COMPUTER FORENSICS LAB

London digital forensics & e-discovery · Established 2007

RAID, NAS and Server Recovery

Reconstructing Arrays, Network Storage and Servers Without Losing the Evidence

A single laptop is one disk. A business runs on something larger and stranger: a RAID array spreading data across several disks at once, a network-attached storage box in a cupboard holding the shared drives, a server with a stack of disks and a controller that knows how they fit together. When one of these holds the evidence, the ordinary image-one-disk approach fails, because no single disk in a RAID contains a readable copy of anything: the data is striped, interleaved and sometimes parity-protected across the set, and reassembling it requires knowing the array's geometry. Get that geometry wrong, or pull the disks in the wrong order, or let a well-meaning administrator rebuild the array onto a failed member, and the evidence is scrambled or destroyed. This guide sets out for UK lawyers how RAID arrays, NAS devices and servers store data, how they are recovered and reconstructed forensically, the specific ways they are damaged by mishandling, and how the shared, always-on nature of server storage shapes preservation, proportionality and disclosure.

⏱ 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. Array and server recovery is core laboratory work: forensic imaging of RAID members, reconstruction of striped and parity arrays from first principles, NAS and server acquisition, and the recovery of data from failed and partly-rebuilt arrays: conducted to preserve integrity and reported under CPR Part 35 and CrimPR Part 19, with the reconstruction method documented so it can be explained and, where needed, repeated.

ESTABLISHED 2007 · LONDON | ISO 17025-ALIGNED PROCEDURES | RAID & SERVER RECONSTRUCTION | NAS & VIRTUALISATION RECOVERY | CPR PART 35 EXPERT REPORTS | FULL CHAIN-OF-CUSTODY DOCUMENTATION

§ CONTENTS

In this guide

§ 0 1 · ORIENTATION

Executive summary

The headline point: on a RAID array, NAS or server, no single disk holds readable data: the evidence exists only when the array's geometry is correctly reconstructed, and the commonest way to lose it is a well-meant rebuild onto a failed set, so the array is preserved and reassembled by an examiner, never rebuilt in place.

How it stores data (§3): RAID spreads data across several disks by striping, with mirroring or parity for redundancy, so a lone member disk is unreadable; a NAS is a small RAID appliance with a file-sharing layer; a server adds a controller, volumes and often virtualisation. **Reconstruction (§4):** recovery means determining the array's geometry, the disk order, stripe size, parity rotation and start offset, imaging every member forensically, and reassembling the logical volume from the images, hashed and documented so the reconstruction can be explained and repeated. **Failed arrays (§5):** a degraded array missing a member can often be rebuilt from parity, but only from a correct, forensic image set, never by forcing the live hardware, and the order in which disks failed matters. **Preservation (§6):** live server storage is shared, always-on and large, so preservation uses snapshots, targeted collection of relevant shares and proportionate scoping rather than pulling a business offline. **Deployment (§7):** ordinary disclosure of server-held material, the spoliation case where an array was deliberately or negligently rebuilt over the evidence, and the honest limit where members are genuinely lost.

- **No single disk is readable:** RAID data is striped across the set; the evidence appears only on correct reassembly.
- **Geometry is everything:** disk order, stripe size, parity rotation and offset must be recovered before the data is.
- **Never rebuild in place:** image every member forensically first; a live rebuild onto a failed set can destroy the evidence.
- **Servers stay up:** shared, always-on storage is preserved by snapshot and targeted collection, not by pulling the business offline.
- **The estate is redundant:** backups, replicas and cloud copies often hold the same data, so a lost array is rarely a lost case.

