



C O M P U T E R F O R E N S I C S L A B

London digital forensics & e-discovery · Established 2007

Google Vault in Litigation

Capabilities, Limits and the Boundary with Independent Forensic Collection

Google Vault is Workspace's built-in preservation, search and export service, and for many disputes it is the first tool anyone reaches for: holds in minutes, searches across mailboxes and Drives, exports with metadata reports. Used well, it is the backbone of defensible Workspace disclosure. Used as if it were a forensic laboratory, it quietly narrows the evidence: services it cannot reach, structures its exports flatten, operators whose independence the dispute itself puts in question. This guide is the Vault counterpart to this series' Purview analysis: what Vault genuinely does, where its reach and fidelity stop, the three-question framework for choosing between Vault and independent collection, the hybrid patterns that combine them, and the disputes: expired-retention claims, contested custodians, revision-level authorship: where the choice decides the evidence.

© Estimated reading time: 26 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. Its Workspace practice runs Vault daily: holds, searches and exports at disclosure scale: and runs past it wherever the matter demands: API collection of revision streams, independent verification of contested custodians, and the countersigned-manifest patterns that let in-house tooling and laboratory independence work together. Its eDiscovery arm, **e-discovery.uk**, processes both into review.

ESTABLISHED 2007 · LONDON

ISO 17025-ALIGNED PROCEDURES

VAULT & API COLLECTION

INDEPENDENT VERIFICATION

CPR PART 35 EXPERT REPORTS

FULL CHAIN-OF-CUSTODY DOCUMENTATION

§ CONTENTS

In this guide

- 01 Executive summary
- 02 The problem in plain English: the excellent tool and the quiet narrowing
- 03 What Vault does well: holds, retention, search, export
- 04 Where Vault stops: reach, fidelity and the operator
- 05 The three questions: trust, structures, downstream
- 06 Hybrid patterns: Vault plus independence
- 07 Vault in correspondence: what to ask and what to offer
- 08 Source architecture: where else the evidence lives
- 09 Worked examples
- 10 Common mistakes and technical limitations
- 11 Questions to ask · Suggested wording
- 12 Checklist and red flags · When to involve a digital forensic expert
- 13 Frequently asked questions
- 14 Glossary · References · Disclaimer · How a specialist laboratory can assist

Executive summary

THE HEADLINE POINT: VAULT IS THE ESTATE'S PRESERVATION BACKBONE AND A BOUNDED COLLECTOR: KNOW WHICH JOB YOU ARE GIVING IT

Vault does four things with real authority. **Holds:** litigation holds by custodian, organisational unit or Shared Drive, across Gmail, Drive, Chat (history-on) and Meet: preserving content through deletion and past retention expiry, with placement dates that become the governing facts of every later recovery question. **Retention:** default and custom rules whose scope and change history are the paperwork behind every "expired" claim. **Search:** term, date and custodian queries across the held estate, with counts that drive scoping and DRD arguments. **Export:** productions with metadata files and account-level manifests: the defensible bulk route. Its boundaries are equally definite. **Reach:** history-off Chat, some service data and everything outside Workspace sit beyond it; deleted-account content survives only where holds or rules already applied. **Fidelity:** exports render native documents as snapshots: revision streams (guide 58's crown jewels) do not travel as series; label taxonomies and some structures need deliberate handling downstream. **Operator:** Vault is driven by the client's administrators inside the client's tenant: where those administrators, or the organisation itself, are within the dispute's perimeter, self-collection trust questions arise exactly as guide 52 framed for Purview. The decision framework therefore transfers whole: **trust** (who operates, who verifies), **structures** (does the matter turn on revisions, threads, labels, settings histories), **downstream** (what will processing, review and the other side's expert need the export to prove about itself). Vault answers the bulk; the API route answers the structures; independence answers the trust: and the hybrid patterns of §6 let one matter use each where it is strongest, stitched by countersigned manifests rather than by hope.

