

ADB(1) MAN PAGE**VERSION**

1.0.41

NAME**adb** CLI Client for ADB (Android Debug Bridge) Server.**SYNOPSIS****adb** [*GLOBAL_OPTIONS*] command [*COMMAND_OPTIONS*]**DESCRIPTION**

Connects to the ADB Server via its smart socket interface. Allows sending requests, receives responses and manages lifecycle of the adb server.

Tasks are performed via commands. Some commands are fulfilled directly by the server while others are "forwarded over to the adbd(ADB daemon) running on the device.

GLOBAL OPTIONS:

- a** Listen on all network interfaces, not just localhost.
- d** Use USB device (error if multiple devices connected).
- e** Use TCP/IP device (error if multiple TCP/IP devices available).
- s SERIAL** Use device with given **SERIAL** (overrides \$ANDROID_SERIAL).
- t ID** Use device with given transport **ID**.
- H** Name of adb server host [default=localhost].
- P **PORT*** Smart socket **PORT** of adb server [default=5037].
- L SOCKET** Listen on given socket for adb server [default=tcp:localhost:5037].
- one-device SERIAL|USB** Server will only connect to one USB device, specified by a **SERIAL** number or **USB** device address (only with 'start-server' or 'server nodaemon').
- exit-on-write-error** Exit if stdout is closed.

GENERAL COMMANDS:

- devices [-l]** List connected devices.
- l** Use long output.
- help** Show this help message.
- version** Show version number.

NETWORKING

- connect HOST[:PORT]** Connect to a device via TCP/IP [default **PORT**=5555].
- disconnect [HOST[:PORT]]** Disconnect from given TCP/IP device [default **PORT**=5555], or all.
- pair HOST[:PORT] [PAIRING_CODE]** Pair with a device for secure TCP/IP communication.
- forward --list | [--no-rebind] LOCAL_REMOTE | --remove LOCAL | --remove-all**
- list** List all forward socket connections.
- [--no-rebind] LOCAL_REMOTE** Forward socket connection using one of the followings.
 - tcp:PORT** (local may be "tcp:0" to pick any open port. **localreserved:UNIX_DOMAIN_SOCKET_NAME**. **localfilesystem:UNIX_DOMAIN_SOCKET_NAME**. **jdwp:PROCESS PID** (remote only). **vsoc:kID:PORT** (remote only). **acceptfd:FD** (listen only). **dev:DEVICE_NAME**. **dev-raw:DEVICE_NAME**. (open device in raw mode)**.
- remove LOCAL** Remove specific forward socket connection.
- remove-all** Remove all forward socket connections.
- reverse --list | [--no-rebind] REMOTE LOCAL | --remove REMOTE | --remove-all**

--list List all reverse socket connections from device.

[--no-rebind] REMOTE LOCAL Reverse socket connection using one of the following.

tcp:**PORT** (**REMOTE** may be “tcp:0” to pick any open port). localabstract:**UNIX_DOMAIN_SOCKET_NAME**. localreserved:**UNIX_DOMAIN_SOCKET_NAME**. localfilesystem:**UNIX_DOMAIN_SOCKET_NAME**.

--remove REMOTE Remove specific reverse socket connection.

--remove-all Remove all reverse socket connections from device.

mdns check | services Perform mDNS subcommands.

check Check if mdns discovery is available.

services List all discovered services.

FILE TRANSFER:

push [--sync] [-z ALGORITHM] [-Z] LOCAL... REMOTE Copy local files/directories to device.

--sync Only push files that are newer on the host than the device.

-n Dry run, push files to device without storing to the filesystem.

-z enable compression with a specified algorithm (any/none/brotli/lz4/zstd).

-Z Disable compression.

pull [-a] [-z ALGORITHM] [-Z] REMOTE... LOCAL Copy files/dirs from device

-a preserve file timestamp and mode.

-z enable compression with a specified algorithm (any/none/brotli/lz4/zstd)

-Z disable compression

sync [-l] [-z ALGORITHM] [-Z] [all|data|odm|oem|product|system|system_ext|vendor] Sync a local build from \$ANDROID_PRODUCT_OUT to the device (default all)

-n Dry run. Push files to device without storing to the filesystem.

-l List files that would be copied, but don't copy them.

-z Enable compression with a specified algorithm (any/none/brotli/lz4/zstd)

-Z Disable compression.

SHELL:

shell [-e ESCAPE] [-n] [-Tt] [-x] [COMMAND...] Run remote shell command (interactive shell if no command given).

-e Choose escape character, or “none”; default ‘~’.

