Is Someone Inside Your Phone?

A practical guide to checking your own iPhone for spyware with Amnesty International's Mobile Verification Toolkit

CTRL ALT Elite · Fortune 100-caliber infrastructure, operations, and security leadership

By René López, Founder and Principal · ORCID https://orcid.org/0009-0008-4993-0088

Why this is no longer someone else's problem

Commercial spyware used to be a story about journalists and dissidents. It is now a story about anyone whose phone holds something worth taking: executives in a deal, lawyers in litigation, founders raising a round, board members, and the family members who share a home network with them. The tools sold by NSO Group, Intellexa, Paragon, and their successors do not need you to click anything. A zero-click exploit arrives as an invisible message, installs itself, and hands a stranger your camera, microphone, messages, and location.

Detection has historically required a forensic lab. In 2021 Amnesty International's Security Lab changed that by releasing the Mobile Verification Toolkit, MVT, the same tooling they used to expose the Pegasus Project. It is free, open source, and maintained with indicators of compromise that now cover more than a dozen spyware families, from Pegasus and Predator to the 2026 additions covering DarkSword, Coruna, Morpheus, and BTMOB, plus indicators for forensic extraction tools like Cellebrite.

This guide walks through running MVT against your own iPhone, end to end, on hardware you probably already own. We wrote it after doing exactly that on an iPhone 16 Pro running iOS 27, and we have included the places where the documentation and reality diverged.

What MVT can and cannot tell you

MVT works by extracting the databases, logs, and caches that iOS writes during normal operation, then comparing what it finds against published indicators: known malicious domains, process names, file paths, and configuration profiles associated with specific spyware campaigns. It does not run on the phone. It runs on a computer, against a backup of the phone, which is why the phone's own security model cannot hide anything from it.

Two limits matter before you start.

- **A clean result means no match against public indicators.** It does not mean the device is clean. Current campaigns use infrastructure that has not been published yet, and Amnesty is explicit that reliance on public indicators alone can give a false sense of security. A clean scan lowers your risk estimate; it does not zero it.
- **MVT is licensed for consensual analysis only.** Its license prohibits adversarial forensics. You may scan your own phone or one whose owner has asked you to. You may not use it to inspect an employee's, partner's, or child's device without their informed consent.

> *A clean scan lowers your risk estimate. It does not zero it.*

What you need

The iPhone path is the easier one, because the analysis runs against an encrypted backup rather than a live device. Everything below is free.

Requirement	Why it matters
A Linux computer	MVT does not officially support Windows. A Linux machine, a Mac, or Windows Subsystem for Linux on Windows 11 all work. A native Linux box with a USB port is the least friction because it can talk to the phone directly.
Docker	Amnesty publishes a prebuilt container with MVT and the iOS tooling already installed. It saves you from building libimobiledevice from source, which is the most fragile step otherwise.
libimobiledevice on the host	The open source tooling that pairs with the phone and pulls the backup. Ubuntu 26.04 ships version 1.4.0, which handles current iOS releases.
Free disk equal to 2x the phone's data	You need room for the encrypted backup and for the decrypted copy MVT analyzes. A full iPhone backup can run to 135 GB or more; plan for 300 GB free on a heavily used phone.
A data-capable USB-C cable	Charge-only cables will enumerate nothing. If the phone charges but the host never sees it, swap the cable first.
A backup password you will remember	Backups must be encrypted or iOS omits most of the artifacts MVT needs. Apple cannot recover this password for you.

The process in five steps

The commands below assume Ubuntu with Docker installed and a working directory at ~/mvt. Substitute your own paths and password. Keep the phone unlocked and connected throughout.

Step 1: Prepare the host

```
sudo apt update && sudo apt install -y usbmuxd libimobiledevice-utils
mkdir -p ~/mvt/backup ~/mvt/results
docker pull ghcr.io/mvt-project/mvt
```

usbmuxd is the daemon that multiplexes USB connections to Apple devices. On Ubuntu it is started by udev the moment an iPhone is plugged in, and its socket appears at /var/run/usbmuxd. It is not something you enable in systemd, and it will not exist until a phone is connected.

Step 2: Connect and pair

```
ls -l /var/run/usbmuxd          # should show a socket once the phone is plugged in
idevicepair pair                # tap Trust on the phone, enter the passcode
idevicepair pair                # run again; you want SUCCESS: Paired with device
ideviceinfo -k ProductVersion   # confirms the host can talk to iOS
```

Pairing records are stored through usbmuxd on the host, so a container that binds the same socket inherits the trust relationship. If usbmuxd has been running for hours from an earlier session and the phone is not found, restart it with sudo systemctl restart usbmuxd and try again.

Step 3: Take an encrypted backup

```
idevicebackup2 encryption on "<password>"
idevicebackup2 backup --full ~/mvt/backup
```

Expect this to take a while. A 135 GB backup over USB on a small-form-factor machine ran for the better part of an hour in our test. The output ends with Backup Successful and a folder named after the device UDID under ~/mvt/backup. If the phone already has backup encryption enabled from an earlier iTunes or Finder backup, the encryption command will tell you so; use that existing password instead.

Step 4: Decrypt and scan inside the container

```
docker run -it -v ~/mvt:/home/mvt/data ghcr.io/mvt-project/mvt
# inside the container:
ldconfig
mvt-ios download-iocs
mvt-ios decrypt-backup -p "<password>" -d /home/mvt/data/decrypted /home/mvt/data/backup/<UDID>
mvt-ios check-backup --output /home/mvt/data/results /home/mvt/data/decrypted
```

The decrypt step writes a full plaintext copy of the backup and takes about as long as the backup itself. The scan is fast by comparison: it loads every indicator file, runs 28 analysis modules across Safari history, SMS, WhatsApp, TCC permissions, configuration profiles, crash logs, and more, and prints a summary.

