord(26325): parseFromBytes
D/ScanRecord(26325): first manudata for manu ID
D/BtGatt.ContextMap(26325): sendClientScanResult for app id 6
D/BtGatt.ContextMap(26325): sendClientScanResult for app id 5
D/ScanRecord(25955): parseFromBytes
D/ScanRecord(25955): first manudata for manu ID
D/BluetoothLeScanner(25955): onScanResult() - ScanResult(mDevice=61:CA:99:02:2B:E9, mScanRecord=ScanRecord [mAdvertiseFlags=26, mServiceU
}, mServiceData={}, mTxPowerLevel=12, mDeviceName=null], mRssi=-91, mTimestampNanos=11791789435718)
D/ScanRecord(26075): parseFromBytes
D/ScanRecord(26075): first manudata for manu ID
D/BluetoothLeScanner(26075): onScanResult() - ScanResult(mDevice=61:CA:99:02:2B:E9, mScanRecord=ScanRecord [mAdvertiseFlags=26, mServiceU
}, mServiceData={}, mTxPowerLevel=12, mDeviceName=null], mRssi=-91, mTimestampNanos=117917894615385)
W/BeaconManager(26075): BLEHelper.LeScanCallback - receive packet
D/QuickConnect[3.0.7253268][33] (25955): BLEHelper.getBTAddress - FLAG:26
D/KeyguardUpdateMonitor(3401): received broadcast android.intent.action.TIME_TICK
D/KeyguardUpdateMonitor(3401): handleTimeUpdate
D/DateView(3401): regionalDateFormat = 22 11 3333 isRTLlanguage = false returnDateFormat = MMMM d, E
D/DateView(3401): regionalDateFormat = 22 11 3333 isRTLlanguage = false returnDateFormat = MMMM d, E
D/KeyguardEffectViewController(3401): onReceive action: android.intent.action.TIME_TICK
I/KeyguardEffectViewController(3401): *** don't update sliding image ***
D/KeyguardUpdateMonitor(3401): received broadcast android.intent.action.BATTERY_CHANGED
D/KeyguardUpdateMonitor(3401): handleBatteryUpdate
D/GpsLocationProvider(3165): receive broadcast intent, action: android.intent.action.BATTERY_CHANGED
D/ScanRecord(26325): parseFromBytes
D/ScanRecord(26325): first manudata for manu ID
D/ScanRecord(26325): parseFromBytes
D/ScanRecord(26325): first manudata for manu ID
D/BtGatt.ContextMap(26325): sendClientScanResult for app id 6
D/BtGatt.ContextMap(26325): sendClientScanResult for app id 5
D/ScanRecord(25955): parseFromBytes
D/ScanRecord(25955): first manudata for manu ID
D/ScanRecord(26075): parseFromBytes
D/ScanRecord(26075): first manudata for manu ID
D/BluetoothLeScanner(25955): onScanResult() - ScanResult(mDevice=61:CA:99:02:2B:E9, mScanRecord=ScanRecord [mAdvertiseFlags=26, mServiceU
}, mServiceData={}, mTxPowerLevel=12, mDeviceName=null], mRssi=-92, mTimestampNanos=117922525945489)
D/BluetoothLeScanner(26075): onScanResult() - ScanResult(mDevice=61:CA:99:02:2B:E9, mScanRecord=ScanRecord [mAdvertiseFlags=26, mServiceU
}, mServiceData={}, mTxPowerLevel=12, mDeviceName=null], mRssi=-92, mTimestampNanos=117922526202862)
W/BeaconManager(26075): BLEHelper.LeScanCallback - receive packet
D/QuickConnect[3.0.7253268][33] (25955): BLEHelper.getBTAddress - FLAG:26
D/ScanRecord(26325): parseFromBytes
D/ScanRecord(26325): first manudata for manu ID
```

f) Bilgisayardan Adb Tool'u ile Fiziksel / Gerçek Android Sisteme Uygulama Yüklenir

Bilgisayardan adb tool'u ile fiziksel / gerçek android sisteme uygulama (.apk) yüklemesi yapılır. Bu işlem için (fiziksel / gerçek android sistemin eskiliği göz önüne alınarak; çünkü son sürüm apk dosyaları hedef fiziksel / gerçek android sistemdeki sdk sürüm eskiliğinden dolayı hata verebilir) eski bir .apk dosyası yüklenecektir. Third party apk dosyaları için bkz. <https://www.apkmirror.com/>.

Third party apk uygulama sitesinden Simple Note adlı bir uygulamanın en eski sürümü (1.3.4) indirilir ve bilgisayardan adb tool'u ile fiziksel / gerçek android sisteme yüklenir.

