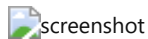


Only the original [README](#) is guaranteed to be up-to-date.

## scrcpy (v1.16)

Aplikasi ini menyediakan tampilan dan kontrol perangkat Android yang terhubung pada USB (atau [melalui TCP/IP](#)). Ini tidak membutuhkan akses *root* apa pun. Ini bekerja pada *GNU/Linux*, *Windows* and *macOS*.



Ini berfokus pada:

- **keringanan** (asli, hanya menampilkan layar perangkat)
- **kinerja** (30~60fps)
- **kualitas** (1920×1080 atau lebih)
- **latensi** rendah ([35~70ms](#))
- **waktu startup rendah** (~1 detik untuk menampilkan gambar pertama)
- **tidak mengganggu** (tidak ada yang terpasang di perangkat)

## Persyaratan

Perangkat Android membutuhkan setidaknya API 21 (Android 5.0).

Pastikan Anda [mengaktifkan debugging.adb](#) pada perangkat Anda.

Di beberapa perangkat, Anda juga perlu mengaktifkan [opsi tambahan](#) untuk mengontrolnya menggunakan keyboard dan mouse.

## Dapatkan aplikasinya

### Linux

Di Debian (*testing* dan *sid* untuk saat ini) dan Ubuntu (20.04):

```
apt install scrcpy
```

Paket [Snap](#) tersedia: [scrcpy](#) .

Untuk Fedora, paket [COPR](#) tersedia: [scrcpy](#) .

Untuk Arch Linux, paket [AUR](#) tersedia: [scrcpy](#) .

Untuk Gentoo, tersedia [Ebuild](#): [scrcpy/](#) .

Anda juga bisa [membangun aplikasi secara manual](#) (jangan khawatir, tidak terlalu sulit).

### Windows

Untuk Windows, untuk kesederhanaan, arsip prebuilt dengan semua dependensi (termasuk `adb`) tersedia :

- [README](#)

Ini juga tersedia di [Chocolatey](#):

```
choco install scrcpy
choco install adb      # jika Anda belum memilikinya
```

Dan di [Scoop](#):

```
scoop install scrcpy
scoop install adb      # jika Anda belum memilikinya
```

Anda juga dapat [membangun aplikasi secara manual](#).

## macOS

Aplikasi ini tersedia di [Homebrew](#). Instal saja:

```
brew install scrcpy
```

Anda membutuhkan `adb`, dapat diakses dari `PATH` Anda. Jika Anda belum memilikinya:

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

Anda juga dapat [membangun aplikasi secara manual](#).

## Menjalankan

Pasang perangkat Android, dan jalankan:

```
scrcpy
```

Ini menerima argumen baris perintah, didaftarkan oleh:

```
scrcpy --help
```

## Fitur

### Menangkap konfigurasi

#### Mengurangi ukuran

Kadang-kadang, berguna untuk mencerminkan perangkat Android dengan definisi yang lebih rendah untuk meningkatkan kinerja.

Untuk membatasi lebar dan tinggi ke beberapa nilai (mis. 1024):

```
scrcpy --max-size 1024
scrcpy -m 1024 # versi pendek
```

Dimensi lain dihitung agar rasio aspek perangkat dipertahankan. Dengan begitu, perangkat 1920×1080 akan dicerminkan pada 1024×576.

### Ubah kecepatan bit

Kecepatan bit default adalah 8 Mbps. Untuk mengubah bitrate video (mis. Menjadi 2 Mbps):

```
scrcpy --bit-rate 2M
scrcpy -b 2M # versi pendek
```

### Batasi frekuensi gambar

Kecepatan bingkai pengambilan dapat dibatasi:

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

Ini secara resmi didukung sejak Android 10, tetapi dapat berfungsi pada versi sebelumnya.

### Memotong

Layar perangkat dapat dipotong untuk mencerminkan hanya sebagian dari layar.

