



Forense en Android: ADB y ANDRILLER

Cristina Martínez Carpintero

¿Qué vamos a ver?

1. ¿Qué es ADB?
2. Comandos ADB
3. Directorios de interés en Android
4. Andriller



ADB (*Android Debug Bridge*)

- Es una herramienta de líneas de comandos que permite la comunicación con una instancia de un emulador o un dispositivo Android conectado. Esta herramienta proporciona diferentes acciones en el dispositivos a través de un acceso vía shell.

<https://developer.android.com/studio/command-line/adb?hl=es-419>

The screenshot shows a web browser window displaying the Android Developer website. The address bar shows the URL <https://developer.android.com/studio/command-line/adb?hl=es-419>. The website header includes the "developers" logo, navigation links for "Plataforma", "Android Studio", "Google Play", "Jetpack", and "Más", a search bar, and a language selector set to "Español ...". Below the header, the "ANDROID STUDIO" section has tabs for "Descarga", "Novedades", "Guía del usuario" (which is active), and "Vista previa". On the left, a sidebar lists various topics, with "adb" highlighted at the bottom. The main content area is titled "Android Debug Bridge (adb)" and includes a breadcrumb trail: "Desarrolladores de Android > Android Studio > Guía del usuario". The text explains that adb is a versatile command-line tool for communicating with a device. It lists three components: a client, a daemon (adb), and a server. On the right, an "Índice" (Index) section provides links to related topics like "Cómo funciona adb", "Cómo habilitar la depuración de adb", and "Cómo conectarse a un dispositivo".

Android Debug Bridge (adb) | D | X +

https://developer.android.com/studio/command-line/adb?hl=es-419

developers Plataforma Android Studio Google Play Jetpack Más Buscar Español ... Acceder

ANDROID STUDIO

Descarga Novedades Guía del usuario Vista previa

Introducción a Android Studio

Conceptos básicos del flujo de trabajo

Cómo administrar tu proyecto

Escribe tu app

Cómo crear y ejecutar tu app

Configura tu compilación

Depura tu app

Cómo probar tu app

Cómo generar perfiles de tu app

Cómo publicar tu app

Herramientas de línea de comandos

Descripción general

aapt2

adb

Desarrolladores de Android > Android Studio > Guía del usuario

Calificar y opinar

Android Debug Bridge (adb)

Android Debug Bridge (adb) es una herramienta de línea de comandos versátil que te permite comunicarte con un dispositivo. El comando adb permite realizar una variedad de acciones en el dispositivo, como instalar y depurar apps, y proporciona acceso a un shell de Unix que puedes usar para ejecutar distintos comandos en un dispositivo. Es un programa cliente-servidor que incluye tres componentes:

- **Un cliente**, que envía comandos. El cliente se ejecuta en tu máquina de desarrollo. Puedes invocar un cliente desde un terminal de línea de comandos emitiendo un comando adb.
- **Un daemon (adb)**, que ejecuta comandos en un dispositivo. El daemon se ejecuta como un proceso en segundo plano en cada dispositivo.
- **Un servidor**, que administra la comunicación entre el cliente y el daemon. El servidor se ejecuta en tu máquina de desarrollo como un proceso en segundo plano.

Índice

- Cómo funciona adb
- Cómo habilitar la depuración de adb en tu dispositivo
- Cómo conectarse a un dispositivo mediante Wi-Fi (Android 11 y versiones posteriores)
- Cómo conectarse a un dispositivo mediante Wi-Fi (Android 10 y versiones anteriores)
- Cómo realizar consultas de dispositivo

platform-tools

Archivo Inicio Compartir Vista

Anclar al Acceso rápido Copiar Pegar Cortar Copiar ruta de acceso Pegar acceso directo

Portapapeles Organizar Mover a Copiar a Eliminar Cambiar nombre Nueva carpeta Nuevo elemento Fácil acceso Propiedades Modificar Historial Abrir Seleccionar todo No seleccionar nada Invertir selección

