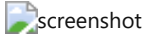


Apri il [README](#) originale e sempre aggiornato.

scrcpy (v1.19)

Questa applicazione fornisce la visualizzazione e il controllo dei dispositivi Android collegati via USB (o [via TCP/IP](#)).
Non richiede alcun accesso *root*. Funziona su *GNU/Linux*, *Windows* e *macOS*.



Si concentra su:

- **leggerezza** (nativo, mostra solo lo schermo del dispositivo)
- **prestazioni** (30~60fps)
- **qualità** (1920×1080 o superiore)
- **bassa latenza** ([35~70ms](#))
- **tempo di avvio basso** (~ 1secondo per visualizzare la prima immagine)
- **non invadenza** (nulla viene lasciato installato sul dispositivo)

Requisiti

Il dispositivo Android richiede almeno le API 21 (Android 5.0).

Assicurati di aver [attivato il debug usb](#) sul(/i) tuo(i) dispositivo(i).

In alcuni dispositivi, devi anche abilitare [un'opzione aggiuntiva](#) per controllarli con tastiera e mouse.

Ottieni l'app

Sommario

- Linux: `apt install scrcpy`
- Windows: [download](#)
- macOS: `brew install scrcpy`

Compila dai sorgenti: [BUILD](#) (in inglese) ([procedimento semplificato](#) (in inglese))

Linux

Su Debian (*testing* e *sid* per ora) e Ubuntu (20.04):

```
apt install scrcpy
```

È disponibile anche un pacchetto [Snap](#): [scrcpy](#) .

Per Fedora, è disponibile un pacchetto [COPR](#): [scrcpy](#) .

Per Arch Linux, è disponibile un pacchetto [AUR](#): [scrcpy](#) .

Per Gentoo, è disponibile una [Ebuild](#): [scrcpy/](#) .

Puoi anche [compilare l'app manualmente](#) (in inglese) ([procedimento semplificato](#) (in inglese)).

Windows

Per Windows, per semplicità è disponibile un archivio precompilato con tutte le dipendenze (incluso `adb`):

- [README](#) (Link al README originale per l'ultima versione)

È anche disponibile in [Chocolatey](#):

```
choco install scrcpy
choco install adb      # se non lo hai già
```

E in [Scoop](#):

```
scoop install scrcpy
scoop install adb      # se non lo hai già
```

Puoi anche [compilare l'app manualmente](#) (in inglese).

macOS

L'applicazione è disponibile in [Homebrew](#). Basta installarlo:

```
brew install scrcpy
```

Serve che `adb` sia accessibile dal tuo `PATH`. Se non lo hai già:

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

È anche disponibile in [MacPorts](#), che imposta `adb` per te:

```
sudo port install scrcpy
```

Puoi anche [compilare l'app manualmente](#) (in inglese).

Esecuzione

Collega un dispositivo Android ed esegui:

```
scrcpy
```

Scrcpy accetta argomenti da riga di comando, essi sono listati con:

```
scrcpy --help
```

Funzionalità

Configurazione di acquisizione

Riduci dimensione

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

Qualche volta è utile trasmettere un dispositivo Android ad una definizione inferiore per aumentare le prestazioni.

Per limitare sia larghezza che altezza ad un certo valore (ad es. 1024):

```
scrcpy --max-size 1024
scrcpy -m 1024 # versione breve
```

L'altra dimensione è calcolata in modo tale che il rapporto di forma del dispositivo sia preservato. In questo esempio un dispositivo in 1920x1080 viene trasmesso a 1024x576.

Cambia bit-rate (velocità di trasmissione)

Il bit-rate predefinito è 8 Mbps. Per cambiare il bitrate video (ad es. a 2 Mbps):

```
scrcpy --bit-rate 2M
scrcpy -b 2M # versione breve
```

Limitare il frame rate (frequenza di fotogrammi)

Il frame rate di acquisizione può essere limitato:

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

Questo è supportato ufficialmente a partire da Android 10, ma potrebbe funzionare in versioni precedenti.

Ritaglio

Lo schermo del dispositivo può essere ritagliato per visualizzare solo parte di esso.

Questo può essere utile, per esempio, per trasmettere solo un occhio dell'Oculus Go:

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

Se anche `--max-size` è specificata, il ridimensionamento è applicato dopo il ritaglio.

