Substantiate, Don't Assume

Why the Auditor's Opinion Matters More Than Ever in the Age of AI

Michael Borck

2026-06-28

Substantiate, Don't Assume
Why the Auditor's Opinion Matters More Than Ever in the Age of AI

Published by Michael Borck
Perth, Western Australia

ISBN: (to be assigned)

First edition, in development.

AI disclosure: This book was written using the methodology it describes. AI tools were used as thinking partners throughout the drafting, iterating, and refining process. The author reviewed, challenged, and took responsibility for every sentence. Full details of the tools used are in the front matter.

Companion website:
https://michael-borck.github.io/substantiate-dont-assume
Source: https://github.com/michael-borck/substantiate-dont-assume

Table of contents

Introduction: The Audit Mindset

> *Every organisation claims its information is secure. The auditor's job is to find out whether that claim survives contact with evidence.*

The polished policy and the unpatched server

Visit almost any company's website and you will find a security page. It is calm, confident, and covered in badges. "ISO 27001 certified." "SOC 2 Type II." "Enterprise-grade encryption." "Your data is safe with us." The page reads as a promise. Read it as an auditor and it reads as an assertion: a claim that has not yet met its evidence.

Tessera has one of those pages. It is a fast-growing Perth B2B software company, runs on AWS, serves customers across the region, and is chasing ISO/IEC 27001 certification. Its security page is impressive. Somewhere behind that page is a server that has not been patched in eleven months, an offboarding step that never runs, and a shared admin account four people know the password to. Or maybe not. That is the point: you do not know yet, and neither does Tessera's board.

This book is about closing the gap between the claim and the evidence. It teaches the discipline that does it: *substantiate, don't assume.*

What information security auditing actually is

Information security auditing is not penetration testing and it is not security monitoring, though it borrows from both. A penetration tester breaks in to show that it can be done. Monitoring watches systems continuously and alerts on trouble. An auditor does something different

and, to many people, less glamorous: an auditor asks an organisation to *prove* that the controls it claims to rely on are actually present, actually working, and actually producing the evidence that says so.

The product of an audit is not a list of vulnerabilities. It is an **opinion**: a defensible, evidenced judgement about whether an organisation's information security arrangements do what they say they do. That opinion is the thing stakeholders cannot get any other way. A vendor will always say it is secure. A standard will always say what secure should look like. Only an independent auditor says whether the two match, here, now, against the evidence.

The founding maxim of the profession captures this in three words.

Trust, but verify

"Trust, but verify" is older than information security; it is the operating principle of any assurance function, from financial audit to nuclear inspections. It does not mean "be suspicious of everyone." It means: take a claim seriously enough to test it. Trust is the starting point, because without some trust there is no engagement at all. Verification is the work, because trust that is never tested is just hope.

The auditor's relationship to the organisation is therefore productive, not adversarial. You are not the security police, and the best auditors are not internal affairs. You are a guest the organisation has invited to find the gaps it suspects but cannot see. That only works as a relationship: open, transparent, and built on trust that runs both ways. Staff who fear you will hand you the polished version. Staff who trust you will show you the real one.

Good auditors behave accordingly long before fieldwork begins. They meet first, run a quick paper assessment so they arrive informed, share the schedule ahead of time, and stay flexible enough to reassess as the picture changes. The engagement is a collaboration with a deadline, not a raid.

To do this credibly you must be willing to find that the polished policy does not match the operational reality, and to say so, clearly, with evidence. That willingness has a name.

A snapshot, not a guarantee

An audit opinion is a judgement about a moment. You gather evidence across a window of days or weeks, and you report what that evidence showed. The environment does not freeze while you write. Controls drift,

staff turn over, configurations change, and a system that was compliant on Thursday can be breached on Friday. Your opinion does not guarantee the future. It guarantees that, at the time you looked, on the evidence you gathered, the claim held (or did not).

This is why a defensible opinion has to be more than correct. It has to be **fair, repeatable, and ethical**, and it has to be documented well enough to serve as evidence itself, sometimes years later, in a board meeting, a regulator's inquiry, or a court. Another auditor following your working papers should be able to reach the same conclusion. A stakeholder reading them should be able to see how you got there.

Which is also why pressure is the auditor's real test. A client facing a deal deadline, a certification cutoff, or an embarrassed executive will want you to sign off on "compliant" before the evidence is in. The professional answer to that pressure is the same as the answer to every other shortcut in this book: substantiate, don't assume. An opinion given under pressure, without evidence, is not an efficiency. It is a liability you have stamped your name on.

Substantiate, don't assume

Substantiate, don't assume is the spine of this book and the habit it tries to build. Every control claim is an assertion until evidence supports it. A policy on the intranet is an assertion. A dashboard showing green checks is an assertion. A manager's assurance that "we do that" is an assertion. None of them is evidence until you have seen it work, on terms you set, in a form you can defend.

