A FIELD MANUAL

Every Door You Handed Out

Third-party and concentration risk — tiering, contracts as controls, vendor access, offboarding, and designing for the day a supplier fails.

Bhavin G Patel

C|CISO · CISSP-ISSAP · CISM

cybersecleader.com

Ten runbooks · Nine templates · Built to be adapted, not admired

About this manual

Some of your most dangerous risk isn't inside your company at all. It is sitting in the systems of vendors you signed a contract with and then mostly forgot about. You can build a beautiful fortress and then hand keys to sixty companies of wildly varying security maturity, and an attacker only needs the weakest one.

Most vendor-risk programmes are theatre. A long questionnaire, filled in optimistically, filed somewhere nobody opens again. This manual is about the parts that actually reduce risk: ruthless tiering so effort lands where the blast radius is, contracts negotiated while you still have leverage, access designed on the assumption the vendor will be breached, and an offboarding process that treats the end of a relationship as a security event rather than a procurement one.

Who it is for

The person who would have to explain to a board that the quarter's biggest single point of failure was a company they had never heard of. A CISO, a third-party risk lead, a procurement partner who wants the security clauses to mean something, or the engineer who just found a live API key for a vendor nobody has done business with in three years.

How to use it

If you inherit a programme that's already theatre, do these in order: run **TP-01** to tier what you have, run **TP-05** against the top tier to find out what access actually exists today, and run **TP-08** against every relationship that ended in the last three years. The third one is where most organisations find something they did not expect.

> **THE PREMISE**
>
> You are not trying to guarantee your vendors are perfect. You cannot, and pretending you can is how these programmes become theatre. You are making sure their worst day does not automatically become yours — not more trust, but less dependence on trust, backed by least privilege, real monitoring and a contract with teeth.

Disclaimer

This manual is general information, not legal, regulatory or procurement advice, and reading it does not create an advisory relationship between us. The contract clause objectives describe outcomes to negotiate for, not drafting to adopt; have your own counsel produce the language. Obligations vary by jurisdiction, sector and contract. I accept no liability for decisions taken on the basis of this document. Verify before you rely.

Contents

PART ONE

The Position

1 · Your Perimeter Includes Companies You Forgot

You spent the year hardening your own environment. Meanwhile the softest way in is a company you onboarded eighteen months ago and have not thought about since.

Every integration is a door. Every SaaS tool with access to your data is a door. Every contractor's laptop is a door. The uncomfortable arithmetic is that you don't defend one castle any more — you defend every door you handed out, and an attacker only has to beat the least-funded vendor in your supply chain before walking in through an entrance you propped open yourself.

The risk is not where the scrutiny is

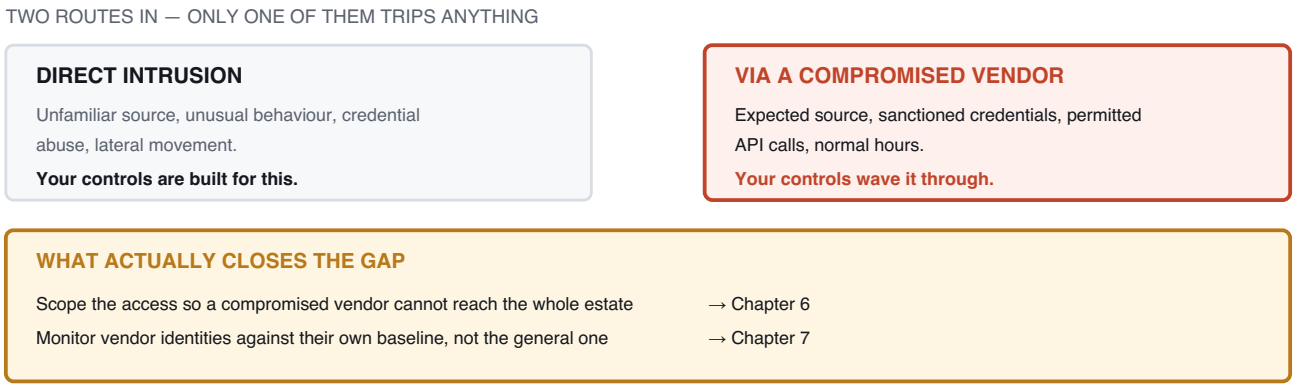
Here is the pattern, repeated across organisations of every size: the vendors that get examined are the large, obvious ones with recognisable names and mature security teams. The vendors that cause incidents are small, deep in a workflow, holding access nobody has re-examined since it was granted. The scrutiny and the risk are inversely correlated, and the mechanism that produces this is entirely mundane — a large vendor triggers a procurement process, a small one gets a credit card and an API key.

> **THE REGISTER YOU DO NOT HAVE**
>
> Most organisations cannot produce a complete list of third parties with access to their data or environment. Not a list of contracts, a list of *access*. Those are different populations and the second is always larger. Producing it is the first honest thing a programme can do, and the gap between the two lists is usually enough to fund the rest of the work.