Ini berguna misalnya untuk mencerminkan hanya satu mata dari Oculus Go:

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

Jika `--max-size` juga ditentukan, perubahan ukuran diterapkan setelah pemotongan.

### Kunci orientasi video

Untuk mengunci orientasi pencerminan:

```
scrcpy --lock-video-orientation 0 # orientasi alami
scrcpy --lock-video-orientation 1 # 90° berlawanan arah jarum jam
scrcpy --lock-video-orientation 2 # 180°
scrcpy --lock-video-orientation 3 # 90° searah jarum jam
```

Ini mempengaruhi orientasi perekaman.

### Rekaman

Anda dapat merekam layar saat melakukan mirroring:

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

Untuk menonaktifkan pencerminan saat merekam:

```
scrcpy --no-display --record file.mp4
scrcpy -Nr file.mkv
# berhenti merekam dengan Ctrl+C
```

"Skipped frames" are recorded, even if they are not displayed in real time (for performance reasons). Frames are *timestamped* on the device, so [packet delay variation] does not impact the recorded file.

"Frame yang dilewati" direkam, meskipun tidak ditampilkan secara real time (untuk alasan performa). Bingkai *diberi stempel waktu* pada perangkat, jadi [variasi penundaan paket](#) tidak memengaruhi file yang direkam.

## Koneksi

### Wireless

`Scrcpy` menggunakan `adb` untuk berkomunikasi dengan perangkat, dan `adb` dapat [terhubung](#) ke perangkat melalui TCP / IP:

1. Hubungkan perangkat ke Wi-Fi yang sama dengan komputer Anda.
2. Dapatkan alamat IP perangkat Anda (dalam Pengaturan → Tentang ponsel → Status).
3. Aktifkan adb melalui TCP / IP pada perangkat Anda: `adb tcpip 5555`.
4. Cabut perangkat Anda.
5. Hubungkan ke perangkat Anda: `adb connect DEVICE_IP: 5555` (*ganti* `DEVICE_IP`).
6. Jalankan `scrcpy` seperti biasa.

Mungkin berguna untuk menurunkan kecepatan bit dan definisi:

```
scrcpy --bit-rate 2M --max-size 800
scrcpy -b2M -m800 # versi pendek
```

### Multi-perangkat

Jika beberapa perangkat dicantumkan di `adb devices`, Anda harus menentukan *serial*:

```
scrcpy --serial 0123456789abcdef
scrcpy -s 0123456789abcdef # versi pendek
```

If the device is connected over TCP/IP:

```
scrcpy --serial 192.168.0.1:5555
scrcpy -s 192.168.0.1:5555 # versi pendek
```

Anda dapat memulai beberapa contoh `scrcpy` untuk beberapa perangkat.

### Mulai otomatis pada koneksi perangkat

Anda bisa menggunakan [AutoAdb](#):

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

### Koneksi via SSH tunnel

Untuk menyambung ke perangkat jarak jauh, dimungkinkan untuk menghubungkan klien `adb` lokal ke server `adb` jarak jauh (asalkan mereka menggunakan versi yang sama dari `adb` protocol):

```
adb kill-server # matikan server adb lokal di 5037
ssh -CN -L5037:localhost:5037 -R27183:localhost:27183 komputer_jarak_jauh_anda
# jaga agar tetap terbuka
```

Dari terminal lain:

```
scrcpy
```

Untuk menghindari mengaktifkan penerusan port jarak jauh, Anda dapat memaksa sambungan maju sebagai gantinya (perhatikan `-L`, bukan `-R`):

```
adb kill-server      # matikan server adb lokal di 5037
ssh -CN -L5037:localhost:5037 -L27183:localhost:27183 komputer_jarak_jauh_anda
# jaga agar tetap terbuka
```

Dari terminal lain:

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

Seperti koneksi nirkabel, mungkin berguna untuk mengurangi kualitas:

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

## Konfigurasi Jendela

### Judul

Secara default, judul jendela adalah model perangkat. Itu bisa diubah:

```
scrcpy --window-title 'Perangkat Saya'
```

### Posisi dan ukuran

Posisi dan ukuran jendela awal dapat ditentukan:

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

### Jendela tanpa batas

Untuk menonaktifkan dekorasi jendela:

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

### Selalu di atas

Untuk menjaga jendela scrcpy selalu di atas:

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

### Layar penuh

Aplikasi dapat dimulai langsung dalam layar penuh::

```
scrcpy --fullscreen
scrcpy -f # versi pendek
```

Layar penuh kemudian dapat diubah secara dinamis dengan `MOD+F`.