ANDRILLER > pla

Acceso rápido OneDrive - Universid Este equipo Red

Símbolo del sistema

```
will automatically reboot the device.
reboot [bootloader|recovery|sideload|sideload-auto-reboot]
reboot the device; defaults to booting system image but
supports bootloader and recovery too. sideload reboots
into recovery and automatically starts sideload mode,
sideload-auto-reboot is the same but reboots after sideloading.
sideload OTAPACKAGE sideload the given full OTA package
root restart adbd with root permissions
unroot restart adbd without root permissions
usb restart adbd listening on USB
tcpip PORT restart adbd listening on TCP on PORT

internal debugging:
start-server ensure that there is a server running
kill-server kill the server if it is running
reconnect kick connection from host side to force reconnect
reconnect device kick connection from device side to force reconnect
reconnect offline reset offline/unauthorized devices to force reconnect

environment variables:
$ADB_TRACE comma-separated list of debug info to log:
all,adb,sockets,packets,rwx,usb,sync,sysdeps,transport,jdwp
$ADB_VENDOR_KEYS colon-separated list of keys (files or directories)
$ANDROID_SERIAL serial number to connect to (see -s)
$ANDROID_LOG_TAGS tags to be used by logcat (see logcat --help)
$ADB_LOCAL_TRANSPORT_MAX_PORT max emulator scan port (default 5585, 16 emus)
$ADB_MDNS_AUTO_CONNECT comma-separated list of mdns services to allow auto-connect (default adb-tls-connect)

C:\Users\Cristina\Desktop>
```

17 elementos

platform-tools

Archivo Inicio Compartir Vista

Anclar al Acceso rápido Copiar Pegar Cortar Copiar ruta de acceso Pegar acceso directo