2 · The Attack Comes Through the Trusted Door

Supply-chain attacks work because the access is already legitimate. From the inside it does not look like an intrusion, it looks like the integration doing exactly what it is allowed to do.

This is what makes third-party compromise structurally harder to detect than direct intrusion. Your controls are built to notice unauthorised behaviour. A compromised vendor performs authorised behaviour, from an expected source, with valid credentials, at a plausible time. The alerting was never tuned to be suspicious of a partner, because being suspicious of partners generates noise the team can't service.



> You will not detect your way out of this. The controls that work are access design and vendor-specific baselining.

3 · Why Questionnaires Answer the Wrong Question

A 200-question spreadsheet tells you what a vendor is willing to claim about themselves on a form. It tells you nothing about what they would actually do the morning they get breached.

The questionnaire isn't worthless. It establishes a baseline, it creates a record, and for regulated firms it is often expected. But it curdles into ritual within a year almost every time, and the reason is structural: it is answered by a sales engineer optimising for the deal, reviewed by someone without the authority to reject, and filed where nobody reads it again. It generates paperwork and the comfortable illusion of diligence while the real access sits unexamined.

Three questions that are worth more than the whole questionnaire

What can this vendor actually reach today? Not what the contract says. What the access token, the API scope, the network route and the integration permit right now. Almost nobody can answer this without going and looking, and going and looking is the exercise.

What happens if they are breached tonight? Concretely — which data, which systems, what could an attacker do through them. If nobody has traced this, the tier assignment is a guess.

How fast would we find out, and how? Their contractual notification window is one answer. Whether you would detect it independently is the more useful one.

> **WHERE EVIDENCE BEATS ATTESTATION**
>
> Independent evidence — an audit report you actually read rather than filed, a penetration test summary, external attack-surface observation, breach history — is worth more than a self-assessment, and is available for most vendors that matter. Reserve the deep version of this for Tier 1, because it costs real time.

4 · Tiering: Spend Scrutiny Where the Blast Radius Is

Not every vendor matters equally, and treating them as if they do is how you burn the whole budget on paperwork before you reach the ones that could sink you.

Ruthless prioritisation is the single change that most improves a vendor programme, and it is unpopular because it means explicitly deciding that most vendors will receive almost no attention. That's the correct decision. Equal effort across sixty vendors means running out of effort before reaching the one that holds your customer data.

Tier by blast radius, not by spend

Contract value is the wrong axis and it is the one most programmes accidentally use, because procurement thresholds drive the process. The vendor that processes your payments may cost less annually than the vendor that supplies office furniture. Tier on what they can reach and what breaks if they fail — data sensitivity, system access, operational criticality, and substitutability.

Tier	Definition	Treatment
Tier 1	Holds regulated or customer data, or has privileged access to production, or the business stops without them.	Technical due diligence, negotiated security terms, named relationship owner, access recertified quarterly, monitored, exit plan, concentration assessed.
Tier 2	Holds internal data or has limited system access. Disruption is material but survivable.	Evidence review rather than questionnaire alone, standard security terms, annual access recertification.
Tier 3	No sensitive data, no system access, easily replaced.	Light touch. Register entry, standard terms, offboarding when it ends. Deliberately minimal.

The tiering must be re-run when circumstances change, because vendors move between tiers silently. A Tier 3 tool that quietly gains an integration with your CRM is now Tier 1 and nobody re-tiered it. TP-06 is the mechanism.

PART TWO

Controlling the Relationship

5 · The Contract Is a Control

The contract is one of the strongest security controls you have with a vendor, because signing is the one moment you hold real leverage. Try to negotiate any of it after an incident and you will discover how little you have left.

Security teams routinely treat contracting as procurement's problem and then spend years working around terms they could have shaped in an afternoon. The clauses that matter are few, and the objectives are more important than the drafting — which is your counsel's job, not this manual's.

Objective	Why it matters, and what weak looks like
Breach notification with a real deadline	Hours, not "promptly" or "without undue delay". Weak language means you learn from a journalist. Specify the clock start — awareness, not confirmation.
Right to audit or to evidence	The right to verify rather than accept attestation. For most vendors an evidence right is achievable where a physical audit right is not; take it.
Minimum security requirements with consequences	Requirements with no remedy attached are aspirations. Tie them to termination rights or service credits.
Data scope, location and retention	Exactly what they may hold, where, and for how long. Ambiguity here is what turns a vendor breach into your regulatory problem.
Sub-processor disclosure and approval	You cannot assess concentration you cannot see. Chapter 8 and TP-10.
Return and deletion on termination	With evidence of deletion. This is what makes offboarding enforceable rather than hopeful.
Cooperation during an incident	An obligation to assist your investigation, provide logs and make people available. Absent this, you are dependent on goodwill at the worst moment.