-n Don't read from stdin.

-T: Disable pty allocation.

-t: Allocate a pty if on a tty (-tt: force pty allocation).

-x Disable remote exit codes and stdout/stderr separation.

emu COMMAND Run emulator console **COMMAND**

APP INSTALLATION

(see also `adb shell cmd package help`):

install [-lrtsdg] [--instant] PACKAGE Push a single package to the device and install it

install-multiple [-lrtsdpg] [--instant] PACKAGE... Push multiple APKs to the device for a single package and install them

install-multi-package [-lrtsdpg] [--instant] PACKAGE... Push one or more packages to the device and install them atomically

-r: Replace existing application.
-t Allow test packages.
-d Allow version code downgrade (debuggable packages only).
-p Partial application install (install-multiple only).
-g Grant all runtime permissions.
--abi ABI Override platform's default ABI.
--instant Cause the app to be installed as an ephemeral install app.
--no-streaming Always push APK to device and invoke Package Manager as separate steps.
--streaming Force streaming APK directly into Package Manager.
--fastdeploy Use fast deploy.
--no-fastdeploy Prevent use of fast deploy.
-force-agent Force update of deployment agent when using fast deploy.
-date-check-agent Update deployment agent when local version is newer and using fast deploy.
--version-check-agent Update deployment agent when local version has different version code and using fast deploy.
--local-agent Locate agent files from local source build (instead of SDK location). See also `adb shell pm help` for more options.
uninstall [-k] APPLICATION_ID Remove this **APPLICATION_ID** from the device.
-k Keep the data and cache directories.

DEBUGGING:

bugreport [PATH] Write bugreport to given **PATH** [default=bugreport.zip]; if **PATH** is a directory, the bug report is saved in that directory. devices that don't support zipped bug reports output to stdout.
jdwp List pids of processes hosting a JDWP transport.
logcat Show device log (logcat -help for more).
server-status Display server configuration (USB backend, mDNS backend, log location, binary path. See `adb_host.proto` (AdbServerStatus) for details.

SECURITY:

disable-verity Disable dm-verity checking on userdebug builds.
enable-verity Re-enable dm-verity checking on userdebug builds.
keygen FILE Generate adb public/private key; private key stored in **FILE**.

SCRIPTING:

wait-for [-TRANSPORT] -STATE... Wait for device to be in a given state.
STATE: device, recovery, rescue, sideload, bootloader, or disconnect. **TRANSPORT:** **usb**, **local**, or **any** [default=**any**].
get-state Print offline | bootloader | device.
get-serialno Print **SERIAL_NUMBER**.
get-devpath Print **DEVICE_PATH**.
remount [-R] Remount partitions read-write.
-R 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 Same as **sideload** but reboots after sideloading.

sideload OTAPACKAGE Sideload the given full OTA package **OTAPACKAGE**.

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 Close connection from host side to force reconnect.

reconnect device Close connection from device side to force reconnect.

reconnect offline Reset offline/unauthorized devices to force reconnect.

USB:

Only valid when running with libusb backend.

attach SERIAL Attach a detached USB device identified by its **SERIAL** number.

detach SERIAL Detach from a USB device identified by its **SERIAL** to allow use by other processes.

Features:

host-features

list features supported by adb server.

features

list features supported by both adb server and device.

ENVIRONMENT VARIABLES

\$ADB_TRACE Comma (or space) separated list of debug info to log: all,adb,sockets,packets,rwx,usb,sync,sysdeps,transport,jdwp,services,auth,fdevent,shell,incremental,mdns.

\$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 emulators).

\$ADB_MDNS_AUTO_CONNECT Comma-separated list of mdns services to allow auto-connect (default adb-tls-connect).

\$ADB_MDNS_OPENSOURCE The default mDNS-SD backend is Bonjour (mdnsResponder). For machines where Bonjour is not installed, adb can spawn its own, embedded, mDNS-SD back end, openscreen. If set to "1", this env variable forces mDNS backend to openscreen.

\$ADB_LIBUSB ADB has its own USB backend implementation but can also employ libusb. **useadb devices -l(usb: prefix is omitted for lib usb)** or **adb host-features** (look for libusb in the output list) to identify which is in use. To override the default for your OS, set **ADB_LIBUSB** to "1" to enable libusb, or "0" to enable the ADB backend implementation.

BUGS

See Issue Tracker: [here](https://issuetracker.google.com/issues/new?component=192795&template=1310483) (https://issuetracker.google.com/issues/new?component=192795&template=1310483).

AUTHORS

See OWNERS file in ADB AOSP repo.