Blocca orientamento del video

Per bloccare l'orientamento della trasmissione:

```
scrcpy --lock-video-orientation # orientamento iniziale (corrente)
scrcpy --lock-video-orientation=0 # orientamento naturale
scrcpy --lock-video-orientation=1 # 90° antiorario
scrcpy --lock-video-orientation=2 # 180°
scrcpy --lock-video-orientation=3 # 90° orario
```

Questo influisce sull'orientamento della registrazione.

La [finestra può anche essere ruotata](#) indipendentemente.

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

Codificatore

Alcuni dispositivi hanno più di un codificatore e alcuni di questi possono provocare problemi o crash. È possibile selezionare un encoder diverso:

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

Per elencare i codificatori disponibili puoi immettere un nome di codificatore non valido e l'errore mostrerà i codificatori disponibili:

```
scrcpy --encoder _
```

Cattura

Registrazione

È possibile registrare lo schermo durante la trasmissione:

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

Per disabilitare la trasmissione durante la registrazione:

```
scrcpy --no-display --record file.mp4
scrcpy -Nr file.mkv
# interrompere la registrazione con Ctrl+C
```

I "fotogrammi saltati" sono registrati nonostante non siano mostrati in tempo reale (per motivi di prestazioni). I fotogrammi sono *datati* sul dispositivo, così una [variazione di latenza dei pacchetti](#) non impatta il file registrato.

v4l2loopback

Su Linux è possibile inviare il flusso video ad un dispositivo v4l2 loopback, cosicchè un dispositivo Android possa essere aperto come una webcam da qualsiasi strumento compatibile con v4l2.

Il modulo `v4l2loopback` deve essere installato:

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

Per creare un dispositivo v4l2:

```
sudo modprobe v4l2loopback
```

Questo creerà un nuovo dispositivo video in `/dev/videoN` dove `N` è un intero (più [opzioni](#) sono disponibili per creare più dispositivi o dispositivi con ID specifici).

Per elencare i dispositivi attivati:

```
# necessita del pacchetto v4l-utils
v4l2-ctl --list-devices
```

```
# semplice ma potrebbe essere sufficiente
ls /dev/video*
```

Per avviare scrcpy utilizzando un v4l2 sink:

```
scrcpy --v4l2-sink=/dev/videoN
scrcpy --v4l2-sink=/dev/videoN --no-display # disabilita la finestra di
trasmissione
scrcpy --v4l2-sink=/dev/videoN -N          # versione corta
```

(sostituisci `N` con l'ID del dispositivo, controlla con `ls /dev/video*`)

Una volta abilitato, puoi aprire il tuo flusso video con uno strumento compatibile con v4l2:

```
ffplay -i /dev/videoN
vlc v4l2:///dev/videoN # VLC potrebbe aggiungere del ritardo per il buffer
```

Per esempio potresti catturare il video in [OBS](#).

Buffering

È possibile aggiungere del buffer. Questo aumenta la latenza ma riduce il jitter (vedi [#2464](#)).

L'opzione è disponibile per il buffer della visualizzazione:

```
scrcpy --display-buffer=50 # aggiungi 50 ms di buffer per la visualizzazione
```

e per il V4L2 sink:

```
scrcpy --v4l2-buffer=500 # aggiungi 50 ms di buffer per il v4l2 sink
```

Connessione

Wireless

Scrcpy usa `adb` per comunicare col dispositivo e `adb` può [connettersi](#) al dispositivo mediante TCP/IP:

1. Connetti il dispositivo alla stessa rete Wi-Fi del tuo computer.
2. Trova l'indirizzo IP del tuo dispositivo in Impostazioni → Informazioni sul telefono → Stato, oppure eseguendo questo comando:

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

3. Abilita adb via TCP/IP sul tuo dispositivo: `adb tcpip 5555` .
4. Scollega il tuo dispositivo.
5. Connetti il tuo dispositivo: `adb connect IP_DISPOSITIVO:5555` (*rimpiazza* `IP_DISPOSITIVO`).
6. Esegui `scrcpy` come al solito.

Potrebbe essere utile diminuire il bit-rate e la definizione

```
scrcpy --bit-rate 2M --max-size 800  
scrcpy -b2M -m800 # versione breve
```

Multi dispositivo

Se in `adb devices` sono listati più dispositivi, è necessario specificare il *seriale*:

```
scrcpy --serial 0123456789abcdef  
scrcpy -s 0123456789abcdef # versione breve
```

Se il dispositivo è collegato mediante TCP/IP:

```
scrcpy --serial 192.168.0.1:5555  
scrcpy -s 192.168.0.1:5555 # versione breve
```

Puoi avviare più istanze di *scrcpy* per diversi dispositivi.