THE LEVERAGE CURVE

Your negotiating position is at its maximum before signature and decays continuously afterwards. It reaches its minimum during an incident, which is precisely when you need these terms. Security must be in the room before the contract is signed, and if it is not. That's an organisational fix rather than a contractual one.

6 · Access Design: Assume They Will Be Breached

The mature move is not trusting your vendors more. It is designing for the day one of them fails — because at sixty vendors, over enough years, one of them will.

Every access decision should be made against a single question: **if this vendor were fully compromised tonight, what could the attacker reach through them, and are we comfortable with that?** If nobody has traced the answer, the access was granted on trust rather than design.

Five principles that survive contact with reality

Scope to the workflow, not to the vendor. Vendors are typically granted access shaped by convenience — a role that already existed, an API key with broad scope because narrowing it required work. Derive the permission set from what the integration actually does.

Prefer time-bounded and just-in-time access for anything human, particularly contractor and support access. Standing vendor access is the category that most often survives the relationship that justified it.

Give each vendor a distinct identity. Shared service accounts across integrations make attribution impossible and revocation destructive. When you need to cut one vendor off in an incident, you will discover what else was using that credential.

Terminate their access at a boundary you control. An integration that terminates inside your general network is harder to contain than one that terminates at a gateway with its own policy enforcement.

Make revocation a single, tested action. If cutting off a compromised vendor requires coordinating four teams and an unknown number of credentials, containment will take days. Know in advance how you would do it, and for Tier 1 vendors, have tested it.

7 · Monitoring What They Can Actually Touch

Their bad day should show up on your radar rather than in a reporter's inbox. That requires monitoring built specifically around vendor identities, because general monitoring will not notice authorised activity.

The practical approach is to baseline each Tier 1 vendor identity separately. An integration that normally reads two hundred records a day, from one source address range, during business hours, has a shape. Deviation from *its own* baseline is meaningful even though nothing it did was unauthorised — and that's the only signal available, because every individual action is permitted.

Signal	Why it matters
Volume deviation	Bulk retrieval through a legitimate integration is what exfiltration looks like from your side.
New source address or geography	The integration moved, or someone else is using its credentials.
Access outside the usual pattern	Data or endpoints the vendor has permission for but has never previously touched. Permission is not usage, and the gap between them is the useful signal.
Timing anomalies	An integration running at 03:00 that has never run outside business hours.
Authentication changes	New credential issued, scope expanded, MFA disabled on a vendor-side account.
External signals	Breach disclosure, credential dumps mentioning their domain, their external attack surface changing.

8 · Concentration, Fourth Parties and Substitutability

Two questions supervisors increasingly ask and firms increasingly cannot answer: how many of your critical services depend on the same underlying provider, and could you actually move if you had to?

Concentration hides behind vendor names. Four different SaaS products, four different contracts, four different account managers — and all four hosted on the same cloud region, or all four using the same authentication provider, or all four relying on one small analytics company nobody has heard of. Diversity at the vendor layer isn't diversity at the dependency layer, and only fourth-party visibility reveals the difference.

Substitutability is the question underneath

A vendor you could replace in a month is a manageable dependency regardless of how critical the function is. A vendor you couldn't realistically replace at all is a strategic exposure that belongs in front of the board, whatever the contract value. Assessing this honestly is uncomfortable because the honest answer for deeply embedded platforms is often "we couldn't," and that conclusion has budget implications people would rather avoid.

> **FOR REGULATED FIRMS**
>
> Material outsourcing and ICT third-party regimes formalise all of this: a register of information, criticality assessment, substitutability, exit planning and testing. If you are in scope, the register is not optional and the exit plan will be examined. If you are not in scope, the discipline is still the right one, you are simply choosing your own thresholds.

PART THREE

The Lifecycle

9 · Onboarding: The Only Moment You Have Leverage

Everything is easier before signature. Almost every failure later in the lifecycle traces back to something that was cheap to fix at onboarding and expensive afterwards.

The onboarding sequence that works is short and its order matters: classify what data and access is involved, tier accordingly, do the diligence appropriate to that tier, negotiate the terms while leverage exists, provision access scoped to the workflow, and register the relationship with a named owner. TP-01 through TP-04 cover each step.

The single most common defect is that this runs *after* the commercial decision has effectively been made, at which point security review becomes a formality with a predetermined outcome. Fixing that's an organisational change, getting the intake trigger attached to the evaluation rather than to the purchase order.

10 · The Middle Years: Drift and Scope Creep

A vendor whose scope crept far beyond what they need today is carrying risk you are getting no value for.

Relationships change in ways nobody records. The integration that started read-only gained write access for a project. The support arrangement that was meant to be temporary became permanent. The product added a feature that now processes a data type the original assessment never contemplated. None of these generate a review trigger, and collectively they mean your risk picture is accurate on the day of onboarding and degrades continuously thereafter.

