Macrium Software

Operationally Resilient

The Recovery By-Design **Blueprint**

A Macrium Framework for **Operationally Resilient Recovery** in Manufacturing and Critical Infrastructure

Manufacturing doesn't stop. And neither do the threats.

Your lines run shifts. Your systems run continuously. Downtime isn't an inconvenience — it's a direct hit to output, contracts, and safety. Every hour offline has a number attached to it, and in most manufacturing environments that number is significant.

But here's the reality most organisations are sitting with: they have backups. They may have a disaster recovery plan, often a business continuity plan too. What they don't have is proof. The backup job runs. The plan sits in a folder. And no one has tested whether either one holds up until the day it has to.

Macrium's *State of Backup and Recovery in Manufacturing found* that **75%** of manufacturing organisations experience unplanned downtime every year. Half experience it quarterly. When it happens, only 1 in 5 meet their recovery time targets. The rest take longer than planned, sometimes much longer. The reason is straightforward: only half of organisations test their backups regularly, and fewer still rehearse the full recovery plan end to end. Most are measuring whether the backup job completed, not whether they can restore, in the right order, to the working state.

This isn't a technology gap. It's a design gap.

The threat picture makes this more urgent, not less.

Dragos recorded 1,693 ransomware attacks against industrial organisations in 2024, an 87% increase year on year. Of the incidents they responded to, 25% caused full OT site shutdowns. The Waterfall 2026 OT Cyber Threat Report puts a number on what that looks like in practice: the Jaguar LandRover attack in 2025 shut down UK factories for over a month, with total economic impact estimated at $2.5 billion. Nation-state and hacktivist attacks on critical infrastructure doubled in the same year. And Waterfall's own analysis warns that the current dip in attack volume is temporary — expect increases to resume at pace in 2026 and 2027.

87% increase year on year.

25% Incidents responded to

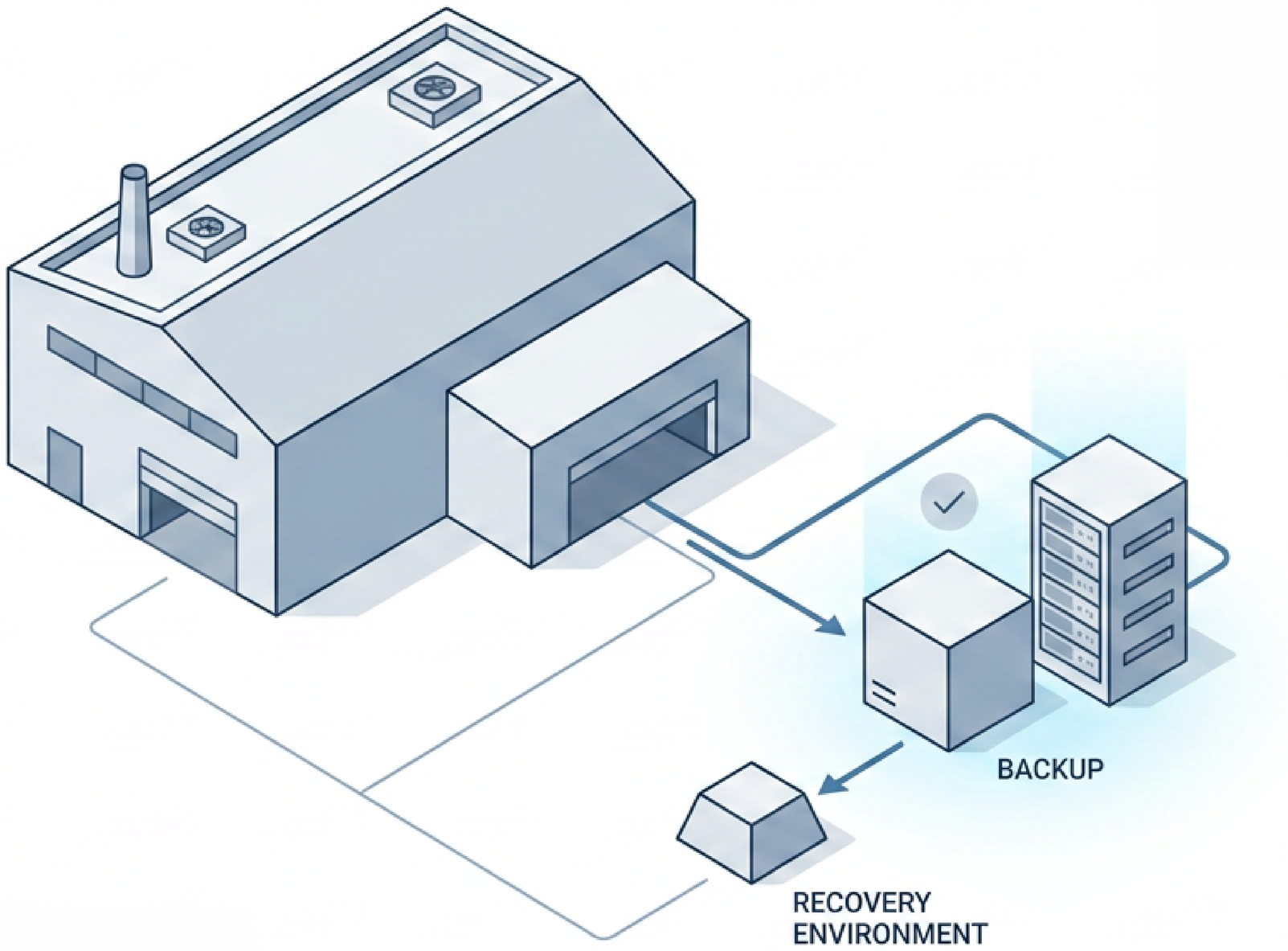
The regulatory environment is catching up too.