Avvio automatico alla connessione del dispositivo

Potresti usare [AutoAdb](#):

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

Tunnel SSH

Per connettersi a un dispositivo remoto è possibile collegare un client `adb` locale ad un server `adb` remoto (assunto che entrambi stiano usando la stessa versione del protocollo *adb*):

```
adb kill-server # termina il server adb locale su 5037  
ssh -CN -L5037:localhost:5037 -R27183:localhost:27183 your_remote_computer  
# tieni questo aperto
```

Da un altro terminale:

```
scrcpy
```

Per evitare l'abilitazione dell'apertura porte remota potresti invece forzare una "forward connection" (notare il `-L` invece di `-R`)

```
adb kill-server # termina il server adb locale su 5037  
ssh -CN -L5037:localhost:5037 -L27183:localhost:27183 your_remote_computer  
# tieni questo aperto
```

Da un altro terminale:

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

Come per le connessioni wireless potrebbe essere utile ridurre la qualità:

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

Configurazione della finestra

Titolo

Il titolo della finestra è il modello del dispositivo per impostazione predefinita. Esso può essere cambiato:

```
scrcpy --window-title 'My device'
```

Posizione e dimensione

La posizione e la dimensione iniziale della finestra può essere specificata:

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

Senza bordi

Per disabilitare le decorazioni della finestra:

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

Sempre in primo piano

Per tenere scrcpy sempre in primo piano:

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

Schermo intero

L'app può essere avviata direttamente a schermo intero:

```
scrcpy --fullscreen  
scrcpy -f # versione breve
```

Lo schermo intero può anche essere attivato/disattivato con `MOD+f`.

Rotazione

La finestra può essere ruotata:

```
scrcpy --rotation 1
```

I valori possibili sono:

- 0 : nessuna rotazione
- 1 : 90 gradi antiorari
- 2 : 180 gradi

- `3` : 90 gradi orari

La rotazione può anche essere cambiata dinamicamente con `MOD+←` (*sinistra*) e `MOD+→` (*destra*).

Notare che *scrcpy* gestisce 3 diversi tipi di rotazione:

- `MOD+r` richiede al dispositivo di cambiare tra orientamento verticale (portrait) e orizzontale (landscape) (l'app in uso potrebbe rifiutarsi se non supporta l'orientamento richiesto).
- `--lock-video-orientation` cambia l'orientamento della trasmissione (l'orientamento del video inviato dal dispositivo al computer). Questo influenza la registrazione.
- `--rotation` (o `MOD+←/MOD+→`) ruota solo il contenuto della finestra. Questo influenza solo la visualizzazione, non la registrazione.

Altre opzioni di trasmissione

"Sola lettura"

Per disabilitare i controlli (tutto ciò che può interagire col dispositivo: tasti di input, eventi del mouse, trascina e rilascia (drag&drop) file):

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

Schermo

Se sono disponibili più schermi, è possibile selezionare lo schermo da trasmettere:

```
scrcpy --display 1
```

La lista degli id schermo può essere ricavata da:

```
adb shell dumpsys display # cerca "mDisplayId=" nell'output
```

Lo schermo secondario potrebbe essere possibile controllarlo solo se il dispositivo esegue almeno Android 10 (in caso contrario è trasmesso in modalità sola lettura).

Mantenere sbloccato

Per evitare che il dispositivo si blocchi dopo un po' che il dispositivo è collegato:

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

Lo stato iniziale è ripristinato quando *scrcpy* viene chiuso.

Spegnere lo schermo

È possibile spegnere lo schermo del dispositivo durante la trasmissione con un'opzione da riga di comando:

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


Oppure premendo `MOD+o` in qualsiasi momento.

Per riaccenderlo premere `MOD+Shift+o`.

In Android il pulsante `POWER` (tasto di accensione) accende sempre lo schermo. Per comodità, se `POWER` è inviato via `scrcpy` (con click destro o con `MOD+p`), si forza il dispositivo a spegnere lo schermo dopo un piccolo ritardo (appena possibile). Il pulsante fisico `POWER` continuerà ad accendere lo schermo normalmente.

Può anche essere utile evitare il blocco del dispositivo:

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

Mostrare i tocchi

Per le presentazioni può essere utile mostrare i tocchi fisici (sul dispositivo fisico).

Android fornisce questa funzionalità nelle *Opzioni sviluppatore*.

`Scrcpy` fornisce un'opzione per abilitare questa funzionalità all'avvio e ripristinare il valore iniziale alla chiusura:

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

Notare che mostra solo i tocchi *fisici* (con le dita sul dispositivo).

