COMPUTER FORENSICS LAB

London digital forensics & e-discovery · Established 2007

Android Secure Folders, Knox and Manufacturer Security Containers

A Second, Separately Locked Phone Inside the Phone, and Why It Is So Often Where the Evidence Is

Many Android phones can carry a second, separately secured space inside them. Samsung's Secure Folder, built on its Knox security platform, is the best known; other manufacturers offer their own private spaces, second spaces and app lockers, and Android's own work profile creates a separate, managed container. Each is, in effect, a second phone inside the phone: its own apps, its own copies of messaging and social apps under separate accounts, its own photos and files, sealed behind its own credential and its own encryption keys, and invisible from the ordinary side of the device. Precisely because it is separate and hidden, a security container is very often where a person keeps what they most want to conceal. For the lawyer, this means a container must be looked for on every Android device, understood as a separately locked space with its own key, and reached lawfully, because its protections are genuine and its contents may be the heart of the case. This guide sets out for UK lawyers what these containers are, how they are secured, how they are found and reached, and how their contents are recovered and read.

⏱ Estimated reading time: 24 min read

§ ABOUT THE AUTHOR

PREPARED BY COMPUTER FORENSICS LAB E-DISCOVERY TEAM

Computer Forensics Lab is a London-based digital forensics and electronic disclosure practice established in 2007. Security-container analysis is a specialist strand of its Android work: detecting Secure Folder, Knox, manufacturer private spaces and work profiles on a device, understanding their separate credentials and keys, reaching them lawfully, and recovering and reading the second-account apps, messages, photos and files they hold. The analysis treats the container as a second phone inside the phone and is reported under CPR Part 35 and CrimPR Part 19.

ESTABLISHED 2007 · LONDON | ISO 17025-ALIGNED PROCEDURES | SECURE FOLDER & KNOX FORENSICS | WORK PROFILES & PRIVATE SPACES | CPR PART 35 EXPERT REPORTS | FULL CHAIN-OF-CUSTODY DOCUMENTATION

§ CONTENTS

In this guide

§ 0 1 · ORIENTATION

Executive summary

The headline point: many Android phones carry a second, separately secured space, Samsung's Knox-based Secure Folder, other makers' private spaces, or Android's work profile, each a second phone inside the phone with its own apps, second-account messaging, photos and files, sealed behind its own credential and keys and hidden from the ordinary side, so it is very often where a person keeps what they most want to conceal, and it must be looked for on every device and reached lawfully.

The containers (§3): Secure Folder on Knox, manufacturer private and second spaces, app lockers, and the work profile, each a separate, managed space. **How they are secured (§4):** a separate credential, separate encryption keys backed by hardware, and isolation from the main profile, so the main unlock does not open them. **Finding and reaching (§5):** detected from system records even when hidden, and reached lawfully through their own credential, by cooperation or compulsion, never by attack. **What they hold (§6):** second-account apps and conversations, hidden photos and files, and their own deleted remnants, read on their own terms. **Deployment (§7):** revealing the concealed second life, and the parallel accounts and communications, that a party kept behind the container.

- **A second phone inside the phone:** separate apps, accounts, photos and files.
- **Its own lock and its own keys:** the main unlock does not open it.
- **Hidden by design:** often where the concealed evidence is.
- **Look for it on every device:** detected from system records even when hidden.
- **Reached lawfully, through its credential:** genuine protections, so the key not the exploit.