Portapapeles Organizar Mover a Copiar a Eliminar Cambiar nombre Nueva carpeta Nuevo elemento Fácil acceso Propiedades Modificar Historial Abrir

Google Nexus 4 (768x1280, 320dpi) - 192.168.21...

Developer options ON

Símbolo del sistema

```
will automatically reboot the device.
reboot [bootloader|recovery|sideload|sideload-auto-reboot]
reboot the device; defaults to booting system image but
supports bootloader and recovery too. sideload reboots
into recovery and automatically starts sideload mode,
sideload-auto-reboot is the same but reboots after sideloading.
sideload OTAPACKAGE sideload the given full OTA package
root restart adbd with root permissions
unroot restart adbd without root permissions
usb restart adbd listening on USB
tcpip PORT restart adbd listening on TCP on PORT

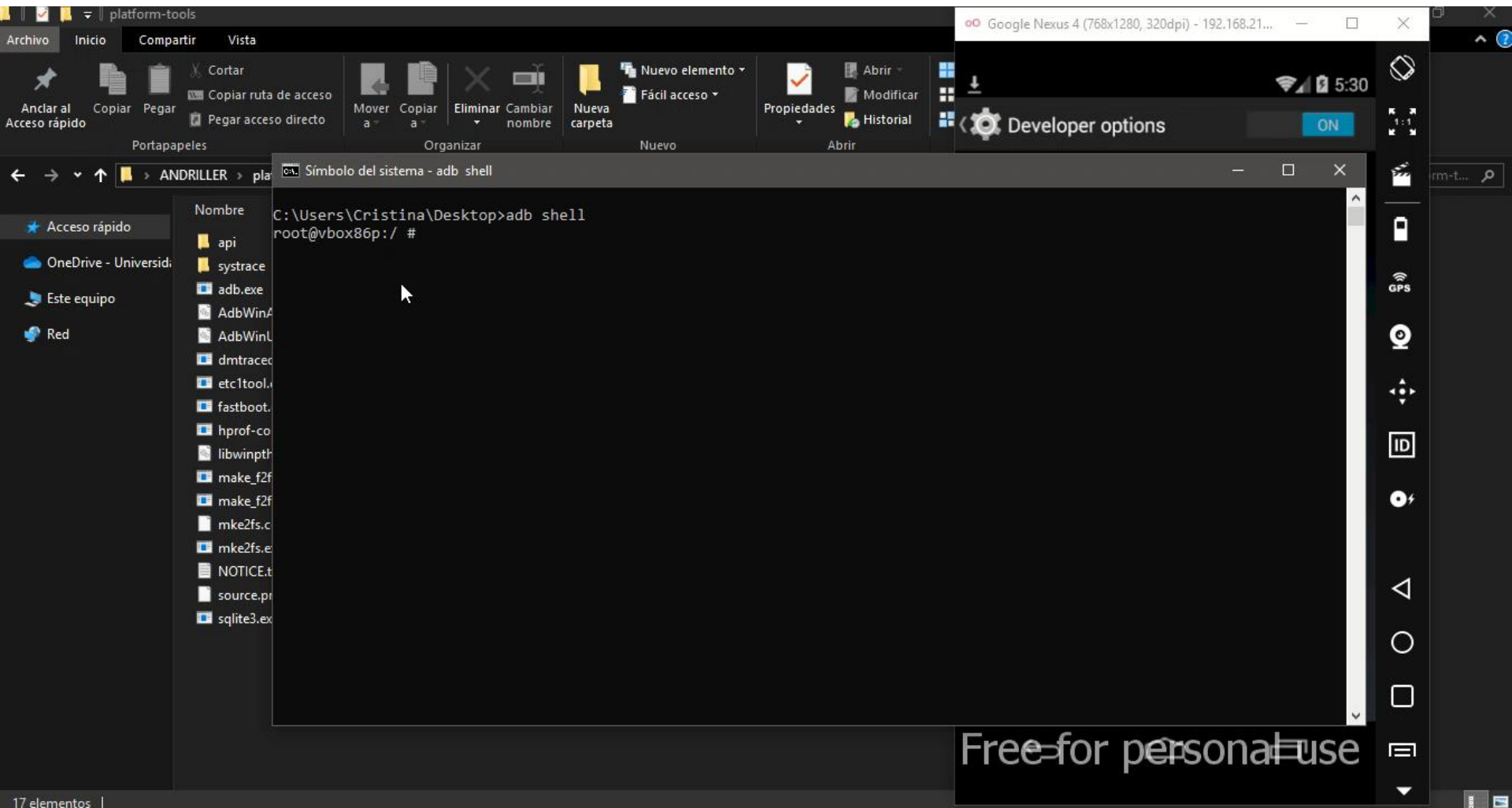
internal debugging:
start-server ensure that there is a server running
kill-server kill the server if it is running
reconnect kick connection from host side to force reconnect
reconnect device kick connection from device side to force reconnect
reconnect offline reset offline/unauthorized devices to force reconnect

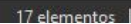
environment variables:
$ADB_TRACE comma-separated list of debug info to log:
all,adb,sockets,packets,rwx,usb,sync,sysdeps,transport,jdwp
$ADB_VENDOR_KEYS colon-separated list of keys (files or directories)
$ANDROID_SERIAL serial number to connect to (see -s)
$ANDROID_LOG_TAGS tags to be used by logcat (see logcat --help)
$ADB_LOCAL_TRANSPORT_MAX_PORT max emulator scan port (default 5585, 16 emus)
$ADB_MDNS_AUTO_CONNECT comma-separated list of mdns services to allow auto-connect (default adb-tls-connect)
```