Two mechanisms address this. **Recertification** (TP-05) asks whether the access that exists is still the access needed, and requires the business owner to actively confirm rather than passively renew. **Reassessment triggers** (TP-06) catch material change — a vendor breach, an acquisition, a new data type, a tier change, between scheduled reviews.

> **RECERTIFICATION ONLY WORKS IF REFUSAL IS POSSIBLE**
>
> A recertification campaign where 99% of access is renewed with a single click is measuring nothing. Present the reviewer with what the access actually permits and what it has recently been used for, and make “remove” the easy option. If nothing is ever removed, the process is decoration.

11 · Offboarding Is Where It Quietly Rots

Everyone focuses on onboarding. Almost nobody focuses on the other end, and that is where the real decay happens.

You stop using a tool. The subscription lapses. The team moves on. And the integration is still live, the API key still works, the access token nobody rotated is still valid. Walk into enough organisations and you will find active credentials for vendors they haven't done business with in years. That's not a door somebody forgot to lock — it is a door to a building they forgot they owned.

The fix is conceptual before it is procedural: **vendor offboarding is a security event, not a procurement afterthought.** When a relationship ends, the access ends with it — keys revoked, tokens killed, integrations torn down, accounts disabled, network rules removed, and someone actually confirming it happened rather than assuming it did. TP-08 is the checklist and Appendix F is the record.

Two failure modes deserve naming. **Silent lapse** is where no offboarding is triggered because no formal termination occurred, the relationship simply stopped. Catching this requires periodically reconciling active vendor access against active contracts, and the difference is always instructive. **Partial revocation** is where the obvious access is removed and the obscure access isn't: the SSO application is deleted but the standing API key, the firewall rule, the VPN account and the data they still hold all persist.

12 · When They Are Breached

At sixty vendors this is a matter of when. The organisations that handle it well are the ones that decided how they would handle it before the notification arrived.

Vendor breach response is genuinely harder than your own incident response, for a reason worth stating plainly: **you control none of the investigation.** You're dependent on their disclosure, their timeline, their competence and their legal posture. Your response therefore has to be built around what you can determine independently — what they could reach, what they did reach, and what you can cut off.

The sequence in TP-07 reflects that. Establish exposure from your own records rather than waiting for their assessment. Contain by revoking access, because that's entirely within your control. Hunt in your own environment for activity through their identity across the full window, not just since disclosure. Then work the notification and contractual questions in parallel, on the clock that started when you became aware.

> **THE QUESTION THAT SHORTENS EVERYTHING**
>
> "If this specific vendor were fully compromised tonight, what could the attacker reach through them?" If that has been answered in advance for every Tier 1 vendor and written down, your first hour is triage. If it has not, your first day is discovery — and the notification clock is running throughout.

PART FOUR

Runbooks

RUNBOOK TP-01

Vendor Intake and Tiering

PURPOSE	Classify a new relationship before it acquires access, so effort lands where the blast radius is.
TRIGGER	Any new vendor evaluation — attached to evaluation, not to the purchase order.
OWNER	Third-party risk. Business sponsor supplies the facts.
EVIDENCE	Register entry with tier, rationale, and named owner.

1. **Capture what actually matters, and refuse incomplete submissions.**
   What data they will hold or see, what systems they will touch, whether access is human or machine, what breaks if they are unavailable, and how easily they could be replaced.

2. **Tier on blast radius, never on contract value.**
   Per Appendix A. The payments processor may cost less than the furniture supplier; procurement thresholds are the wrong axis and the one most programmes drift into.

3. **Check for an existing relationship first.**
   The same vendor is frequently onboarded three times by three teams under three tiers. Consolidate to the highest.

4. **Assign a named business owner.**
   An individual, not a department. This is who recertifies access and who is called when the vendor is breached.

5. **Route by tier.**
   Tier 1 to TP-02 technical diligence and TP-03 contract negotiation. Tier 2 to evidence review. Tier 3 straight to registration with standard terms.

6. **Register before access is provisioned, always.**
   An unregistered vendor with live access is invisible to every subsequent control in this manual.

RUNBOOK TP-02

Technical Due Diligence

PURPOSE	Establish what a Tier 1 vendor's security actually looks like, beyond what they are willing to claim.
TRIGGER	Tier 1 onboarding, and on reassessment.
OWNER	Security, with third-party risk coordinating.
EVIDENCE	Assessment record, evidence reviewed, findings and conditions of approval.