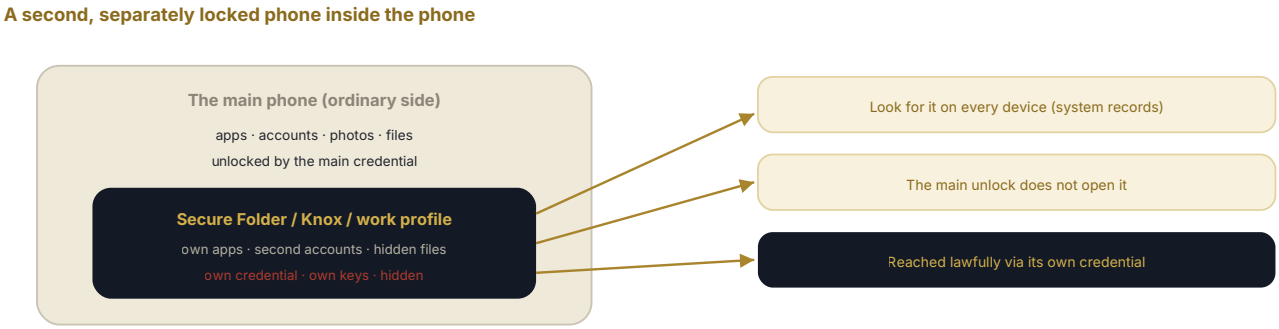

Figure 1 - a security container is a second, separately locked phone inside the phone, with its own apps, second accounts and hidden files behind its own credential and keys, looked for on every device and reached lawfully.

§ 0 2 · FIRST PRINCIPLES

The problem in plain English: a second phone inside the phone

An examiner who unlocks an Android phone and examines everything on it may still have missed the most important part, because many Android phones can hold a second, separate space that the ordinary unlock does not open. Samsung's Secure Folder, built on its Knox security platform, is the best known: a user can create it in minutes, and it becomes a sealed area with its own apps, its own separate copies of messaging and social apps signed into different accounts, its own photos, videos and files, and its own credential. Other manufacturers offer their own versions, private spaces, second spaces, app lockers and hidden areas, and Android itself provides the work profile, a managed container that an employer or the user can create to keep a separate set of apps and data. Whatever the name, the effect is the same: a second phone inside the phone, with its own lock, its own keys and, very often, its own hidden existence.

For the lawyer this matters for a simple reason. A space that is separate, separately locked and hidden from the ordinary side of the device is exactly where a person keeps what they most want to conceal: the second messaging account, the conversations with someone they are not supposed to be talking to, the photographs they would not want found, the files that would embarrass or incriminate them. Because it is invisible unless one knows to look, a party can produce their phone, unlocked, and appear entirely open while the container stays shut and unmentioned. Three things follow. A container must be looked for on every Android device, and it can be detected from the system's own records even when the user has hidden it. It must be understood as what it is, a genuinely separate space secured by its own credential and its own hardware-backed encryption keys, so that the main phone's credential does not open it and the same lawful, credential-led approach that governs the locked phone (guide 163) governs the container within it. And once opened, it is examined as a phone in its own right, its apps, accounts, messages, photos, files and deleted remnants recovered and read on their own terms, and attributed to the person with the same care as anything else. This guide sets out what these containers are, how they are secured, how they are found and reached lawfully, and how their contents are recovered and read.

§ 0 3 · THE CONTAINERS

The containers: Secure Folder, Knox, private spaces and work profiles

- **Samsung Secure Folder on Knox:** a user-created sealed space on Samsung devices, built on the Knox platform, with its own apps, accounts, files and credential, the most common consumer container (§4): the Samsung container.
- **Manufacturer private and second spaces:** other makers' equivalents, private spaces, second spaces, app lockers and hidden-file areas, differing in name and strength but sharing the second-space idea (guide 161): the maker-specific containers.
- **Android's work profile:** a managed container, created by an employer's management system or by the user, holding a separate set of apps and data under separate policies, and increasingly a private space for consumers (guide 144): the managed container.
- **Second-account apps:** the container's separate copies of messaging and social apps signed into different accounts, so one phone holds parallel accounts and conversations (guide 167): the parallel accounts.
- **Own storage and remnants:** the container's own storage holding its photos, files and app databases, with its own deleted remnants, examined as a phone in its own right (guide 170): the container's own evidence.

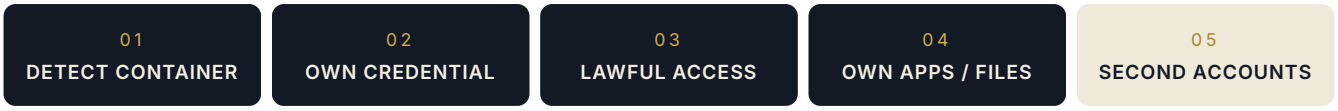

Figure 2 - detect the container from system records, identify its own credential, reach it lawfully, and recover its own apps, files and second-account conversations.

