Countering Forensics Software by Baiting Them

Dr. Weihan Goh | Joseph Lim | Isaac Soon

DEF CON 33 Creator Talk, organized by Adversary Village

Agenda

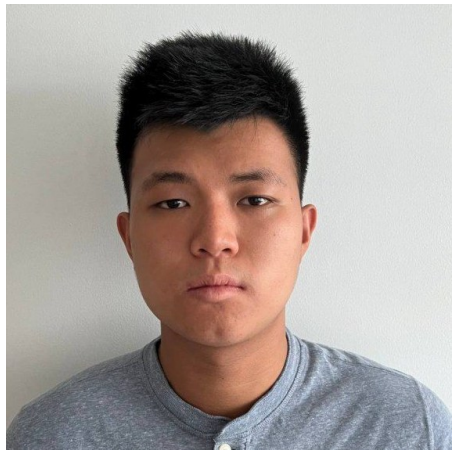
The big picture

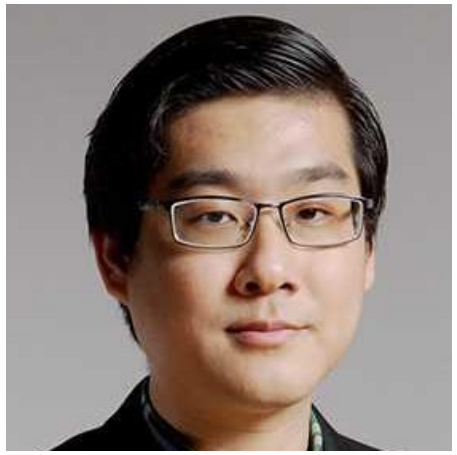
Honey tokens 101

POC or it didn't happen

Defensive strategies

Conclusion





New Team, Who Dis?

Is This Talk For You?

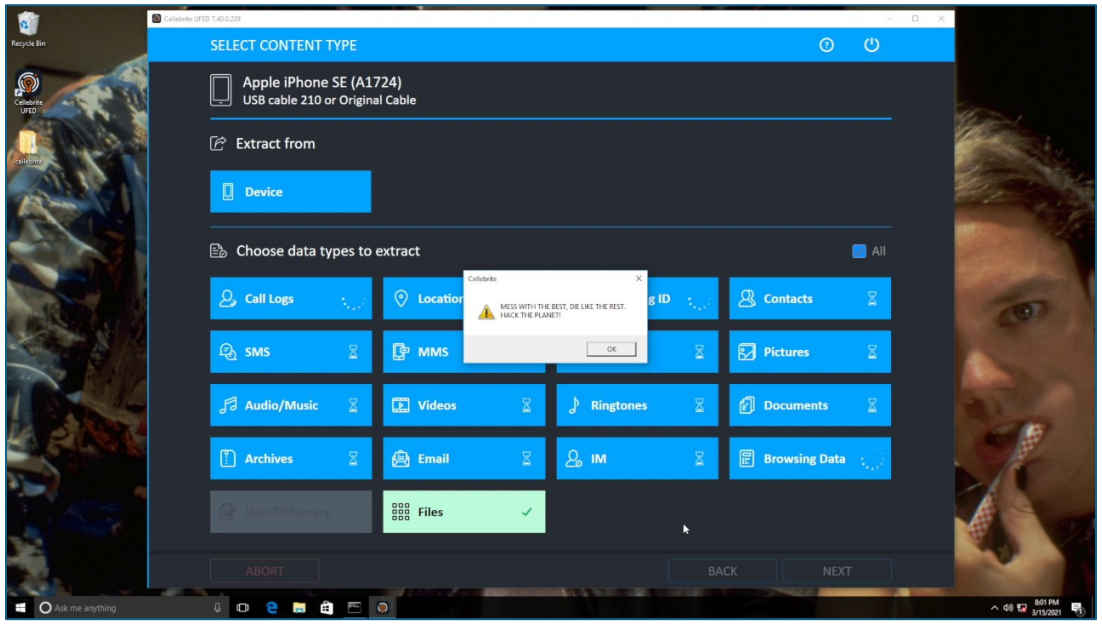
https://www.linkedin.com/posts/abrignoni_dfir-activity-6928451100442304512-QabW

Why "First Stage" Access Isn't Enough

- **Logical extraction workflow**
  1. Gain access to the device (Unlock device / bypass security)
  2. Extract data
  3. ???
  4. PROFIT!

- This assumption underpins the workflow of a number of mainstream forensic software

- Reality check: **Second-layer countermeasures!**



https://signal.org/blog/cellebrite-vulnerabilities/

This Whole Concept Isn't New

Photo by Mkckim / CC BY-SA 4.0

IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 29, NO. 7, AUGUST 2011 1377

A Plausibly-Deniable, Practical Trusted Platform Module Based Anti-Forensics Client-Server System

Weihan Goh, Peng Chor Leong, and Chai Kiat Yeo

***Abstract*—A Trusted Platform Module (TPM) contains a unique identity and provides a range of security functions. This paper demonstrates a novel approach of using a TPM-enabled computer in a client-server system to hinder forensic examination. The prime motivation for this is to highlight the implications of such TPM-based approach in digital forensics for law enforcement agents. The system allows for data confidentiality, plausible deniability, and hiding of traces that data containing incriminating information was present on the client. The server will attest the client before allowing it to submit or receive encrypted data, and encrypted data containing incriminating information can only be decrypted 1) by the encrypting client, and 2) if and only if the encrypting client's platform configuration matches that during encryption. The client's state can always be established via measurement results, and they cannot be tampered to fake attestation.**

***Index Terms*—anti-forensics, information confidentiality, plausible deniability, trusted platform module.**

I. Introduction

ELECTRONIC records such as computer and network logs, e-mails, word processing files, and image files increasingly provide essential and important evidence in criminal cases. As a result, guidelines for handling and processing have been established to ensure that electronic evidences seized from sources adhere to the standards of evidence that are admissible in courts of law [1]–[3]. In order to obtain these records, law enforcement agents employ forensic examination of seized electronic sources. While thus far, forensics have proven useful in extracting incriminating evidences, it is possible to develop a system that hinders this effort.

The TCG specification [4] is a specification developed by the Trusting Computing Group (TCG). This specification outlines the concept of trusted computing, which essentially ensures that a system will consistently behave in specific, pre-arranged ways, verifiable by external entities using a combination of hardware and software support.

At the core of the TCG specification is the Trusted Platform

While the TCG specification and TPM could have many positive uses such as in the realms of digital rights management, integrity verification, and remote system auditing, it is also possible that they be misused by criminals to encrypt data in such a way that effectively denies its access to law enforcement agents even with forensic aid. In addition, it is also possible for such systems to conceal the existence of encrypted data. Since the keys are never exposed to the criminal, it is practically impossible for enforcement agents to coerce or force a criminal to reveal it. Such an undesirable though possible usage has been conveyed to some law enforcement agencies in 2006 [6].

This paper presents specifications and analysis of an anti-forensics system constructed by utilizing TPM-enabled security on a client-server system. It extends the basic system specifications presented in [7] and provides detailed analysis of its anti-forensic capabilities. Although presented as a criminal-centric system, this work is important to the field of digital forensics as it helps to answer the following questions:

- What are the *requirements* for such anti-forensics system, and how could such a system be *realized*?
- What are the *implications for law enforcement agents* if such system is *used in real-world crime scenarios*?
- Are there any *positive uses for such a system*?