- **Vault for preservation, always and immediately:** whatever collection route follows, the holds go in through Vault on day one: nothing else stops the clocks.
- **Vault for bulk, by design:** scoped, counted, manifested exports across custodians: the defensible default for the uncontested mass.
- **Not Vault alone for the contested core:** disputed documents' revision streams, contested custodians' estates, and structures the issues turn on go the structural or independent route.
- **Vault's own operations are logged:** holds, releases, searches, exports: guide 58 §7's conduct record, requested from opponents in every retention dispute.
- **Fidelity stated, never assumed:** every production says what its route preserved and flattened: the honesty that pre-empts the export-mechanics cross-examination.

The problem in plain English: the excellent tool and the quiet narrowing

Vault's virtue is availability: it is already there, already licensed, already understood by the client's IT team: and in the first week of a dispute that availability is priceless, because holds placed in minutes beat perfect collections planned for later. The hazard arrives afterwards, when availability quietly becomes strategy: the whole disclosure run through the built-in tool because it was to hand, custodians under suspicion exporting their own estates, revision-level questions answered from snapshot exports, and nobody having decided: as opposed to defaulted: that Vault was the right instrument for each job it was given.

The correction is not scepticism about Vault; it is assignment of roles. Preservation: Vault, always. Bulk collection: Vault, usually. The contested core, the structural evidence, the trust-sensitive custodians: decided deliberately, with the guide 52 questions asked out loud and the answer documented. The difference between the two postures is a paragraph in the collection specification and, some months later, the difference between an export that proves itself and one that gets cross-examined.

What Vault does well: holds, retention, search, export

- **Holds that mean it:** custodian-, OU- and Shared Drive-scoped, preserving through user deletion and retention expiry: with the placement date as the recoverability watershed guide 58 keeps citing; hold reports exhibit scope and dates cleanly.
- **Retention rules as governance:** defaults and custom rules per service, with effective dates: read together with their change history, they are the estate's memory policy in writing: exactly the paperwork "expired" disputes demand.
- **Search that scopes:** terms, dates, custodians, services: fast counts for proportionality arguments, sampling for term testing (the guides 84-85 disciplines ahead), and matter-workspace organisation that keeps parallel disputes separated.
- **Export with self-description:** productions accompanied by metadata and account manifests: counts reconcilable, gaps visible: the properties that let a Vault production answer completeness challenges with documents rather than assurances.
- **Operations that are themselves audited:** every hold, release, search and export an event in the admin layer: the tool watches its operators, which §7 turns into correspondence leverage.

Where Vault stops: reach, fidelity and the operator

- **Reach:** history-off Chat was never kept (guide 58's Example 2); some service data and third-party apps signed in with Google sit outside; deleted accounts' content survives only where holds or rules pre-dated deletion: Vault preserves the kept, not the never-kept.
- **Fidelity:** exports snapshot native documents: revision streams stay behind unless the API route collects them as series; label structures, settings histories and some relational context need deliberate downstream handling: the flattening is documented behaviour, not a defect, and the specification is where it is managed.
- **The operator:** Vault runs with the client's admin credentials, inside the client's tenant, usually by the client's people: for uncontested bulk this is fine and normal; for custodians whose conduct is the case, it is guide 52's self-collection problem restated, and the independence patterns of §6 exist for it.
- **Scope illusions:** a Vault search covers what Vault reaches: an "all custodians, all services" claim inherits every reach boundary above, and completeness statements are drafted with the boundary map open.
- **The device-side silence:** Vault knows nothing of sync mirrors, app caches and local artefacts: the ensemble sources guides 54 and 58 rescue cases from live entirely outside it.