§ 0 4 · HOW CONTAINERS ARE SECURED

How containers are secured

- **A separate credential:** the container locked by its own PIN, pattern, password or biometric, distinct from the main lock, so unlocking the phone does not unlock the container (guide 163): the second lock.
- **Separate, hardware-backed keys:** the container's data encrypted with its own keys, derived from its credential and protected by the device's security hardware (Knox or equivalent), so its contents are sealed even to someone with the main phone open (guide 163): the second vault.
- **Isolation from the main profile:** the container's apps and data isolated from the main side, so the main profile's apps, backups and views cannot reach them (§3): the separation.
- **Hiding and tamper features:** the container able to be hidden from the app list, and some containers wiping or locking after failed attempts or tampering, so both concealment and destruction are designed in (§5): the concealment and self-protection.
- **Strength varies by maker:** Knox-backed containers strong, some other makers' lockers weaker, so the specific container's protections are assessed (guide 161): the per-container strength.

§ 0 5 · FINDING AND REACHING

Finding the container and reaching it lawfully

- **Look for it on every device:** a container sought on every Android device examined, since its existence is often undisclosed, and its absence confirmed rather than assumed (§2): the routine check.
- **Detect from system records:** the container's existence detected from the system's user and profile records, package lists and storage layout, even when the user has hidden it from the launcher (guide 167): the detection.
- **Identify its credential:** the container's own credential identified as the key, distinct from the main lock, and its type (PIN, pattern, password, biometric) established (§4): the second key.
- **Reach it lawfully:** the credential obtained by the owner's cooperation or lawful compulsion, and no attempt made that could trigger the container's lockout or wipe (guides 163, 117): the credential-led access.
- **Preserve state and acquire promptly:** the device kept powered and isolated, and the container acquired at depth once opened, since some container data is reachable only while it is unlocked (guides 162, 163): the window used.

§ 0 6 · WHAT IT HOLDS

What a container holds, and honest limits

- **Second-account conversations:** the container's messaging and social apps holding conversations under separate accounts, recovered with their deleted remnants (guides 165, 170): the parallel communications.
- **Hidden photos, videos and files:** media and documents moved into the container to hide them, with their metadata read from the originals (guide 166): the concealed material.
- **Its own apps and activity:** apps installed only in the container, with their own data and usage records (guides 167, 169): the container's own life.
- **Read on its own terms:** the container examined as a phone in its own right, its apps understood individually and its data attributed to the person with care, since a container implies intent to separate but not automatically wrongdoing (guides 151, 105): the honest reading.
- **Honest limits:** a container whose credential cannot be lawfully obtained remaining sealed, and that limit stated rather than overpromised, with the estate pursued for what synced out of it (guide 100): the honest boundary.

§ 0 7 · DEPLOYMENT

Deployment: the concealed second life

- **Revealing the undisclosed account:** a second messaging or social account, and its conversations, found in a container a party did not disclose (guide 167): the parallel account revealed.
- **Recovering the hidden material:** photographs, videos and files hidden in the container recovered with their metadata (guide 166): the concealed material recovered.
- **Evidencing concealment itself:** the creation, use and hiding of the container, and what was moved into it and when, evidencing intent to conceal, read in context (guide 110): the concealment evidenced.
- **Exposing incomplete disclosure:** a phone produced "unlocked" while its container stayed shut and unmentioned, so disclosure is shown to be incomplete (guide 98): the incomplete production exposed.
- **Lawful, honest presentation:** the container reached lawfully, its contents recovered at depth, attributed with care, and presented at honest confidence under CPR Part 35 or CrimPR Part 19 (guides 100, 130): the container evidence deployed defensibly.