The remainder of this paper is organized as follow:

- Section II gives an overview of electronic evidences and current anti-forensic techniques, as well as information on the TPM and its related specification and software.
- Section III outlines the requirements for the anti-forensics system.
- Section IV documents the system and communication protocol design.
- Section V presents a detailed security analysis of the design.
- Section VI shows the system implementation as a proof-of-concept and presents evaluation results of the implemented system.

Rethinking Traditional Anti-Forensics

Conventional anti-forensic tactics

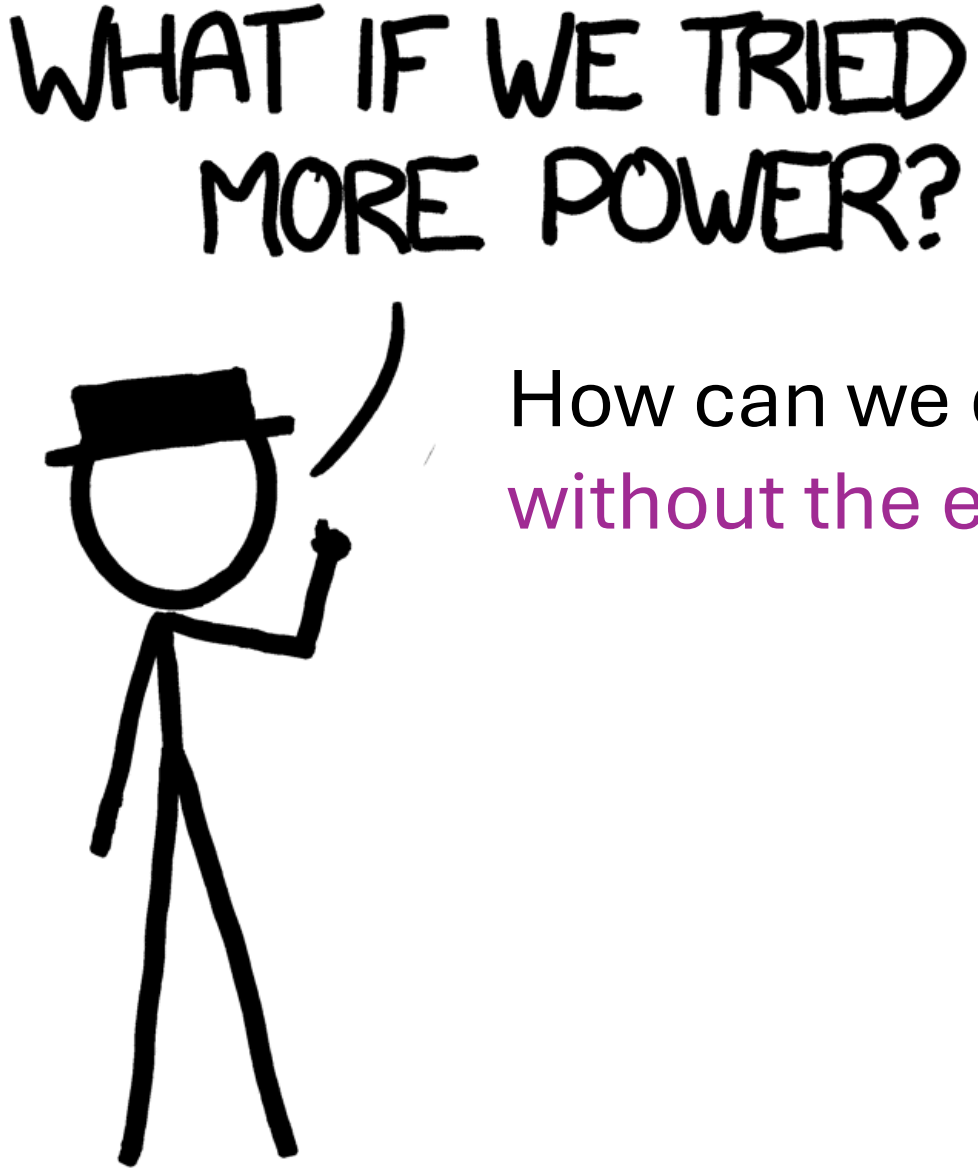
- Disabling USB debugging entirely
- Monitor for forensic software installation, then block ADB connections
- Kill the forer

But…

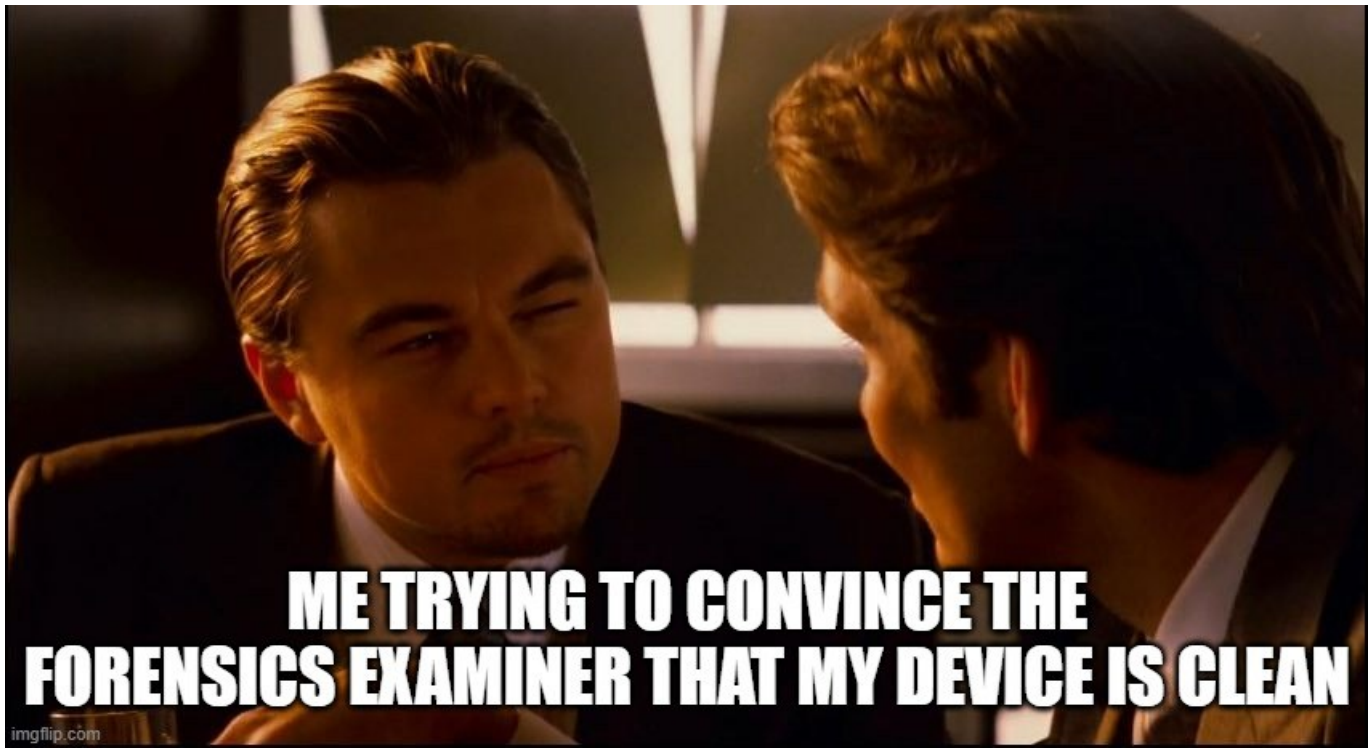
- 🚩🚩🚩🚩🚩🚩🚩🚩
- Once alerte iners switch to at or more s, leading tiny that can l subterfuge

```
if (packageName.contains("belkasoft") || packageName.contains( ebrite")){
    //String commandDenyPackage= "su -c pm disable " + packageName;
    String commandDenyPackage= "su -c do pkill -f " + packageName;
    Process denyPackage = Runtime.getRuntime().exec(commandDenyPackage);
    denyPackage.waitFor();
}
```

This is not the way...



