COMPUTER FORENSICS LAB

London digital forensics & e-discovery · Established 2007

Gatekeeper, XProtect and macOS Malware Artefacts

What Ran, What Was Blocked, and Whether "a Virus Did It" Holds Up

Macs are not immune to malware, and the security systems Apple builds in, Gatekeeper, which checks whether software is allowed to run, and XProtect, which detects and blocks known malware, leave artefacts that a forensic examiner can read. Those artefacts matter in two very different situations. Where a Mac is genuinely compromised, they help establish what ran, what was blocked, and what a compromise did, relevant to breach, liability and causation. And where a party raises the "a virus did it" defence, claiming that incriminating material or activity was the work of malware rather than the user, the same artefacts, together with the wider record, allow that claim to be tested rather than accepted. The discipline is even-handed: sometimes malware genuinely is responsible, and the evidence should show it; sometimes the defence is an excuse, and the evidence should test it honestly. This guide sets out for UK lawyers how Gatekeeper and XProtect work, what malware artefacts show, how a compromise is investigated, and how the "a virus did it" claim is examined at honest confidence.

⏱ Estimated reading time: 25 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 Mac security work reads the artefacts left by Gatekeeper and XProtect, and the wider system record, to establish what ran on a Mac, what was blocked, and whether a compromise occurred, and to test the "a virus did it" claim even-handedly. The analysis distinguishes genuine compromise from unfounded excuse, states causation and attribution at honest confidence, and is reported under CPR Part 35 and CrimPR Part 19. The laboratory does not create, deploy or assist with malware.

ESTABLISHED 2007 · LONDON | ISO 17025-ALIGNED PROCEDURES | GATEKEEPER & XPROTECT ARTEFACTS | COMPROMISE & "VIRUS DEFENCE" ANALYSIS | CPR PART 35 EXPERT REPORTS | FULL CHAIN-OF-CUSTODY DOCUMENTATION

§ CONTENTS

In this guide

§ 0 1 · ORIENTATION

Executive summary

The headline point: Macs get malware, and Apple's Gatekeeper and XProtect leave artefacts showing what ran, was blocked or was quarantined, so those artefacts, with the wider record, both help investigate a genuine compromise and allow the "a virus did it" defence to be tested even-handedly, at honest confidence.

How they work (§3): Gatekeeper checks whether software is allowed to run (signing, notarisation, quarantine), and XProtect detects and blocks known malware, both leaving records. **What the artefacts show (§4):** what applications ran or were allowed, what XProtect detected or blocked, what was quarantined, and when, a record of the machine's security events. **Genuine compromise (§5):** where a Mac is truly compromised, the artefacts and system record help establish what ran, how it got in, and what it did, for breach and causation. **The virus defence (§6):** where a party claims malware, not them, was responsible, the same artefacts and the wider record test whether malware capable of the alleged act was actually present and active, or whether the claim is unfounded. **Deployment (§7):** breach and liability, causation, and the honest, even-handed assessment of the malware defence.

- **Macs get malware:** they are not immune, and Apple's defences leave readable artefacts.
- **Artefacts show security events:** what ran, what was blocked, what was quarantined, and when.
- **Investigate real compromise:** establish what got in, what it did, and the causation.
- **Test the virus defence:** check whether malware capable of the act was actually present and active.
- **Be even-handed:** sometimes malware did it, sometimes the claim is an excuse; the evidence decides.