## Rotasi

Jendela mungkin diputar:

```
scrcpy --rotation 1
```

Nilai yang mungkin adalah:

- `0` : tidak ada rotasi
- `1` : 90 derajat berlawanan arah jarum jam
- `2` : 180 derajat
- `3` : 90 derajat searah jarum jam

Rotasi juga dapat diubah secara dinamis dengan `MOD+←` (*kiri*) and `MOD+→` (*kanan*).

Perhatikan bahwa *scrcpy* mengelola 3 rotasi berbeda::

- `MOD+r` meminta perangkat untuk beralih antara potret dan lanskap (aplikasi yang berjalan saat ini mungkin menolak, jika mendukung orientasi yang diminta).
- `--lock-video-orientation` mengubah orientasi pencerminan (orientasi video yang dikirim dari perangkat ke komputer). Ini mempengaruhi rekaman.
- `--rotation` (atau `MOD+←/MOD+→`) memutar hanya konten jendela. Ini hanya mempengaruhi tampilan, bukan rekaman.

## Opsi pencerminan lainnya

### Hanya-baca

Untuk menonaktifkan kontrol (semua yang dapat berinteraksi dengan perangkat: tombol input, peristiwa mouse, seret & lepas file):

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

## Layar

Jika beberapa tampilan tersedia, Anda dapat memilih tampilan untuk cermin:

```
scrcpy --display 1
```

Daftar id tampilan dapat diambil dengan::

```
adb shell dumpsys display # cari "mDisplayId=" di keluaran
```

Tampilan sekunder hanya dapat dikontrol jika perangkat menjalankan setidaknya Android 10 (jika tidak maka akan dicerminkan dalam hanya-baca).

## Tetap terjaga

Untuk mencegah perangkat tidur setelah beberapa penundaan saat perangkat dicolokkan:

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

Keadaan awal dipulihkan ketika scrcpy ditutup.

## Matikan layar

Dimungkinkan untuk mematikan layar perangkat saat pencerminan mulai dengan opsi baris perintah:

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

Atau dengan menekan `MOD+o` kapan saja.

Untuk menyalakannya kembali, tekan `MOD+Shift+o`.

Di Android, tombol `POWER` selalu menyalakan layar. Untuk kenyamanan, jika `POWER` dikirim melalui scrcpy (melalui klik kanan atau `MOD+p`), itu akan memaksa untuk mematikan layar setelah penundaan kecil (atas dasar upaya terbaik). Tombol fisik `POWER` masih akan menyebabkan layar dihidupkan.

Ini juga berguna untuk mencegah perangkat tidur:

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

## Render frame kedaluwarsa

Secara default, untuk meminimalkan latensi, *scrcpy* selalu menampilkan frame yang terakhir didekodekan tersedia, dan menghapus frame sebelumnya.

Untuk memaksa rendering semua frame (dengan kemungkinan peningkatan latensi), gunakan:

```
scrcpy --render-expired-frames
```

## Tunjukkan sentuhan

Untuk presentasi, mungkin berguna untuk menunjukkan sentuhan fisik (pada perangkat fisik).

Android menyediakan fitur ini di *Opsi Pengembang*.

*Scrcpy* menyediakan opsi untuk mengaktifkan fitur ini saat mulai dan mengembalikan nilai awal saat keluar:

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

Perhatikan bahwa ini hanya menunjukkan sentuhan *fisik* (dengan jari di perangkat).