RAID reconstruction: disks in, geometry solved, logical volume out

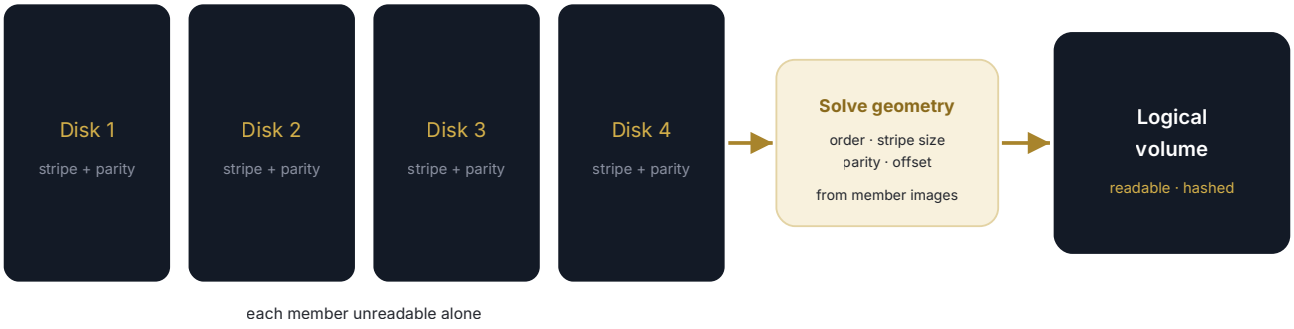

Figure 1 - a RAID yields nothing from any single member; the data appears only when the geometry is solved from forensic images of every disk and the logical volume is reassembled.

§ 0 2 · FIRST PRINCIPLES

The problem in plain English: the data is not on any one disk

Imaging a laptop is conceptually simple: there is one disk, it holds the files, you copy it bit for bit. Server storage breaks that intuition completely. A RAID array, the technology underneath most business storage, deliberately spreads each file across several disks at once, cutting it into stripes and writing successive pieces to different members, often adding parity information so that if one disk fails the missing pieces can be calculated back. The point of the design is speed and resilience. The consequence for evidence is that no single disk in the array holds a readable copy of anything: pull one member out and image it, and you get a meaningless fraction, every file sliced through the middle.

To recover the data you must reassemble the array, and to reassemble it you must know its geometry: which disk came first, how big each stripe is, how the parity rotates across the set, where the data starts on each disk. Sometimes the controller records this and it can be read; sometimes it must be worked out forensically from the content of the disks themselves. This is why server recovery is a specialist reconstruction rather than a simple copy, and why the ordinary handling that is safe for a laptop is dangerous here. The single most destructive act is the well-meant rebuild: an administrator, trying to help, tells the array to repair itself onto a replacement or a failed member, and the controller overwrites the very stripes the case needed. The disciplined answer is always the same: image every member forensically, untouched, and reassemble from the copies, never rebuild the original in place. This guide explains how that is done and how it goes wrong.

§ 0 3 · HOW IT STORES DATA

How RAID, NAS and servers store data

- **Striping (RAID 0 family):** data cut into stripes and written across members in turn for speed, with no redundancy: fast, but the loss of any one disk loses the whole array, and reconstruction needs the exact stripe size and disk order.
- **Mirroring (RAID 1):** each disk a full copy of another: the most recoverable arrangement, because a single mirror member is a readable whole, though the array's own metadata still matters for currency and consistency.
- **Parity arrays (RAID 5, 6 and nested levels):** data striped with one or two disks' worth of parity distributed across the set, so the array survives one or two failed members: the commonest business arrangement, and the one whose geometry, stripe size, parity rotation, order and offset, must be solved to reconstruct.
- **NAS appliances:** a network-attached box that is, internally, a small RAID array with a file-sharing and management layer on top, often running a Linux-based file system: recovered by imaging the member disks and reconstructing both the array and its file system, with the appliance's own configuration and logs as evidence.
- **Servers and virtualisation:** a server adds a hardware or software RAID controller, logical volumes, and frequently a virtualisation layer where whole machines exist as large disk-image files on the array: the recovery reaching through array, volume, file system and virtual-disk layers, each of which must be reconstructed in turn.

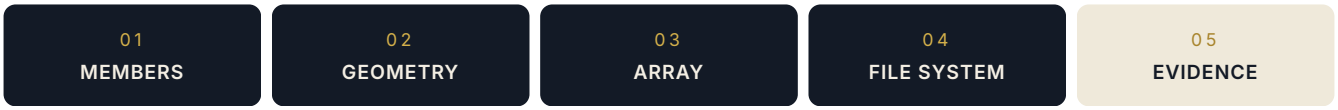

Figure 2 - the reconstruction ladder: forensic images of the members, solved geometry, reassembled array, mounted file system (and virtual disks), then the readable evidence.