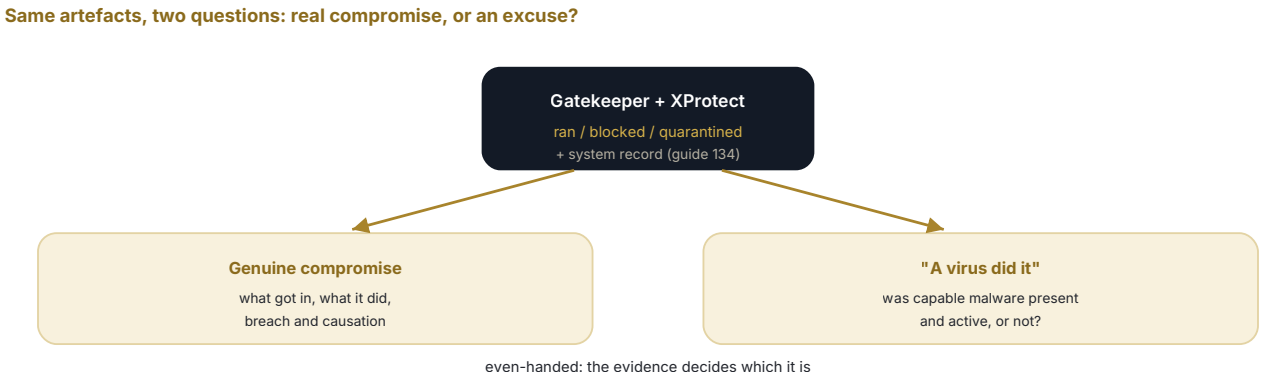

Figure 1 - the same Gatekeeper, XProtect and system artefacts answer two questions even-handedly: whether a genuine compromise occurred, and whether a "a virus did it" defence holds up.

§ 0 2 · FIRST PRINCIPLES

The problem in plain English: Macs get malware, and "a virus did it" gets raised

A persistent myth holds that Macs do not get viruses. They do. Malware for macOS is real and increasingly common, and Apple accordingly builds several defences into the operating system. Gatekeeper checks software before it runs, verifying that an application is signed by a known developer, has been through Apple's notarisation, and, for downloaded software, handling the quarantine flag that triggers checks on first launch. XProtect is Apple's built-in malware detection, which recognises known malicious software by signature and can block or remove it. These systems operate quietly in the background, and, importantly for evidence, they leave artefacts: records of what was checked, what was allowed to run, what was blocked, and what was quarantined or removed.

Those artefacts become relevant in two very different kinds of case, and it is essential to keep both in view. In the first, a Mac has genuinely been compromised, by malware, a targeted intrusion, ransomware, and the question is what happened: what got in, how, what it did, and who is liable. Here the security artefacts and the wider system record help reconstruct the compromise. In the second, a party facing incriminating evidence, illicit material, unauthorised activity, fraudulent transactions, raises the "a virus did it" defence: the claim that malware, not the user, was responsible. This is a legitimate defence that is sometimes true and sometimes a convenient excuse, and the same artefacts, combined with the wider record, allow it to be tested rather than simply accepted or dismissed. The forensic discipline is even-handedness. The examiner does not assume the defence is false, nor take it at face value; they look at whether malware capable of the alleged act was actually present and active on the machine at the relevant time, and let the evidence decide. This guide sets out how Gatekeeper and XProtect work, what the artefacts show, how a genuine compromise is investigated, and how the malware defence is tested honestly. The laboratory investigates malware; it does not create or deploy it.

§ 0 3 · HOW THEY WORK

How Gatekeeper and XProtect work

- **Gatekeeper:** the macOS system that checks, before software runs, whether it is signed by an identified developer and notarised by Apple, and that enforces the quarantine of downloaded software, allowing, warning or blocking accordingly (§4): the gate on what may run.
- **Quarantine:** the flag macOS applies to downloaded files (guide 138's origin data) that triggers Gatekeeper checks and user prompts on first launch, so the handling of downloaded software is recorded (§4): the first-run checkpoint.
- **XProtect:** Apple's built-in malware detection, which recognises known malware by signature and can block or remove it, updated by Apple over time, leaving records of detections and actions (§4): the built-in scanner.
- **Notarisation and signing:** Apple's developer signing and notarisation, so legitimate software is identifiable and unsigned or unnotarised software is flagged, relevant to whether a given program was trusted or suspect (§5): the trust markers.
- **Artefacts read on the decrypted device:** the Gatekeeper, quarantine and XProtect records, with the unified logs and system artefacts (guide 134), recovered from the acquired, decrypted Mac (guides 131, 133), with integrity preserved (guides 2, 3): the security record recovered.

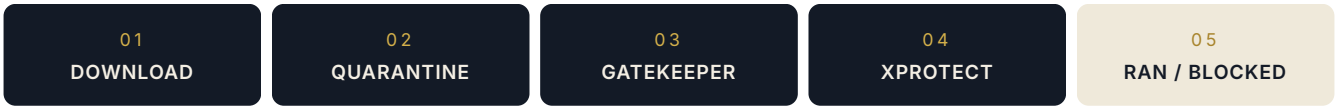

Figure 2 - the macOS security path from download and quarantine through Gatekeeper and XProtect to whether software ran or was blocked, each step leaving artefacts.

