RANDORISEC

MOBILE HACKING
FORENSICS

iOS CHEATSHEET
V0.1

MAIN STEPS

- Check iOS and configurations (VPN, Certificates, ...)
- Verify applications
- Verify iCloud settings
- Check Data
- Check Logs

RESOURCES

- SANS FOR585: Smartphone Forensic Analysis In-Depth – https://for585.com/
- Sarah Edwards iOS Forensic Research Blog – https://www.mac4n6.com/

TOOLS

- libimobiledevice
- 3uTools
- mvt
- APOLLO
- cheeky4n6monkey

Backup: Steps

Once the device connected through USB:

1. Activate backup encryption (optional, allows more data to be retrieved)
```
idevicebackup2 encryption on <PASSWORD>
```
2. Perform a full backup
```
idevicebackup2 backup --full <OUTPUT_DIR>
```
3. Decrypt backup
```
mvt-ios decrypt-backup -p <PASSWORD> -d <OUTPUT_DIR> <BACKUP_DIR>
```

Backup: Structure

The iOS backup folder consists of four files: **Info.plist, Manifest.db, Manifest.plist** and **Status.plist,** with folder names ranging from 00 to ff (0 to 255 in hexadecimal).

00..ff folders
These are the backed-up files location. Their content will be encrypted if the backup encryption was turned on.

Info.plist
A property list file containing metadata on the backed-up iOS device, such as the device name, iOS version, device type, and other information.

Manifest.db
An SQLite database that provides a table of all the files in the backup, as well as their metadata. The "Files" table includes information on every file and directory in the backup. Entries in the database include domain, relativePath, flags, fileID, and fileSize, which are used to identify and validate the backup files.

Manifest.plist
A property list file containing information on the current backup's properties. It also keeps a set of keys used to encrypt the backup.

Info.plist
A property list file containing information about the backup and restore processes. It includes information such as the backup date and time, the backup status, and any errors that occurred during the backup process.

Artifacts: Data

CallHistory.storedata at `/private/var/mobile/Library/CallHistoryDB/`, keeps records of incoming and outgoing calls, including from messaging applications
AddressBook.sqlitedb, AddressBookImages.sqlitedb at `/private/var/mobile/Library/AddressBook/` phone's address book records and pictures
sms.db at `/private/var/mobile/Library/SMS/`, sms sqlite database
ChatStorage.sqlite at `/private/var/mobile/Containers/Shared/AppGroup/*/`, whatsapp chat db
voicemail.db at `/private/var/mobile/Library/Voicemail/`, voicemail db
Accounts3.sqlite at `/private/var/mobile/Library/Accounts/`, Accounts information
healthdb.sqlite, healthdb_secure.sqlite at `/private/var/mobile/Library/Health/` health records, trainings, personal data, medical information, etc.

Artifacts: Internet

History, BrowserState.db at `*/Library/Safari/`, visited URLs history and open tabs (Safari)
Favicons.db at `*/Library/Image Cache/Favicons/`, favicons mapping database (Safari)
History, Favicons at `/private/var/mobile/Containers/Data/Application/*/Library/Application Support/Google/Chrome/Default/`, visited URLs history and favicons (Chrome)
browser.db at `/private/var/mobile/profile.profile/`, visited URLs history and favicons (Firefox)

Artifacts: Networking

com.apple.wifi.plist WiFi settings information
preferences.plist networking preferences (vpn, dns, proxies, interfaces, etc.)
com.apple.wifi-private-mac-networks.plist List of scanned networks with private mac
com.apple.wifi.known-networks.plist list of known networks information
com.apple.networkextension.plist network extension information
com.apple.MobileBluetooth.ledevices.other.db seen bluetooth devices
com.apple.MobileBluetooth.ledevices.paired.db paired bluetooth devices
com.apple.MobileBluetooth.devices.plist paired bluetooth devices

Artifacts: Configuration

MCProfileEvents.plist configuration profile events: process, operation (install, update, remove) and timestamp
ProfileTruth.plist configuration profile settings, restrictions and configuration
PayloadManifest.plist keeps track of ordered configuration profiles and hidden profiles
PayloadDependency.plist dependencies needed for configuration profiles payloads
profile-* configuration profile
com.apple.Preferences.plist system preferences

Artifacts: Applications

Shortcuts.sqlite at `/private/var/mobile/Library/Shortcuts/`, apps shortcuts
interactionC.db at `/private/var/mobile/Library/CoreDuet/People/`, user-app interactions
clients.plist at `/private/var/mobile/Library/Caches/locationd/`, apps location access requests
TCC.db at `/private/var/mobile/Library/TCC/`, applications permissions and access
mobile_installation.log.* apps installation logs