NIS2, CAF 4.0, the UK Cyber Security and Resilience Bill, NIST CSF 2.0, and IEC 62443 all converge on the same expectation: not "do you have backups?" but "can you prove you can recover safely, quickly, and predictably?" The compliance question has changed. Most organisations' recovery programmes haven't.

In OT, availability is everything.

There is also something specific to OT environments that pure IT thinking misses entirely. In manufacturing and critical infrastructure availability is the priority. Operations being online is the mission. Backup and recovery activity cannot disrupt that. It has to be designed around it — non-intrusive, operationally achievable, and built for the reality of air-gapped networks, segmented zones, legacy control systems, and the subsystems that run quietly in the background until they don't.

That is what recovery by design means. Not backup by habit. Not Disaster Recovery as an afterthought. A deliberate, structured approach to recovery that starts with understanding what matters, builds protection around it, proves it works under realistic conditions, and governs it as an ongoing operational programme.

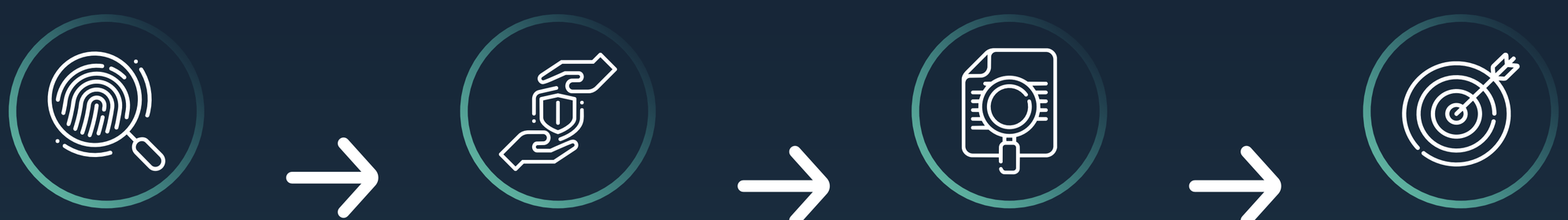


The Recovery by Design Blueprint

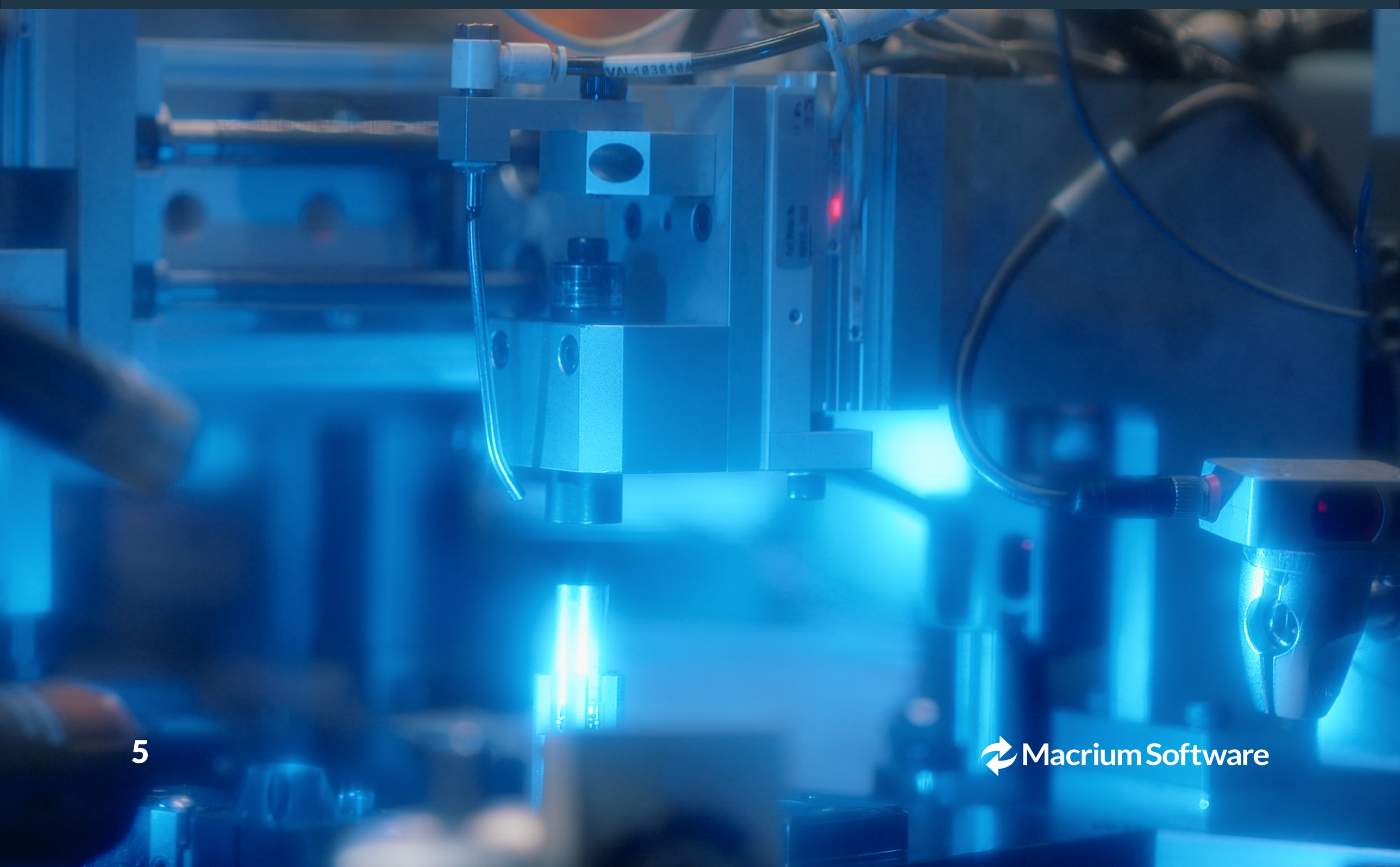
4 Pillars of Operationally Resilient Recovery

Most recovery programmes in manufacturing are built backwards. They start with the backup tool, work outward, and hope that covers it. The Recovery by Design Blueprint starts from the other end (with business impact, operational priorities, and the reality of your environment) and builds a recovery capability that is structured, provable, and repeatable.