This sounds obvious. In practice it is hard. The most dangerous control is the one that looks perfect on paper, because it lowers your guard. Tessera is written to be plausible: impressive in its marketing, rougher underneath, exactly like the real organisations you will audit. Good auditing does not assume the worst. It does something more disciplined: it substantiates what is actually there, names what is missing, and lets the evidence set the opinion.

By the end of this book you will have done exactly that, one week at a time, against Tessera.

The auditor's moat in the age of AI

Here is the harder question this book takes seriously: *what is a human auditor actually for when a machine can draft the work?*

The answer is not "the same thing, faster." The answer reshapes the job.

The book takes a deliberately pragmatic stance on AI, steering clear of both alarmism and hype, and lands it on the auditor's desk. Four moves make the argument.

The technical craft is being commoditised

Listing ISO/IEC 27001:2022's 93 controls. Mapping them to NIST CSF 2.0. Drafting a test procedure for access reviews. Writing up a finding in the standard format. This is the technical craft of auditing, and it is exactly the kind of work generative AI is rapidly commoditising. An AI can enumerate the controls faster and more completely than any junior auditor. If the only thing an auditor does is *produce* that list, the auditor has no edge. Every organisation's auditor can now produce the same list.

Variation is the moat

If an AI is equally good at running every company, then by definition there is no variation between companies that use it generically, and no variation means no competitive edge. The edge comes from variation: human agency, individual judgement, a genuine point of view about what matters. For an auditor, that variation shows up as **professional scepticism**: the willingness to look at a polished policy and ask whether the evidence supports it, to decide which of a hundred small gaps actually matters, to form a defensible opinion rather than a confident average. The machine produces the draft. The auditor supplies the judgement. That judgement is the moat.

From drafter to evaluator

As AI moves from chatbots to agentic systems that can run procedures on their own, the auditor's job shifts: from creator of the "shitty first draft" to **expert evaluator and editor** of AI-generated work. You will read AI-drafted risk assessments, AI-drafted control mappings, AI-drafted findings, and your value will be in spotting where they are generic, where they confabulated compliance, and where the nuance they smoothed over was the whole point. This book makes that discipline explicit and teaches it. The course's assessments already embody it: you are required to document where you *disagreed with or expanded on* an AI's analysis.

Augmentation, not automation

The stance is proactive. Use these tools for **augmentation and personalised learning**, not passive automation that quietly atrophies the critical thinking the profession depends on. Efficiency is never the goal; *defensible judgement* is. If you use AI to skip the thinking, you have done

the work wrong, no matter how good the output looks. The companion book, *Conversation, Not Delegation*, teaches the craft of working with AI well; this book teaches what you must verify once the drafting is done.

> ⚠ Watch out: your own AI hygiene
>
> The AI you use is a third party. Do not paste Tessera's evidence, configurations, logs, or customer data into a consumer AI tool. That is a disclosure you cannot take back, and it may breach the engagement's confidentiality clauses and the Privacy Act. Use the approved or enterprise tooling your firm provides, or a locally hosted model. Treat every prompt as if it could become someone else's training data, because it might. The same scepticism you apply to Tessera's AI applies to your own: the tool is a thinking partner, not a safe drop box for client secrets.

This premise is the thread that runs through every chapter.

Who this book is for

This is the core text for **ISYS6018: Information Security Audit and Control** at Curtin University, and it is written for the student taking that unit: someone learning to audit against international standards for the first time. You do not need to be an auditor already. You need to be willing to form opinions and defend them with evidence.

It is also for any practitioner, new or experienced, who wants a readable, opinionated spine for the discipline of information security auditing, and a clear stance on where the human auditor fits once the machines can draft. If you audit, manage auditors, or rely on audit opinions, the argument here is for you.

How this book works

The book is one engagement told twelve ways. You audit **Tessera**, a fast-growing B2B software company built on AWS and chasing ISO/IEC 27001 certification. Each chapter is one topic, a single weekly question from *Why?* through *Risk?*, *Which?*, *Scope?* and on to *Report?* and *Conclude?*. **Chapter N = Topic N, delivered in Week N of the course**: read the chapter for a topic in the week you live that topic.

The arc has two acts:

Act	Weeks	What it does	Ends in
I · Engage	1–6	Build the case against Tessera with *limited access*	A preliminary readiness opinion (desk audit)
II · Prove	7–12	*Full access granted*, move from suspicion to proof	A defensible certification-readiness opinion

Act I forms judgement from documents alone. Act II opens with full auditor access (interviews, configurations, logs, incident records) and drives to evidence. The break between the acts is the break between suspicion and proof.

