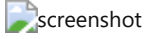


scrcpy (v1.18)

Bu uygulama Android cihazların USB (ya da [TCP/IP](#)) üzerinden görüntülenmesini ve kontrol edilmesini sağlar. *root* erişimine ihtiyaç duymaz. *GNU/Linux*, *Windows* ve *macOS* sistemlerinde çalışabilir.



Öne çıkan özellikler:

- **hafiflik** (doğal, sadece cihazın ekranını gösterir)
- **performans** (30~60fps)
- **kalite** (1920×1080 ya da üzeri)
- **düşük gecikme süresi** ([35~70ms](#))
- **düşük başlangıç süresi** (~1 saniye ilk kareyi gösterme süresi)
- **müdahaleci olmama** (cihazda kurulu yazılım kalmaz)

Gereksinimler

Android cihaz en düşük API 21 (Android 5.0) olmalıdır.

[Adb hata ayıklamasının](#) cihazınızda aktif olduğundan emin olun.

Bazı cihazlarda klavye ve fare ile kontrol için [ilave bir seçenek](#) daha etkinleştirmeniz gerekebilir.

Uygulamayı indirin

Özet

- Linux: `apt install scrcpy`
- Windows: [indir][direct-win64]
- macOS: `brew install scrcpy`

Kaynak kodu derle: [BUILD](#) ([basitleştirilmiş süreç](#))

Linux

Debian (şimdilik *testing* ve *sid*) ve Ubuntu (20.04) için:

```
apt install scrcpy
```

[Snap](#) paketi: [scrcpy](#) .

Fedora için, [COPR](#) paketi: [scrcpy](#) .

Arch Linux için, [AUR](#) paketi: [scrcpy](#) .

Gentoo için, [Ebuild](#) mevcut: [scrcpy/](#) .

Ayrıca [uygulamayı el ile de derleyebilirsiniz](#) ([basitleştirilmiş süreç](#)).

Windows

Windows için (`adb` dahil) tüm gereksinimleri ile derlenmiş bir arşiv mevcut:

- [README](#)

[Chocolatey](#) ile kurulum:

```
choco install scrcpy
choco install adb      # if you don't have it yet
```

[Scoop](#) ile kurulum:

```
scoop install scrcpy
scoop install adb      # if you don't have it yet
```

Ayrıca [uygulamayı el ile de derleyebilirsiniz](#).

macOS

Uygulama [Homebrew](#) içerisinde mevcut. Sadece kurun:

```
brew install scrcpy
```

adb , PATH içerisinde erişilebilir olmalıdır. Eğer değilse:

```
brew install android-platform-tools
```

[MacPorts](#) kullanılarak adb ve uygulamanın birlikte kurulumu yapılabilir:

```
sudo port install scrcpy
```

Ayrıca [uygulamayı el ile de derleyebilirsiniz](#).

Çalıştırma

Android cihazınızı bağlayın ve aşağıdaki komutu çalıştırın:

```
scrcpy
```

Komut satırı argümanları aşağıdaki komut ile listelenebilir:

```
scrcpy --help
```

Özellikler

Ekran yakalama ayarları

Boyut azaltma

Bazen, Android cihaz ekranını daha düşük seviyede göstermek performansı artırabilir.