§ 0 4 · RECONSTRUCTION

Reconstructing an array: geometry, parity and forensic imaging

- **Image every member first:** each disk imaged forensically with a write-blocker, hashed, before any reconstruction is attempted: the reassembly performed on the copies, so the originals are never altered and any error is recoverable (guides 2, 3).
- **Determine the geometry:** the disk order, stripe size, parity type and rotation, and data start offset established, from the controller's metadata where readable, or worked out forensically from the content and structure on the member images: the parameters without which the array cannot be reassembled correctly.
- **Reassemble the logical volume:** the member images combined per the solved geometry into a single logical image of the array, which is then mounted and examined like an ordinary volume: the striped fragments made whole in software, the original hardware untouched.
- **Reach through the layers:** the file system reconstructed on the logical volume, and any virtual-disk files within it mounted in turn to reach the virtual machines' own file systems: the layered recovery servers require (§3).
- **Verify and document:** the reconstruction validated (does the file system mount cleanly, do the files make sense), the logical image hashed, and the geometry and method recorded so the reconstruction can be explained in evidence and, if necessary, reproduced by another examiner: the reconstruction defensible, not a black box.

§ 0 5 · FAILED AND DEGRADED ARRAYS

Failed and degraded arrays: recovery from partial sets

- **The degraded array:** a parity array running with a failed member is degraded but functional, its missing data calculable from the remaining disks and parity: recoverable from a correct forensic image set, provided no further member is lost and no rebuild has overwritten the parity.
- **Order and timing of failure:** which member failed, and when, matters: a disk that dropped out early holds stale data, and reintroducing it (as a naive rebuild may) can corrupt the array with out-of-date stripes: the failure history established before reconstruction, from controller logs and disk state.
- **Multiple failures:** an array that has lost more members than its parity can cover (two in a single-parity array) cannot be fully reconstructed from the survivors alone: the honest limit stated, with partial recovery of what the survivors and parity still cover, and the estate reach-around (§8) for the rest.
- **The dangerous rebuild:** the single most destructive act, a rebuild of the live array onto a replacement or a failed member, which recalculates and overwrites stripes and can destroy recoverable data: forbidden until every member is forensically imaged, so recovery is attempted on copies where a failed rebuild costs nothing (guide 119 §5's one-attempt principle).
- **Physical failure of a member:** a member with a mechanical or electronic fault recovered by the damaged-media techniques of guide 119 (board repair, transplant, chip-off, cleanroom) before it can take its place in the reconstruction: the array recovery and the damaged-disk recovery combined.

§ 0 6 · PRESERVING LIVE SERVER STORAGE

Preserving live server storage: snapshots, shares and proportionality

- **The always-on problem:** a business server cannot usually be powered off and its disks pulled without halting the business, so preservation must often happen while the system runs: the acquisition designed around continuity, not against it.
- **Snapshots and live acquisition:** storage and virtualisation snapshots capturing a consistent point-in-time state of the volumes, and live forensic acquisition of the running system, preserving the relevant data without downing the server (guide 99's volatile-capture discipline, at storage scale).
- **Targeted collection of shares:** the relevant network shares, user home directories and mailboxes collected in a documented, reproducible way (guide 71), rather than imaging a multi-terabyte array wholesale: the collection scoped to relevance and proportionality (guide 20).
- **Proportionality and cost:** full forensic imaging of a large array is expensive and slow and is justified where the array itself, its deleted data, its structure, is in issue; targeted collection of the relevant shares suffices where it is the documents, not the array, that matter: the method sized to what the dispute needs.
- **Configuration and logs as evidence:** the server's and NAS's own configuration, access logs, share permissions and audit records preserved alongside the data (guides 105, 114): the who-accessed-what record that the storage itself keeps.

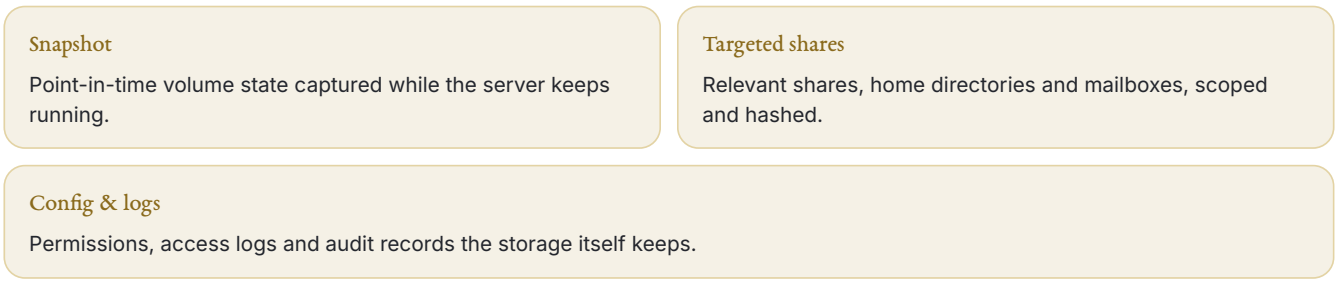

Figure 3 - preserving live server storage without pulling the business offline: snapshot the state, collect the relevant shares, and keep the configuration and logs.