Running through both acts is the **Evidence Locker**: a cumulative working-paper file you build one artefact at a time. Each chapter ends with an Evidence Locker prompt, the one thing to lock in this week. By the final week the locker is not a study aid; it *is* the audit.

The companion site

Tessera is not just a name in this book. It is a live, simulated company at **tessera.locoensayo.org**, part of the *LocoEnsayo* rehearsal environments. Their tagline is the right one for auditing: *rehearse the real world; the cast is AI; the judgement is yours.* The site is open all hours, with no password, so you can follow along wherever and whenever you read. By the end of the book you will have audited it.

Two things make Tessera worth auditing, and both are deliberate. Its people are AI characters you can interview, each with their own role, knowledge, and gaps, like the staff of any real company. And its document library is full, not curated. Not every document is evidence. Not every employee knows the answer. Some files are red herrings that look relevant and lead nowhere. Deciding which document to test and which person to ask is an audit skill in its own right, and Tessera gives you room to practise it. A scenario engineered to hand you the obvious evidence teaches you nothing. A messy, realistic one teaches you to find it.

Every chapter's Tessera Case narrates the worked example; the companion site is where you run your own.

The companion book

Conversation, Not Delegation (https://michael-borck.github.io/conversation-not-delegation/) is the companion on method. Where this book teaches the audit discipline (the *what we verify*), that book teaches *how to work with the machine well*: the conversation loop, staying in the loop, compensating for AI's predictable weaknesses. The two are complementary. Every chapter here assumes you are using AI, and assumes you are using it the way that book describes: as a thinking partner, not a delegate.

Conventions used in this book

Every chapter carries the same set of recurring features so you learn to recognise them. Each is rendered differently.

> **AI Field Note**
>
> **Blue boxes** show AI used as a thinking partner for the chapter's task: a worked example against Tessera, including, crucially, where the AI got it wrong and how the auditor corrected it. This is the book version of the course's weekly AI Field Notes task.

> **Auditing AI**
>
> **Red boxes** apply the chapter's concept *to AI itself*: which controls an AI deployment falls under, how you audit an AI-assisted control, what evidence an AI system produces. These appear where the topic demands.

> **Evidence Locker: lock this in**
>
> **Green boxes** are the weekly prompt: the one artefact to add to your cumulative working-paper file this chapter. By Week 12 the locker *is* the audit.

Tessera Case

Shaded case boxes carry the running Tessera engagement, the concrete artefact each chapter lands on. Tessera is written to be plausible: impressive on paper, rougher underneath. Good auditing does not assume the worst. It substantiates what is there.

You will also see standard callouts for emphasis: **Watch out** (yellow) for

traps that look right but are not, plus occasional **Key idea** and **Try this** boxes.

How this book was written

This book was written using the methodology its companion describes. AI tools were used as thinking partners at every stage: drafting, stress-testing, and being corrected. The relationship between author and AI in producing this book is exactly what the book argues the auditor's relationship to AI-generated work should be. The machine produces the draft; the author supplies the judgement, the evidence, and the defensible opinion. Every sentence reflects the author's judgement. The AI made the work faster. It did not do the thinking.

A note on where this book is at

This is an open-access book in active development. The Introduction and Chapter 1 are the vertical slice: they exist to prove the voice and the recurring-feature format before the remaining eleven chapters are drafted. The online edition is always the most current. The standards targeted (ISO/IEC 27001:2022 and NIST CSF 2.0) are fixed from the outset; where standards drift, the book handles it by edition, not errata.

Part I

Act I · Engage: Build the Case

Chapter 1

Why? The Auditor's Mandate and Ethics

Why does Tessera need an audit, and why should they trust you?

Tessera Case: The engagement opens

The email arrives on a Monday. Tessera's CTO, racing toward an enterprise deal that requires ISO/IEC 27001 certification, has engaged your firm for a certification-readiness audit. The tone is confident: "We're pretty much there, we just need the paperwork signed off." That sentence is your first piece of evidence, and it is not the one the CTO thinks it is. An organisation that believes it is "pretty much there" is an organisation that has not yet tested whether it is. Your engagement opens with a question, not an answer: *why does Tessera need this audit, and why should they trust the person delivering the opinion?*

1.1 The auditor's mandate

An auditor does not audit by accident. You audit because someone with the authority to ask for assurance has engaged you to provide it, on agreed terms, against agreed criteria. That is the mandate, and it has three parts that matter from day one.

You are engaged by a party that wants the opinion. In Tessera's case, that is a board and a CTO chasing certification. You audit **against criteria**: a standard or framework that defines what "good" looks like, here ISO/IEC 27001:2022. And you deliver an **opinion** on whether the subject matter

(Tessera's information security management system) conforms to those criteria, supported by evidence you gathered on terms you set.