Packaging status

| | |
|------------------------|---------------|
| Alpine Linux 3.16 | 1.24 |
| Alpine Linux Edge | 1.24 |
| ALT Linux p9 | 1.16 |
| ALT Linux p10 | 1.21 |
| ALT Sisyphus | 1.21 |
| antiX-19 | 1.12.1 |
| AOSC | 1.24 |
| Arch | 1.24 |
| Arch Linux 32 i686 | 1.24 |
| Arch Linux 32 pentium4 | 1.24 |
| Arch Linux ARM aarch64 | 1.24 |
| AUR | 1.17.r3.ge... |
| Chocolatey | 1.24 |
| Debian 11 | 1.17 |
| Debian 11 Backports | 1.23 |
| Debian 12 | 1.24 |
| Debian Unstable | 1.24 |
| Devuan 4.0 | 1.17 |
| Devuan Unstable | 1.24 |
| DPorts | 1.9 |
| FreeBSD Ports | 1.24 |
| Funtoo 1.4 | 1.24 |
| Gentoo | 1.24 |
| Homebrew | 1.24 |
| Kali Linux Rolling | 1.24 |
| LiGurOS stable | 1.24 |
| LiGurOS develop | 1.24 |
| MacPorts | 1.24 |
| Manjaro Stable | 1.24 |
| Manjaro Testing | 1.24 |
| Manjaro Unstable | 1.24 |
| MPR | 1.24 |
| MSYS2 mingw | 1.24 |
| MX Linux MX-17 | 1.12.1 |
| MX Linux MX-19 | 1.12.1 |
| nixpkgs stable 21.05 | 1.17 |
| nixpkgs stable 21.11 | 1.20 |
| nixpkgs stable 22.05 | 1.24 |
| nixpkgs unstable | 1.24 |
| OpenMandriva 4.1 | 1.12.1 |
| OpenMandriva 4.2 | 1.17 |
| OpenMandriva Rolling | 1.24 |
| OpenMandriva Cooker | 1.24 |

Hem genişliği hem de yüksekliği bir değere sabitlemek için (ör. 1024):

```
scrcpy --max-size 1024
scrcpy -m 1024 # kısa versiyon
```

Diğer boyut en-boy oranı korunacak şekilde hesaplanır. Bu şekilde ekran boyutu 1920x1080 olan bir cihaz 1024x576 olarak görünür.

Bit-oranı değiştirme

Varsayılan bit-oranı 8 Mbps'dir. Değiştirmek için (ör. 2 Mbps):

```
scrcpy --bit-rate 2M
scrcpy -b 2M # kısa versiyon
```

Çerçeve oranı sınırlama

Ekran yakalama için maksimum çerçeve oranı için sınır koyulabilir:

```
scrcpy --max-fps 15
```

Bu özellik Android 10 ve sonrası sürümlerde resmi olarak desteklenmektedir, ancak daha önceki sürümlerde çalışmayabilir.

Kesme

Cihaz ekranının sadece bir kısmı görünecek şekilde kesilebilir.

Bu özellik Oculus Go'nun bir gözünü yakalamak gibi durumlarda kullanışlı olur:

```
scrcpy --crop 1224:1440:0:0 # (0,0) noktasından 1224x1440
```

Eğer `--max-size` belirtilmişse yeniden boyutlandırma kesme işleminden sonra yapılır.

Video yönünü kilitleme

Videonun yönünü kilitlemek için:

```
scrcpy --lock-video-orientation # başlangıç yönü
scrcpy --lock-video-orientation=0 # doğal yön
scrcpy --lock-video-orientation=1 # 90° saatin tersi yönü
scrcpy --lock-video-orientation=2 # 180°
scrcpy --lock-video-orientation=3 # 90° saat yönü
```

Bu özellik kaydetme yönünü de etkiler.

[Pencere ayrı olarak döndürülmüş](#) olabilir.

Kodlayıcı

Bazı cihazlar birden fazla kodlayıcıya sahiptir, ve bunların bazıları programın kapanmasına sebep olabilir. Bu durumda farklı bir kodlayıcı seçilebilir:

| | |
|-------------------|--------|
| Parabola | 1.24 |
| Pardus 21 | 1.17 |
| Parrot | 1.17 |
| Pisi Linux | 1.24 |
| PureOS landing | 1.17 |
| Raspbian Stable | 1.17 |
| Raspbian Testing | 1.24 |
| RPM Sphere | 1.24 |
| Scoop | 1.24 |
| SlackBuilds | 1.24 |
| Solus | 1.24 |
| Trisquel 10.0 | 1.12.1 |
| Ubuntu 20.04 | 1.12.1 |
| Ubuntu 22.04 | 1.21 |
| Ubuntu 22.10 | 1.24 |
| Void Linux x86_64 | 1.24 |

```
scrcpy --encoder OMX.qcom.video.encoder.avc
```