How can we deploy anti-forensic measures without the examiner realizing they're in play?



https://what-if.xkcd.com/13/

Let Them Fight

Forensic Software

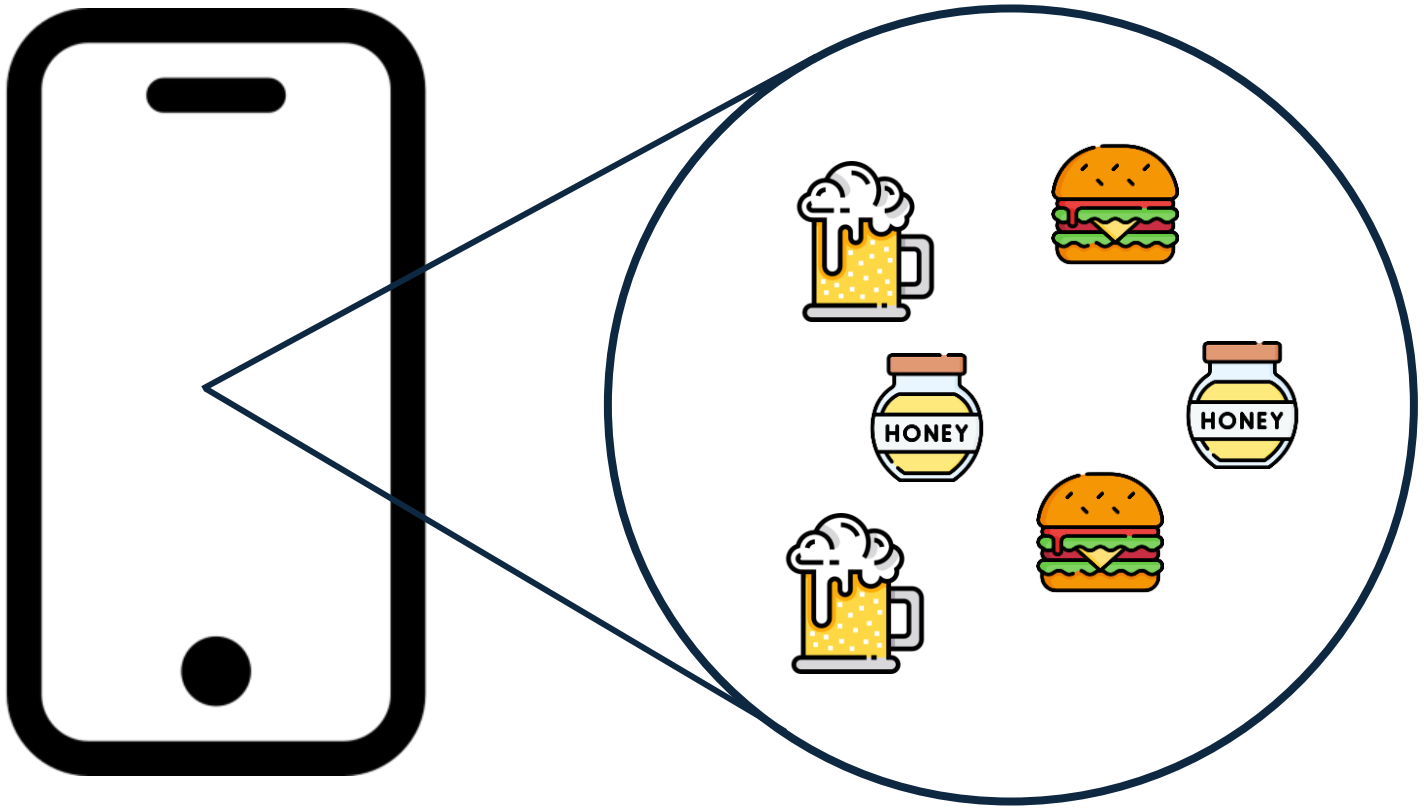
1. Gain access to the device (Unlock device / bypass security)
2. Extract data
3. ???
4. WTF?

Adversary (i.e., Us)