§ 0 4 · WHAT THE ARTEFACTS SHOW

What the artefacts show: ran, blocked, quarantined

- **What ran or was allowed:** records of applications that passed Gatekeeper and ran, with timing, so what executed on the machine, and when, can be established, corroborating the activity artefacts (guide 134): the executed-software record.
- **What was blocked or warned:** records of software Gatekeeper or XProtect blocked or warned about, so attempts to run suspect or malicious software are evidenced, whether or not they succeeded (§5): the blocked-attempt record.
- **What was quarantined or removed:** quarantine entries and XProtect detections and removals evidencing that particular files were treated as suspect or malicious, and when (§6): the flagged-as-malware record.
- **Origins of downloaded software:** the quarantine and origin data (guide 138) showing where a program came from, relevant both to a compromise and to whether the user deliberately obtained something (guide 129): the where-it-came-from record.
- **Timing on the security timeline:** the security events placed on a timeline with the activity and communication artefacts (guides 130, 134), so what ran, was blocked or was quarantined sits in the sequence of the machine's use (guide 96): the security events in context.

§ 0 5 · INVESTIGATING A COMPROMISE

Investigating a genuine compromise

- **Establish what got in:** the malware or intrusion identified from the artefacts, the running processes, the persistence mechanisms and the system record, so the nature of the compromise is characterised (guide 99's volatile-data context): the intrusion identified.
- **Establish how:** the entry route reconstructed, a downloaded and run program, a malicious document, an exploited service, from quarantine, Gatekeeper and log artefacts, so the vector is understood (§4): the how-it-got-in answer.
- **Establish what it did:** the compromise's actions, data accessed or exfiltrated, changes made, connections opened, reconstructed from the system record and network artefacts (guides 110, 134), so impact and causation are assessed: the what-it-did answer.
- **Preserve volatile evidence:** the live, running state, memory and active processes captured where the machine is still running (guide 99), since much malware evidence is volatile and lost at shutdown, so a live-compromised Mac is handled with care: the volatile record secured.
- **Scope, breach and liability:** the compromise scoped for breach-notification, liability and remediation, with honest limits on what the artefacts establish, so the legal consequences rest on sound findings (guide 20): the compromise turned into assessment.

§ 0 6 · TESTING THE VIRUS DEFENCE

Testing the "a virus did it" claim

- **Approach it even-handedly:** the claim neither assumed false nor accepted at face value, but tested against the evidence, since malware genuinely is sometimes responsible and sometimes not (guide 100): the neutral starting point.
- **Was capable malware present?:** the central question, whether malware capable of the alleged act, creating the material, performing the activity, was actually present on the machine at the relevant time, from XProtect, artefacts and a malware examination (§4): the presence question.
- **Was it active then?:** if malware was present, whether it was running and active at the time of the act, not merely dormant or long-removed, so mere presence is not mistaken for causation (guide 108): the activity-timing question.
- **Does its behaviour fit?:** whether the malware's actual capabilities and behaviour match the alleged act, since generic malware often cannot do what the defence claims, so capability is matched to the act (§5): the capability-fit question.
- **Weigh against user activity:** the malware hypothesis weighed against the evidence of deliberate user activity (guides 125, 134), so the account, malware or user, that the whole record best supports is identified at honest confidence: the even-handed conclusion.

§ 0 7 · DEPLOYMENT

Deployment: breach, causation and the malware defence

- **Breach and liability:** a genuine compromise investigated for breach-notification obligations, liability and remediation, with what the malware did, and what data it reached, established for the legal response (guides 20, 110): the compromise assessed.
- **Causation:** the security artefacts and system record establishing whether a compromise, or a user's act, caused the outcome in issue, so causation rests on evidence rather than assertion (guide 100): the cause identified.
- **Rebutting a false malware defence:** the evidence showing that no malware capable of the alleged act was present or active, so a "a virus did it" excuse is honestly rebutted where the record does not support it (Example 2, §6): the excuse tested and answered.
- **Supporting a genuine malware defence:** where the evidence does show a capable, active compromise, that finding fairly supporting the defence, since even-handedness cuts both ways (Example 3, §6): the true defence supported.
- **Honest, corroborated presentation:** the findings, compromise or its absence, causation, the malware defence, corroborated across artefacts and the estate and presented at honest confidence under CPR Part 35 or CrimPR Part 19 (guides 100, 130): the security evidence deployed defensibly.