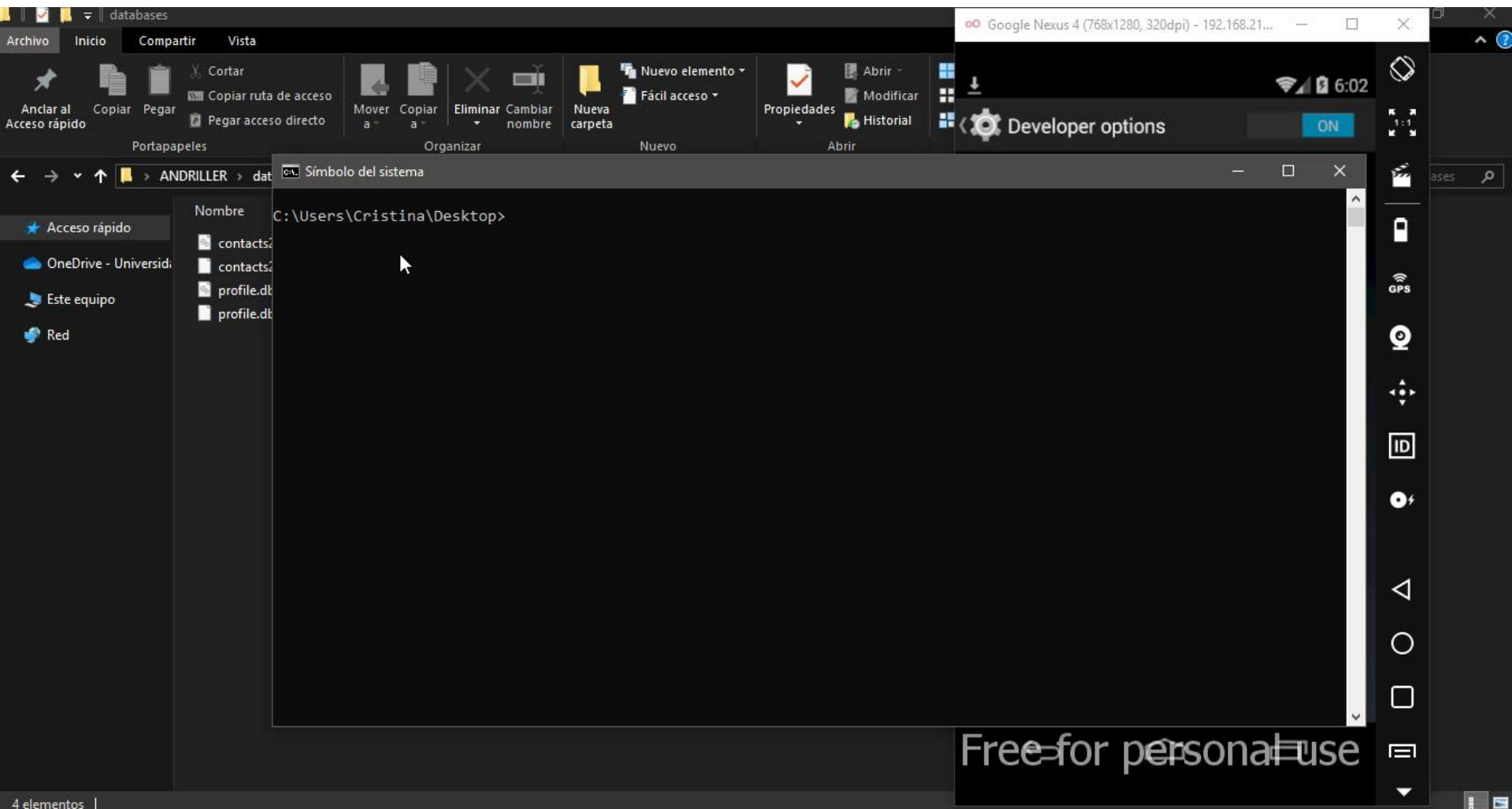
C:\Users\Cristina\Desktop>

Free-for personal use

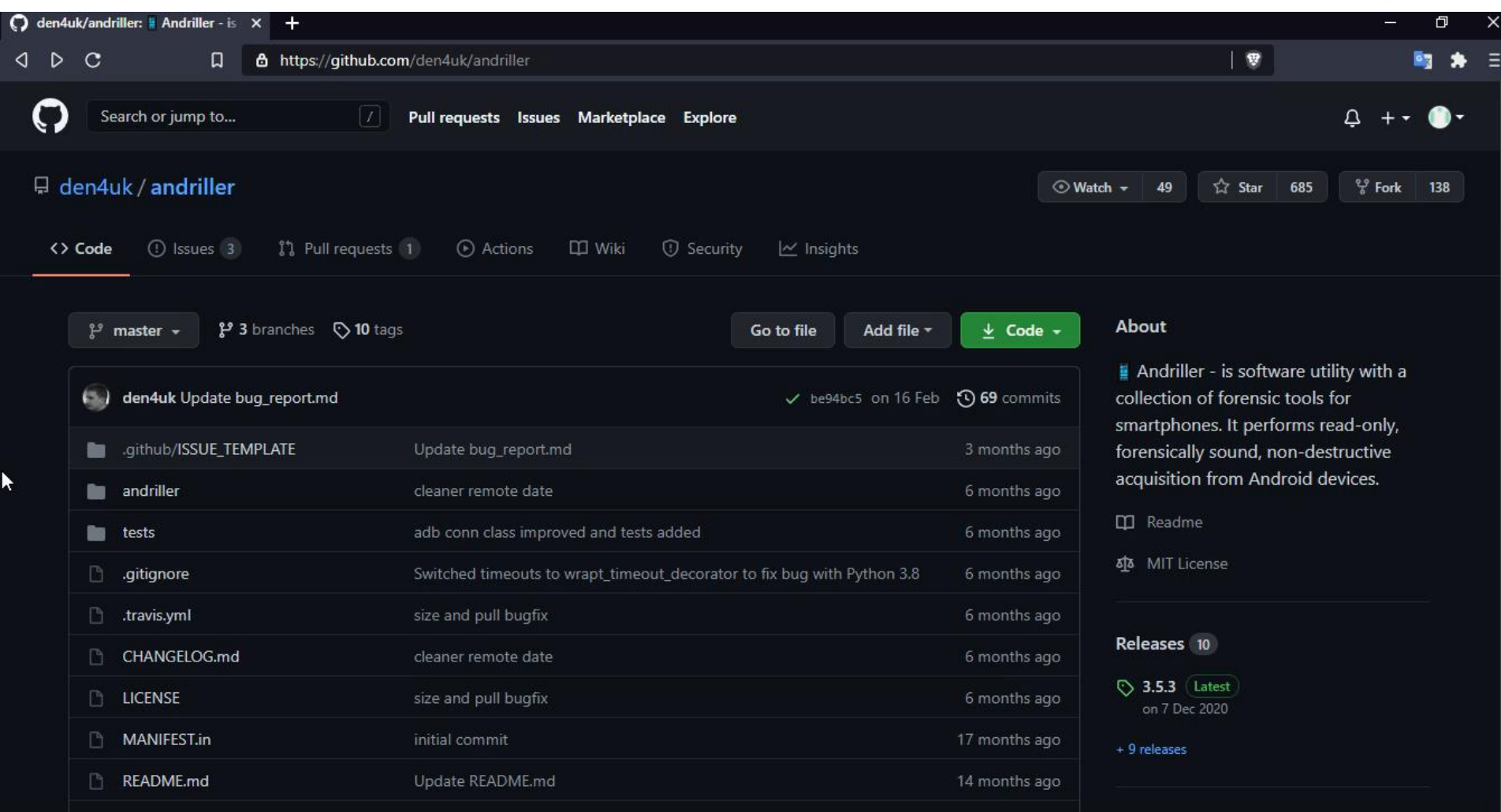
17 elementos







<https://github.com/den4uk/andriller>



den4uk/andriller: Andriller - is

Search or jump to... Pull requests Issues Marketplace Explore

den4uk / andriller Watch 49 Star 685 Fork 138

<> Code Issues 3 Pull requests 1 Actions Wiki Security Insights

master 3 branches 10 tags Go to file Add file Code

den4uk Update bug_report.md ✓ be94bc5 on 16 Feb 69 commits

.github/ISSUE_TEMPLATE	Update bug_report.md	3 months ago
andriller	cleaner remote date	6 months ago
tests	adb conn class improved and tests added	6 months ago
.gitignore	Switched timeouts to wrapt_timeout_decorator to fix bug with Python 3.8	6 months ago
.travis.yml	size and pull bugfix	6 months ago
CHANGELOG.md	cleaner remote date	6 months ago
LICENSE	size and pull bugfix	6 months ago
MANIFEST.in	initial commit	17 months ago
README.md	Update README.md	14 months ago