The Blueprint is built on four pillars. They are sequential by design. You cannot govern what you have not proven. You cannot prove what you have not protected. You cannot protect what you have not identified. Each pillar depends on the one before it, and together they form a complete operational programme — not a checklist, not a project, but an ongoing practice.



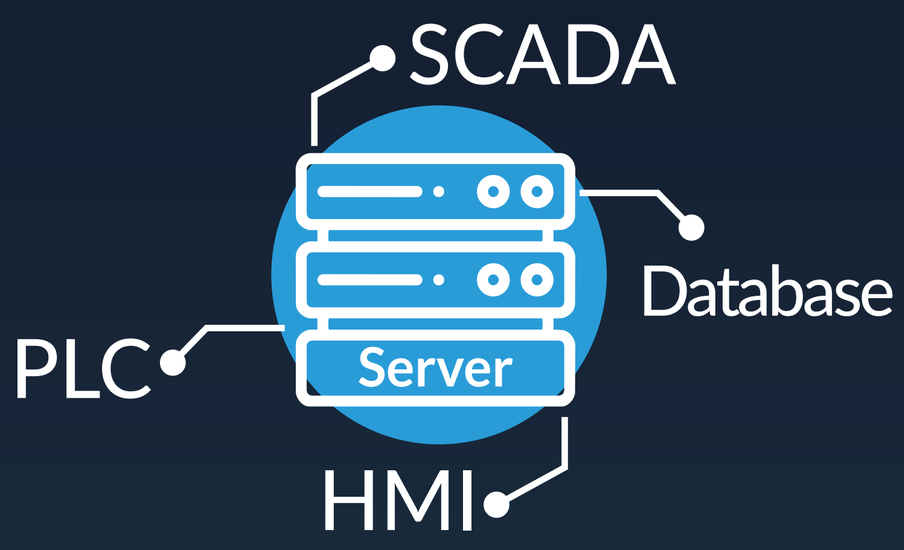
Identify **Protect** **Prove** **Govern**



PILLAR 1

Identify and Prioritise Recovery

Know what you have, what it depends on, and what must come back first. This is where recovery starts — not with technology, but with a clear-eyed view of your assets, your dependencies, and your recovery priorities across every zone of your environment.



PILLAR 2

Protect: Plan and Implement Your Backup Strategy

Build a backup strategy that fits your operational reality. One that is non-intrusive, appropriately protected, and simple enough that anyone on your team can execute it under pressure.

PILLAR 3

Prove Recovery

A backup you have never tested is not a recovery capability. It is an assumption. This pillar is about turning that assumption into a verified, evidenced, deterministic capability — knowing you can restore, in the right order, to a known-good state.

PILLAR 4

Govern Recovery

Recovery is not a project with an end date. It is a programme that needs ownership, metrics, audit evidence, and a lessons learned loop. This pillar makes recovery measurable, reportable, and repeatable over time.

Follow the four pillars and you are also addressing the core recovery expectations of NIS2, CAF 4.0, the UK Cyber Security and Resilience Bill, NIST CSF 2.0, and IEC 62443. Compliance becomes a consequence of doing the right thing — not a separate workstream.

The Blueprint in Practice

The following sections take each pillar in depth. For each one you will find the key principles, the practical steps, and the specific considerations that apply to OT and manufacturing environments. Read them in order the first time — the logic builds. After that, use them as a working reference for whichever pillar you are focused on.

Pillar 1
Identify And Prioritise Recovery

Recovery starts with business impact and clear priorities — not with the backup tool.

Most OT environments have a reasonable handle on their most visible, most critical systems. They are documented, maintained, and treated with the seriousness they deserve. What is often less well understood are the supporting systems around them — the assets that run quietly in the background, with no service agreements, no regular maintenance visits, and sometimes no backup at all. These are the gaps that turn a contained incident into a prolonged outage.

Pillar 1 is about closing those gaps before you need to.