§ 0 8 · THE WIDER MAP

Source architecture: where else the evidence lives

Per this series' source-architecture discipline: the Gatekeeper and XProtect artefacts on a Mac are one part of a security record that spans the machine and its wider environment, and both a compromise investigation and a malware defence rest on corroborating across that whole record. The Mac holds the Gatekeeper, quarantine and XProtect artefacts, the running-process and memory state, and the system logs (guide 134). But network sources, firewall, proxy and DNS logs on a corporate network, hold an independent record of the connections a compromise made; endpoint-security and antivirus tools hold their own detections; the system artefacts and Spotlight metadata (guides 134, 138) show what ran and where software came from; backups and snapshots (guides 132, 137) hold earlier states from before or during a compromise; and a managed device's administrative console holds central security logs. The discipline is to corroborate the Mac's security artefacts across these, so that a compromise is characterised from more than the machine alone, and so that a malware defence is tested against the network and endpoint record as well as the device. When the on-machine artefacts are thin, network and endpoint logs evidence what a compromise did; when the live state is lost, a backup or snapshot preserves the earlier position; and the estate together shows whether capable malware was truly present and active, or not.

EVIDENCE	MAC SECURITY ARTEFACTS	SYSTEM LOGS / MEMORY	NETWORK / FIREWALL LOGS	ENDPOINT / AV TOOLS	BACKUPS / SNAPSHOTS	MANAGED CONSOLE
What ran	Y: Gatekeeper	Y: process record	n/a	Execution logs	Earlier state	Admin logs
What was blocked	Y: XProtect	Log entries	Blocked connections	Y: detections	Earlier detections	Central alerts
Malware present	Y: detections	Running processes	C2 traffic	Y: identified	Y: at backup time	Managed alerts
What it did	Limited	Y: activity	Y: exfil / connections	Behaviour logs	Before/after state	Network logs
Origin of software	Quarantine data	Download traces	Download source	n/a	Earlier metadata	n/a

Fallback strategy in one line: corroborate the Mac's security artefacts across system, network, endpoint and backup sources; whether investigating a compromise or testing a malware defence, the estate shows whether capable malware was truly present and active.

§ 0 9 · IN THE WILD

Worked examples

EXAMPLE 1 · THE GENUINE COMPROMISE RECONSTRUCTED

A firm's Mac was compromised, and the question was what had happened and what data was reached. The §5 investigation reconstructed it: the §4 Gatekeeper and quarantine artefacts showed a malicious application, downloaded from a deceptive source, that had been run despite a warning; the §4 XProtect and system record showed what it did; and the §8 network logs evidenced the connections it made and the data it exfiltrated (guide 110). The §5 volatile capture of the running state, taken before shutdown (guide 99), preserved active processes central to the analysis. The compromise was characterised, its entry, actions and impact, for the §7 breach and liability assessment (guide 20). The lesson is §5's: Macs do get compromised, and the Gatekeeper, XProtect and system artefacts, with the network record and a live capture, allow a genuine compromise to be reconstructed, what got in, how, and what it did, so breach, causation and liability rest on evidence rather than speculation.

EXAMPLE 2 · THE "A VIRUS DID IT" CLAIM THAT DID NOT HOLD

A party facing evidence of deliberate activity on his Mac claimed a virus had been responsible. The §6 even-handed test did not assume the claim false but examined it: the §4 XProtect and security artefacts recorded no malware detection at the relevant time; a malware examination found no malicious software present; and no process capable of the alleged activity existed, while the §6 weighing against user activity showed deliberate actions, searches, logins, file handling (guides 125, 134), consistent with the user, not malware. The §6 capability-fit question was decisive: even had generic malware been present, it could not have done what was alleged. The claim was honestly rebutted on the evidence. The lesson is §6's: the malware defence is tested, not assumed, and where no capable malware was present or active and the record shows deliberate user activity, the "a virus did it" claim is fairly rebutted, but only after a genuine, even-handed examination.