Notice what the mandate is not. It is not a licence to poke around wherever you like. It is not a guarantee that you will find wrongdoing. It is a scoped, criteria-bound engagement whose entire value rests on one quality you bring to it: the credibility of the person giving the opinion. That credibility has a name, and it is the rest of this chapter.

1.2 Three hats: counsellor, partner, investigator

Auditors wear three hats, sometimes in the same meeting, and the skill is knowing which one the moment needs.

The **counsellor** helps the organisation understand its obligations before an opinion is formed. A client asks, "Do we really need separate incident-response and business-continuity plans?" You explain the standard, the intent, and the trade-offs. You are advising, not judging, and you must remember the difference, because advice given today shapes the evidence you evaluate tomorrow.

The **partner** works *with* the organisation to make the audit productive. You coordinate access, agree a realistic programme, and treat Tessera's staff as people running a business, not suspects to be interrogated. Most audits fail on logistics, not on findings.

The **investigator** is the hat people picture: following the evidence where it leads, asking the uncomfortable question, refusing to accept "we do that" without proof. This is where professional scepticism lives, and it is the hat this book spends the most time on.

The three hats share one requirement: whichever you wear, your judgement has to be yours, and it has to be defensible. That brings us to the foundation the whole profession stands on.

1.3 Independence and objectivity

An audit opinion is only worth what the auditor's independence is worth. If the opinion can be bought, pressured, or quietly shaped by self-interest, it is worthless, and everyone who relies on it is misled.

Independence is the *fact* of being free from relationships and interests that could compromise the opinion. **Objectivity** is the *state of mind*: the discipline to judge evidence on its merits, without bias, even when the answer is inconvenient. Independence is something an outsider can check;

objectivity is something only you can guarantee, on every engagement, every day.

The profession catalogues the threats so you can spot them.

> ⚠ Watch out: threats to independence that look reasonable
>
> The dangerous threats are the ones that feel like normal business. A friendship with the CTO. A consulting contract that pays well and renews if the audit goes smoothly. A prior role inside the organisation. A desire to land next year's engagement. None of these is a crime. Each of them is a pressure on the opinion, and each requires a safeguard, or a refusal of the engagement, before you begin.

The safeguards are familiar: rotate the engagement team, separate the audit from any consulting work, document the threats you identified and how you addressed them, and walk away when the threat cannot be reduced to an acceptable level. The discipline is not to pretend you have no interests. It is to be honest about them and act accordingly.

1.4 The ISACA Code of Ethics

Information security auditors work within a professional code. ISACA's Code of Ethics is short and worth knowing by heart, because it is the standard your professional conduct will be measured against. Its obligations come down to a few commitments: support the organisation's legitimate objectives while maintaining your independence; act with due care and competence; keep information confidential and protect privacy; and uphold the reputation of the profession.

Read it once and it sounds like generic good behaviour. Read it on the day a client asks you to soften a finding so a deal closes, and you find out what it actually costs. The code is not a poster. It is the answer you are expected to give when the pressure arrives, and the reason "no" is sometimes the only professional response.

1.5 Why would anyone invite an audit?

Step back to the question Tessera's CTO has not fully asked: *why pay someone to come and find your problems?*

The honest answer is that an organisation invites an audit because the opinion is worth more than the cost. To someone. For Tessera, certification unlocks enterprise customers whose procurement teams will not sign

without it. For a board, an independent opinion is protection against the day something goes wrong: evidence that oversight was exercised, not neglected. For regulators, it is compliance made visible. And, less flatteringly but just as truly, organisations invite audits because the people inside them already suspect the gaps and want a credible outsider to name them.

The auditor's job is to be worth that trust: to deliver an opinion that is genuinely independent, even when the organisation that paid for it hopes for a different answer. That is the whole deal. Tessera is not paying you to say they are ready. Tessera is paying you to find out whether they are.

1.6 Three kinds of control

Before the engagement goes further you need the vocabulary every audit uses. Controls (the safeguards an organisation puts in place to manage risk) come in three kinds, and knowing which kind you are looking at tells you what evidence will prove it works.

Preventive controls stop something going wrong. An access-review process that removes leavers' accounts before they can be misused. A firewall rule. Mandatory training before a user gets privileges.

Detective controls notice that something has gone wrong, after the fact. A log that records a failed login. An alert when an admin account is created out of hours. A reconciliation that finds a discrepancy.

Corrective controls fix something once it is found. An incident-response plan. A backup you restore from. A patch you push out.

Most real defences are a chain of all three, because no single control is reliable on its own. A preventive control fails silently; you only find out it failed because a detective control caught it; you only recover because a corrective control exists. When you audit, you are testing the whole chain, and you are always asking the question that separates a control that *exists* from one that *works*.