§ 0 8 · THE WIDER MAP

Source architecture: where else the evidence lives

Per this series' source-architecture discipline: a security container is a sealed node within the device, but its contents often reach beyond it, and its existence leaves traces the container cannot hide. The container holds its own apps, second-account data, hidden files and remnants behind its own credential. But the second accounts it runs sync to their own clouds and providers (guide 164 for Google, and the app providers for the rest), reachable lawfully whether or not the container opens; the counterparts of its conversations hold their side; a Samsung account or manufacturer cloud may back the container up, with its own credential; the main profile's system records reveal the container's existence and history even when hidden; and the device's usage artefacts (guide 169) record when the container was used. The discipline is to detect the container from the main side's records, reach it lawfully through its own credential, and in parallel pursue the second accounts' clouds, providers and counterparts, so that a container whose credential cannot be obtained does not seal the evidence that synced out of it. When the container stays shut, the accounts' clouds and the counterparts hold the conversations; when it opens, it is examined as a second phone; and its existence and use are evidenced from the main side either way.

EVIDENCE	INSIDE THE CONTAINER	MAIN PROFILE (SYSTEM RECORDS)	SECOND ACCOUNTS' CLOUDS / PROVIDERS	COUNTERPARTS	CONTAINER BACKUP (MAKER CLOUD)	USAGE ARTEFACTS
Container exists	n/a	Y: profile / package records	Account sign-ins	n/a	Y: backup presence	Y: use events
Second-account messages	Y: if opened	No (isolated)	Y: if synced / backed up	Y: their side	Y: if backed up	Notification traces
Hidden photos / files	Y: if opened	Move traces (media store)	Cloud copies (if synced)	Shared copies	Y: if backed up	n/a
Container apps / activity	Y: if opened	Package records	App accounts	n/a	Some	Y: usage records
If credential unobtainable	Sealed	Y: existence proven	Y: contents via cloud	Y: their copies	Sealed (own credential)	Y: use proven

Fallback strategy in one line: detect the container from the main side's records, reach it lawfully via its own credential, and in parallel pursue the second accounts' clouds and counterparts, so a sealed container does not seal the evidence that synced out of it.

§ 0 9 · IN THE WILD

Worked examples

EXAMPLE 1 · THE UNLOCKED PHONE WITH THE LOCKED FOLDER INSIDE

A party produced their Samsung phone unlocked, apparently fully open, and its main side held nothing of note. The §5 routine check found what the production had omitted: the §5 system records showed a Secure Folder, hidden from the app list, with its §3 own copies of two messaging apps signed into accounts the party had not disclosed. Its §4 own credential was §5 obtained by lawful compulsion (guide 117), and once opened the container was examined as a second phone: the §6 second-account conversations, including §6 deleted remnants (guide 170), and hidden photographs were recovered. The §7 incomplete disclosure was exposed. The open phone had concealed a closed one. The lesson is §2's and §5's: a container is looked for on every Android device and detected from the system's records even when hidden, because a phone produced "unlocked" may still hold a separately locked space where the undisclosed accounts and material live.

EXAMPLE 2 · THE CONTAINER THAT STAYED SHUT, AND THE ACCOUNTS THAT DID NOT

A container was detected on a device, but its §4 credential could not be lawfully obtained and its Knox-backed protections made it a genuine §4 vault that no attempt should be made to attack. The §6 honest limit was stated: the container remained sealed. The §8 estate was pursued instead: the §3 second accounts the container ran had synced their conversations to their §8 providers' clouds, reached by lawful process (guide 164), and the §8 counterparts held their side, so the conversations were recovered without opening the container, and the §8 main profile's records and usage artefacts (guide 169) proved the container's existence and use. The vault stayed shut; its contents were reached around it. The lesson is §8's: a container's contents often sync beyond it, so when its credential cannot be obtained, the second accounts' clouds, providers and counterparts supply the conversations, and the container's existence and use are evidenced from the main side regardless.

EXAMPLE 3 · THE CONTAINER THAT WAS A WORK PROFILE, READ FAIRLY

A device held a container that a party urged as evidence of concealment. The §3 examination identified it as an employer-created §3 work profile, holding the party's work apps and email under the employer's management (guide 144), not a hidden personal space. Its contents were §6 read on their own terms and attributed fairly, and the §6 principle applied: a container implies separation, not automatically wrongdoing. The work profile evidenced a job, not a secret. The lesson is §6's: containers come in kinds, and a container's existence is read in context, a managed work profile is not a hidden vault, so the container is identified for what it is and its contents attributed and interpreted fairly rather than treated as inherently suspicious.