1. Let it happen 🧘 (Show nothing, but do something)
2. Let it happen
3. ???
4. PROFIT!!

The device intended user only eats burgers and drink beer…

But the forensic software consumes all…

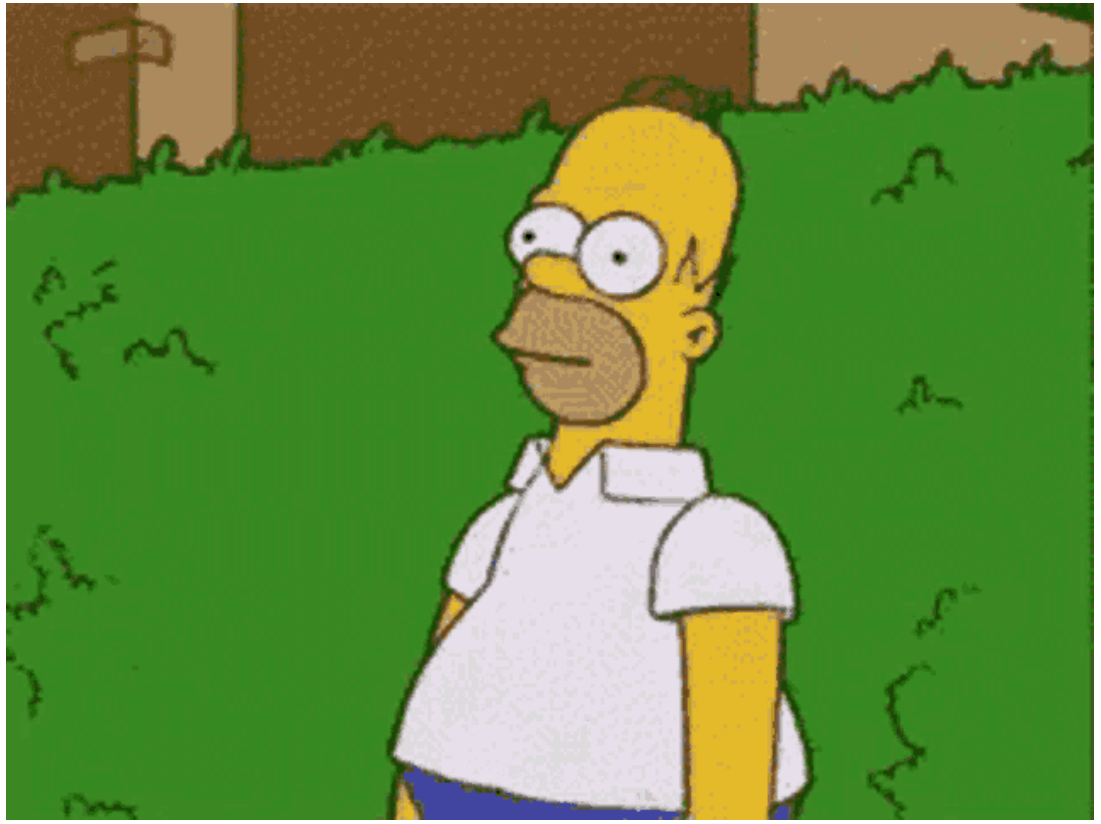


Honey Tokens 101

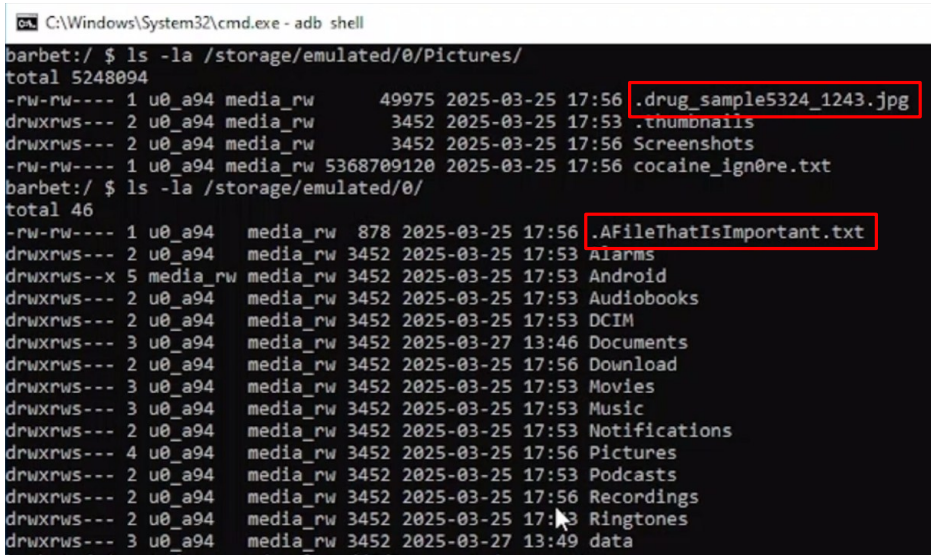
- We can **litter bait files** containing rubbish information
    1. Deep within filesystem directories
    2. In top-level directories (e.g., Documents) but files lead with "." suffix

- Once these bait files are **"accessed"**, we activate our **anti-forensic payloads**

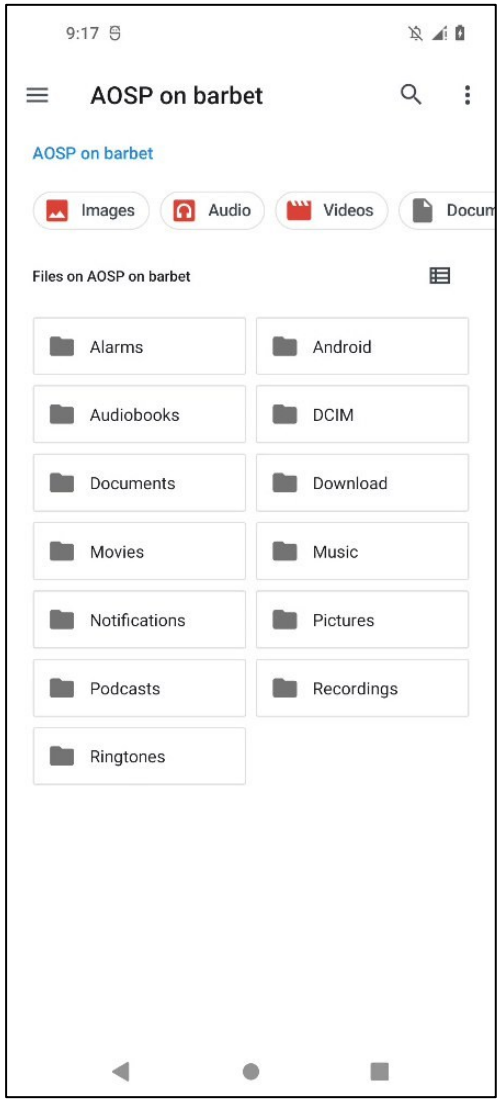
- Can be done programmatically too! When a new app is installed.