Tessera Case: Where Tessera's hats show

In your opening week you wear all three hats at once. As **counsellor**, you walk Tessera's CTO through what ISO/IEC 27001:2022 actually requires, and watch the confidence drain when "pretty much there" meets "93 controls across four themes." As **partner**, you agree how the engagement will run and what access you will get (limited, for now). As **investigator**, you file away the CTO's opening claim ("we're basically ready") as the first assertion the evidence will later confirm or refute. Every chapter from here tests that claim a different way.

AI Field Note: Exploring the control categories

Ask an AI to "explain preventive, detective, and corrective controls with examples for a SaaS company on AWS." It will give you a competent, well-structured answer in seconds, and that is exactly the trap. The examples will be *generic*: "firewall rules," "log monitoring," "incident response plan." Correct, and useless, because they describe every company.

This is where you, the auditor, add the variation the machine cannot. Push back: "Give me a *preventive* control specific to a multi-tenant SaaS where customers configure their own IAM roles, and tell me how it would fail silently." Now the answer has to engage with Tessera's reality, not a textbook. The AI's first cut was a draft; your refinement is the work that matters. You have just done, in miniature, the drafter-to-evaluator shift the Introduction described.

Auditing AI: Can an AI be independent?

Independence is a fact and objectivity is a state of mind, and an AI has neither. It has no relationships to disclose, true, but it also has no stake, no accountability, and no capacity to *refuse* an engagement when a threat cannot be managed. More subtly, an AI trained on an organisation's own data, or embedded in the tooling an organisation pays for, has interests of a new kind: commercial, algorithmic, and invisible.

So can an AI deliver an audit opinion? It can *produce* one. It cannot *stand behind* one, which is the entire point of an opinion. As AI-assisted controls and AI-generated evidence spread through the estates you audit, the question stops being theoretical. We return to it in Chapter 7, when we ask whether you can trust a log an AI produced.

Evidence Locker: lock this in (Week 1)

Open your Evidence Locker with the engagement's foundation. Record: the **mandate** (who engaged you, against what criteria, to deliver what opinion); the **independence threats** you identified for this engagement and the **safeguards** applied; and your one-line restatement of Tessera's opening claim, the assertion the rest of the engagement will substantiate or refute. This is not note-taking. It is the first entry in a defensible audit trail.

1.7 The question, answered

Why does Tessera need an audit? Because certification, customers, and a board all need a credible opinion they cannot give themselves, and because the people inside Tessera already suspect the gaps. Why should they trust *you*? Because your mandate is clear, your independence is real, your ethics are not for sale, and your discipline is to substantiate every conclusion with evidence rather than assume it.

That is the audit mindset, stated. The next eleven chapters enact it, one question at a time, until the polished policy has met its evidence and you have formed an opinion you can defend.

Next week, the first real question: *what could go wrong, and how do we measure it?*

Chapter 2

Risk? Foundations of Risk Management

What could go wrong, and how do we measure it?

Tessera Case: "We've covered the big risks"

Day three of the engagement, an email lands from Tessera's CTO. Attached is a polished spreadsheet: the draft risk register, thirty-odd rows, colour-coded green and amber, not a red cell in sight. "We've covered the big risks," he writes. "Let me know if you need anything else for the risk assessment." That attachment is your first real artefact of the engagement, and the claim stapled to it is, as always, the thing to test. A register that rates everything medium is not evidence of low risk. It is evidence of optimism, or of a process that has never had to defend its ratings in front of a board. Your job this chapter is to find out which.

2.1 What risk actually is

Risk is the effect of uncertainty on objectives. That is the ISO definition, stripped to its bones, and it is better than it sounds. Two words do the work. **Uncertainty** means you do not know whether something will happen, only that it might. **Objectives** means risk is always measured against something an organisation cares about: revenue, reputation, customer trust, compliance, availability. A threat that cannot touch any objective is not a risk worth managing. A threat that can is, no matter how unlikely it feels on a calm day.

Notice what risk is *not*. It is not a threat, and it is not a vulnerability, though both create it. A threat is an actor or event that could cause

harm: a ransomware crew, a flood, a careless insider. A vulnerability is a weakness that a threat could exploit: an unpatched server, a shared admin password, an offboarding step that never runs. Risk is what you get when a threat meets a vulnerability and the result could damage an objective. Tessera's unpatched server is not a risk by itself. The ransomware crew that could reach it, and the revenue loss and certification delay that would follow, is the risk.

This three-part model, threat plus vulnerability plus impact on an objective, is the engine under every risk register you will ever read. Hold it in mind. Most of the optimism you will spend this chapter correcting comes from someone quietly dropping one of the three: naming a threat with no vulnerability, a vulnerability with no credible threat, or an impact with no objective attached.