## Nonaktifkan screensaver

Secara default, `scrcpy` tidak mencegah screensaver berjalan di komputer.

Untuk menonaktifkannya:

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

## Kontrol masukan

### Putar layar perangkat

Tekan `MOD+R` untuk beralih antara mode potret dan lanskap.

Perhatikan bahwa itu berputar hanya jika aplikasi di latar depan mendukung orientasi yang diminta.

### Salin-tempel

Setiap kali papan klip Android berubah, secara otomatis disinkronkan ke papan klip komputer.

Apa saja `Ctrl` pintasan diteruskan ke perangkat. Khususnya:

- `Ctrl+C` biasanya salinan
- `Ctrl+X` biasanya memotong
- `Ctrl+V` biasanya menempel (setelah sinkronisasi papan klip komputer-ke-perangkat)

Ini biasanya berfungsi seperti yang Anda harapkan.

Perilaku sebenarnya tergantung pada aplikasi yang aktif. Sebagai contoh, *Termux* mengirim SIGINT ke `Ctrl+C` sebagai gantinya, dan *K-9 Mail* membuat pesan baru.

Untuk menyalin, memotong dan menempel dalam kasus seperti itu (tetapi hanya didukung di Android >= 7):

- `MOD+C` injeksi `COPY` (*salin*)
- `MOD+X` injeksi `CUT` (*potong*)
- `MOD+V` injeksi `PASTE` (setelah sinkronisasi papan klip komputer-ke-perangkat)

Tambahan, `MOD+Shift+V` memungkinkan untuk memasukkan teks papan klip komputer sebagai urutan peristiwa penting. Ini berguna ketika komponen tidak menerima penempelan teks (misalnya di *Termux*), tetapi dapat merusak konten non-ASCII.

**PERINGATAN:** Menempelkan papan klip komputer ke perangkat (baik melalui `Ctrl+V` or `MOD+V`) menyalin konten ke clipboard perangkat. Akibatnya, aplikasi Android apa pun dapat membaca kontennya. Anda harus menghindari menempelkan konten sensitif (seperti kata sandi) seperti itu.

### Cubit untuk memperbesar/memperkecil

Untuk mensimulasikan "cubit-untuk-memperbesar/memperkecil": `Ctrl+klik-dan-pindah`.

Lebih tepatnya, tahan `Ctrl` sambil menekan tombol klik kiri. Hingga tombol klik kiri dilepaskan, semua gerakan mouse berskala dan memutar konten (jika didukung oleh aplikasi) relatif ke tengah layar.

Secara konkret, `scrcpy` menghasilkan kejadian sentuh tambahan dari "jari virtual" di lokasi yang dibalik melalui bagian tengah layar.

### Preferensi injeksi teks

Ada dua jenis [peristiwa](#) dihasilkan saat mengetik teks:



- *peristiwa penting*, menandakan bahwa tombol ditekan atau dilepaskan;
- *peristiwa teks*, menandakan bahwa teks telah dimasukkan.

Secara default, huruf dimasukkan menggunakan peristiwa kunci, sehingga keyboard berperilaku seperti yang diharapkan dalam game (biasanya untuk tombol WASD).

Tapi ini mungkin [menyebabkan masalah](#). Jika Anda mengalami masalah seperti itu, Anda dapat menghindarinya dengan:

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

(tapi ini akan merusak perilaku keyboard dalam game)

### Ulangi kunci

Secara default, menahan tombol akan menghasilkan peristiwa kunci yang berulang. Ini dapat menyebabkan masalah kinerja di beberapa game, di mana acara ini tidak berguna.

Untuk menghindari penerusan peristiwa penting yang berulang:

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

### Seret/jatuhkan file

#### Pasang APK

Untuk menginstal APK, seret & lepas file APK (diakhiri dengan `.apk`) ke jendela *scrcpy*.

Tidak ada umpan balik visual, log dicetak ke konsol.

#### Dorong file ke perangkat

Untuk mendorong file ke `/sdcard/` di perangkat, seret & jatuhkan file (non-APK) ke jendela *scrcpy*.