Mevcut kodlayıcıları listelemek için geçerli olmayan bir kodlayıcı ismi girebilirsiniz, hata mesajı mevcut kodlayıcıları listeleyecektir:

```
scrcpy --encoder _
```

Yakalama

Kaydetme

Ekran yakalama sırasında kaydedilebilir:

```
scrcpy --record file.mp4  
scrcpy -r file.mkv
```

Yakalama olmadan kayıt için:

```
scrcpy --no-display --record file.mp4  
scrcpy -Nr file.mkv  
# Ctrl+C ile kayıt kesilebilir
```

"Atlanan kareler" gerçek zamanlı olarak gösterilmese (performans sebeplerinden ötürü) dahi kaydedilir. Kareler cihazda *zamandamgası* ile saklanır, bu sayede [paket gecikme varyasyonu](#) kayıt edilen dosyayı etkilemez.

v4l2loopback

Linux'ta video akışı bir v4l2 loopback cihazına gönderilebilir. Bu sayede Android cihaz bir web kamerası gibi davranabilir.

Bu işlem için `v4l2loopback` modülü kurulu olmalıdır:

```
sudo apt install v4l2loopback-dkms
```

v4l2 cihazı oluşturmak için:

```
sudo modprobe v4l2loopback
```

Bu komut `/dev/videoN` adresinde `N` yerine bir tamsayı koyarak yeni bir video cihazı oluşturacaktır. (birden fazla cihaz oluşturmak veya spesifik ID'ye sahip cihazlar için diğer [seçenekleri](#) inceleyebilirsiniz.)

Aktif cihazları listelemek için:

```
# v4l-utils paketi ile  
v4l2-ctl --list-devices  
  
# daha basit ama yeterli olabilecek şekilde  
ls /dev/video*
```

v4l2 kullanarak scrpy kullanmaya başlamak için:

```
scrpy --v4l2-sink=/dev/videoN
scrpy --v4l2-sink=/dev/videoN --no-display # ayna penceresini kapatarak
scrpy --v4l2-sink=/dev/videoN -N # kısa versiyon
```

(N harfini oluşturulan cihaz ID numarası ile değiştirin. `ls /dev/video*` cihaz ID'lerini görebilirsiniz.)

Aktifleştirildikten sonra video akışını herhangi bir v4l2 özellikli araçla açabilirsiniz:

```
ffplay -i /dev/videoN
vlc v4l2:///dev/videoN # VLC kullanırken yükleme gecikmesi olabilir
```

Örneğin, [OBS](#) ile video akışını kullanabilirsiniz.

Bağlantı

Kablosuz

Scrpy cihazla iletişim kurmak için `adb` 'yi kullanır, Ve `adb` bir cihaza TCP/IP kullanarak [bağlanabilir](#).

1. Cihazınızı bilgisayarınızla aynı Wi-Fi ağına bağlayın.
2. Cihazınızın IP adresini bulun. Ayarlar → Telefon Hakkında → Durum sekmesinden veya aşağıdaki komutu çalıştırarak öğrenebilirsiniz:

```
adb shell ip route | awk '{print $9}'
```

3. Cihazınızda TCP/IP üzerinden adb kullanımını etkinleştirin: `adb tcpip 5555` .
4. Cihazınızı bilgisayarınızdan sökün.
5. Cihazınıza bağlanın: `adb connect DEVICE_IP:5555` (*DEVICE_IP* değerini değiştirin).
6. `scrpy` komutunu normal olarak çalıştırın.