2.2 The risk lifecycle

Risk management is not a document you produce once. It is a cycle you run, repeatedly, for as long as the organisation exists. ISO/IEC 27005 lays the loop out, and NIST SP 800-30 walks the same path in slightly different language. The shape is the same.

You **establish context**: what is in scope, which objectives matter, what appetite for risk the organisation has set. You **identify** risks: the threats, the vulnerabilities, the scenarios where the two meet. You **analyse** each one, estimating likelihood and impact. You **evaluate** them, ranking the analysed risks against your criteria to decide which deserve treatment. You **treat** them, choosing from the four options we will come to. And you **monitor and review**, because risks change, controls drift, and a register that was right in January is stale by July.

The cycle matters more than the artefact. A risk register is a snapshot of one trip around the loop. The organisation that updates it annually, to satisfy an auditor, has a document. The organisation that runs the loop continuously has risk management. When you audit, part of what you are testing is which of those Tessera is.

2.3 Likelihood × impact

Every quantitative or semi-quantitative risk rating comes down to the same arithmetic: **risk is a function of likelihood and impact**. How likely is this to happen, and how bad would it be if it did? Score each on a scale, combine them, and you have a rating you can rank, plot on a matrix, and defend.

Likelihood is usually a blend of two things the register rarely separates:

how often the threat *tries*, and how likely it is to *succeed* given the controls in place. A phishing campaign hits Tessera's staff hundreds of times a week; that is high threat activity. Whether it becomes an incident depends on whether the training, the mail filter, and the MFA hold. Conflating the two is how registers end up with every externally-facing risk marked "high likelihood." It is also how they end up low: rating the attempt frequency and then quietly assuming the controls work, without ever testing them.

Impact is where the optimism lives. Organisations under-rate impact because they measure the obvious cost and miss the cascading one. A lost laptop is a replacement cost. A lost laptop holding an unencrypted customer list is a breach notification, a regulator's interest, a churned enterprise account, and a line item on next quarter's board report. The first number is small. The second is the one that ends careers. Your job, when you read an impact rating, is to ask which of those two numbers the author actually wrote down.

The matrix is seductive because it produces a tidy colour. Green, amber, red. Treat the colour as a hypothesis, not a conclusion. Two risks sitting in the same amber cell can be wildly different in what they demand of you, because the matrix has flattened a judgement into a square. We will come back to why, in the Watch out at the end.

2.4 The risk register and the risk owner

The risk register is where the lifecycle leaves a paper trail. Every identified risk gets a row: a description, the assets and objectives it threatens, a likelihood and impact rating, the treatment chosen, the controls that implement the treatment, and a residual rating once those controls are in place. Done well, it is the most honest document in the organisation. Done poorly, it is theatre.

There is one column that gets neglected more than any other, and it is the one that determines whether the register functions: **the risk owner**. Every risk needs a named individual, not a team, not "the business," accountable for keeping it managed. Ownership is not the same as blame. It is the answer to the question "who will notice when this risk changes, and who will act?" A risk without an owner is an orphan. It will drift until it becomes an incident, and then someone will discover that nobody owned it, which is exactly when ownership would have mattered.

ISO/IEC 27001:2022 makes this explicit in its organisational controls: roles and responsibilities for information security must be defined and assigned. The risk owner is the sharpest test of whether that control is real. When you audit Tessera's register, look at the owner column first. If it is full of "IT," "Engineering," and "Management," you have found

your first finding before you have rated a single risk.

Tessera Case: The preliminary Tessera risk matrix

Here is the register as it arrives, trimmed to the rows that matter. Looks complete. Looks disciplined. Read it as the auditor, not the recipient.

Risk	Likelihood	Impact	Rating	Owner
Tenant data leakage between customers in multi-tenant platform	Low	Medium	**Medium**	Engineering
Ransomware via phishing of staff	Medium	Medium	**Medium**	IT
Insider misuse of production DB access	Low	Medium	**Medium**	Management
AWS region outage, platform unavailable	Low	High	**Medium**	Engineering
Loss of certification due to failed audit	Low	High	**Medium**	CTO

The pattern gives the game away before you read a single rationale. Five risks, five different scenarios, one universal verdict: medium. The register is not measuring risk. It is averaging anxiety. Each rating is defensible in isolation and indefensible in aggregate, because nobody has asked whether the controls that justify those "Low" likelihoods have ever been tested, or whether the "Medium" impacts account for the enterprise contracts and the Privacy Act obligations that make a single breach a board-level event. This matrix needs re-rating, and the rationale needs to be written down in a form another auditor could follow. That is this chapter's Evidence Locker.

2.5 Treatment: avoid, mitigate, transfer, accept

Once a risk is evaluated, you treat it. The vocabulary is small and worth memorising, because every control you audit for the rest of this book is implementing one of these four decisions.