Tidak ada umpan balik visual, log dicetak ke konsol.

Direktori target dapat diubah saat mulai:

```
scrcpy --push-target /sdcard/foo/bar/
```

### Penerusan audio

Audio tidak diteruskan oleh *scrcpy*. Gunakan [sndcpy](#).

Lihat juga [Masalah #14](#).

### Pintasan

Dalam daftar berikut, `MOD` adalah pengubah pintasan. Secara default, ini (kiri) `Alt` atau (kiri) `Super`.

Ini dapat diubah menggunakan `--shortcut-mod`. Kunci yang memungkinkan adalah `lctrl`, `rctrl`, `lalt`, `ralt`, `lsuper` dan `rsuper`. Sebagai contoh:

```
# gunakan RCtrl untuk jalan pintas
sccrpy --shortcut-mod=rctrl

# gunakan baik LCtrl+LAlt atau LSuper untuk jalan pintas
sccrpy --shortcut-mod=lctrl+lalt,lsuper
```

[Super](#) biasanya adalah *Windows* atau *Cmd key*.

Aksi	Pintasan
Alihkan mode layar penuh	MOD+f
Putar layar kiri	MOD+← ( <i>kiri</i> )
Putar layar kanan	MOD+→ ( <i>kanan</i> )
Ubah ukuran jendela menjadi 1:1 (piksel-sempurna)	MOD+g
Ubah ukuran jendela menjadi hapus batas hitam	MOD+w   <i>klik-dua-kali</i> <sup>1</sup>
Klik HOME	MOD+h   <i>Klik-tengah</i>
Klik BACK	MOD+b   <i>Klik-kanan</i> <sup>2</sup>
Klik APP_SWITCH	MOD+s
Klik MENU (buka kunci layar)	MOD+m
Klik VOLUME_UP	MOD+↑ ( <i>naik</i> )
Klik VOLUME_DOWN	MOD+↓ ( <i>turun</i> )
Klik POWER	MOD+p
Menyalakan	<i>Klik-kanan</i> <sup>2</sup>
Matikan layar perangkat (tetap mirroring)	MOD+o
Hidupkan layar perangkat	MOD+Shift+o
Putar layar perangkat	MOD+r
Luaskan panel notifikasi	MOD+n
Ciutkan panel notifikasi	MOD+Shift+n
Menyalin ke papan klip <sup>3</sup>	MOD+c
Potong ke papan klip <sup>3</sup>	MOD+x
Sinkronkan papan klip dan tempel <sup>3</sup>	MOD+v
Masukkan teks papan klip komputer	MOD+Shift+v
Mengaktifkan/menonaktifkan penghitung FPS (di stdout)	MOD+i
Cubit-untuk-memperbesar/memperkecil	Ctrl+klik-dan-pindah

<sup>1</sup>Klik-dua-kali pada batas hitam untuk menghapusnya.

<sup>2</sup>Klik-kanan akan menghidupkan layar jika mati, tekan **BACK** jika tidak.

<sup>3</sup>Hanya di Android >= 7.

Semua **Ctrl+key** pintasan diteruskan ke perangkat, demikian adanya ditangani oleh aplikasi aktif.

## Jalur kustom

Untuk menggunakan biner *adb* tertentu, konfigurasi jalurnya di variabel lingkungan `ADB` :

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

Untuk mengganti jalur file `scrpcy-server` , konfigurasi jalurnya di `SCRCPY_SERVER_PATH` .

## Mengapa *scrpcy*?

Seorang kolega menantang saya untuk menemukan nama yang tidak dapat diucapkan seperti [gnirehtet](#).

`scrpcy` menyalin sebuah **string**; `scrpcy` menyalin sebuah **screen**.

## Bagaimana Cara membangun?

Lihat [BUILD](#).

## Masalah umum

Lihat [FAQ](#).

## Pengembang

Baca [halaman.pengembang](#).

## Lisensi

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.

## Artikel

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