The three questions: trust, structures, downstream

- **Trust:** who will operate the collection, and does the dispute put them inside its own perimeter? Client IT running Vault for uncontested custodians: routine. The finance director exporting the finance director's mailbox: the pattern every self-collection authority warns about, answered by independent operation or supervised operation with the supervision recorded.
- **Structures:** does the matter turn on things the compliance export flattens: revision streams, Chat threading, label states, settings change histories, sharing configurations? If yes, the API route collects those sources as structures, whatever Vault does for the rest.
- **Downstream:** what will processing, review, and the opposing expert need the collection to demonstrate about itself: manifests, hashes, query records, fidelity statements? The route that can produce its own paperwork wins the argument that eventually comes.
- **Asked per source, not per matter:** the framework's output is a routing table: this custodian Vault, that custodian independent, these documents API: guide 52's discipline, and the specification records the answers.
- **Re-asked when the dispute moves:** a custodian who becomes contested mid-matter moves routes; the countersigned patterns below make the move cheap.

Hybrid patterns: Vault plus independence

- **Vault-collect, laboratory-verify:** client IT exports; the laboratory re-runs sampled queries, reconciles counts and countersigns the manifest: independence purchased at verification cost, not duplication cost: the workhorse pattern for cooperative clients with contested edges.
- **Split routing:** Vault for the bulk; API/independent for the contested core (named custodians, disputed documents' revision streams, settings histories): one specification, two routes, joined by hashes.
- **Supervised operation:** the laboratory directs and witnesses the client's Vault work in real time where credentials cannot transfer: the supervision note travels with the export.
- **Preserve-now, decide-later:** Vault holds everything on day one; routing decisions follow at leisure: the pattern that exists because preservation cannot wait for strategy and never has to.
- **Cross-checking the opponent:** their Vault production tested against its own metadata files, the admin-layer operations log, and targeted requests for structures their route flattened: §7's programme.

Vault in correspondence: what to ask and what to offer

- **Ask for the hold report:** scope, custodians, services, placement dates: the two-line disclosure that dates every later recoverability argument.
- **Ask for the retention rules with change history:** and read them against the duty's chronology: guide 57's Example 2 pattern, Google edition.
- **Ask for the Vault operations log:** holds, releases, searches, exports, with acting accounts: the conduct record their own tenant kept.
- **Ask route-and-fidelity questions:** which tool produced the disclosure, what did it preserve of revisions, labels and threading, and where are the metadata files: one paragraph that surfaces flattening before review finds it the hard way.
- **Offer the same:** volunteering your own hold report, route statement and fidelity note costs little, builds the cooperation record PD 57AD expects, and makes your later requests hard to refuse.

§ 0 8 · THE WIDER MAP

Source architecture: where else the evidence lives

Per this series' source-architecture discipline: never treat Vault's reach as the estate. The Workspace evidence Vault preserves coexists with: the structures only the API route exports (revision streams as series, activity records, settings histories); the admin-layer reports (the diary, with Vault's own operations inside it); device-side sync mirrors and app caches (outside Vault entirely, and the rescue layer of guide 58's Example 2); counterpart tenants and external participants (their holds, their clocks); personal Google accounts where sync or forwarding crossed over; third-party backup platforms with their own point-in-time sets; and the exported-copy trail (prior Takeouts, circulated productions, migration exports) whose existence the operations log reveals. When Vault cannot reach it: never kept, deleted pre-hold, outside Workspace: the walk runs structures, devices, counterparts, backups, in that order, with the operations log framing what once existed.

EVIDENCE	VAULT (HOLDS + EXPORT)	API / STRUCTURAL ROUTE	ADMIN REPORTS	DEVICE MIRRORS / CACHES	COUNTERPARTS + BACKUPS	DELETED / RECOVERABLE
Mail + Drive content	Y: the backbone	Y, with structures	Activity events	As last synced	Y	Hold date governs
Revision streams	Snapshots only	Y: as series	Edit events	Cached renderings	As-shared generations	Pruning noted
History-off Chat	n/a: never kept	n/a	Setting- change events	Y: the rescue layer	External participants: Y	Tenant: no; ensemble: often
Litigation conduct	Is the subject	n/a	Y: operations log	n/a	n/a	The paperwork layer
Pre-hold deletions	Outside reach	Trash within window	Deletion events	Mirror divergence	Recipient copies, backup sets	The architecture's job

Fallback strategy in one line: what Vault cannot give, ask the API for; what the tenant never kept, ask the devices and counterparts for; and whatever the answer, ask the operations log what happened to it.