Ubuntu 18.04 LTS Terminal:

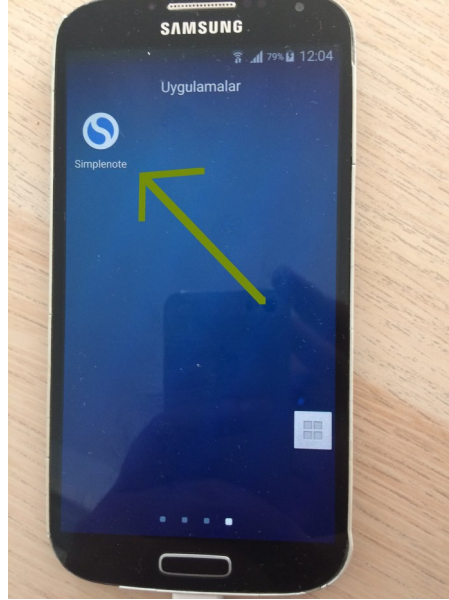
```
> ./adb install "/home/hefese/Desktop/com.automattic.simplenote_1.3.4-44_minAPI15(nod pi).apk"
```

(Not: Bu SimpleNote isimli apk dosyası <https://www.apkmirror.com/apk/automattic-inc/simplenote/simplenote-1-3-4-release/> adresinden en eski sürümü seçilmek suretiyle indirilmiştir. ~/Downloads/Mobil Sızma Testi Gereçler/ dizini altında mevcuttur.).

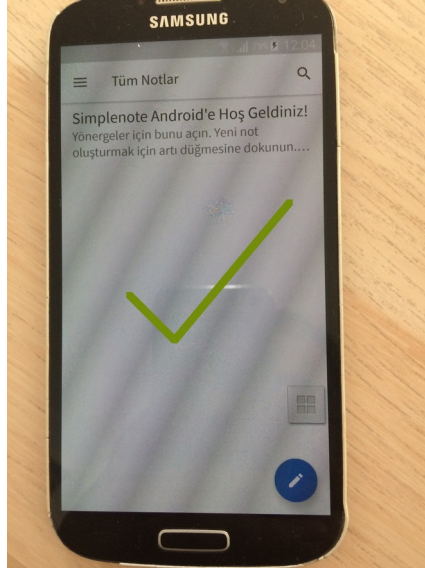
Çıktı:

```
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools
File Edit View Search Terminal Tabs Help
hefese@TUBITA... x hefese@TUBITA... x hefese@TUBITA... x hefese@TUBITAK... x
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$ ./adb install "/
home/hefese/Desktop/com.automatic.simplenote_1.3.4-44_minAPI15(nodpi).apk"
Performing Push Install
/home/hefese/Desktop/com.automatic.si...ed. 6.1 MB/s (4578077 bytes in 0.710s)
  pkg: /data/local/tmp/com.automatic.simplenote_1.3.4-44_minAPI15(nodpi)
.apk
Success
hefese@TUBITAK-HASANFSIMSEK3: /opt/genymobile/genymotion/tools$
```

Uygulamanın fiziksel / gerçek android sisteme yüklenip yüklenmediğini test ettiğimizde ana menüde uygulama görünecektir.



Uygulamaya girdiğimizde uygulama sorunsuz açılacaktır.



Kaynaklar

- <https://www.howtogeek.com/125769/how-to-install-and-use-abd-the-android-debug-bridge-utility/>
- <https://www.howtogeek.com/258788/what-is-usb-debugging-and-is-it-safe-to-leave-it-enabled-on-android/>
- <https://stackoverflow.com/questions/23415746/how-to-resolve-the-adb-server-didnt-ack-error>
- <https://developer.android.com/studio/command-line/adb.html>
- <https://gurelahmet.com/mobil-android-s%C4%B1zma-testine-giri%C5%9F/>
- <https://developer.android.com/studio?pkg=tools>
- https://wiki.lineageos.org/adb_fastboot_guide.html
- <https://www.makeuseof.com/tag/what-is-usb-debugging-mode-on-android-makeuseof-explains/>
- <https://developer.android.com/studio/releases/platform-tools>
- <https://stackoverflow.com/questions/2507960/does-android-keep-the-apk-files-if-so-where/32312241>
- <https://stackoverflow.com/questions/39731695/can-multiple-adb-clients-connect-to-a-single-android-adbd-over-wireless>
- <https://www.raywenderlich.com/621437-android-debug-bridge-adb-beyond-the-basics>
- <https://www.xda-developers.com/what-is-adb/>
- <https://developer.android.com/studio/command-line/adb.html>
- <https://stackoverflow.com/questions/43973838/how-to-connect-multiple-android-devices-with-adb-over-wifi>