- **Avoid.** Eliminate the risk by removing the source. Decommission the system, switch off the feature, exit the market. Decisive, often expensive, sometimes the only honest answer.
- **Mitigate.** Reduce likelihood or impact by putting controls in place. Most of information security is mitigation: patching, MFA, encryption, access reviews, logging, training. Mitigation rarely removes a risk; it lowers it to a level the organisation can live with.
- **Transfer.** Move the risk, or its cost, to another party. Cyber-insurance is the obvious example. A managed-service contract that carries an SLA is another. Transfer is never total: insurance covers the money, not the reputational damage, and a contract does not return your customers' data. You have transferred the financial consequence, not the underlying exposure.
- **Accept.** Acknowledge the risk and choose to live with it, explicitly, with a reason, signed by someone with the authority to accept it. Acceptance is legitimate. *Implicit* acceptance, where a risk is simply never treated and never owned, is the most common failure mode in the registers you will read.

The choice is rarely pure. A real treatment plan combines them: mitigate hard, transfer what remains through insurance, and accept the residual, with the owner's name on the acceptance. The discipline is to make the choice deliberately and record it. A control with no treatment rationale attached is a control someone installed on instinct and never justified.

2.6 Residual risk

Here is the distinction that separates people who have read a standard from people who understand one. **Inherent risk** is the risk before controls: the raw exposure, threat meeting vulnerability with nothing in the way. **Residual risk** is what remains *after* treatment. The whole point of mitigation is to move a risk from its inherent level down to a residual level the organisation has decided is acceptable.

Residual risk is where auditors live. Inherent risk tells you about the world. Residual risk tells you whether the controls are doing their job. A risk that sits at "High inherent, High residual" has controls that are not working, or do not exist, or exist only on paper. A risk at "High

inherent, Low residual" has controls you now need to test, because the entire justification for that comfortable residual rating rests on them. Every residual rating in a register is a claim about a control. Your job, for the rest of the engagement, is to substantiate those claims one by one.

⚠ Watch out: residual is not eliminated, and one rating can hide a swarm

Two traps, both common, both dangerous.
First, treating *residual* as if it meant *eliminated*. It does not. Residual risk is the risk that remains after treatment, and it remains because no control is perfect, because controls fail, and because the threat landscape moves. A register full of "residual: low" is not a register of solved problems. It is a register of bets on controls, and bets can lose. If the organisation talks as though treatment closed the risk, the treatment is probably aspirational.
Second, the single-point likelihood × impact rating hiding aggregation and velocity. One phishing email rated "medium" tells you almost nothing, because phishing is not a one-off; it is a continuous stream, and the relevant question is the likelihood that *at least one* succeeds over a year, not the likelihood of any single attempt. Likewise, a risk whose impact compounds fast (a tenant-data leak that spreads as customers discover it, a ransomware payload that moves laterally in minutes) cannot be captured by a static impact score. A matrix cell is a still photograph. Real risk has a frame rate. When you re-rate, ask not just "how likely, how bad" but "how often, and how fast does it get worse."

2.7 Two standards, one loop: ISO/IEC 27005 and NIST SP 800-30

You will meet two risk standards repeatedly, and it helps to know how they differ and why it rarely matters which one you reach for.

ISO/IEC 27005 is the information security risk management standard that sits beside ISO/IEC 27001. If 27001 tells you to *do* risk management (it is mandatory: Clause 6.1.2 requires a risk assessment and Clause 6.1.3 requires a risk treatment plan, and the Statement of Applicability is built directly from the treatment decisions), then 27005 tells you *how*, in process terms: context, assessment, treatment, acceptance, communication, monitoring. It is the method document for the ISMS. For Tessera, chasing 27001 certification, 27005 is the natural home for the methodology their register should follow.

NIST SP 800-30 is the US federal equivalent, "Guide for Conducting Risk Assessments." It is more granular on the assessment step itself: it breaks the analysis into identifying threat sources and events, vulnerabilities, and predisposing conditions, then determining likelihood, then impact, then risk. It is the document to reach for when you want a rigorous, repeatable walk through a single assessment.

Both describe the same loop. Both use likelihood and impact. Both feed treatment. The honest difference is tone and lineage, not substance. An auditor who can read a Tessera register and map its columns to either standard, fluently, has the skill; the partisan who insists one is correct and the other is not has missed the point. The risk-based audit approach does not care which standard produced the register. It cares whether the register survives scrutiny.

2.8 The risk-based audit approach

Everything in this chapter converges on a method. The **risk-based audit approach** uses risk to decide where the audit looks hardest. You do not test every control with equal effort, because no engagement has the time or the budget for that, and because uniform effort is wasted effort: it spends the same care on a low-impact nicety as on a load-bearing control that protects the crown jewels.