Worked examples

EXAMPLE 1 · THE SNAPSHOT EXPORT AND THE VANISHED AUTHORSHIP

An agency dispute turned on who inserted an indemnity clause into a shared Doc. The producing party's Vault export rendered the document as a PDF and a current-state file: authorship invisible, and their expert's statement that "the available metadata does not identify the editor" was, for that export, true. The requesting party's expert asked the route question, then the structural one: the API collection of the same document produced the revision series: the clause entered in three revisions by a named account across twenty minutes, eleven days after the version the counterparty had approved. Nothing had been hidden; the first route simply could not carry the answer. The costs of the second collection were argued about; the clause was not.

EXAMPLE 2 · THE FINANCE DIRECTOR'S OWN EXPORT

In a misfeasance claim, the respondent's disclosure of the finance director's mailbox had been exported from Vault by the finance director himself, wearing his admin hat: technically clean, procedurally indefensible. The applicant did not allege tampering; it asked for the trust pattern instead: laboratory re-run of the same Vault queries under supervision, count reconciliation, countersigned manifest. The re-run found the counts short: a date-range error in the original export, human and innocent, had dropped five months: including the quarter the claim centred on. The missing months contained the case's best documents. No conduct finding was ever needed; the verification pattern had done its work, and the judgment's disclosure narrative commends it dryly.

EXAMPLE 3 · THE OPERATIONS LOG THAT DATED THE STRATEGY

A respondent resisted specific disclosure of a Shared Drive, asserting its contents had "long since been deleted under routine retention". The applicant's request list included the Vault operations log and retention rules with change history. The log showed a custom retention rule: 30 days, scoped precisely to that Shared Drive: created eight days after the letter of claim, by an account belonging to the respondent's head of IT; the drive's prior coverage had been indefinite. The deletion was real; "routine" was not. The specific disclosure application succeeded on the log alone, the backup layer supplied the drive's pre-rule state, and the retention rule's creation date appeared in the costs judgment.

Common mistakes and technical limitations

Common mistakes

- **Vault as the whole strategy:** availability defaulted into every role, the routing table never drawn.
- **Snapshot exports in structure disputes:** Example 1's first production: authorship questions answered by a route that cannot carry authorship.
- **Interested operators:** Example 2's original export: custodians collecting themselves because the credentials were nearest.
- **Completeness claims past the reach map:** "everything" asserted over an estate with history-off spaces and pre-hold deletions.
- **The operations log forgotten:** in both directions: never requested from opponents; never checked before your own "routine retention" assertions go out.
- **Holds delayed for strategy:** preserve-now-decide-later inverted, and the clocks running meanwhile.
- **Metadata files discarded:** the export's self-description: the completeness answer: not retained with the production.
- **Fidelity silence:** productions that do not say what their route flattened, discovered at review.

Technical limitations

- **Reach is what it is:** never-kept content has no preservation answer: only the architecture walk.
- **Export formats constrain:** the snapshot behaviour is inherent to the route: managed by routing, not by wishing.
- **Search semantics have edges:** operator behaviour, attachment handling and service differences are tested (sampling, per guides 84-85 ahead) before counts become commitments.
- **Edition differences matter:** Vault availability and admin-report reach vary by licence: the entitlement facts precede every promise.
- **Vault is tenant-eyed:** the device and counterpart layers are invisible to it: the ensemble remains the case's true perimeter.

§ 1 1 · INTERROGATORIES & DRAFTING AIDS

Questions to ask · Suggested wording

Ask your client

- Are the holds in, with what scope and dates: and who, by name and role, will operate any Vault collection?
- Do the issues touch structures (revisions, threading, settings histories) that need the API route?
- Has anyone connected to the dispute run Vault searches or exports already: and what does our own operations log show?