Know what you have — all of it

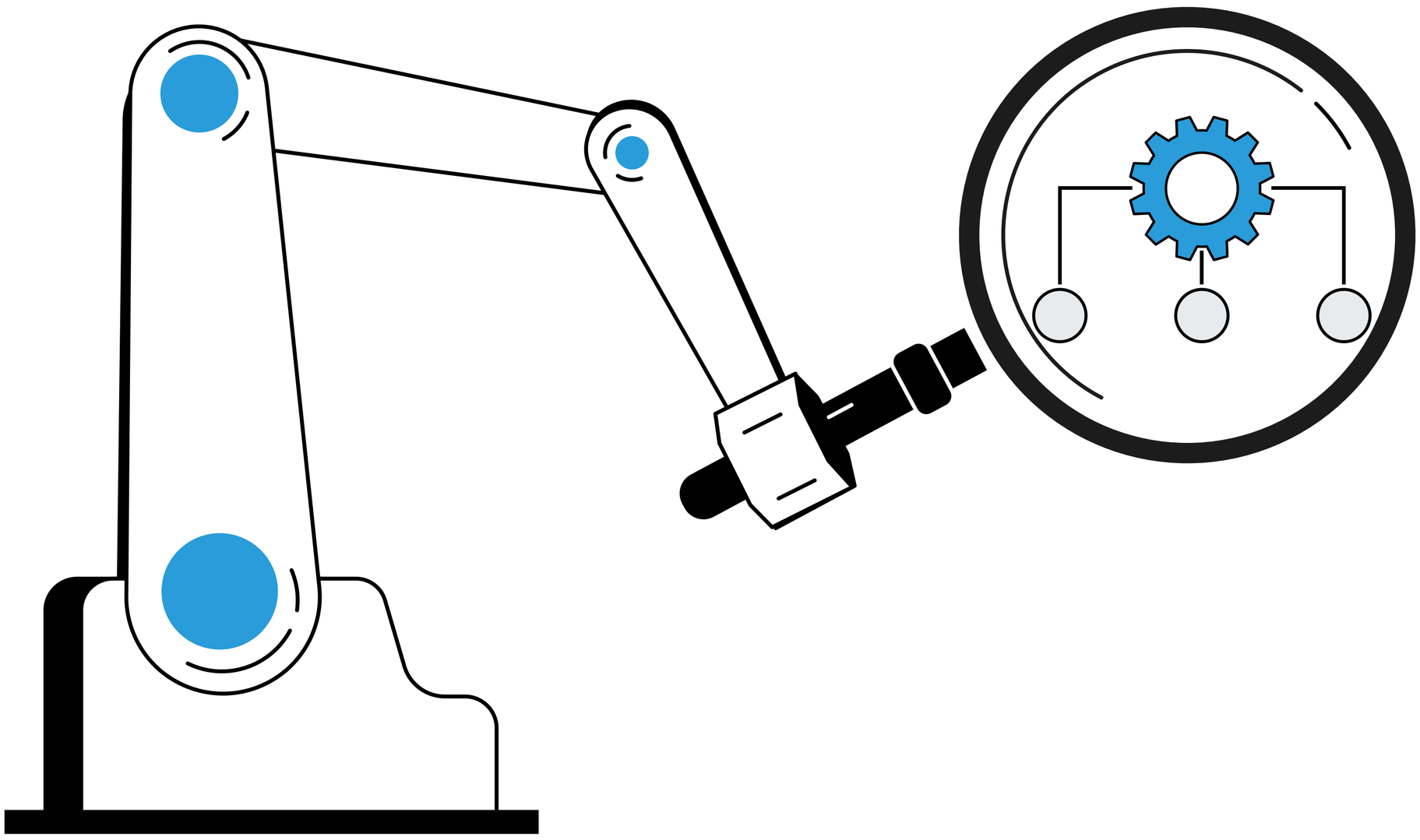
Start with a full asset inventory: data, systems, identities, configurations, licenses, and OT engineering assets including PLC logic, SCADA and HMI configurations, historian data, MES, network configs, and keys. If a vendor owns the backup process for a system, that system still needs to be on your map — because if it fails, the recovery problem lands with you.