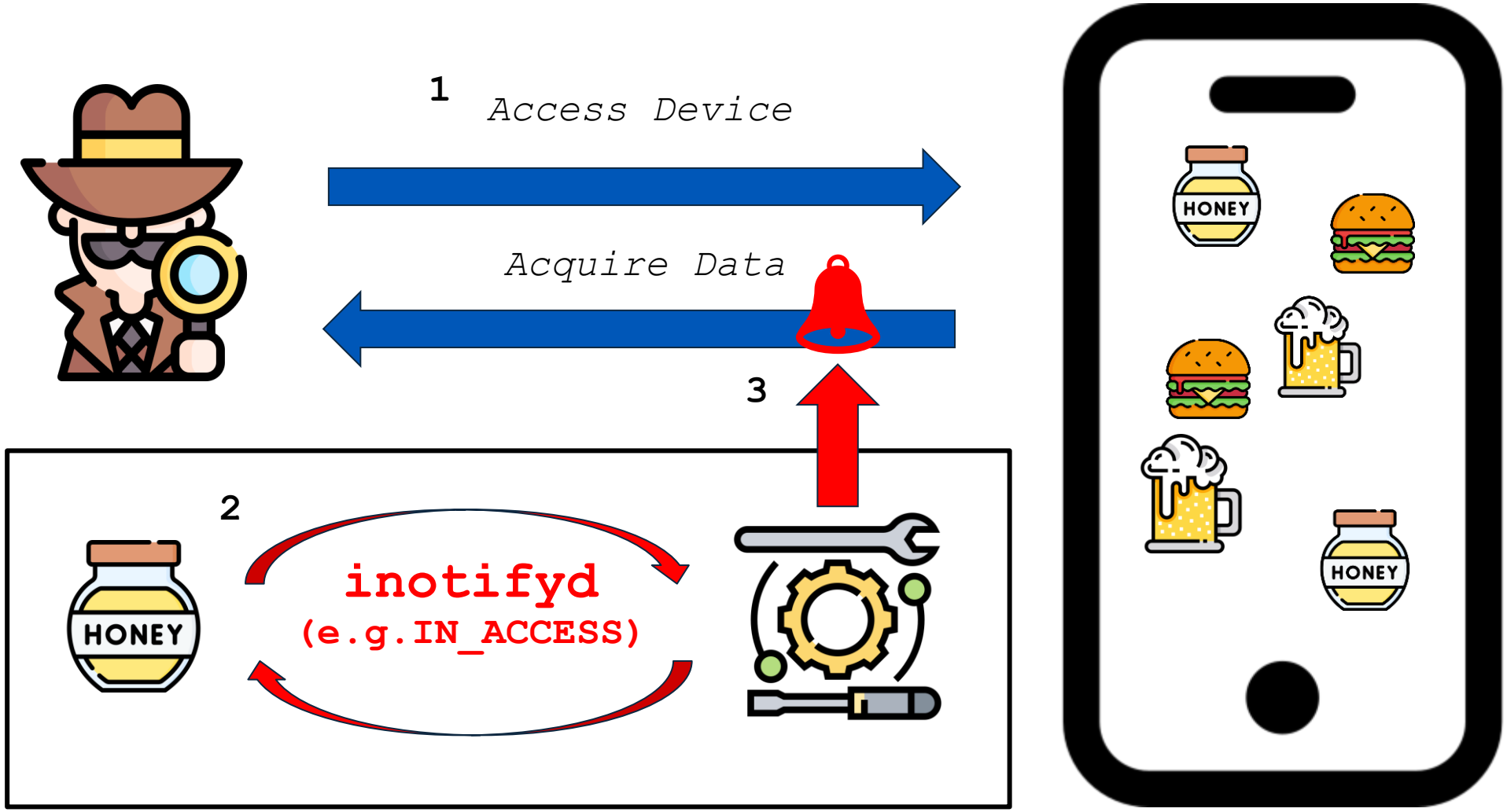
Honey Token Deployment




```
barbet:/ # ls -la /storage/emulated/0/Android/data/org.telegram.messenger.web/files/
total 13
drwxrws--- 3 u0_a102 ext_data_rw 3452 2025-07-07 21:18 .
drwxrws--- 4 u0_a102 ext_data_rw 3452 2025-07-07 21:18 ..
drwxrws--- 6 u0_a102 ext_data_rw 3452 2025-07-07 21:18 Telegram
-rw-rw---- 1 u0_a102 ext_data_rw 1326 2025-07-07 21:18 action_history_20250707_211850.log
barbet:/ # ls -la /data/data/org.telegram.messenger.web/files/
total 1048
drwxrwx--x 4 u0_a102 u0_a102          3452 2025-07-07 21:18 .
drwx------ 8 u0_a102 u0_a102          3452 2025-07-07 21:18 ..
drwx------ 2 u0_a102 u0_a102          3452 2025-07-07 21:18 account1
drwx------ 2 u0_a102 u0_a102          3452 2025-07-07 21:18 account2
-rw-r--r-- 1 u0_a102 u0_a102_cache     1772 2025-07-07 21:18 action_history_20250707_211850.log
-rw------- 1 u0_a102 u0_a102          5929 2025-07-07 21:18 bluebubbles.attheme
-rw------- 1 u0_a102 u0_a102          4096 2025-07-07 21:18 cache4.db
-rw------- 1 u0_a102 u0_a102         32768 2025-07-07 21:18 cache4.db-shm
-rw------- 1 u0_a102 u0_a102       1005312 2025-07-07 21:18 cache4.db-wal
-rw-rw---- 1 u0_a102 u0_a102           912 2025-07-07 21:18 tgnet.dat
barbet:/ #
```



Honey Token Implementation



inotifyd

- **`inotifyd`** is a command in the userland that is built on top of **`inotify`** Linux kernel subsystem which is a useful mechanism for monitoring filesystems!
- **`inotifyd`** is part of **`toybox`** which is included in **Android 6.0+**!

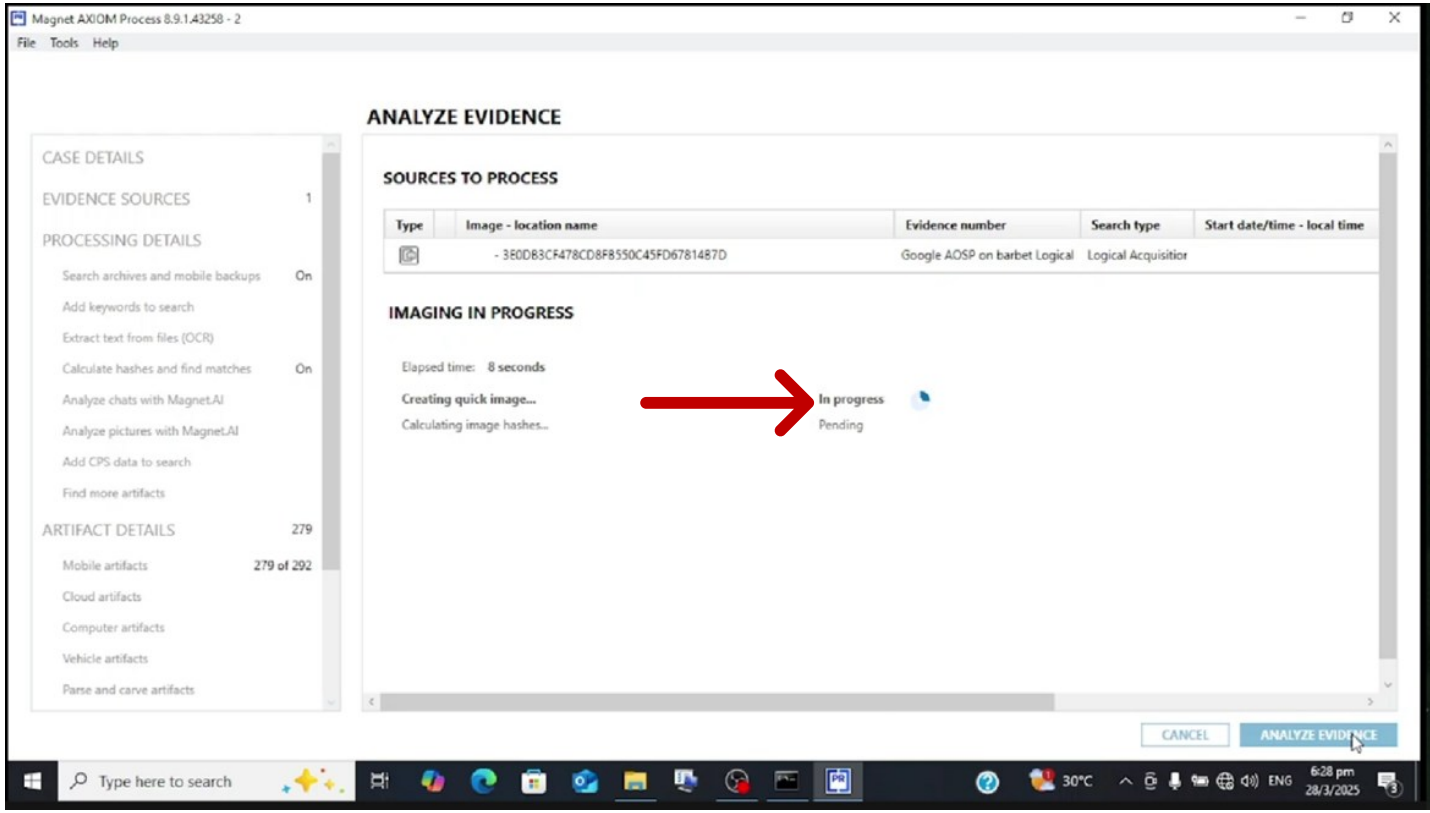
```
com.example.sweetdeceit  I  File event detected: r  /storage/emulated/0/Download/.drug_shipping_schedule_mapletree_1914.txt
com.example.sweetdeceit  I  File event detected: a  /storage/emulated/0/Download/.drug_shipping_schedule_mapletree_1914.txt
com.example.sweetdeceit  I  File event detected: 0  /storage/emulated/0/Download/.drug_shipping_schedule_mapletree_1914.txt
```