Bit-oranını ve büyüklüğü azaltmak yararlı olabilir:

```
scrpy --bit-rate 2M --max-size 800
scrpy -b2M -m800 # kısa versiyon
```

Birden fazla cihaz

Eğer `adb devices` komutu birden fazla cihaz listeliyorsa *serial* değerini belirtmeniz gerekir:

```
scrpy --serial 0123456789abcdef
scrpy -s 0123456789abcdef # kısa versiyon
```

Eğer cihaz TCP/IP üzerinden bağlanmışsa:

```
scrcpy --serial 192.168.0.1:5555
scrcpy -s 192.168.0.1:5555 # kısa version
```

Birden fazla cihaz için birden fazla *scrcpy* uygulaması çalıştırabilirsiniz.

Cihaz bağlantısı ile otomatik başlatma

[AutoAdb](#) ile yapılabilir:

```
autoadb scrcpy -s '{}'
```

SSH Tünel

Uzaktaki bir cihaza erişmek için lokal `adb` istemcisi, uzaktaki bir `adb` sunucusuna (aynı *adb* sürümünü kullanmak şartı ile) bağlanabilir :

```
adb kill-server # 5037 portunda çalışan lokal adb sunucusunu kapat
ssh -CN -L5037:localhost:5037 -R27183:localhost:27183 your_remote_computer
# bunu açık tutun
```

Başka bir terminalde:

```
scrcpy
```

Uzaktan port yönlendirme ileri yönlü bağlantı kullanabilirsiniz (`-R` yerine `-L` olduğuna dikkat edin):

```
adb kill-server # 5037 portunda çalışan lokal adb sunucusunu kapat
ssh -CN -L5037:localhost:5037 -L27183:localhost:27183 your_remote_computer
# bunu açık tutun
```

Başka bir terminalde:

```
scrcpy --force-adb-forward
```

Kablosuz bağlantı gibi burada da kalite düşürmek faydalı olabilir:

```
scrcpy -b2M -m800 --max-fps 15
```

Pencere ayarları

İsim

Cihaz modeli varsayılan pencere ismidir. Değiştirmek için:

```
scrcpy --window-title 'Benim cihazım'
```

Konum ve

Pencerenin başlangıç konumu ve boyutu belirtilebilir:

```
scrcpy --window-x 100 --window-y 100 --window-width 800 --window-height 600
```

Kenarlıklar

Pencere dekorasyonunu kapatmak için:

```
scrcpy --window-borderless
```

Her zaman üstte

Scrcpy penceresini her zaman üstte tutmak için:

```
scrcpy --always-on-top
```

Tam ekran

Uygulamayı tam ekran başlatmak için:

```
scrcpy --fullscreen  
scrcpy -f # kısa versiyon
```

Tam ekran `MOD+f` ile dinamik olarak değiştirilebilir.

Döndürme

Pencere döndürülebilir:

```
scrcpy --rotation 1
```

Seçilebilecek değerler:

- 0 : döndürme yok
- 1 : 90 derece saat yönünün tersi
- 2 : 180 derece
- 3 : 90 derece saat yönü

Döndürme `MOD+←(sol)` ve `MOD+→(sağ)` ile dinamik olarak değiştirilebilir.

`scrcpy`'de 3 farklı döndürme olduğuna dikkat edin:

- `MOD+r` cihazın yatay veya dikey modda çalışmasını sağlar. (çalışan uygulama istenilen oryantasyonda çalışmayı desteklemiyorsa döndürme işlemini reddedebilir.)
- [--lock-video-orientation](#) görüntü yakalama oryantasyonunu (cihazdan bilgisayara gelen video akışının oryantasyonu) değiştirir. Bu kayıt işlemini etkiler.
- `--rotation` (or `MOD+←/MOD+→`) pencere içeriğini dönderir. Bu sadece canlı görüntüyü etkiler, kayıt işlemini etkilemez.