Artifacts: Processes – Files – Paths - Logs

DataUsage.sqlite at `/private/var/wireless/Library/Databases/`, keeps records about processes data usage
com.apple.osanalytics.addaily.plist at `/private/var/mobile/Library/Preferences/`, keeps records about processes data usage

RANDORISEC

MOBILE HACKING
FORENSICS

iOS CHEATSHEET
V0.1

MAIN STEPS

- Check iOS and configurations (VPN, Certificates, ...)
- Verify applications
- Verify iCloud settings
- Check Data
- Check Logs

RESOURCES

- SANS FOR585: Smartphone Forensic Analysis In-Depth – https://for585.com/
- Sarah Edwards iOS Forensic Research Blog – https://www.mac4n6.com/

TOOLS

- libimobiledevice
- 3uTools
- mvt
- APOLLO
- cheeky4n6monkey

Sysdiagnose: Steps

Sysdiagnose should be triggered by the user and then shared to be analyzed.

1. Hold volume UP and volume DOWN and Power Button for about 1 second.
2. Wait for the sysdiagnose to finish. The process can take several minutes.
3. Verify the sysdiagnose is ready at Settings > Privacy > Analytics > Analytics Data > `sysdiagnose_YYYY.MM.DD_hh-mm-ss-TZD_DEVICETYPE_OSBUILD`
4. Extract sysdiagnose archive from device

```
idevicecrashreport -e -k <OUTPUT_DIR>
```

Sysdiagnose: Structure

The sysdiagnose result will be generated at `/var/mobile/Library/Logs/CrashReporter/DiagnosticLogs/sysdiagnose/` as a TAR.GZ archive. It is a big collection of logs and the most important are the following:

/
The root folder has many .txt files which are basically commands output performed when the sysdiagnose was triggered. It also has log file extracted from the device and microstackshots of the device state.

logs
The logs folder is one of the juiciest folders. It has logs and configuration files about most of the services running on iOS.

WiFi
WiFi logs, configuration files and command results. A subfolder **CoreCapture** provides capture data.

summaries
All log summaries for every sysdiagnose process.

crashes_and_spins
Crash reports and spins. These files try to explain what happened in a crash/panic by providing stacktrace, memory usage and relevant system logs, or jusitifes system events like JetsamEvent.

system_logs.archive
Device's log archive collection

errors
Errors occurred in the sysdiagnose process

Artifacts: Applications

TCC.db applications permissions and access
mobile_installation.log.* apps installation logs
containermanagerd.log.* app container manager logs
powerlog_YYYY-MM-DD_*.PLSQL power usage logs (per app, process, etc.)

Artifacts: Networking

com.apple.wifi.plist WiFi settings information
preferences.plist networking preferences (vpn, dns, proxies, interfaces, etc.)
com.apple.wifi-private-mac-networks.plist List of scanned networks with private mac
com.apple.wifi.known-networks.plist list of known networks information
com.apple.networkextension.plist network extension information
com.apple.wifi.recent-networks.json recent wifi networks
wifi_status.txt WiFi information (interface, mac address, ip, etc.)
network_status.txt Network information (ipv4, ipv6, interface, dns, internet)
bluetooth_status.txt Bluetooth information (paired devices, mac address, etc.)
CoreCapture WiFi/BT data captures
ifconfig.txt All interfaces information
arp.txt ARP cache
3bars.txt Cellular signal information

Artifacts: Configuration

MCProfileEvents.plist configuration profile events: process, operation (install, update, remove) and timestamp
ProfileTruth.plist configuration profile settings, restrictions and configuration
PayloadManifest.plist keeps track of ordered configuration profiles and hidden profiles
PayloadDependency.plist dependencies needed for configuration profiles payloads
profile-* configuration profile
com.apple.Preferences.plist system preferences
PublicEffectiveUserSettings.plist user settings
EffectiveUserSettings.plist user settings
UserSettings.plist user settings

Artifacts: Processes

ps.txt Info about current processes
ps_thread.txt Info about current processes threads
taskinfo.txt Info about threads and thread priorities
launchctl-print-system.txt system domain description
launchctl-list-*.txt information about launched services
spindump-nosymbols.txt processes stack
Microstackshots user and kernel stacks
powerlog_YYYY-MM-DD_*.PLSQL power usage logs (per app, process, etc.)