1. **Read the independent evidence before sending any questionnaire.**
   Audit reports, penetration test summaries, certifications with their scope statements. Read the scope, a certification covering one product line tells you nothing about the one you are buying.
2. **Check the scope of every attestation against what you are actually consuming.**
   This is the most commonly skipped step and the most commonly wrong assumption.
3. **Observe what you can from outside.**
   External attack surface, certificate hygiene, exposed services, breach history, credential exposure. Cheap, independent, and occasionally decisive.
4. **Trace the data path concretely.**
   Where the data goes, who processes it, which sub-processors are involved, which jurisdictions. Feeds TP-10.
5. **Use the questionnaire to fill gaps only,**
   targeted at what the evidence did not answer. A short specific set produces better information than two hundred generic questions.
6. **Answer the blast-radius question explicitly and record it.**
   "If compromised, an attacker could reach X." This sentence is what makes TP-07 fast later.
7. **Approve with conditions where needed,**
   each with an owner and a date. Unconditional approval of a vendor with known gaps is how findings disappear.

RUNBOOK TP-03

Contract Security Negotiation

PURPOSE	Secure enforceable terms while you still have leverage.
TRIGGER	Before signature, and at every renewal.
OWNER	Legal drafts. Security specifies objectives. Business owner accepts residual gaps.
EVIDENCE	Clause checklist with achieved/not achieved, and risk acceptance for gaps.

1. **Engage before the commercial decision is effectively made.**
   Security arriving after the vendor has been chosen has advisory influence and no leverage.

2. **Work from objectives, not from your standard paper.**
   Appendix C lists what each clause must achieve. Vendors reject unfamiliar drafting and accept familiar drafting that achieves the same outcome.

3. **Fix the notification clock precisely.**
   Hours, measured from awareness rather than from confirmation. “Promptly” and “without undue delay” are unenforceable in practice.

4. **Get an evidence right where an audit right is refused.**
   Most vendors will not accept physical audit. Most will accept an obligation to provide reports, test summaries and answers on request. Take the achievable version rather than losing the whole clause.

5. **Attach consequences to the security requirements.**
   Termination rights or service credits. Requirements without remedy are aspirations.

6. **Secure sub-processor disclosure and change notification.**
   Without it, Chapter 8 and TP-10 are impossible.

7. **Get return-and-deletion with evidence.**
   This is what makes offboarding enforceable rather than a request.

8. **Record what you did not get, and have it accepted by name.**
   A gap accepted by a named business owner is a risk decision. An unrecorded gap is a surprise waiting for an incident.

RUNBOOK TP-04

Vendor Access Provisioning

PURPOSE	Grant the minimum access the workflow requires, in a form you can revoke in one action.
TRIGGER	Approved vendor requires access.
OWNER	Identity and platform teams. Security approves Tier 1 scope.
EVIDENCE	Access record (Appendix D), scope justification, revocation test result.

1. **Derive permissions from the workflow, not from an existing role.**
   Reusing a role that already exists is the most common source of over-provisioning, and it is invisible afterwards.

2. **Issue a dedicated identity per vendor.**
   Never shared across vendors or with internal services. Shared credentials make revocation destructive and attribution impossible.

3. **Prefer federated, short-lived credentials over static keys.**
   Where a static key is unavoidable, record where it lives, who holds it, and its rotation schedule — and treat those as commitments rather than intentions.

4. **Time-bound anything human.**
   Contractor and support access expires by default. Permanent human vendor access is the category most likely to outlive the relationship.

5. **Terminate the connection at a boundary you control**
   with its own policy enforcement, rather than inside the general estate.

6. **Enable vendor-specific logging and baselining**
   at provisioning time, per Chapter 7. Retrofitting it later rarely happens.

7. **Test revocation now, while nothing is wrong.**
   Confirm you can cut this vendor off in a single action and know what breaks when you do. For Tier 1 this is not optional — discovering the answer during an incident costs hours.

8. **Record it in the access register**
   with scope, expiry, owner and revocation method.

RUNBOOK TP-05

Vendor Access Recertification

PURPOSE	Confirm the access that exists is still the access needed — and remove what is not.
TRIGGER	Tier 1 quarterly, Tier 2 annually. Also on business-owner change.
OWNER	Business owner decides. Identity executes removals.
EVIDENCE	Campaign record with decisions, removals executed, and the removal rate.

1. **Enumerate what access actually exists,**
from the systems rather than from the register. The two differ, and the difference is the finding.

2. **Present permissions in business language, with recent usage attached.**
“Can read all customer records; last accessed 14 months ago” produces a different decision from a list of role names.

3. **Make removal the easy path.**
Default to revoke, require justification to retain. The inverse produces 99% renewal and measures nothing.

4. **Flag unused access explicitly.**
Anything with no usage in the period is a strong removal candidate regardless of what anyone believes it is for.

5. **Reconcile against active contracts.**
Access belonging to a relationship that has ended goes straight to TP-08. This reconciliation is how silent lapses are caught.

6. **Execute removals within a defined window and verify them.**
A recertification that decides to remove access and does not is worse than none, because the record now says it is gone.

7. **Report the removal rate.**
Near-zero removal across a large campaign means the process is decoration, and that is the number worth showing leadership.