In manufacturing this is harder than it sounds. Sites grow organically, often over decades, and the asset picture grows with them. An IT view alone won't surface everything. Walk the shop floor. Trace what each system actually does to the operation, not just where it sits on the network. And don't skip the systems no one wants to touch, the unlabelled box running a process no one fully remembers is exactly the kind of asset that turns a contained incident into a prolonged outage.

Every asset needs a tier, and that tier drives everything that follows — recovery targets, backup frequency, testing priority, and spend. You can't tier everything at once, so start where it matters. Identify your priority-one (P1) assets first (the systems that stop the operation if they fail), then work outward through what supports them, mapping dependencies and wider impact as you go.

Map your dependencies

An asset does not exist in isolation. Map the people, processes, infrastructure, vendors, and network connections each system depends on. Understand what talks to what, and what the knock-on effect looks like if something in that chain fails or is compromised. Include identity and admin control planes, hypervisor and storage layers, and your backup platform itself.



Plan for your threat landscape

Distinguish between routine failures, site-wide events, and cyber incidents. Your recovery approach for a single failed system is different from your approach for a network compromise. Plan for both.

Record what you find

The output of this pillar is a documented asset inventory, criticality tiers, dependency map, and recovery objectives. Write it down clearly enough that a new team member could follow it. If it only lives in someone's head, it doesn't exist when you need it most. Documentation isn't a closing step, it runs through all four pillars, and what you capture here is the foundation everything else records against.

Pillar 2

Protect: Plan And Implement Your Backup Strategy

Engineer the backup strategy and ensure it doesn't compromise production

You know what you have and what matters. Now you need to protect it. Pillar 2 is where strategy meets execution — building a backup approach that is appropriately engineered, operationally achievable, and simple enough to run consistently without becoming a burden on the people and systems that keep production running.

Plan your strategy around your requirements

Your asset inventory and criticality tiers from Pillar 1 are the direct input here. Different assets have different backup requirements — frequency, retention, storage location, and recovery method will all vary. Resist the temptation to apply one approach across the board. A backup and recovery strategy has to be realistic. Ground it in your operational constraints, not an ideal no one can sustain. Be proactive: decide in advance how often each tier is backed up, where the copies live, and how much redundancy a critical system needs — before it goes down, not after. Frequency follows the RPO you set in Pillar 1. The tighter the RPO, the more often you back up.

Two key constraints shape every decision that follows, and in OT both are non-negotiable.

The first is availability. A backup job that slows a system or risks interrupting a control process is not acceptable. Schedule backups around operational cycles, validate their impact through performance testing, and treat any intrusion into live operations as a design failure to fix — not a trade-off to accept. Protecting an asset can never threaten its availability.

The second is connectivity, or the lack of it. Plan for the moment the network is unavailable or untrusted. Keep installers, configuration files, and local repositories available offline, so recovery never depends on systems that may themselves be compromised or unreachable during an incident. Build this in now, not when the network is already down.

Use a proven architecture

The 3-2-1-1 rule is the right baseline: **three copies of data, across two different media types, with one copy offsite and one protected from tampering.** In OT environments where cloud connectivity is limited or unwanted (often for physical or security reasons), the offsite copy often means physical media. That is fine, but it has to be deliberate, documented, and tested.

One copy needs to be on media that can't be tampered with. Ransomware that reaches your backup platform can destroy unprotected copies fast. The fix is straightforward: air-gapped storage, offline media, or immutable storage all achieve the same goal. Pick what fits your environment, and make sure at least one copy is beyond reach.

Keep it simple and (again) document everything

A backup and recovery strategy only its designer can operate is a liability. When an incident hits, pressure is high and time is short. Keep it simple. Make sure more than one person can execute it. And document it. Documentation in this pillar isn't a job log — it's the runbook: what gets backed up, on what schedule, to where, and the exact steps to restore. Clear enough that someone who didn't build it can run it under pressure. Backups must be protected and operationally achievable. If they're not both, they're neither.

Pillar 3

Prove Recovery

A backup you have never tested is not a recovery capability. It is an assumption.