§ 0 7 · DEPLOYMENT

Deployment: disclosure, spoliation and the mishandled rebuild

- **Ordinary disclosure:** the common case, relevant material held on a server or NAS collected and produced like any other source, with the array reconstructed or the shares targeted as the matter requires: most server instructions are simply this, done correctly.
- **Spoliation by rebuild:** where an array was rebuilt over the evidence, deliberately, or negligently by an administrator who "just tried to fix it", the loss assessed and, where the duty had arisen, the spoliation consequences (adverse inference, strike-out, costs) pursued (guides 95 §7, 110 §7): the rebuild dated against the duty and its avoidability examined.
- **The negligent-versus-deliberate line:** a genuine attempt to restore a failed business system distinguished from a convenient destruction of evidence, from the timing, the instructions given and whether preservation had been requested: the same duty-clock analysis as the wiped device (guide 110), applied to the array.
- **The honest limit:** where members are genuinely lost beyond parity's reach and no backup exists, the data is gone, and the examiner states it plainly while pursuing the estate for copies (§8): no false promise of reconstruction from an impossible set.
- **Proportionality in dispute:** the cost of full array reconstruction weighed against what targeted collection and the estate already yield, so heroic reconstruction is reserved for where the array uniquely holds the central evidence (guide 20).

§ 0 8 · THE WIDER MAP

Source architecture: where else the evidence lives

Per this series' source-architecture discipline: a server or array is one node, and its data is usually reflected across a redundant estate that no single reconstruction depends on. The array holds the primary shared data and its deleted residue. But the backups, nightly, offsite, and cloud, hold prior states of the same data; replicas and secondary sites mirror it; the individual endpoints hold synced and cached copies of the shares people worked on; the cloud platforms hold the mailboxes and SaaS records the server does not; and the counterparties hold their side of shared material. The discipline is to map these before committing to a difficult or expensive reconstruction: often a backup holds the relevant data cleanly and cheaply, and the array reconstruction is needed only for what exists nowhere else, deleted material, or the array's own structure and logs. When the array is degraded beyond recovery, the backup and the endpoints carry the case; when a rebuild destroyed the live data, the last good backup predates it; when the server is too large to image whole, the targeted shares and the estate together answer the question proportionately.

EVIDENCE	THE ARRAY / SERVER	BACKUPS & REPLICAS	ENDPOINTS (SYNCED/ CACHED)	CLOUD PLATFORMS	COUNTERPARTS	DELETED / RECOVERABLE
Shared files	Y: primary copy	Y: prior states	Synced / cached copies	Cloud drive copies	Shared copies	Varies (reconstruct + carve)
Mailboxes	Where server-hosted	Y: mail backups	Local caches (OST)	Y: cloud mailbox	Recipients	Varies
Databases / apps	Y: on the volumes	Y: DB backups	n/a	SaaS equivalents	n/a	Limited
Access / activity	Y: server & NAS logs	Backup logs	Endpoint artefacts	Cloud audit logs	n/a	Limited
Array structure	Y: geometry & config	n/a	n/a	n/a	n/a	Y: from member images

Fallback strategy in one line: image every member before any rebuild, reconstruct from the copies; and where the array is beyond recovery, the backups, replicas and endpoints hold the same data untouched by the failure.

§ 0 9 · IN THE WILD

Worked examples

EXAMPLE 1 · THE HELPFUL REBUILD THAT NEARLY LOST THE CASE

A parity array holding the shared drives central to a commercial dispute had a member fail. Before anyone thought about preservation, the client's IT contractor slotted in a replacement disk and told the controller to rebuild, standard practice for keeping a business running, catastrophic for the evidence, because the rebuild began recalculating and overwriting stripes. The laboratory was called mid-rebuild and stopped it at once. The §4 discipline salvaged the position: every member, including the partially-rebuilt replacement and the failed original, was imaged forensically before anything further was touched, and the reconstruction was attempted on the copies. Because the rebuild had been caught early, enough original stripes survived on the imaged members to reassemble the array with the solved geometry, and the relevant data was recovered. Had it run to completion, the evidence would have been gone. The lesson is §5's: never rebuild the original, image every member first, and recover on copies where a mistake costs nothing.