About

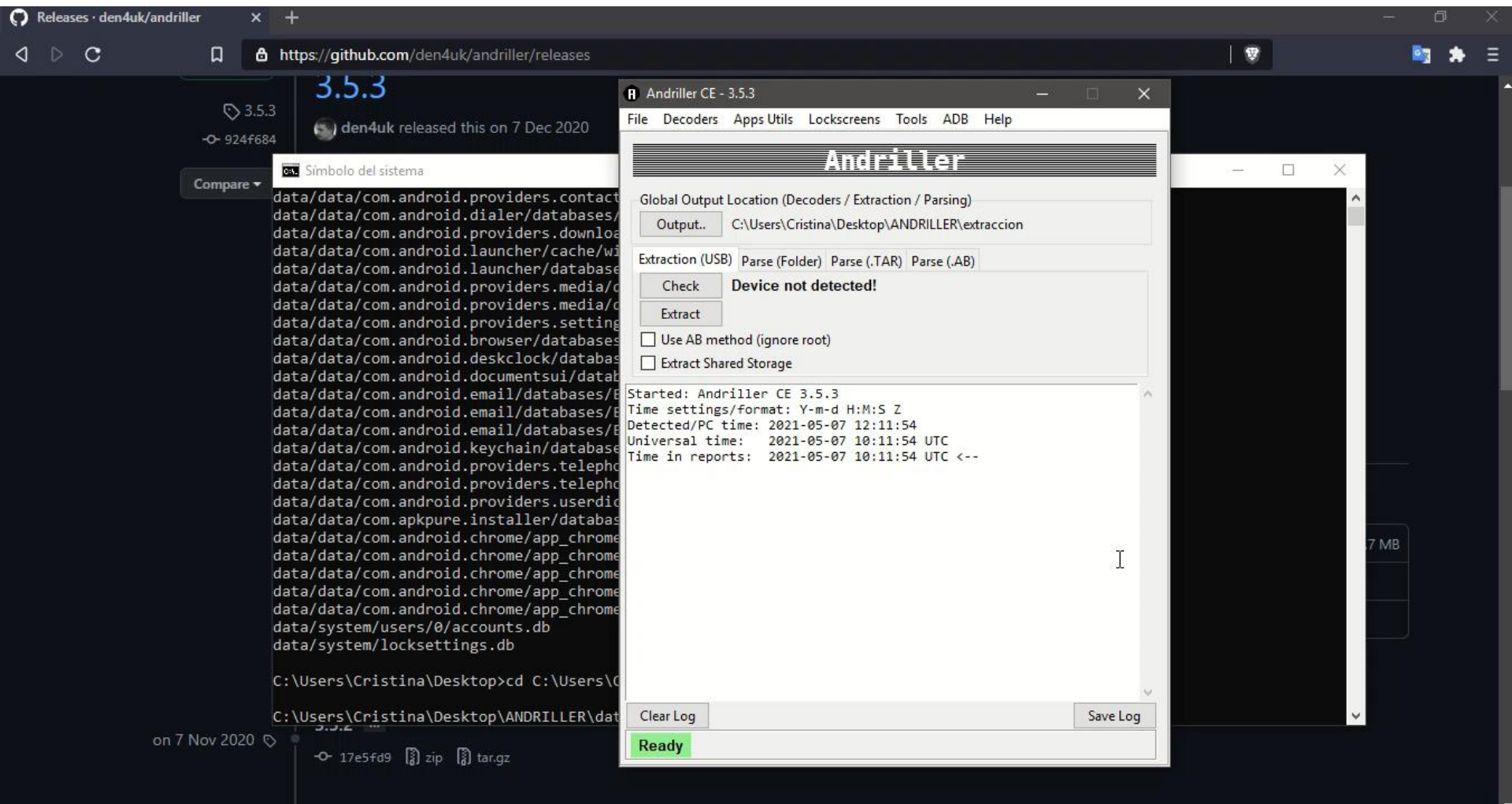
Andriller - is software utility with a collection of forensic tools for smartphones. It performs read-only, forensically sound, non-destructive acquisition from Android devices.

Readme MIT License

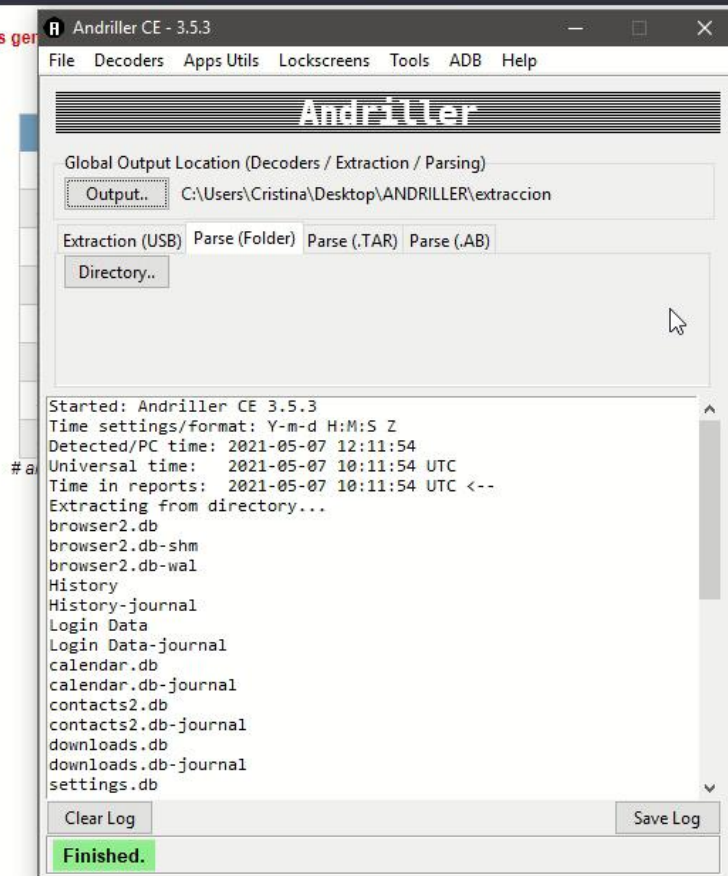
Releases 10

3.5.3 Latest on 7 Dec 2020

+ 9 releases



This report was generated by Andriller CE 3.5.3





Forense en Android: ADB y ANDRILLER

Cristina Martínez Carpintero