Having backups is not the same as having a recovery capability. Pillar 3 turns that assumption into something deterministic — a verified, evidenced ability to restore the right systems, in the right order, to a known-good state. **The "0" in 3-2-1-1-0 is earned here. Zero errors on verified recovery. Not assumed. Proven.**

Backup completion is not enough — as a minimum, verify

Most organisations measure whether backup jobs complete. That is the wrong metric. Completion tells you data was copied. It does not tell you whether it can be restored, whether the system will function correctly, or whether it is safe to return to production.

As a minimum, run integrity verification after every backup, i.e. hash verification, confirming the data is intact and uncorrupted. It is a simple step, but it is the difference between a backup and a proven backup.

Move beyond verification

Verification is the starting point, not the destination. From there, build toward progressively more meaningful tests.

- **Virtual boot testing (Sandbox testing)** comes first: boot the backup image in an isolated sandbox, away from your live environment, to prove the operating system loads and applications start. It is low cost, practical, and achievable for most teams — and it confirms what completion never tells you, that the image actually runs.
- **Hardware restore** testing goes further, validating recovery on physical equipment under real conditions.
- **Full disaster recovery exercises** test the whole process — technology, team, procedures, and timing — against a realistic scenario.

You don't need a full replica of your operation to do this. Most OT sites already run representative test environments for patch and upgrade testing. Reuse them. The environment you maintain to trial a firmware update is the same one that lets you prove a recovery without touching the live line. Not every system needs the same depth of testing. Apply your criticality tiers from Pillar 1, and start with the systems that would stop the operation.

In OT, safe to reconnect is not the same as safe to operate

Before any restored system goes back into production, verify that control logic, setpoints, interlocks, and safety configurations are correct. A system that boots is not necessarily a system that is safe to run. In manufacturing environments this distinction matters. Verify it explicitly, every time.

Make testing routine

Once a year is not enough. Schedule regular testing, automate verification where possible, and retest after any significant change.

Once a year is not enough. Schedule regular testing, automate verification where possible, and retest after any significant change.

In OT you can't simply test in production, no one does that more than once. Work with the reality instead. Planned downtime, like an annual maintenance shutdown, is your window for the heavier exercises. And don't wait for the calendar: when a specific system changes, test that system then, while you know exactly what moved. Recovery capability degrades silently — new systems, configuration changes, and infrastructure updates all introduce gaps that only surface when you test. Document every test. Record what was tested, the results, the time taken, and any issues found. This evidence supports your audit trail and shows your recovery programme is real, not theoretical.

For a practical step-by-step guide to building your validation approach, see the Macrium OT Engineer's Playbook for Backup Validation.

Pillar 4
Govern Recovery

The first three pillars build your recovery capability. Pillar 4 keeps it alive — relevant, operational, and provable. Without governance, recovery programmes drift. Testing gets skipped. Documentation goes stale. Ownership blurs. A capability that existed six months ago may not exist today, and you won't know until you need it.

Governance is also where compliance is won. Every framework that touches OT — IEC 62443, NIS2, CAF 4.0, NIST CSF 2.0 — asks for the same thing: documented proof that you can recover, maintained over time. That proof is a governance output. The evidence an auditor wants is the same evidence a real recovery depends on. Govern recovery properly and compliance stops being a separate workstream — it becomes a by-product of the programme.

Running recovery like a programme means treating it with the same operational rigour you apply to production itself.

Own it

Recovery needs a named owner and clear roles. Who declares an incident? Who makes the call to restore? Who communicates with the business? Define these in advance and make sure the right people know what is expected of them.

Include your vendor dependencies too. OEM contacts, integrator relationships, tool licenses, and offline support options all need to be documented and maintained. In an incident, time spent tracking down a support contact is time you cannot afford.

Build your incident readiness

Have a first 24 hours playbook. When an incident happens, the first decisions are the most consequential — contain, assess, protect your backups, and establish a recovery decision point. These steps need to be written down and practiced, not improvised on the day.

Align your incident processes to the reporting timelines your organisation is subject to. NIS2, the UK Cyber Security and Resilience Bill, and CAF 4.0 all carry 24 to 72 hour notification expectations. Meeting those timelines requires triage and decision-making that is already practiced.