Ask your opponent

- Please disclose the Vault hold report (scope, services, placement dates), the retention rules in force with change history since [date], and the Vault operations log for [period].
- Please identify the route(s) used for Workspace collection, produce the export metadata files, and state what the route preserved of revision histories, labels and threading.
- Where any custodian within the issues operated their own collection, please say so and confirm what verification was applied.

Ask your eDiscovery / forensic provider

- Draw the routing table: which sources Vault, which API, which independent: and why, on the record.
- For a Vault-collect / laboratory-verify pattern: what sampling and reconciliation will the countersigned manifest rest on?
- What does their production's own metadata say about its completeness: before we draft the challenge?

SUGGESTED WORDING · ROUTE-AND-VERIFICATION CLAUSE FOR THE DISCLOSURE PROTOCOL

"Workspace collection shall proceed as follows: (a) Google Vault litigation holds over the custodians and Shared Drives at Schedule A were placed on [date] and shall be maintained; (b) bulk collection shall be by Vault export with metadata files and account manifests retained and exchanged; (c) the sources at Schedule B (including the revision histories of the documents there listed, collected as series) shall be collected by API-based means preserving structure; (d) collection concerning any custodian within the issues shall be operated or supervised independently, with supervision recorded; and (e) each party shall on request disclose its Vault operations log, retention rules with change history, and a statement of what its chosen routes preserve and flatten. Counts shall be reconciled and manifests countersigned where verification is agreed."

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

The Vault checklist

- ☐ Holds in on day one; hold report exhibited with dates
- ☐ Routing table drawn per source: Vault / API / independent: and recorded
- ☐ Interested operators identified; independence or supervision arranged
- ☐ Structures the issues turn on routed to the API collection
- ☐ Export metadata files retained; counts reconciled; manifests countersigned where agreed
- ☐ Fidelity statement accompanies every production
- ☐ Reach map (history-off, pre-hold, out-of-scope services) documented before completeness claims
- ☐ Operations log checked on your side; requested from theirs
- ☐ Retention rules and change history read against the duty's chronology
- ☐ Device and counterpart layers scoped for what Vault cannot see

Red flags

- A custodian in the issues exporting their own estate.
- Retention rules created or shortened after the duty arose (Example 3's signature).
- Structure disputes answered with snapshot exports.
- Productions without metadata files or route statements.
- Completeness claims over never-kept content.
- An operations log the opponent will not produce.
- Holds "pending strategy" while trash clocks run.

When to involve a digital forensic expert

At routing, where the three questions are answered per source and documented; at verification, where the Vault-collect / laboratory-verify pattern turns cooperative collection into countersigned evidence (Example 2's rescue); at structural collection, where revision streams and settings histories are exported as the series Example 1 needed; and at the conduct seam, where operations logs and retention histories meet the correspondence file (Example 3's application). Reported under CPR Part 35 with reach and fidelity stated: and through **e-discovery.uk**, both routes' outputs processed into one coherent review.

§ 13 · COMMON QUESTIONS

Frequently asked questions

Is a Vault export defensible disclosure on its own?

For the uncontested bulk, with metadata files retained, counts reconciled and a fidelity statement attached: yes, routinely. What makes it indefensible is misassignment: contested custodians self-operating, structure disputes answered with snapshots, completeness asserted past the reach map. Defensibility is a property of the routing decision and its paperwork, not of the tool: which is this guide's single sentence.

When do we genuinely need collection beyond Vault?

Three triggers, from §5: trust (a perimeter custodian would otherwise operate), structures (the case turns on revisions, threading, settings or sharing histories), downstream (the production must prove itself to a hostile expert). Any one suffices for the affected sources; none moves the uncontested bulk off Vault. The routing table, not an all-or-nothing instinct, is the deliverable.

Can Vault exports be manipulated by the operator?

The realistic risks are scope and query manipulation (or error: Example 2's five months) rather than content forgery: exports carry self-describing metadata, and Vault's operations are audit-logged. Which is precisely why the safeguards are procedural: documented queries, retained metadata files, count reconciliation, supervision or verification for interested operators: and why an opponent's refusal to produce the operations log is itself information.