Diğer ekran yakalama seçenekleri

Yazma korumalı

Kontrolleri devre dışı bırakmak için (cihazla etkileşime geçebilecek her şey: klavye ve fare girdileri, dosya sürükleyip bırakma):

```
scrcpy --no-control  
scrcpy -n
```

Ekran

Eğer cihazın birden fazla ekranı varsa hangi ekranın kullanılacağını seçebilirsiniz:

```
scrcpy --display 1
```

Kullanılabilecek ekranları listelemek için:

```
adb shell dumpsys display # çıktı içerisinde "mDisplayId=" terimini arayın
```

İkinci ekran ancak cihaz Android sürümü 10 veya üzeri olmalıdır (değilse yazma korumalı olarak görüntülenir).

Uyanık kalma

Cihazın uyku moduna girmesini engellemek için:

```
scrcpy --stay-awake  
scrcpy -w
```

scrcpy kapandığında cihaz başlangıç durumuna geri döner.

Ekranı kapatma

Ekran yakalama sırasında cihazın ekranı kapatılabilir:

```
scrcpy --turn-screen-off  
scrcpy -S
```

Ya da **MOD+O** kısayolunu kullanabilirsiniz.

Tekrar açmak için ise **MOD+Shift+O** tuşlarına basın.

Android'de, **GÜÇ** tuşu her zaman ekranı açar. Eğer **GÜÇ** sinyali scrcpy ile gönderilseyse (sağ tık veya **MOD+P**), ekran kısa bir gecikme ile kapanacaktır. Fiziksel **GÜÇ** tuşuna basmak hala ekranın açılmasına sebep olacaktır.

Bu cihazın uykuya geçmesini engellemek için kullanılabilir:

```
scrcpy --turn-screen-off --stay-awake  
scrcpy -Sw
```

Dokunuşları gösterme

Sunumlar sırasında fiziksel dokunuşları (fiziksel cihazdaki) göstermek faydalı olabilir.

Android'de bu özellik *Geliştirici seçenekleri* içerisinde bulunur.

Scrcpy bu özelliği çalışırken etkinleştirebilir ve kapanırken eski haline geri getirebilir:

```
scrcpy --show-touches  
scrcpy -t
```

Bu opsiyon sadece *fiziksel* dokunuşları (cihaz ekranındaki) gösterir.

Ekran koruyucuyu devre dışı bırakma

Scrcpy varsayılan ayarlarında ekran koruyucuyu devre dışı bırakmaz.

Bırakmak için:

```
scrcpy --disable-screensaver
```

Girdi kontrolü

Cihaz ekranını dönderme

MOD+r tuşları ile yatay ve dikey modlar arasında geçiş yapabilirsiniz.

Bu kısayol ancak çalışan uygulama desteklediği takdirde ekranı döndürecektir.

Kopyala yapıştır

Ne zaman Android cihazdaki pano değişse bilgisayardaki pano otomatik olarak senkronize edilir.

Tüm Ctrl kısayolları cihaza iletilir:

- Ctrl+c genelde kopyalar
- Ctrl+x genelde keser
- Ctrl+v genelde yapıştırır (bilgisayar ve cihaz arasındaki pano senkronizasyonundan sonra)

Bu kısayollar genelde beklediğiniz gibi çalışır.

Ancak kısayolun gerçekten yaptığı eylemi açık olan uygulama belirler. Örneğin, *Termux* Ctrl+c ile kopyalama yerine SIGINT sinyali gönderir, *K-9 Mail* ise yeni mesaj oluşturur.

Bu tip durumlarda kopyalama, kesme ve yapıştırma için (Android versiyon 7 ve üstü):

- MOD+c KOPYALA
- MOD+x KES
- MOD+v YAPIŞTIR (bilgisayar ve cihaz arasındaki pano senkronizasyonundan sonra)

Bunlara ek olarak, MOD+Shift+v tuşları bilgisayar pano içeriğini tuş basma eylemleri şeklinde gönderir. Bu metin yapıştırmayı desteklemeyen (*Termux* gibi) uygulamalar için kullanışlıdır, ancak ASCII olmayan içerikleri bozabilir.