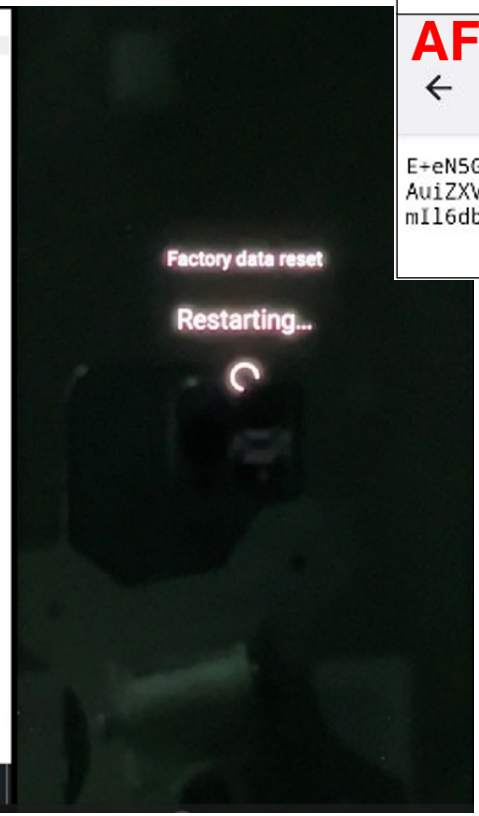
Legend

`r -> opened IN_OPEN`
`a -> accessed IN_ACCESS`
`0 -> closed (unwritable) IN_CLOSE_NOWRITE`

We Have a Tripwire...
So Now What?