The opponent says the content expired under retention before any hold. What is the sequence?

Paper first: retention rules with change history against the duty's date (Example 3's seam), the hold report's placement dates, the operations log. Then reach: trash windows, any pre-existing holds from other matters (guide 58's Example 3), backup platforms. Then the ensemble: devices, counterparts, circulated copies. "Expired" is a checkable claim with a paper trail, and this series' standing rule applies: test it, never inherit it.

How does Vault compare with Purview for a firm running both ecosystems?

The strategy is identical; the boundaries differ in detail: what each reaches, what each export flattens, where each logs its operations. The transferable core is the framework: preserve through the native tool immediately, route by trust/structures/downstream, verify where operators are interested, and demand the operations paperwork in disputes. Guides 51-52 and 58-59 are deliberately parallel so teams can carry one method across both estates.

What single Vault artefact should we secure first in any new matter?

The hold report: scope, services, placement dates: created by placing the holds themselves, today. It is simultaneously the preservation act, the recoverability watershed for every later question, and the first exhibit of a defensible process. Everything else in this guide can be decided next week; the holds cannot.

§ 1 4 · REFERENCE

Glossary

Countersigned manifest

A collection manifest verified and signed by an independent examiner.

Fidelity statement

The production's declaration of what its route preserved and flattened.

Hold report

Vault's record of hold scope, services and placement dates.

Matter (Vault)

The workspace organising holds, searches and exports per dispute.

Operations log

The audited record of Vault holds, releases, searches and exports.

Reach map

The documented boundary of what Vault covers in this estate.

Routing table

The per-source assignment: Vault, API or independent collection.

Snapshot export

A current-state rendering of a living document; the fidelity boundary.

Supervised operation

Client-credential collection directed and witnessed by the laboratory.

Verification pattern

Re-run, reconcile, countersign: independence at verification cost.

Sources and authoritative references

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

- cflab.uk, digital forensics services (Vault and API collection, independent verification, operations-log analysis, CPR Part 35 reporting): cflab.uk/digital-forensics-services · guides library: cflab.uk/guides
- e-discovery.uk, EDRM Phase 03 · Collection and Phase 04 · Processing (route selection, manifests and reconciliation, structured outputs into review): e-discovery.uk/edrm/collection · e-discovery.uk/edrm/processing
- e-discovery.uk, practice method and Knowledge Centre: e-discovery.uk · e-discovery.uk/knowledge
- Google documentation: Vault help centre: support.google.com/vault · Vault audit / operations: support.google.com, audit and investigation
- PD 57AD and CPR Part 35: justice.gov.uk, PD 57AD · justice.gov.uk, Part 35
- ICO, UK GDPR guidance: ico.org.uk

DISCLAIMER

This publication provides general information about Google Vault as at August 2026. Product behaviour, reach and editions change on Google's schedule and vary by tenant and licence; verify current behaviour and entitlements in the matter at hand. The worked examples are fictional. This guide is not legal advice, does not create a professional relationship, and should not be relied upon in place of advice on the facts of a specific matter from a qualified lawyer or a suitably qualified forensic practitioner. Links were live at the date of publication.

§ HOW A SPECIALIST LABORATORY CAN ASSIST

Working with Computer Forensics Lab

The laboratory's Workspace practice is built around the routing table: Vault holds placed with clients in day one; bulk export supported and verified (the re-run, reconcile, countersign pattern of Example 2 is standing service); API collection of revision streams, settings histories and structures wherever the issues demand the series (Example 1's rescue); and the conduct analyses: operations logs, retention histories, hold chronologies: that turn "expired" into either explanation or application material (Example 3). Cross-ecosystem matters run the same method over Purview and Vault in parallel. Reporting is CPR Part 35 with reach and fidelity stated; processing through **e-discovery.uk** merges both routes' outputs into one review. Conflicts checks, confidential scoping discussions and fixed-fee estimates are routine; and if the holds are not yet in, that is the first ten minutes of the engagement.



I N S T R U C T T H E L A B

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