§ 10 · WHERE IT GOES WRONG

Common mistakes and technical limitations

Common mistakes

- **Not looking for a container:** the cardinal error, accepting an unlocked phone as fully open (Example 1, §5).
- **Assuming the main credential opens it:** not recognising the container's separate lock and keys (§4).
- **Attacking the container:** attempting to defeat Knox-backed protections and risking lockout or wipe (§5, guide 163).
- **Giving up when it stays shut:** not pursuing the second accounts' clouds and counterparts (Example 2, §8).
- **Treating every container as concealment:** reading a work profile as a hidden vault (Example 3, §6).
- **Missing the second accounts:** not recognising that container apps run separate accounts and conversations (§3).
- **Shallow extraction:** a logical extraction that never reaches the container's data (§5, guide 162).
- **Not evidencing existence and use:** failing to use the main side's records to prove the container was there and used (§8).

Technical limitations

- **Genuine protection:** a Knox-backed container without its credential may be inaccessible: the core limit (§4).
- **Lockout and wipe risk:** attempts can destroy the container's contents (§4).
- **Isolation:** the main profile's backups and views do not reach container data (§4).
- **Strength varies:** some makers' lockers are weaker than Knox (§4).
- **Containers evolve:** features, hiding and protections change across makers and versions (§3).

§ 11 · INTERROGATORIES & DRAFTING AIDS

Questions to ask · Suggested wording

Ask your client

- Does the device have a Secure Folder, private space, second space or work profile, and what accounts and material are in it?
- Is the container's own credential known or obtainable, since the main unlock does not open it?
- Do the container's accounts sync to clouds we can reach if the container stays shut?

Ask your opponent

- Please confirm whether the device at Schedule 1 contains any Secure Folder, Knox container, private or second space, app locker or work profile, disclose its contents and accounts, and provide its credential for lawful access.
- Please confirm all accounts used in any messaging or social app within any such container.
- Please preserve the device powered on and isolated, and do not delete, hide or reset any container.

Ask your eDiscovery / forensic provider

- Has every device been checked for a container from its system records, including hidden ones?
- What are the container's protections and credential, and what is the lawful route to it?
- If it stays shut, what do the second accounts' clouds, counterparts and the main side's records supply?

SUGGESTED WORDING · INSTRUCTION FOR A SECURITY-CONTAINER ANALYSIS

> "We instruct you to examine the Android device at Schedule 1 for security containers relevant to Schedule 2. Please, preserving the device's state and integrity: (1) detect from the system's user, profile, package and storage records whether the device contains any Secure Folder, Knox container, manufacturer private or second space, app locker or work profile, including any hidden from the launcher, and identify each container's kind and protections; (2) identify each container's own credential and obtain it lawfully by cooperation or compulsion, making no attempt that could trigger lockout or wipe; (3) once opened, acquire the container at sufficient depth and examine it as a phone in its own right, recovering its apps, second-account conversations, hidden photos and files, and deleted remnants; (4) where a container's credential cannot be obtained, state that limit honestly and pursue the second accounts' clouds and providers, the counterparts, and any container backup; (5) evidence the container's existence, creation, hiding and use from the main profile's records and usage artefacts; and (6) identify each container for what it is (a personal vault or a managed work profile), attribute its contents with care, and present findings at honest confidence."

§ 12 · QUICK CONTROL

Checklist and red flags · When to involve a digital forensic expert

The security-container checklist

- ☐ Every device checked for a container from system records
- ☐ Hidden containers detected; kind and protections identified
- ☐ Container's own credential identified as distinct from the main lock
- ☐ Credential obtained lawfully; no lockout or wipe risked
- ☐ Device kept powered and isolated; container acquired promptly once open
- ☐ Container examined as a second phone: apps, accounts, files, remnants
- ☐ Second-account conversations recovered and attributed
- ☐ Sealed container's contents pursued via clouds, providers and counterparts
- ☐ Existence, creation, hiding and use evidenced from the main side
- ☐ Work profiles distinguished from personal vaults; contents read fairly

Red flags

- An unlocked phone accepted as fully open without a container check: Example 1.
- The main credential assumed to open the container.
- An attempt to attack a Knox-backed container.
- A sealed container treated as the end of the enquiry: Example 2.
- A work profile read as a hidden vault: Example 3.
- Second accounts in the container missed.
- A shallow extraction relied on for container data.