Disabilitare il salvaschermo

In maniera predefinita `scrcpy` non previene l'attivazione del salvaschermo del computer.

Per disabilitarlo:

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

Input di controlli

Rotazione dello schermo del dispositivo

Premere `MOD+r` per cambiare tra le modalità verticale (portrait) e orizzontale (landscape).

Notare che la rotazione avviene solo se l'applicazione in primo piano supporta l'orientamento richiesto.

Copia-incolla

Quando gli appunti di Android cambiano, essi vengono automaticamente sincronizzati con gli appunti del computer.

Qualsiasi scorciatoia `Ctrl` viene inoltrata al dispositivo. In particolare:

- `Ctrl+c` copia
- `Ctrl+x` taglia
- `Ctrl+v` incolla (dopo la sincronizzazione degli appunti da computer a dispositivo)

Questo solitamente funziona nella maniera più comune.

Il comportamento reale, però, dipende dall'applicazione attiva. Per esempio *Termux* invia SIGINT con `Ctrl+c`, e *K-9 Mail* compone un nuovo messaggio.

Per copiare, tagliare e incollare in questi casi (ma è solo supportato in Android >= 7):

- `MOD+c` inietta `COPY`
- `MOD+x` inietta `CUT`
- `MOD+v` inietta `PASTE` (dopo la sincronizzazione degli appunti da computer a dispositivo)

In aggiunta, `MOD+Shift+v` permette l'iniezione del testo degli appunti del computer come una sequenza di eventi pressione dei tasti. Questo è utile quando il componente non accetta l'incollaggio di testo (per esempio in *Termux*), ma questo può rompere il contenuto non ASCII.

AVVISO: Incollare gli appunti del computer nel dispositivo (sia con `Ctrl+v` che con `MOD+v`) copia il contenuto negli appunti del dispositivo. Come conseguenza, qualsiasi applicazione Android potrebbe leggere il suo contenuto. Dovresti evitare di incollare contenuti sensibili (come password) in questa maniera.

Alcuni dispositivi non si comportano come aspettato quando si modificano gli appunti del dispositivo a livello di codice. L'opzione `--legacy-paste` è fornita per cambiare il comportamento di `Ctrl+v` and `MOD+v` in modo tale che anch'essi iniettino il testo gli appunti del computer come una sequenza di eventi pressione dei tasti (nella stessa maniera di `MOD+Shift+v`).

Pizzica per zoomare (pinch-to-zoom)

Per simulare il "pizzica per zoomare": `Ctrl+click e trascina`.

Più precisamente, tieni premuto `Ctrl` mentre premi il pulsante sinistro. Finché il pulsante non sarà rilasciato, tutti i movimenti del mouse ridimensioneranno e ruoteranno il contenuto (se supportato dall'applicazione) relativamente al centro dello schermo.

Concretamente `scrcpy` genera degli eventi di tocco aggiuntivi di un "dito virtuale" nella posizione simmetricamente opposta rispetto al centro dello schermo.

Preferenze di iniezione del testo

Ci sono due tipi di [eventi](#) generati quando si scrive testo:

- *eventi di pressione*, segnalano che tasto è stato premuto o rilasciato;
- *eventi di testo*, segnalano che del testo è stato inserito.

In maniera predefinita le lettere sono "iniettate" usando gli eventi di pressione, in maniera tale che la tastiera si comporti come aspettato nei giochi (come accade solitamente per i tasti WASD).

Questo, però, può [causare problemi](#). Se incontri un problema del genere, puoi evitarlo con:

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

(ma questo romperà il normale funzionamento della tastiera nei giochi)

Ripetizione di tasti

In maniera predefinita tenere premuto un tasto genera una ripetizione degli eventi di pressione di tale tasto. Questo può creare problemi di performance in alcuni giochi, dove questi eventi sono inutilizzati.

Per prevenire l'inoltro ripetuto degli eventi di pressione:

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

Click destro e click centrale

In maniera predefinita, click destro aziona BACK (indietro) e il click centrale aziona HOME. Per disabilitare queste scorciatoie e, invece, inviare i click al dispositivo:

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

Rilascio di file

Installare APK

Per installare un APK, trascina e rilascia un file APK (finisce con `.apk`) nella finestra di *scrcpy*.

Non c'è alcuna risposta visiva, un log è stampato nella console.