EXAMPLE 3 · THE MALWARE DEFENCE THE EVIDENCE SUPPORTED

In another matter, a party raised the same defence, and even-handedness required following the evidence where it led. The §6 examination found that it led in the party's favour: a §4 XProtect record and a malware examination showed that capable malware had indeed been present and active on the Mac at the relevant time, downloaded and run through a §4 quarantine-flagged file, and its §6 behaviour matched the activity in issue, which the §8 network logs corroborated. Weighed against the user-activity evidence, the malware hypothesis was the better-supported account. The finding fairly supported the defence. The lesson is §6's: even-handedness cuts both ways, sometimes malware genuinely is responsible, and an honest examination that would rebut a false defence must equally support a true one, so the credible expert follows the artefacts to whichever conclusion they support, stated at honest confidence, and does not start from the assumption that the defence is an excuse.

§ 10 · WHERE IT GOES WRONG

Common mistakes and technical limitations

Common mistakes

- **Assuming Macs cannot be infected:** the myth that leads to malware being overlooked entirely (§2).
- **Taking the virus defence at face value:** accepting "a virus did it" without testing whether capable malware was present and active (Example 2, §6).
- **Assuming the defence is always an excuse:** the opposite error, dismissing a genuine compromise without examination (Example 3, §6).
- **Confusing presence with causation:** treating the mere presence of malware as proof it caused the act, without checking activity and capability (§6).
- **Shutting down a live-compromised Mac:** losing volatile evidence by powering off before capture (Example 1, §5, guide 99).
- **Ignoring capability fit:** not asking whether the malware could actually do what is alleged (§6).
- **Not corroborating with the network:** relying on the machine alone when firewall and DNS logs would confirm or refute (§8).
- **Overstating conclusions:** asserting compromise or its absence beyond what the artefacts support (§6).

Technical limitations

- **Artefacts are not exhaustive:** Gatekeeper and XProtect record known threats and checks, so novel or removed malware may leave limited traces: an honest limit (§4).
- **Presence is not causation:** malware present is not malware responsible; activity and capability must be shown (§6).
- **Volatile evidence is fragile:** much compromise evidence is lost at shutdown (§5, guide 99).
- **Attribution still applies:** user activity is an account's, and malware findings must be weighed against it with care (§6, guide 105).
- **macOS and threats evolve:** the defences and the malware change constantly, so analysis is current-aware (§3).

§ 11 · INTERROGATORIES & DRAFTING AIDS

Questions to ask · Suggested wording

Ask your client

- Is this a suspected compromise, or a case where a malware defence has been or may be raised, so the analysis is framed correctly?
- Is the Mac still running, so volatile compromise evidence can be captured before shutdown?
- Are network, endpoint and backup records available to corroborate the machine?

Ask your opponent

- Please preserve the device without shutting it down where a compromise is live, and disclose the Gatekeeper, XProtect and system security artefacts and any endpoint or antivirus records.
- If a malware cause is asserted, please identify the specific malware said to be responsible and disclose the evidence of its presence and activity at the relevant time.
- Please preserve and disclose the relevant network and firewall logs.

Ask your eDiscovery / forensic provider

- Do the artefacts show a genuine compromise, and if so what got in, how, and what it did?
- If a virus is blamed, was capable malware actually present and active at the relevant time?
- Does the whole record best support malware or deliberate user activity, and at what confidence?

SUGGESTED WORDING · INSTRUCTION FOR A MAC SECURITY AND MALWARE ANALYSIS

> "We instruct you to examine the security and malware position of the device at Schedule 1 in relation to Schedule 2, even-handedly. Please, preserving integrity and, where a compromise may be live, capturing the volatile state before shutdown: (1) recover the Gatekeeper, quarantine and XProtect artefacts and the system security record, establishing what ran, what was blocked, what was quarantined or detected, and when; (2) where a compromise is alleged, identify what malware or intrusion was present, how it entered, and what it did, corroborating with network, endpoint and backup records; (3) where a malware cause is asserted for the activity in issue, test it neutrally, determining whether malware capable of the alleged act was actually present and active at the relevant time, whether its behaviour fits the act, and how the malware hypothesis weighs against the evidence of deliberate user activity; (4) do not assume the defence either true or false, but follow the evidence to whichever conclusion it supports; and (5) state findings on compromise, causation and the malware defence at honest confidence, distinguishing presence from causation. We understand you do not create or deploy malware."