RUNBOOK TP-06

Ongoing Monitoring and Reassessment

PURPOSE	Keep the risk picture current between scheduled reviews, and catch tier changes.
TRIGGER	Continuous for signals; scheduled reassessment by tier; event-driven on material change.
OWNER	Third-party risk, with security operations for the signals.
EVIDENCE	Signal log, reassessment records, tier change decisions.

1. **Define the reassessment triggers explicitly.**
Vendor breach, acquisition or ownership change, new data type, new integration, material incident, adverse external signal, tier change. Write them down; unwritten triggers are never fired.

2. **Watch for scope creep as a first-class signal.**
New integrations, expanded API scope, new data flows. This is how Tier 3 vendors silently become Tier 1 without anyone re-tiering them.

3. **Monitor Tier 1 vendor identities against their own baseline,**
per Chapter 7. Deviation from their normal shape is the only available signal when every action is authorised.

4. **Track external signals**
— breach disclosure, credential exposure, attack-surface change, financial distress. Financial distress is an underrated predictor of security degradation.

5. **Re-tier when the facts change,**
and route to full diligence if the new tier requires it.

6. **Report vendors overdue for reassessment**
by tier and owner, rather than reporting how many assessments were completed.

RUNBOOK TP-07

Vendor Breach Response

PURPOSE	Respond to a compromised vendor when you control none of the investigation.
TRIGGER	Vendor notification, or independent indication of compromise.
OWNER	Incident commander. Vendor owner manages the relationship track.
EVIDENCE	Timeline, exposure assessment, containment actions, notification decisions.

1. **Record the moment of awareness.**
   Your notification clocks start here, not when the vendor confirms details.
2. **Establish exposure from your own records first.**
   Access register, data flows, the blast-radius sentence from TP-02. Do not wait for the vendor's assessment — it will be slow, incomplete, and shaped by their counsel.
3. **Contain what is within your control: revoke.**
   Suspend the vendor's access pending clarity. This is the only lever you fully own. Weigh operational disruption explicitly and decide fast, hesitation here is expensive.
4. **Hunt in your own environment across the full plausible window.**
   Every action taken by the vendor's identity, going back well before disclosure. Their dwell time is your exposure window and their disclosure date is not the start of it.
5. **Invoke the contract deliberately.**
   Cooperation obligations, log provision, notification terms. This is the moment Chapter 5 pays for itself or reveals its gaps.
6. **Assume shared credentials and reused secrets are compromised.**
   Anything they held, and anything derived from it.
7. **Run notification assessment in parallel, not afterwards.**
   Your obligations may be triggered by their breach; the clock does not wait for their forensics.
8. **Decide the relationship's future separately and later,**
   once facts are in. Termination during the incident is occasionally right and usually premature.
9. **Post-incident, revisit the tiering and access design**
   for every comparable vendor, not only this one.

RUNBOOK TP-08

Vendor Offboarding

PURPOSE	Ensure the access ends when the relationship does — verified, not assumed.
TRIGGER	Termination, non-renewal, or discovery of a lapsed relationship with live access.
OWNER	Third-party risk drives. Identity, platform and data teams execute.
EVIDENCE	Completed offboarding checklist (Appendix F) with verification per item and deletion evidence.

1. **Treat it as a security event with an owner and a deadline,**
not a procurement close-out task. This reframing is the whole runbook.

2. **Enumerate every access path from the systems, not the register.**
Accounts, federated identities, API keys, OAuth grants, certificates, VPN accounts, firewall rules, allow-listed addresses, webhooks, shared mailboxes, physical access. The obscure ones are what persist.

3. **Revoke and verify each one individually.**
Confirm the credential no longer authenticates, do not accept that a ticket was closed. Partial revocation is the normal failure mode.

4. **Tear down the integration, not just the credential.**
Disabled integrations get re-enabled by someone who assumes they are dormant rather than terminated.

5. **Invoke return and deletion, and obtain evidence.**
Per the contract. Record what they confirmed and when.

6. **Rotate anything shared.**
Secrets they held, certificates they were issued, any credential that touched their environment.

7. **Remove network and email trust.**
Allow-list entries, domain trust, mail rules and connectors.

8. **Verify independently after a delay.**
Two weeks later, confirm no authentication attempts and no live routes. This is what catches what the checklist missed.

9. **Close the register entry with the completion record.**
Not before verification.

> **RUN THIS RETROSPECTIVELY FIRST**
>
> Before relying on this process going forward, run it against every relationship that ended in the last three years. Most organisations find live access for at least one vendor they no longer work with, and that finding is usually what secures the funding for everything else in this manual.

RUNBOOK TP-09

Concentration Risk Assessment

PURPOSE	Identify where multiple critical dependencies converge on one underlying provider.
TRIGGER	Annually, and on any Tier 1 onboarding.
OWNER	Third-party risk, with architecture.
EVIDENCE	Concentration map, substitutability assessment, exit position per critical service.