EXAMPLE 2 · THE NAS IN THE CUPBOARD AND THE BACKUP THAT SAVED IT

A departed director was alleged to have deleted key files from a small firm's NAS on his way out. Two of the NAS's four disks had since failed, and the array was beyond parity's reach, a genuine §5 multiple-failure limit, honestly stated. The reconstruction could not be completed from the survivors alone. But the §8 estate carried the case: the NAS had been backed up nightly to a cloud backup service, and the backup taken the evening before the deletions held the files intact, while the backup taken the following morning showed them gone, dating the deletion precisely and proving its content. The director's deletion was established not from the wrecked array but from the backup that bracketed it. The lesson is §8's: the array is rarely the only source, and a redundant estate turns an unrecoverable array from a lost case into a well-evidenced one.

EXAMPLE 3 · THE VIRTUALISED SERVER AND THE LAYERED RECONSTRUCTION

A fraud investigation needed the accounting system from a company's virtualised server, a parity array holding large virtual-disk files, each containing a whole virtual machine. The §4 layered recovery was required in full: the member disks were imaged, the array's geometry solved and the logical volume reassembled, the host file system mounted, the relevant virtual-disk file extracted, and that virtual disk mounted in turn to reach the guest machine's own file system and the accounting database within it. At each layer the reconstruction was validated and hashed. A deleted-data pass on the reassembled volume also recovered a prior virtual-machine snapshot the company had removed, which held an earlier state of the accounts. The reconstruction was documented layer by layer so it could be explained in evidence and reproduced. The example shows why server recovery is a reconstruction through several layers, not a single copy, and why each layer is preserved and documented in turn.

§ 10 · WHERE IT GOES WRONG

Common mistakes and technical limitations

Common mistakes

- **Rebuilding the live array:** the fatal error, an administrator "fixing" the array onto a replacement or failed member and overwriting the evidence before imaging (Example 1): the one act to prevent above all.
- **Imaging one disk and expecting files:** a single RAID member imaged and found to contain gibberish, the striping misunderstood: the array not reassembled (§2).
- **Pulling disks without recording order:** members removed without labelling their positions, the geometry needlessly harder to solve: the order recorded before anything is disturbed.
- **Reintroducing a stale member:** a disk that failed early slotted back in, corrupting the array with out-of-date stripes: the failure history unread (§5).
- **Imaging a multi-terabyte array wholesale:** disproportionate full imaging where targeted collection of the relevant shares would answer the question (§6).
- **Powering down a business server needlessly:** the array pulled offline and the business halted where a snapshot and live acquisition would have preserved the evidence (§6).
- **The estate ignored:** a heroic reconstruction pursued while a clean backup held the same data (Example 2, §8).
- **Undocumented reconstruction:** the array reassembled as a black box, the geometry and method unrecorded, so it cannot be explained or reproduced (§4).

Technical limitations

- **Beyond-parity loss is final:** an array that has lost more members than its parity covers cannot be fully reconstructed from the survivors: the honest limit, stated (§5).
- **Rebuild overwrite is irreversible:** stripes recalculated and overwritten by a completed rebuild are gone: caught early they may be salvageable, complete they are not (Example 1).
- **Geometry can be hard to solve:** unusual or proprietary controllers, non-standard stripe layouts and encryption complicate reconstruction: solvable often, but not always, and the effort is stated.
- **Encryption sits on top:** an encrypted array or volume still needs its key after reconstruction (guides 117, 118): the reconstruction and the decryption are separate problems.
- **Scale bounds proportionality:** multi-terabyte arrays make full imaging costly and slow: targeted collection and the estate are the proportionate route unless the array itself is in issue (§6).

§ 11 · INTERROGATORIES & DRAFTING AIDS

Questions to ask · Suggested wording

Ask your client

- What server, NAS and array storage holds the relevant data, what RAID level and how many disks, and has any member failed or been replaced?
- Has anyone attempted, or been asked to attempt, a rebuild, restore or repair since the dispute arose?
- What backups, replicas and cloud copies of the same data exist, and how far back do they reach?