§ 12 · QUICK CONTROL

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

The Mac security and malware checklist

- ☐ Live-compromised Mac preserved; volatile state captured before shutdown
- ☐ Gatekeeper, quarantine and XProtect artefacts recovered
- ☐ What ran, was blocked and was quarantined established, with timing
- ☐ Genuine compromise characterised: entry, actions, impact
- ☐ Malware defence tested neutrally, not assumed either way
- ☐ Presence, activity-timing and capability-fit all examined
- ☐ Malware hypothesis weighed against deliberate user activity
- ☐ Findings corroborated with network, endpoint and backup records
- ☐ Presence distinguished from causation
- ☐ Conclusions stated at honest confidence

Red flags

- Malware overlooked on the assumption Macs cannot be infected: §2.
- A virus defence accepted without testing: Example 2.
- A genuine compromise dismissed without examination: Example 3.
- Presence of malware treated as proof of causation: §6.
- A live-compromised Mac shut down before capture: §5.
- Capability fit not assessed.
- The network record not used to corroborate.

When to involve a digital forensic expert

Whenever a compromise is suspected or a malware defence is in play, and urgently where a compromise may be live, because volatile evidence is lost at shutdown (Example 1, §5): the expert recovers the Gatekeeper, XProtect and system security artefacts to establish what ran, was blocked and was quarantined, and, for a genuine compromise, reconstructs what got in, how, and what it did, corroborated across the network and endpoint record (§5, §8). Where a "a virus did it" defence is raised, the expert tests it even-handedly, neither assuming it false nor accepting it, by asking whether malware capable of the alleged act was actually present and active at the relevant time, whether its behaviour fits, and how the malware hypothesis weighs against deliberate user activity, following the evidence to whichever conclusion it supports (Examples 2 and 3, §6). Presence is distinguished from causation, and findings are reported at honest confidence under CPR Part 35 or CrimPR Part 19. The laboratory investigates malware and does not create or deploy it. Through **cflab.uk** and **e-discovery.uk**, Mac security work does justice to both truths: Macs really do get malware, and "a virus did it" is sometimes real and sometimes an excuse, and only an even-handed, evidence-led examination can tell which.

§ 13 · COMMON QUESTIONS

Frequently asked questions

Do Macs really get malware?

Yes (§2). The idea that Macs are immune is a myth; macOS malware is real and increasingly common, which is why Apple builds in defences like Gatekeeper and XProtect. Those defences, and the wider system record, leave artefacts that let a genuine compromise be investigated and a malware defence be tested. Treating a Mac as incapable of infection leads to malware being overlooked, so a competent analysis takes the possibility seriously in both directions, real compromise and false excuse alike.

What do Gatekeeper and XProtect actually record?

Their security decisions (§3, §4). Gatekeeper checks whether software is signed, notarised and quarantined before it runs, and its handling, allowed, warned, blocked, leaves traces. XProtect detects known malware and can block or remove it, recording detections and actions. Together with quarantine data and the unified logs (guide 134), these artefacts show what ran, what was blocked, what was flagged as malware, and when, a security record that supports both compromise investigation and testing of the malware defence.

Someone says "a virus did it". How do you test that?

Even-handedly, by asking specific questions (§6, Examples 2 and 3). Rather than assuming the claim true or false, the examiner asks: was malware capable of the alleged act actually present on the machine at the relevant time; was it active then, not merely dormant or long-removed; does its real behaviour fit what is alleged; and how does the malware hypothesis weigh against the evidence of deliberate user activity? The evidence, on the machine and across the network and endpoint record, then decides which account it best supports.

If malware is found, does that prove it was responsible?

No, presence is not causation (§6). Finding malware on a machine does not by itself prove that the malware, rather than the user, caused the act in issue. The malware must have been active at the relevant time and capable of doing what is alleged, and generic malware often cannot do what a defence claims. So the analysis distinguishes the mere presence of malware from its actually causing the outcome, and weighs it against the evidence of user activity, stating the conclusion at honest confidence rather than leaping from presence to blame.