When to involve a digital forensic expert

On every Android device, because a container can be hidden and a phone produced unlocked may still hold a separately locked space where the concealed evidence lives: the expert checks every device for Secure Folder, Knox containers, manufacturer private and second spaces, app lockers and work profiles, detecting them from the system's records even when hidden (Example 1, §5), and identifies each container's own credential and protections, reaching it lawfully by cooperation or compulsion and never by an attack that risks lockout or wipe (§4, §5, guide 163). Once open, the container is acquired at depth and examined as a second phone, its second-account conversations, hidden media, files and deleted remnants recovered and attributed with care (§6); where it stays sealed, the second accounts' clouds, providers and counterparts are pursued and the container's existence and use evidenced from the main side (Example 2, §8); and each container is identified for what it is, a personal vault or a managed work profile, and read fairly (Example 3). Findings are reported at honest confidence under CPR Part 35 or CrimPR Part 19. This guide continues the Android block. Through **cflab.uk** and **e-discovery.uk**, container work finds the second phone inside the phone, opens it lawfully where the key can be

obtained, and reaches around it where it cannot, so that the space built for concealment does not succeed in concealing.

§ 13 · COMMON QUESTIONS

Frequently asked questions

What is a Secure Folder or security container?

A second, separately locked space inside an Android phone (§2, §3). Samsung's Secure Folder, built on its Knox platform, is the best known: a sealed area with its own apps, its own copies of messaging and social apps under separate accounts, its own photos and files, and its own credential. Other manufacturers offer private spaces, second spaces and app lockers, and Android's work profile is a managed container. Each is, in effect, a second phone inside the phone, sealed behind its own lock and keys and often hidden from the ordinary side of the device.

If the phone is unlocked, isn't everything accessible?

No, and that assumption is the commonest mistake (§4, Example 1). A container is secured by its own credential and its own hardware-backed encryption keys, distinct from the main lock, and isolated from the main profile, so unlocking the phone does not unlock the container, and the main side's apps, backups and views cannot reach its contents. A party can therefore produce their phone unlocked, apparently fully open, while the container stays shut and unmentioned. So every device is checked for a container from its system records, and the container's own credential is sought.

How do you find a hidden container?

From the system's own records (§5). Even when a user has hidden the container from the app list, its existence is recorded in the device's user and profile records, package lists and storage layout, and its use in the usage artefacts, so it is detected regardless of the hiding. This is why a container is looked for on every Android device as a routine step and its absence confirmed rather than assumed. Once found, its kind, Secure Folder, private space, work profile, and its protections are identified so the lawful route to it can be planned.

Can the container be broken into?

Not safely, and it should not be attempted (§4, §5). A Knox-backed container is a genuine vault: its keys are protected by the device's security hardware, and some containers lock or wipe after failed attempts or tampering, so an attack risks destroying the very contents sought. The container is reached lawfully through its own credential, obtained by the owner's cooperation or by lawful compulsion where the law provides, exactly as for the locked phone. Where the credential cannot be obtained, the container stays sealed, that limit is stated honestly, and the estate is pursued for what synced out of it.

If the container stays locked, is the evidence lost?

Often not (§8, Example 2). The second accounts a container runs usually sync their conversations to their own providers' clouds, reachable by lawful process whether or not the container opens; the counterparts hold their side of every conversation; a manufacturer cloud may back the container up; and the main profile's records and usage artefacts prove the container's existence, creation, hiding and use. So a sealed container rarely seals the evidence that synced out of it, and the concealment itself can still be evidenced from the main side of the device.

Does having a container prove someone was hiding something?