UYARI: Bilgisayar pano içeriğini cihaza yapıştırmak (Ctrl+v ya da MOD+v tuşları ile) içeriği cihaz panosuna kopyalar. Sonuç olarak, herhangi bir Android uygulaması içeriğe erişebilir. Hassas içerikler (parolalar gibi) için bu özelliği kullanmaktan kaçınin.

Bazı cihazlar pano değişiklikleri konusunda beklenen şekilde çalışmayabilir. Bu durumlarda --legacy-paste argümanı kullanılabilir. Bu sayede Ctrl+v ve MOD+v tuşları da pano içeriğini tuş basma eylemleri şeklinde gönderir (MOD+Shift+v ile aynı şekilde).

İki parmak ile yakınlaştırma

"İki parmak ile yakınlaştırma" için: `Ctrl+tıkla-ve-sürükle`.

Daha açıklayıcı şekilde, `Ctrl` tuşuna sol-tık ile birlikte basılı tutun. Sol-tık serbest bırakılıncaya kadar yapılan tüm fare hareketleri ekran içeriğini ekranın merkezini baz alarak dönderir, büyütür veya küçültür (eğer uygulama destekliyorsa).

Scrcpy ekranın merkezinde bir "sanal parmak" varmış gibi davranır.

Metin gönderme tercihi

Metin girilirken iki çeşit [eylem](#) gerçekleştirilir:

- *tuş eylemleri*, bir tuşa basıldığı sinyali verir;
- *metin eylemleri*, bir metin girildiği sinyali verir.

Varsayılan olarak, harfler tuş eylemleri kullanılarak gönderilir. Bu sayede klavye oyunlarda beklenilene uygun olarak çalışır (Genelde WASD tuşları).

Ancak bu [bazı problemlere](#) yol açabilir. Eğer bu problemler ile karşılaşırsanız metin eylemlerini tercih edebilirsiniz:

```
scrcpy --prefer-text
```

(Ama bu oyunlardaki klavye davranışlarını bozacaktır)

Tuş tekrarı

Varsayılan olarak, bir tuşa basılı tutmak tuş eylemini tekrarlar. Bu durum bazı oyunlarda problemlere yol açabilir.

Tuş eylemlerinin tekrarını kapatmak için:

```
scrcpy --no-key-repeat
```

Sağ-tık ve Orta-tık

Varsayılan olarak, sağ-tık GERİ (ya da GÜÇ açma) eylemlerini, orta-tık ise ANA EKRAN eylemini tetikler. Bu kısayolları devre dışı bırakmak için:

```
scrcpy --forward-all-clicks
```

Dosya bırakma

APK kurulumu

APK kurmak için, bilgisayarınızdaki APK dosyasını (`.apk` ile biten) *scrcpy* penceresine sürükleyip bırakın.

Bu eylem görsel bir geri dönüt oluşturmaz, konsola log yazılır.

Dosyayı cihaza gönderme

Bir dosyayı cihazdaki `/sdcard/Download/` dizinine atmak için, (APK olmayan) bir dosyayı *scrcpy* penceresine sürükleyip bırakın.

Bu eylem görsel bir geri dönüt oluşturmaz, konsola log yazılır.

Hedef dizin uygulama başlatılırken değiştirilebilir:

```
scrcpy --push-target=/sdcard/Movies/
```

Ses iletimi

Scrcpy ses iletimi yapmaz. Yerine [sndcpy](#) kullanabilirsiniz.

Ayrıca bakınız [issue #14](#).

Kısayollar

Aşağıdaki listede, MOD kısayol tamamlayıcısıdır. Varsayılan olarak (sol) Alt veya (sol) Super tuşudur.