Isn't the expert just there to defeat the virus defence?

No, even-handedness cuts both ways (§6, Example 3). An honest examination that can rebut a false malware defence must equally be able to support a true one, and sometimes malware genuinely is responsible. The credible expert does not start from the assumption that the defence is an excuse; they follow the artefacts to whichever conclusion the evidence supports, and will say plainly when a capable, active compromise did occur. That neutrality is exactly what makes the analysis reliable to a court, in either direction.

Why the urgency about not shutting the Mac down?

Because compromise evidence is often volatile (§5, guide 99). Much of what matters in a live compromise, running malicious processes, in-memory artefacts, active network connections, exists only while the machine is running and is lost at shutdown. So where a compromise may be live, the machine is kept running and its volatile state captured before it is powered off. As with Mac acquisition generally (guide 131), the early handling decision, here, capturing the live state, can determine how much of the compromise can be reconstructed.

§ 14 · REFERENCE

Glossary

Gatekeeper
The macOS system checking whether software may run.

XProtect
Apple's built-in malware detection.

Quarantine
A flag on downloaded files triggering checks on launch.

Notarisation
Apple's check of software for known malicious content.

Code signing
A developer's identifiable signature on software.

Malware
Malicious software; a virus, trojan, ransomware and more.

Compromise
A successful malware infection or intrusion.

The malware defence
The claim that malware, not the user, is responsible.

Capability fit
Whether malware could do what is alleged.

Volatile evidence
Live data lost at shutdown (see guide 99).

Sources and authoritative references

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

- cflab.uk, digital forensics services (Gatekeeper and XProtect artefacts, compromise investigation, even-handed testing of the malware defence, CPR Part 35 and CrimPR Part 19 reporting): cflab.uk/digital-forensics-services · guides library: cflab.uk/guides
- e-discovery.uk, EDRM Phase 03 · Preservation and Phase 06 · Analysis (live capture and security analysis as practised): e-discovery.uk/edrm/preservation · e-discovery.uk/edrm/analysis
- e-discovery.uk, practice method and Knowledge Centre: e-discovery.uk · e-discovery.uk/knowledge
- Apple Platform Security guide (Gatekeeper, notarisation, XProtect and malware protection on macOS): support.apple.com, Platform Security
- NCSC guidance on malware and incident response (UK): ncsc.gov.uk · Forensic Science Regulator, Code of Practice v2 (analysis and honest reporting): gov.uk, FSR Code
- ISO/IEC 27037 · PD 57AD · CPR Part 35: iso.org, 27037 · justice.gov.uk, PD 57AD · justice.gov.uk, Part 35

§ HOW A SPECIALIST LABORATORY CAN ASSIST

Working with Computer Forensics Lab

Mac security work at the laboratory does justice to two truths that are easy to get wrong: Macs really do get malware, and "a virus did it" is sometimes real and sometimes an excuse. Where a compromise is suspected, and especially where it may be live, the machine is kept running and its volatile state captured before shutdown (guide 99), and the Gatekeeper, quarantine and XProtect artefacts and the system record are recovered to establish what ran, was blocked and was quarantined, so a genuine compromise, its entry, its actions and its impact, is reconstructed and corroborated across the network and endpoint record for the breach, causation and liability assessment (Example 1). Where a malware defence is raised, it is tested even-handedly: neither assumed false nor accepted, but examined by asking whether malware capable of the alleged act was actually present and active at the relevant time, whether its behaviour fits, and how the malware hypothesis weighs against the evidence of deliberate user activity, with the examiner following the artefacts to whichever conclusion they support, rebutting a false defence and equally supporting a true one (Examples 2 and 3). Presence is always distinguished from causation, and findings are reported at honest confidence under CPR Part 35 or CrimPR Part 19. The laboratory investigates malware and does not create or deploy it. This guide completes the macOS artefact run of the Apple block. Conflicts checks, confidential scoping discussions and fixed-fee estimates are routine; and the guiding principle is neutrality: only an evidence-led, even-handed examination can say whether a compromise occurred, and whether a virus, or a person, did it.

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