Trasferimento di file verso il dispositivo

Per trasferire un file in `/sdcard/Download` del dispositivo trascina e rilascia un file (non APK) nella finestra di *scrcpy*.

Non c'è alcuna risposta visiva, un log è stampato nella console.

La cartella di destinazione può essere cambiata all'avvio:

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

Inoltro dell'audio

L'audio non è inoltrato da *scrcpy*. Usa [sndcpy](#).

Vedi anche la [issue #14](#).

Scorciatoie

Nella lista seguente, `MOD` è il modificatore delle scorciatoie. In maniera predefinita è `Alt` (sinistro) o `Super` (sinistro).

Può essere cambiato usando `--shortcut-mod`. I tasti possibili sono `lctrl`, `rctrl`, `lalt`, `ralt`, `lsuper` and `rsuper` (`l` significa sinistro e `r` significa destro). Per esempio:

```
# usa ctrl destro per le scorciatoie
scrcpy --shortcut-mod=rctrl

# use sia "ctrl sinistro"+"alt sinistro" che "super sinistro" per le scorciatoie
scrcpy --shortcut-mod=lctrl+lalt,lsuper
```

[Super](#) è il pulsante *Windows* o *Cmd*.

Azione	Scorciatoia
Schermo intero	MOD+f

Rotazione schermo a sinistra	MOD+← (<i>sinistra</i>)
Rotazione schermo a destra	MOD+→ (<i>destra</i>)
Ridimensiona finestra a 1:1 (pixel-perfect)	MOD+g
Ridimensiona la finestra per rimuovere i bordi neri	MOD+w <i>Doppio click sinistro</i> ¹
Premi il tasto HOME	MOD+h <i>Click centrale</i>
Premi il tasto BACK	MOD+b <i>Click destro</i> ²
Premi il tasto APP_SWITCH	MOD+s <i>4° click</i> ³
Premi il tasto MENU (sblocca lo schermo)	MOD+m
Premi il tasto VOLUME_UP	MOD+↑ (<i>su</i>)
Premi il tasto VOLUME_DOWN	MOD+↓ (<i>giù</i>)
Premi il tasto POWER	MOD+p
Accendi	<i>Click destro</i> ²
Spegni lo schermo del dispositivo (continua a trasmettere)	MOD+o
Accendi lo schermo del dispositivo	MOD+Shift+o
Ruota lo schermo del dispositivo	MOD+r
Espandi il pannello delle notifiche	MOD+n <i>5° click</i> ³
Espandi il pannello delle impostazioni	MOD+n+n <i>Doppio 5° click</i> ³
Chiudi pannelli	MOD+Shift+n
Copia negli appunti ⁴	MOD+c
Taglia negli appunti ⁴	MOD+x
Sincronizza gli appunti e incolla ⁴	MOD+v
Inietta il testo degli appunti del computer	MOD+Shift+v
Abilita/Disabilita il contatore FPS (su stdout)	MOD+i
Pizzica per zoomare	Ctrl+click e trascina

¹Doppio click sui bordi neri per rimuoverli.

²Il tasto destro accende lo schermo se era spento, preme BACK in caso contrario.

³4° e 5° pulsante del mouse, se il tuo mouse ne dispone.

⁴Solo in Android >= 7.

Le scorciatoie con pulsanti ripetuti sono eseguite rilasciando e premendo il pulsante una seconda volta. Per esempio, per eseguire "Espandi il pannello delle impostazioni":

1. Premi e tieni premuto MOD.
2. Poi premi due volte n.
3. Infine rilascia MOD.

Tutte le scorciatoie `ctrl+tasto` sono inoltrate al dispositivo, così sono gestite dall'applicazione attiva.

Path personalizzati

Per utilizzare dei binari *adb* specifici, configura il suo path nella variabile d'ambiente `ADB` :

```
ADB=/percorso/per/adb scrpcy
```

Per sovrascrivere il percorso del file `scrpcy-server` , configura il percorso in `SCRPCY_SERVER_PATH` .

Perchè *scrpcy*?

Un collega mi ha sfidato a trovare un nome tanto impronunciabile quanto [gnirehtet](#).

[strcpv](#) copia una **string** (stringa); `scrpcy` copia uno **screen** (schermo).

Come compilare?

Vedi [BUILD](#) (in inglese).

Problemi comuni

Vedi le [FAQ](#).

Sviluppatori

Leggi la [pagina per sviluppatori](#).

Licenza (in inglese)

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.

Articoli (in inglese)

- [Introducendo scrpcy](#)
- [Scrpcy ora funziona wireless](#)