Bu tuş --shortcut-mod argümanı kullanılarak lctrl , rctrl , lalt , ralt , lsuper ve rsuper tuşlarından biri ile değiştirilebilir. Örneğin:

```
# Sağ Ctrl kullanmak için
scrcpy --shortcut-mod=rctrl

# Sol Ctrl, Sol Alt veya Sol Super tuşlarından birini kullanmak için
scrcpy --shortcut-mod=lctrl+lalt,lsuper
```

[Super](#) tuşu genelde Windows veya Cmd tuşudur.

| Action | Shortcut |
|--|-----------------------------------|
| Tam ekran modunu değiştirme | MOD+f |
| Ekrani sola çevirme | MOD+← (sol) |
| Ekrani sağa çevirme | MOD+→ (sağ) |
| Pencereyi 1:1 oranına çevirme (pixel-perfect) | MOD+g |
| Penceredeki siyah kenarlıkları kaldırma | MOD+w Çift-sol-tık ¹ |
| ANA EKRAN tuşu | MOD+h Orta-tık |
| GERİ tuşu | MOD+b Sağ-tık ² |
| UYGULAMA_DEĞİŞTİR tuşu | MOD+s 4.tık ³ |
| MENÜ tuşu (ekran kilidini açma) | MOD+m |
| SES_AÇ tuşu | MOD+↑ (yukarı) |
| SES_KIS tuşu | MOD+↓ (aşağı) |
| GÜÇ tuşu | MOD+p |
| Gücü açma | Sağ-tık ² |
| Cihaz ekranını kapatma (ekran yakalama durmadan) | MOD+o |
| Cihaz ekranını açma | MOD+Shift+o |

| | |
|--|-----------------------------------|
| Cihaz ekranını dönderme | MOD+x |
| Bildirim panelini genişletme | MOD+n 5.tık ³ |
| Ayarlar panelini genişletme | MOD+n+n Çift-5.tık ³ |
| Panelleri kapatma | MOD+Shift+n |
| Panoya kopyalama ⁴ | MOD+c |
| Panoya kesme ⁴ | MOD+x |
| Panoları senkronize ederek yapıştırma ⁴ | MOD+v |
| Bilgisayar panosundaki metini girme | MOD+Shift+v |
| FPS sayacını açma/kapatma (terminalde) | MOD+i |
| İki parmakla yakınlaştırma | Ctrl+tıkla-ve-sürükle |

¹Siyah kenarlıkları silmek için üzerine çift tıklayın.

²Sağ-tık ekran kapalıysa açar, değilse GERİ sinyali gönderir.

³4. ve 5. fare tuşları (eğer varsa).

⁴Sadece Android 7 ve üzeri versiyonlarda.

Tekrarlı tuşu olan kısayollar tuş bırakılıp tekrar basılarak tekrar çalıştırılır. Örneğin, "Ayarlar panelini genişletmek" için:

1. MOD tuşuna basın ve basılı tutun.
2. n tuşuna iki defa basın.
3. MOD tuşuna basmayı bırakın.

Tüm Ctrl+tuş kısayolları cihaza gönderilir. Bu sayede istenilen komut uygulama tarafından çalıştırılır.

Özel izinler

Varsayılandan farklı bir adb programı çalıştırmak için ADB ortam değişkenini ayarlayın:

```
ADB=/path/to/adb scrpcy
```

scrpcy-server programının dizinini değiştirmek için SCRCPY_SERVER_PATH değişkenini ayarlayın.

Neden scrpcy?

Bir meslektaşım [gnirehtet](#) gibi söylenmesi zor bir isim bulmam için bana meydan okudu.

[strcpv](#) string kopyalıyor; [scrpcy](#) screen kopyalıyor.

Nasıl derlenir?

Bakınız [BUILD](#).

Yaygın problemler

Bakınız [FAQ](#).

Geliştiriciler

[Geliştiriciler sayfası](#)nı okuyun.

Lisans

Copyright (C) 2018 Genymobile

Copyright (C) 2018-2022 Romain Vimont

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Makaleler

- [Introducing scrpcy](#)
- [Scrcpy now works wirelessly](#)