Step 5: Read the results

Every module writes a JSON file to the results folder. Any module that matched an indicator also writes a file ending in _detected.json. The summary box at the end of the run counts alerts by severity.

```
ls /home/mvt/data/results | grep detected
```

Interpreting what you see

Result	What it means
No _detected files, 0 HIGH or CRITICAL alerts	No trace of any published spyware indicator in the backup. This is the expected outcome for most devices. It is reassurance, not proof.
LOW alert: Lockdown mode disabled	Not a finding about compromise. MVT is noting that Apple's high-risk hardening mode is off. Most people leave it off; targeted individuals should turn it on.
One or more _detected.json files	MVT found a match: a known malicious domain in browser history, a suspicious process in crash logs, an unexpected configuration profile. Open the file. Some matches are false positives from shared infrastructure; many are not.
MEDIUM, HIGH, or CRITICAL alerts	Treat the device as compromised until an expert says otherwise. See the next section before you do anything else.
Module errors about corrupted databases	Usually means the backup was not decrypted, or a database format changed in a new iOS release. Confirm you scanned the decrypted folder, then check the MVT GitHub issues for your iOS version.

If MVT finds something

The instinct is to wipe the phone. Resist it. A factory reset destroys the evidence that an expert needs to confirm the infection, identify the operator, and understand what was taken. Instead:

- **Preserve the backup and the results folder.** Copy them to an encrypted external drive and keep the plaintext decrypted folder only as long as an investigator needs it.
- **Stop using the device for anything sensitive.** Assume the microphone, camera, and messaging apps are live to a third party. Move to a different device for calls and messages about the incident itself.
- **Get expert help.** Amnesty International's Security Lab supports human rights defenders and civil society through securitylab.amnesty.org. Access Now runs a 24/7 Digital Security Helpline for at-risk users. Corporate targets should engage their incident response retainer or a mobile forensics firm; the private sector versions of these teams exist and are used to this call.
- **Notify counsel before notifying anyone else.** If the target is an executive or the device holds privileged material, the legal exposure starts the moment you know.

Cleanup, whether or not you found anything

When the scan is done you are holding a complete, unencrypted copy of a phone on a hard drive. Treat it accordingly.

```
rm -rf ~/mvt/decrypted                    # the plaintext copy; delete it first
rm -rf ~/mvt/backup                       # or keep it as an encrypted baseline for rescans
idevicebackup2 changepw "<old>" "<new>"   # if the password touched a chat log or shell history
```

Keep the results folder. It is small, contains extracted metadata rather than message content, and gives you a baseline to compare against on the next run. Rescans are incremental and much faster: plug in, run idevicebackup2 backup without --full, then repeat the decrypt and check steps.

Field notes: where the documentation and reality diverged

Four things cost us time on a current Ubuntu 26.04 host with the September 2026 MVT container. None are documented, and all are quick once you know.

- **The container's idevicepair fails to load libimobiledevice-1.0.so.6.** Running ldconfig inside the container fixes it. Alternatively, pair and back up with the host's own libimobiledevice-utils, which is what we ended up doing.
- **Order matters with the usbmuxd socket.** If you start the container with a bind mount on /var/run/usbmuxd before the phone is plugged in, Docker creates an empty directory at that path and the container will never see a device. Plug in first, confirm the socket exists, then start the container.
- **check-backup will not decrypt on the fly.** The current release rejects an encrypted backup outright and has no password flag; the MVT_IOS_BACKUP_PASSWORD environment variable is also ignored. Budget the disk and time for a full decrypt-backup pass.
- **iOS 27 backs up fine with libimobiledevice 1.4.0.** Older distributions ship older versions that may fail to pair with a current iPhone. If pairing fails on an older host, the Docker container's bundled tooling, once ldconfig has run, is the fallback.

Beyond the scan

MVT is a check, not a control. If you have reason to believe you are a target, four measures do more than any scan:

- **Turn on Lockdown Mode.** It closes most of the attack surface zero-click exploits rely on. The cost is some broken link previews and blocked attachments.
- **Install iOS updates the day they ship.** The majority of spyware infections we see in published cases exploited vulnerabilities Apple had already patched.
- **Restart the phone daily.** Several spyware families do not survive a reboot and must re-infect, which costs the operator money and creates fresh forensic traces.
- **Rescan on a cadence.** Quarterly for most executives, and after any travel to a jurisdiction where mercenary spyware is deployed domestically.

Work with CTRL ALT Elite

We spent 25 years running the systems that do not get to go down, ever, at enterprise scale: airlines, cruise lines, and healthcare platforms serving hundreds of millions of customers. We help founders, boards, and operators build the security posture their stage demands, including executive device hygiene programs, mobile threat assessments, and incident response planning for the day a scan comes back with a _detected file.

CTRL ALT Elite

reboot@ctrlaltelite.io · 954-516-6909 · ctrlaltelite.io · Pompano Beach, FL

René López, Founder and Principal · ORCID https://orcid.org/0009-0008-4993-0088

Sources: Amnesty International Security Lab, Mobile Verification Toolkit documentation (docs.mvt.re) and mvt-project/mvt on GitHub; Amnesty International, Forensic Methodology Report: How to catch NSO Group's Pegasus, July 2021; Access Now Digital Security Helpline. Commands verified September 2026 on Ubuntu 26.04, Docker, MVT 2026.9.22, libimobiledevice 1.4.0, iPhone 16 Pro on iOS 27.0.