1. **Map dependencies by underlying provider, not by vendor name.**
   Four vendors on one cloud region is one dependency wearing four badges.

2. **Include the non-obvious shared layers.**
   Authentication providers, CDNs, DNS, payment rails, certificate authorities, and the security tooling that sits privileged across everything.

3. **Assess substitutability honestly per critical service.**
   Could you move, how long would it take, and has anyone tried? “We could not realistically move” is a legitimate and important answer.

4. **Identify single points of failure with no viable alternative**
   and escalate them as strategic exposures rather than vendor findings.

5. **Document the exit position for each critical service,**
   even where exit is unattractive. An unattractive documented position beats an undiscovered one.

6. **Report concentration to the board as a named list,**
   not a heat map. “Six of our nine critical services depend on one provider” is a sentence that changes decisions.

RUNBOOK TP-10

Fourth-Party and Sub-Outsourcing Review

PURPOSE	See past your direct vendors to the dependencies behind them.
TRIGGER	Tier 1 onboarding and annual review; on sub-processor change notification.
OWNER	Third-party risk.
EVIDENCE	Sub-processor list per Tier 1 vendor, with data flow and jurisdiction.

1. **Obtain the sub-processor list under the contractual right** secured in TP-03. Without that clause this runbook cannot be executed, which is why the clause matters.
2. **Establish which sub-processors touch your data specifically,** rather than accepting a generic corporate list.
3. **Record jurisdiction and data flow per sub-processor.** This is frequently where residency commitments quietly fail.
4. **Feed the results into TP-09.** Fourth parties are where concentration most often hides, several unrelated vendors depending on one small provider nobody has heard of.
5. **Assess whether your vendor manages them at all.** A vendor with no sub-processor governance has effectively extended your trust to companies neither of you has assessed.
6. **Act on change notifications rather than filing them.** A new sub-processor in a new jurisdiction is a reassessment trigger under TP-06.

PART FIVE

Templates and Reference

A · Vendor Tiering Criteria

Tier on the highest criterion met, not on an average. Adjust thresholds to your sector, but keep the axes.

Axis	Tier 1	Tier 2	Tier 3
Data held	Regulated, customer or payment data	Internal or commercially sensitive	None or public
System access	Privileged or production write	Limited or read-only	None
Operational criticality	Business stops or is impaired	Material but survivable disruption	Inconvenience
Substitutability	Hard or impossible to replace quickly	Replaceable with effort	Readily replaceable
Network position	Connectivity into the estate	Defined integration only	No connectivity

B · The Vendor Register

One row per relationship. If you are subject to a material outsourcing or ICT third-party regime, align the fields to the required register rather than maintaining two.

Field	Notes
Vendor / legal entity	The contracting entity, which is often not the trading name.
Service provided	In business terms.
Tier and rationale	Which criterion drove it.
Business owner	Named individual. Recertifies access, called on breach.
Data held / classification	Categories and sensitivity.
Access held	Cross-reference to the access record. Contract scope and actual access differ.
Blast radius statement	One sentence: if compromised, an attacker could reach X. Written at diligence, used in incident.
Sub-processors	From TP-10, with jurisdictions.
Contract dates and notice period	Renewal is your next leverage point.
Security terms achieved	Which clause objectives were secured, and which gaps were accepted by whom.
Substitutability and exit position	For Tier 1. Feeds TP-09.
Last assessment / next due	Overdue is the number worth reporting.
Status	Active / In offboarding / Terminated-verified.

C · Contract Security Clause Objectives

Objectives, not drafting. Your counsel produces the language; these are the outcomes worth insisting on.

Clause	Objective — and the weak version to reject
Breach notification	A fixed number of hours from awareness. Reject “promptly”, “without undue delay”, and clocks that start at confirmation.
Incident cooperation	Obligation to assist your investigation, provide logs and make personnel available. Absent this you depend on goodwill.
Audit or evidence right	Right to verify. Where physical audit is refused, secure an evidence-on-request obligation with a response deadline.
Minimum security requirements	Specific and testable, with a remedy attached. Requirements with no consequence are aspirations.
Data scope, location, retention	Exactly what, where, how long. Ambiguity becomes your regulatory problem.
Sub-processor disclosure	List on request, notification of change, right to object. Enables TP-10.
Return and deletion	On termination, with evidence, within a stated period. Makes offboarding enforceable.
Exit assistance	For Tier 1: obligation to assist migration, in a usable format, at defined cost. Negotiate before dependency exists.
Subcontracting limits	Restrictions on onward transfer, particularly across jurisdictions.