Create an auditable evidence trail

IEC 62443 places significant weight on auditability — the ability to demonstrate that recovery processes are controlled, repeatable, and safe to operate. Standardise your evidence pack. For every recovery test, capture what was restored, the time taken, validation results, issues identified, and sign-off. Retain it. Ensure time synchronisation across zones so logs align when you need them.

Ensure time synchronisation

Keep it current

Recovery documentation goes stale quickly. Retest after change — patches, configuration updates, new systems, and segmentation changes. Maintain your assets in an active lifecycle. Equipment that is out of support or unpatched is a recovery risk before it is anything else.

Close the loop

After every test and every real incident, document what happened, what worked, and what did not. Feed that back into your recovery strategy, update your documentation, and schedule a retest. This is also where the full DR plan lives — written clearly enough that a new team member could follow it under pressure, covering asset inventory, criticality tiers, dependency map, recovery order, targets, assumptions, and escalation paths.

Built for Operations.
Ready for Compliance.

Follow the Blueprint. The compliance follows.

Every major framework that touches OT and industrial environments has something to say about backup and recovery. The language differs. The scope differs. But the underlying expectation is consistent across all of them — you must be able to demonstrate that you can recover, not just that you have backups.

The table below captures what each framework is really asking. If you are following the Recovery by Design Blueprint, you are already answering these questions.

Framework	Where it applies	Who it applies to	The recovery question it is asking
IEC 62443	Global	OT asset owners, system integrators, and industrial control system operators across manufacturing, energy, utilities, and critical infrastructure	Can you safely restore industrial operations to a known-good state?
NIST CSF 2.0	North America	Any organisation — commonlyexpected in regulated sectors and supply chains	Can you recover business operations predictably?
IEC 62443	European Union	Medium and large entities in 18 critical sectors, plus key digital and ICT service providers	Can you prove you can recover essential services?
CAF 4.0	United Kingdom	Operators of essential services and public sector bodies assessed by regulators	Can you restore the essential function under attack conditions?

The question has shifted. It is no longer "do you have backups?" It is "can you prove you can recover safely, quickly, and predictably?" The Recovery by Design Blueprint is built around exactly that question — and working through its four pillars gives you the evidence, the structure, and the operational capability to answer it with confidence.

Recovery by Design Self-Assessment

How well does your recovery programme stand up? Work through the checklist below. It is not an audit — it is a starting point. The questions are designed to surface the gaps that matter most.

Pillar 1 — Identify and Prioritise Recovery

- Do you have a complete inventory of every OT asset in your environment, including subsystems and vendor-managed systems?
- Have you assigned criticality tiers to your assets — and do those tiers drive your recovery priorities?
- Do you have documented recovery targets (RTO and RPO) for your critical systems?
- Do you know what your recovery sequence looks like — what needs to come back first, and what depends on what?

Pillar 2 — Create, Schedule and Protect Backups

- Does your backup strategy reflect the criticality of each asset — or is it the same approach applied across the board?
- Do you have at least one copy of your most critical backups on media that cannot be tampered with or encrypted by ransomware?
- Have you validated that your backup jobs do not impact production systems or operational processes?
- If your network was unavailable or untrusted today, could your team still execute a recovery?

Pillar 3 — Prove Recovery

- Have you tested the restoration of your most critical systems in the last three months?
- Do you test beyond backup completion — verifying that restored systems actually function correctly?
- For your OT systems, have you confirmed that restored configurations, control logic, and safety settings are correct — not just that the system boots?
- Do you have documented evidence of your recovery tests that you could present to an auditor today?

Pillar 4 — Govern Recovery

- Is there a named owner for recovery in your organisation — someone accountable for keeping the programme current?
- Do you have a first 24 hours playbook that your team has practiced?
- Does your recovery documentation get updated when systems, configurations, or infrastructure change?
- Could a new team member follow your DR plan under pressure without needing to ask for help?

Macrium Software

If you answered **no** to several of these questions,

YOU ARE NOT ALONE.

Most manufacturing and OT environments have gaps — the important thing is knowing where they are**. Speak to our team about how to strengthen your recovery posture.**

Speak to our team