Not by itself, and containers are read in context (§6, Example 3). A container implies an intention to keep something separate, but not automatically wrongdoing: an employer-created work profile holds a person's job, not a secret, and many people keep private spaces for ordinary privacy. So each container is identified for what it is, its contents are attributed and interpreted fairly, and what was moved into it and when is read in context. Where a container does hold undisclosed accounts or material, that is evidenced; where it holds a job or ordinary privacy, it is reported as such.

§ 14 · REFERENCE

Glossary

Secure Folder
Samsung's sealed second space on a device.

Knox
Samsung's hardware-backed security platform.

Private / second space
Other makers' equivalents of a container.

App locker
A feature locking or hiding selected apps.

Work profile
Android's managed container for work apps and data.

Container credential
The container's own lock, distinct from the main one.

Isolation
Separation of container data from the main profile.

Second-account app
A container's app signed into a different account.

Hidden container
A container removed from the app list.

Profile records
System records revealing a container's existence.

Sources and authoritative references

The container disciplines in this guide draw on materials published by our laboratory and e-discovery practice, referenced here as sources:

- cflab.uk, digital forensics services (Secure Folder and Knox forensics, work profiles and private spaces, lawful container access, CPR Part 35 and CrimPR Part 19 reporting): cflab.uk/digital-forensics-services · guides library: cflab.uk/guides
- e-discovery.uk, EDRM Phase 02 · Identification and Phase 04 · Collection (identifying and collecting from separate spaces as practised): e-discovery.uk/edrm/identification · e-discovery.uk/edrm/collection
- e-discovery.uk, practice method and Knowledge Centre: e-discovery.uk · e-discovery.uk/knowledge
- Samsung Knox and Secure Folder documentation; Android work profile documentation: docs.samsungknox.com · samsung.com, Secure Folder · developer.android.com, work profiles
- Regulation of Investigatory Powers Act 2000, Part III (disclosure of protected information); Forensic Science Regulator, Code of Practice v2; ISO/IEC 27037: legislation.gov.uk, RIPA Part III · gov.uk, FSR Code · iso.org, 27037
- PD 57AD and CPR Part 31 · CPR Part 35: justice.gov.uk, PD 57AD · justice.gov.uk, Part 35

§ HOW A SPECIALIST LABORATORY CAN ASSIST

Working with Computer Forensics Lab

Container work at the laboratory finds the second phone inside the phone. Every Android device is checked, as a routine step, for Secure Folder, Knox containers, manufacturer private and second spaces, app lockers and work profiles, detected from the system's user, profile, package and storage records even when the user has hidden them from the launcher, so that a phone produced "unlocked" is never accepted as fully open until its containers are accounted for (Example 1). Each container is identified for its kind and protections, and its own credential, distinct from the main lock, is obtained lawfully by cooperation or compulsion, with no attempt made that could trigger the container's lockout or wipe (guides 163, 117). Once open, the container is acquired at sufficient depth and examined as a phone in its own right, its apps, second-account conversations, hidden photos and files, and deleted remnants recovered and attributed with care (guides 162, 165, 166, 170). Where a container's credential cannot be obtained, that limit is stated honestly and the second accounts' clouds, providers and counterparts are pursued, while the container's existence, creation, hiding and use are evidenced from the main profile's records and usage artefacts (Example 2, guides 164, 169). Each container is read fairly for what it is, a personal vault or a managed work profile, and findings are reported at honest confidence under CPR Part 35 or CrimPR Part 19 (Example 3). This guide continues the Android block. Conflicts checks, confidential scoping discussions and fixed-fee estimates are routine; and the guiding rule is simple: look for the second phone on every device, open it lawfully where the key can be had, and reach around it where it cannot.

INSTRUCT THE LAB

Speak to a forensic examiner, not a salesperson.

Describe your matter in confidence and an examiner will respond the same working day. Conflicts checks and scoping calls with US counsel are routine.

NEW ENQUIRIES
+44 (0)20 7164 6971

EMAIL
info@cflab.uk

E-DISCOVERY
e-discovery.uk

Computer Forensics Lab Ltd · Euro House, 133 Ballards Lane, Finchley, London N3 1LJ, United Kingdom
cflab.uk · Customer service +44 (0)20 7164 6915 · ICO ZB395023 · NPCC · ISO 17025 aligned practice · UK Gov Digital Marketplace