D · Vendor Access Record

Field	Notes
Vendor / integration name	
Identity used	Must be dedicated. Shared identities are a finding.
Credential type	Federated / short-lived / static key. Static keys need a rotation owner and date.
Permissions granted	In terms of what it can do, not role names.
Permissions actually used	From logs. The gap between granted and used is your reduction opportunity.
Data reachable	What this access can read or write.
Network path and termination point	Where it enters and what enforces policy there.
Expiry	Mandatory for human access.
Revocation method	The single action that cuts it off — and the date it was last tested.
Baseline established?	Per Chapter 7, for Tier 1.
Last recertified / by whom	

E · Vendor Breach Response Checklist

Phase	Actions
Hour one	Record awareness time. Pull register and access record. Retrieve the blast-radius statement. Convene owner, security, legal. Decide on immediate revocation.
Day one	Enumerate all access paths. Hunt across the full plausible window in your own logs. Invoke contractual cooperation. Begin notification assessment. Establish an update cadence with the vendor.
Week one	Rotate anything they held. Confirm exposure scope. Complete regulatory and customer notification decisions. Assess whether their remediation is credible.
After	Decide the relationship’s future. Re-tier. Re-examine comparable vendors. Update the blast-radius statement with what you learned.

F · Offboarding Checklist

Every line requires verification, not assertion. “Ticket closed” is not verification.

Item	Verified by	Method
Named user accounts disabled and deleted		Authentication attempt fails
Federated / SSO application removed		IdP configuration confirmed
API keys and tokens revoked		Call returns 401
OAuth grants and webhooks removed		Grant list inspected
Certificates revoked		Revocation confirmed
VPN / remote access removed		Connection attempt fails
Firewall rules and allow-list entries removed		Configuration diff
Integration torn down, not disabled		Configuration removed
Mail connectors and domain trust removed		Configuration confirmed
Shared secrets rotated		Rotation record
Data return and deletion evidenced		Written confirmation retained
Physical access and assets returned		Asset record
Independent recheck after two weeks		No auth attempts, no live routes

G · Board Reporting: Third-Party Risk

Measure	Current	Commentary prompt
Tier 1 vendors with a current assessment		Which are overdue, and who owns each?
Tier 1 vendors with a documented blast radius		Blank means the first day of a breach is discovery.
Vendor access removed at recertification (%)		Near zero means the process is decoration.
Relationships ended with verified offboarding		Any gap is live access to a former vendor.
Critical services with a single unsubstitutable provider		Name them. This is a strategic exposure, not a vendor finding.
Vendor incidents affecting us this period		What did we learn, and what changed as a result?

H · Where This Gets Hard

The judgement calls this manual does not resolve, because the right answer depends on your sector, your leverage and your appetite.

What do you do when you have no leverage? Much of Chapter 5 assumes you can negotiate. Against a hyperscaler, a market-standard SaaS platform or a monopoly data provider, you can't, you take their paper or you don't buy. The honest response is to treat the gap as an accepted risk recorded by name, and to compensate through access design and monitoring. Pretending you negotiated is worse than admitting you couldn't.

How much access reduction is worth the friction? Tightening vendor access breaks integrations, generates support tickets, and makes you unpopular with the business owner who just wants the tool to work. The reduction is real and so is the cost. Where that line sits is a genuine judgement and it differs by tier.

Do you terminate after a vendor breach? Instinct says yes. Frequently the vendor that just had a breach and responded well is now better run than the comparable vendor that hasn't been tested. Migration also carries its own risk and cost. This decision deserves more thought than it usually gets in the two weeks after.

How far down the chain do you go? Fourth parties are visible with effort. Fifth parties generally aren't. There is a point where the return on tracing dependencies falls below the cost, and it is lower than the frameworks imply. Decide where yours is deliberately rather than discovering it through exhaustion.

What do you do about concentration you can't fix? If six critical services depend on one provider and no alternative exists at acceptable cost, the assessment produces a finding nobody can close. That's still worth documenting — an accepted, visible, board-level exposure is a different position from an undiscovered one, and it is the honest end state for many firms.

I · Glossary

Blast radius statement	One sentence recording what an attacker could reach through a given vendor. Written at diligence, used in the first hour of a breach.
Concentration risk	Multiple critical dependencies converging on one underlying provider, often hidden behind different vendor names.
Fourth party	Your vendor’s vendor. Where concentration most often hides.
Material outsourcing	An arrangement whose failure would materially impair the firm. Triggers register, exit-planning and notification obligations in regulated sectors.
Register of information	The formal record of third-party arrangements required under ICT third-party regimes. Align your vendor register to it rather than keeping two.
Silent lapse	A relationship that ended without a formal termination, so offboarding never triggered and access persists.
Substitutability	How realistically a provider could be replaced, and how quickly. Usually the question underneath concentration.

USING THIS MANUAL

Take it apart. Rename the runbooks, delete the tiers that do not fit, and argue with Appendix H in particular, those are the places where your leverage and your sector should produce different answers than mine would.