1. Encrypt data
2. Factory reset the device
3. **Sky's the limit**





BEFORE ENCRYPT

← secret.txt

```
shhhh this is my password: S3cUr3P@sswOrd
```

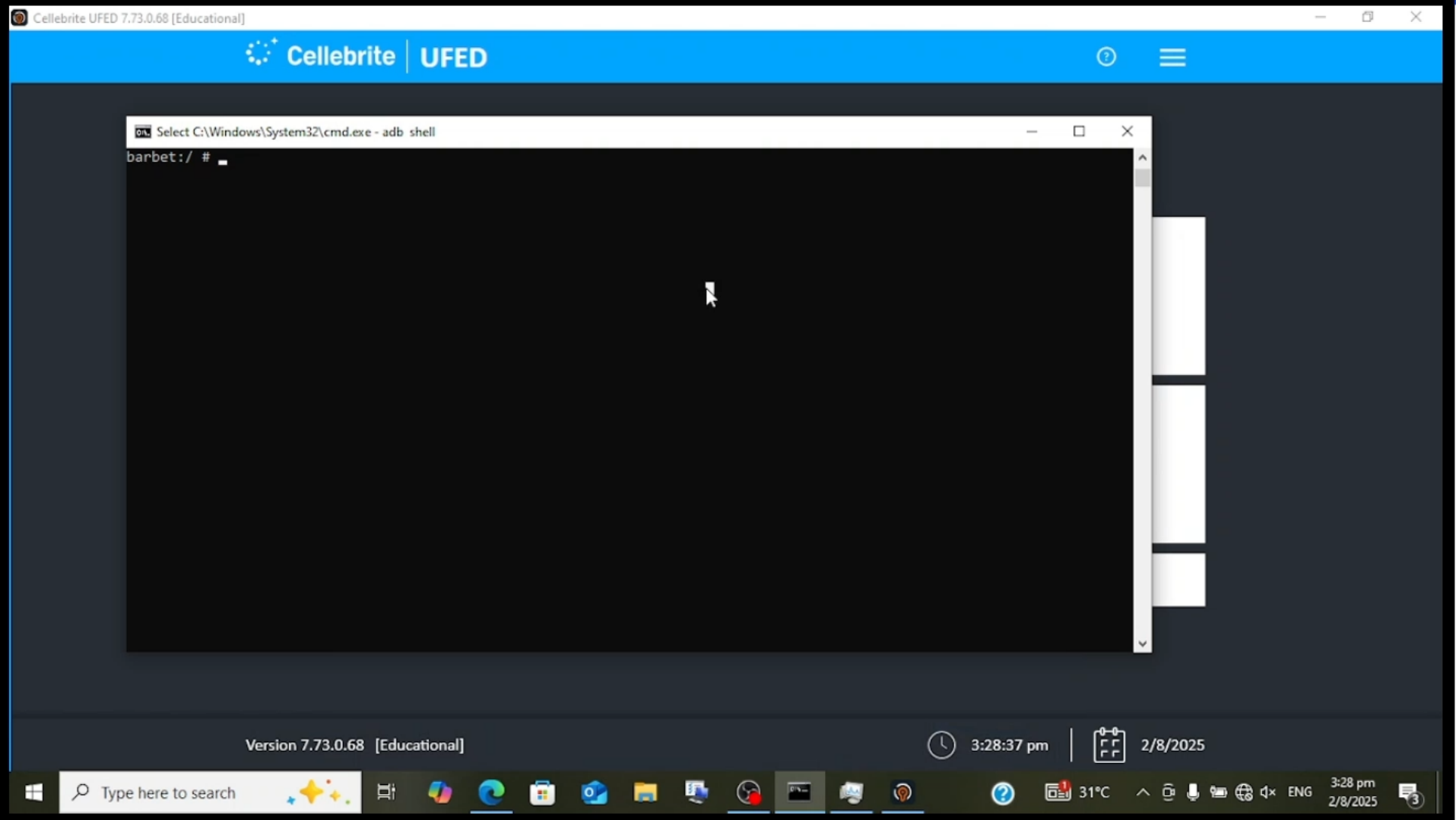
AFTER ENCRYPT

← secret.txt

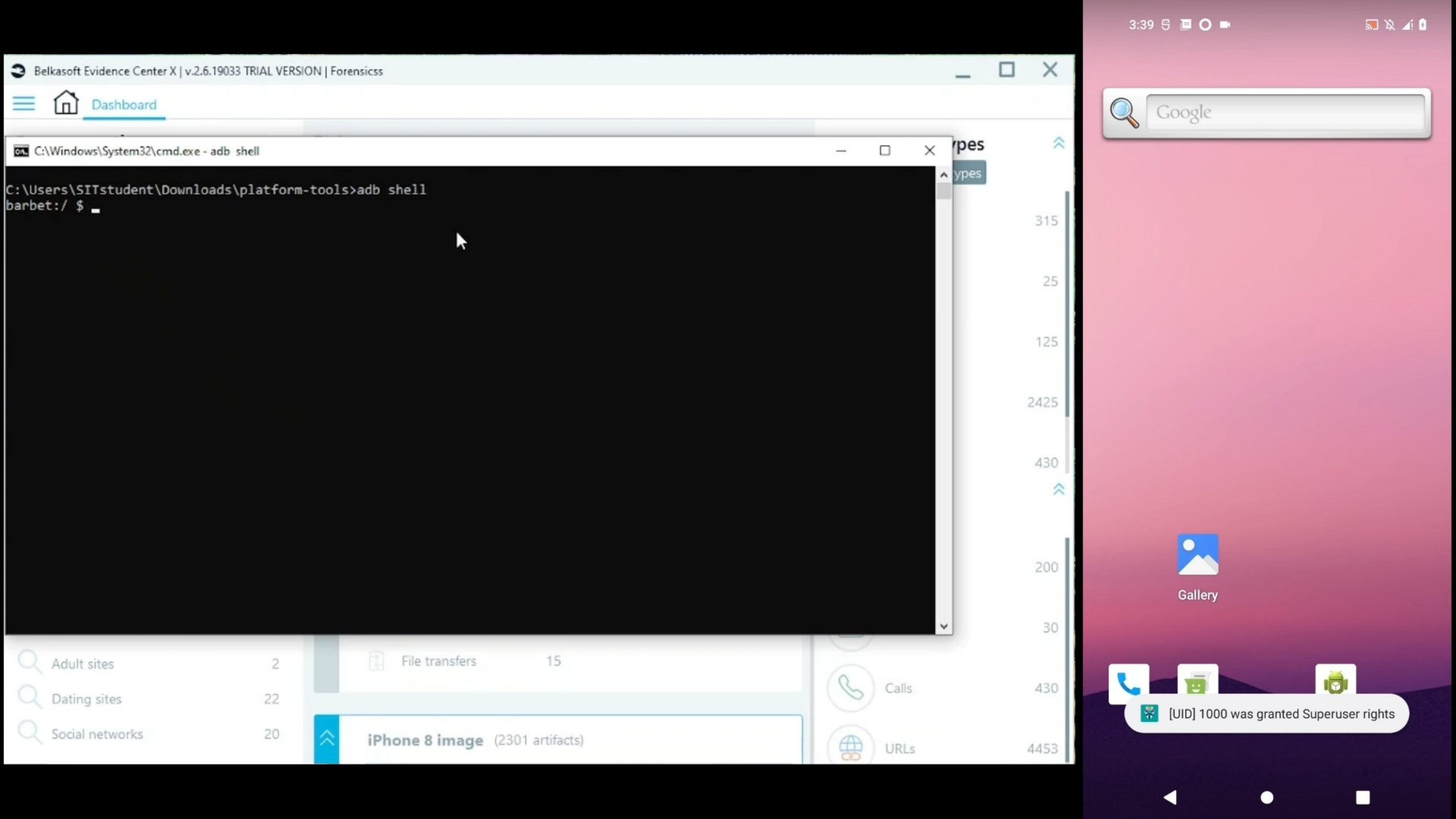
```
E+eN5GGDS2nJmBYS
AuiZXVODWnPHvl+N3j9j5O/uQYllJfXW14YWANzzlFopVymr4aHC
mIl6dbQN5YbeM65dw/Bkow==
```