Ask your opponent

- Please preserve the relevant array, NAS and server storage without rebuilding, restoring, repairing or reconfiguring it, retain all member disks including any failed or replaced members in their original state, and confirm the RAID configuration and the identity of everyone who has handled the storage since [date].
- Please confirm whether any rebuild, restore, reconfiguration or member replacement has occurred since [date], with dates, the person responsible and the reason, and preserve all backups, replicas and snapshots of the relevant data.
- Please permit forensic imaging of the member disks and reconstruction by an agreed or independent examiner, and disclose the storage configuration, access logs and share permissions.

Ask your eDiscovery / forensic provider

- What RAID level and geometry is this, and can it be solved from the members or the controller?
- Is a full reconstruction needed, or does targeted collection of the relevant shares answer the question proportionately?
- What do the backups and estate already hold, so reconstruction is reserved for what exists nowhere else?

SUGGESTED WORDING · INSTRUCTION FOR AN ARRAY / SERVER RECOVERY

> "We instruct you to preserve, recover and examine the RAID array, NAS or server storage identified at Schedule 1. Please: (1) before any reconstruction, forensically image every member disk, including any failed or replaced members, with a write-blocker, hashing each image, and do not rebuild, restore or reconfigure the original storage; (2) determine the array geometry (disk order, stripe size, parity type and rotation, and offset) from the controller metadata or the member images, and reassemble the logical volume from the images, reaching through the file system and any virtual-disk layers as required; (3) validate and hash the reconstruction and document the geometry and method so it can be explained and reproduced; (4) where the array is degraded or has failed members, establish the failure history before reconstruction and state honestly any data beyond recovery; and (5) advise whether targeted collection of the relevant shares, preserved by snapshot or live acquisition without downing the system, is the proportionate approach, and identify the backups, replicas and cloud copies holding the same data. Do not permit any rebuild or restore of the original storage pending imaging."

§ 12 · QUICK CONTROL

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

The array / server checklist

- ☐ No rebuild, restore or reconfiguration of the original array permitted
- ☐ Disk positions and order recorded before any member is disturbed
- ☐ Every member forensically imaged, including failed and replaced disks
- ☐ Array geometry solved from controller or member images
- ☐ Logical volume reassembled from copies; layers reached in turn
- ☐ Reconstruction validated, hashed and documented for reproducibility
- ☐ Failure history established before any degraded-array recovery
- ☐ Live servers preserved by snapshot and targeted collection, not downtime
- ☐ Backups, replicas and cloud copies mapped and preserved
- ☐ Proportionality assessed: full reconstruction versus targeted shares

Red flags

- An administrator who "already started a rebuild" to fix the array: Example 1.
- Failed or replaced members discarded rather than retained.
- Disks pulled and shuffled without their order recorded.
- A rebuild, restore or reconfiguration around a key date.
- An array degraded beyond its parity with no backup identified.
- A multi-terabyte array being imaged wholesale where shares would do.
- A reconstruction offered as a black box with no documented geometry.

When to involve a digital forensic expert

Immediately, and before anyone touches the array, because the destructive act, a rebuild onto a failed or replacement member, is exactly what a well-meant administrator does next (Example 1, §5); at imaging, where every member is preserved forensically before reconstruction so recovery is attempted on copies; at reconstruction, where the geometry is solved and the layered reassembly performed, validated and documented (Example 3); at preservation of live servers, where snapshots and targeted collection keep the business running while the evidence is secured (§6); and at deployment, where the reconstruction and any spoliation-by-rebuild are explained under CPR Part 35 or CrimPR Part 19, alongside the estate that often holds the same data (Example 2). Through **cflab.uk** and **e-discovery.uk**, array and server recovery runs from write-blocked imaging of every member to a documented, reproducible reconstruction: because the data is on no single disk, and the way an array is handled in the first hour decides whether the evidence survives.

§ 13 · COMMON QUESTIONS

Frequently asked questions

Can you just image one disk from the server?

Not usefully, on a striped or parity RAID: a single member holds only interleaved fragments, every file sliced through, and reads as gibberish (§2). The array must be reassembled from forensic images of every member, using the solved geometry (§4). The exception is a mirror (RAID 1), where one member is a readable whole, but even then the array metadata matters for currency.

A disk failed and IT put in a new one and rebuilt the array. Is the evidence gone?