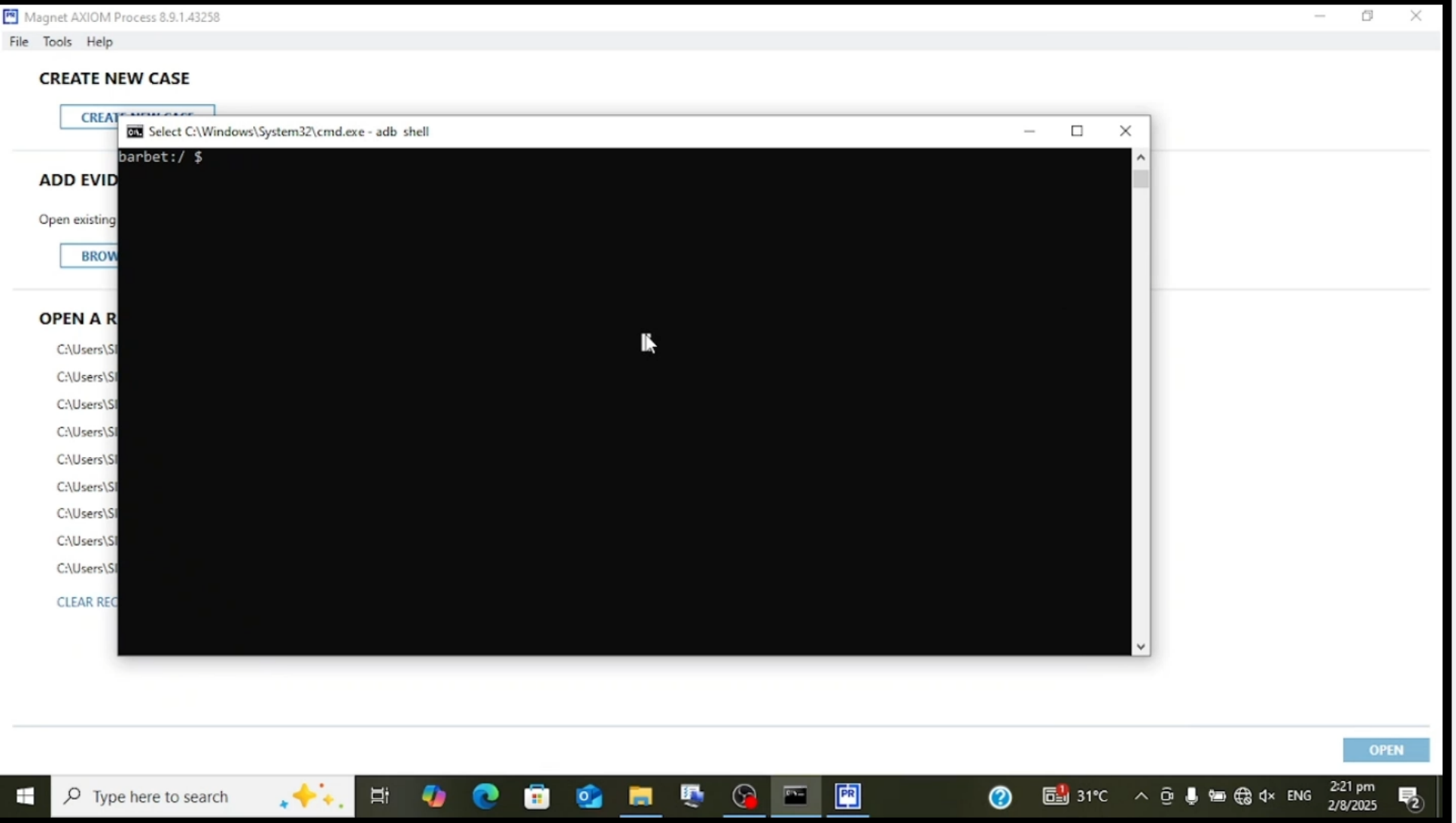
POC or it didn't happen

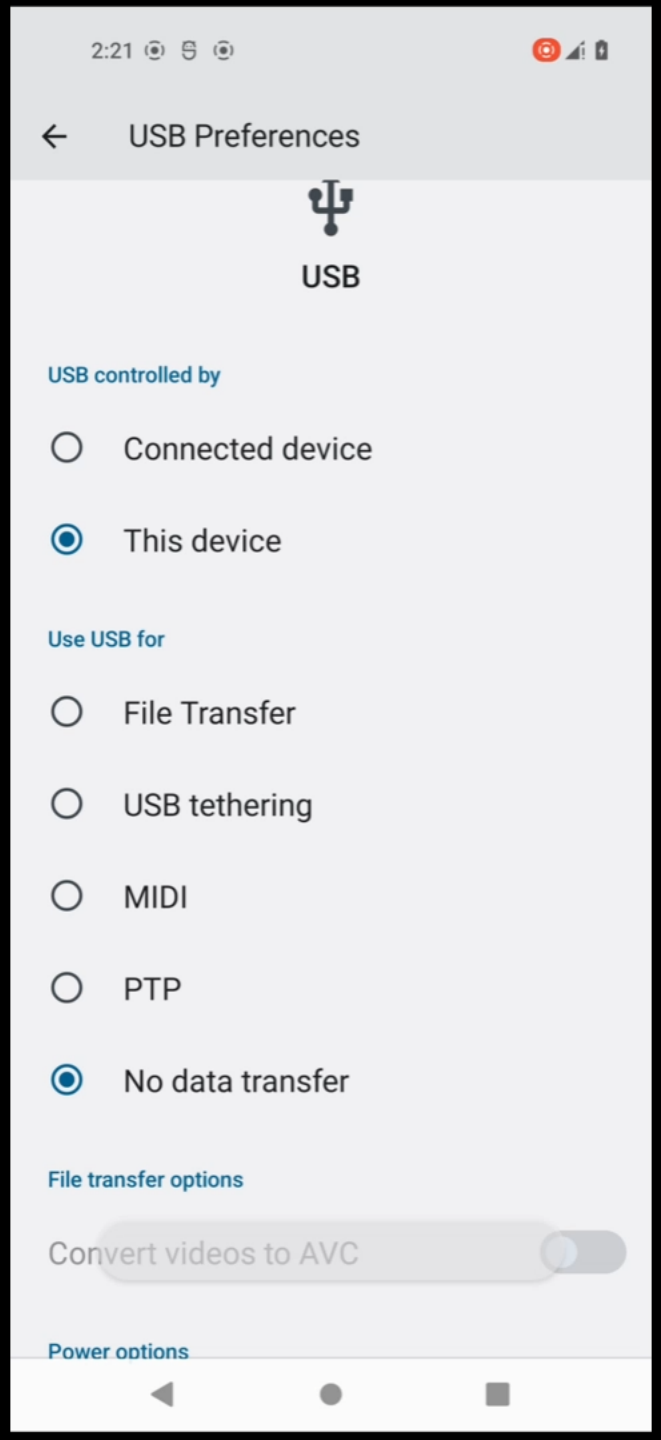
Proof-of-concept code available at https://github.com/Ins1ght32/SweetDeceit


Cellebrite UFED 7.73.0.68 [Educational]
Cellebrite | UFED
Select C:\Windows\System32\cmd.exe - adb shell
barbet:/ #
Version 7.73.0.68 [Educational]
3:28:37 pm
2/8/2025
Type here to search
31°C
ENG
3:28 pm
2/8/2025

3:28
LTE
Search contacts
Create new contact
J
John Doe
Favorites
Recents
Contacts


Belkasoft Evidence Center X | v.2.6.19033 TRIAL VERSION | Forensicss
Dashboard
C:\Windows\System32\cmd.exe - adb shell
C:\Users\SITstudent\Downloads\platform-tools>adb shell
barbet:/ $
Adult sites 2
Dating sites 22
Social networks 20
File transfers 15
iPhone 8 image (2301 artifacts)
Calls 430
URLs 4453
315
25
125
2425
430
200
30
3:39
Google
Gallery
[UID] 1000 was granted Superuser rights


Magnet AXIOM Process 8.9.1.43258
File Tools Help
CREATE NEW CASE
ADD EVID
Open existing
OPEN A R
CLEAR REC
OPEN
Select C:\Windows\System32\cmd.exe - adb shell
barbet:/ $
Type here to search
31°C
ENG
2:21 pm
2/8/2025


2:21
USB Preferences
USB
USB controlled by
Connected device
This device
Use USB for
File Transfer
USB tethering
MIDI
PTP
No data transfer
File transfer options
Convert videos to AVC
Power options

So, now what?

Mitigations and Defensive Strategies

- Selective vs bulk acquisition
- Exploit weakness in anti-forensics mechanism
  - Accelerated acquisition
- Cold system forensics (if ever possible)
- Know your tool

Conclusion

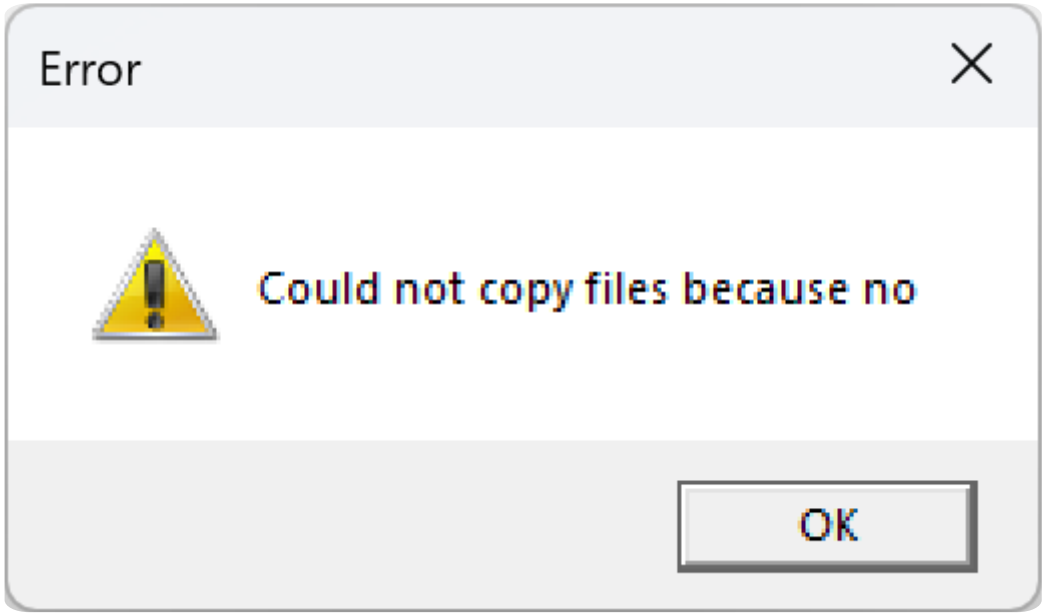
- Little prior research on how mobile forensic tools handle adversarially modified environments
- Our tests show Android devices can run stealthy, second-layer countermeasures that can compromise evidence
- Such countermeasures can silently manipulate or destroy data after extraction begins
- Longstanding forensic workflows may need rethinking to account for adversarial environments

Gaining device access is just the beginning - what happens next might not be what you think

ADVERSARY
VILLAGE

DEFCON

Thank You!


Error
Could not copy files because no
OK