Possibly, and this is the single most damaging thing that can happen (§5, Example 1): a rebuild recalculates and overwrites stripes. Caught early, enough original stripes may survive on forensic images of the members to reconstruct; run to completion, the overwritten data is gone. Stop any rebuild at once, image every member including the failed and replacement disks, and recover on the copies.

The array has lost two disks. Can it be recovered?

It depends on the parity: a single-parity array (RAID 5) survives one failure, not two, so two losses put it beyond reconstruction from the survivors alone (§5). A double-parity array (RAID 6) survives two. Where the array is beyond parity, partial recovery of what the survivors cover is attempted, and the estate, backups, replicas, endpoints, carries the rest (Example 2, §8).

Do we have to shut the business server down to preserve it?

Usually not: live server storage is preserved by snapshots and live forensic acquisition that capture the relevant data while the system keeps running (§6), and by targeted, documented collection of the relevant shares rather than wholesale imaging. Downtime is reserved for where the array itself must be imaged member by member, which is planned to minimise disruption.

The whole array is huge. Do we really need to image all of it?

Often not: where it is the documents that matter, targeted collection of the relevant shares, home directories and mailboxes is the proportionate route (§6, guide 20). Full member-by-member reconstruction is reserved for where the array itself is in issue, its deleted data, its structure, or where the file system must be rebuilt. The method is sized to what the dispute actually needs.

Is a reconstructed array admissible as evidence?

Yes, when done properly: the members are imaged and hashed, the reconstruction is validated and its geometry and method documented so it can be explained and reproduced by another examiner (§4). A reconstructed logical volume is as sound as any other image; what must be avoided is the undocumented black-box reassembly that cannot be explained, and the rebuild-in-place that alters the original.

§ 14 · REFERENCE

Glossary

RAID
Redundant array of independent disks; storage spread across several disks.

Striping
Cutting data into pieces written across members for speed.

Parity
Redundancy data letting a failed member's content be recalculated.

Geometry
An array's order, stripe size, parity rotation and offset.

Degraded array
An array running with a failed member, still functional.

Rebuild
Recalculating a member's data; overwrites stripes and endangers evidence.

NAS
Network-attached storage; a small RAID appliance with a sharing layer.

Logical volume
The single readable image reassembled from the members.

Snapshot
A point-in-time capture of a live volume's state.

Virtual disk
A file on the array holding a whole virtual machine.

Sources and authoritative references

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

- cflab.uk, digital forensics services (RAID and server reconstruction, NAS and virtualisation recovery, 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 04 · Collection (server and array preservation, snapshot and targeted collection as practised): e-discovery.uk/edrm/preservation · e-discovery.uk/edrm/collection
- e-discovery.uk, practice method and Knowledge Centre: e-discovery.uk · e-discovery.uk/knowledge
- Forensic Science Regulator, Code of Practice v2 (integrity and method in recovery and reconstruction): gov.uk, FSR Code · ISO/IEC 27037 (identification, collection, acquisition and preservation): iso.org, 27037
- NIST SP 800-86, Guide to Integrating Forensic Techniques into Incident Response (acquisition of complex storage): csrc.nist.gov, SP 800-86
- PD 57AD (preservation and proportionality) and CPR Part 31 / PD 31B: justice.gov.uk, PD 57AD · justice.gov.uk, Part 31 · CPR Part 35: justice.gov.uk, Part 35

§ HOW A SPECIALIST LABORATORY CAN ASSIST

Working with Computer Forensics Lab

Array and server work at the laboratory begins with the instruction that saves most cases: do not rebuild, restore or reconfigure the storage, because the well-meant repair is what destroys the evidence (Example 1). Every member is imaged forensically with a write-blocker, including the failed and replacement disks, and the reconstruction is performed on the copies: the geometry solved from the controller or the member images, the logical volume reassembled, and the file-system and virtual-disk layers reached in turn, each validated, hashed and documented so the reconstruction can be explained in evidence and reproduced (Example 3). Degraded arrays are recovered from the parity where the failure history allows, and the honest limit is stated where members are lost beyond recovery, with the estate, backups, replicas, endpoints and cloud, worked for the same data (Example 2). Live business servers are preserved by snapshot and targeted collection so the business keeps running, and the method is sized proportionately to whether it is the array or merely the documents that is in issue. Reports run under CPR Part 35 or CrimPR Part 19. Conflicts checks, confidential scoping discussions and fixed-fee estimates are routine; and if an array has failed or a rebuild has been started, the first message is the urgent one: stop, and image every disk before anything else.

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