Instead you let the risk register guide the programme. High residual risks get deep testing: you trace the controls that justify the low rating back to evidence, and you test that they actually work. Medium risks get proportionate testing. Low risks may be confirmed-present and moved on. The audit effort follows the risk, not the alphabet.

There is a second, subtler dimension. You are not only using risk to plan the audit; you are auditing risk management itself. Is the register complete? Are the owners real? Do the ratings survive a challenge? Is the cycle actually running, or was it last touched the week before you arrived? For an organisation certifying to ISO/IEC 27001, a sound risk management process is not optional; it is the spine of the whole management system. An opinion on Tessera's readiness is, in large part, an opinion on whether their risk management is real. You cannot certify an organisation that manages controls but not risk.

This is also where risk connects to everything that follows. The frameworks you will meet next chapter exist to give you a structured way to treat the risks this chapter identifies. The audit plan in Chapter 4 is built on the risk-based priorities you set here. Risk is the load-bearing concept of the entire engagement.

AI Field Note: Drafting the matrix, then fixing it

Ask an AI to "draft an information security risk register for a multi-tenant B2B SaaS company on AWS seeking ISO 27001 certification." It returns a clean, well-structured register in seconds, with sensible columns and plausible risks. That is the trap the Introduction warned you about: it is correct, and it is generic.

Look at the row every AI writes the same way: *cross-tenant data leakage*. The AI rates it **Medium**: medium likelihood, medium impact. Its reasoning, if you push it, is that multi-tenant platforms use logical isolation, encryption, and IAM, so a leak is possible but contained, and the impact is a single customer's data. That reasoning is sound for a *generic* SaaS. It is wrong for Tessera, and here is the judgement the machine cannot supply.

Tessera's enterprise contracts carry data-isolation clauses: a cross-tenant leak is not a breach of one customer's data, it is a breach of a contractual guarantee, with liquidated damages in several of them. Tessera is also subject to the Australian Privacy Principles and the Notifiable Data Breaches scheme, so the same leak is a mandatory notification and, depending on the data, a reportable incident affecting multiple parties. The impact is not "one customer." It is contracts, regulators, and reputation, simultaneously. Re-rate it **High**, and write the rationale down: *impact elevated from Medium to High on the basis of multi-party contractual exposure and mandatory notification obligations under the APPs, which the generic rating did not consider.*

That is the move, repeated across the register. The AI drafted in thirty seconds. You supplied, in an hour, the customer-contract context, the regulatory context, and the operational reality that turn a generic register into Tessera's register. The drafter-to-evaluator shift, on a single artefact. Lock the re-rated version, with rationale, in your Evidence Locker.

Auditing AI: Rating the risk of the model itself

The likelihood × impact framework was built for threats that steal data or break systems. It applies, with a twist, to AI systems themselves, because an AI deployment is a new source of risk that an organisation can own, treat, and rate like any other.

Treat the model as an asset with its own threats and vulnerabilities. A threat source might be a prompt-injection attack, a training-data extraction attempt, or simply drift as the model is retrained. The vulnerabilities are familiar to anyone who has worked with these

systems: they confabulate, they leak training data under the right queries, and they fail in ways that look confident rather than loud. The impact? A Tessera support chatbot that hallucinates a refund policy it just invented is a customer-trust risk. An AI assistant that summarises customer tickets and leaks one tenant's details into another tenant's summary is the cross-tenant leak from the Field Note, delivered through a new channel.
When you audit an organisation that has deployed AI, the risk register should contain rows for the model: who owns it, how it was assessed before deployment, what monitoring catches drift or misuse, what the residual risk is. If those rows are missing, the organisation has not managed the risk; it has assumed it. That is the very failure this chapter is about, wearing a new face.

Evidence Locker: lock this in (Week 2)

Lock in your **re-rated Tessera risk matrix**. For every rating you changed, document the rationale in a sentence another auditor could follow: *what context the original rating missed* (contractual, regulatory, operational, or aggregation), *what you re-rated it to*, and *on what evidence*. Note the **owner** you would assign where the register has none, and flag any risk you believe is **missing entirely**. By the end, your matrix should read less like the CTO's "we've covered the big risks" and more like a defensible argument about what could actually go wrong at Tessera. This is the artefact the rest of the engagement will keep returning to.

2.9 The question, answered

What could go wrong at Tessera? Quite a lot: a cross-tenant leak that breaches contracts and triggers a regulator, a phishing payload that reaches an admin account, a region outage that takes the platform down for the customers whose deals depend on it, an insider with DB access and a grievance, a certification failure that stalls the next funding round. The risk register's job is to name these, rate them, own them, and treat them honestly. Your job is to test whether it does.

How do we measure it? With the same two questions, asked sceptically, over and over: *how likely, and how bad*, accounting for aggregation and velocity, and always measured against the objectives and obligations that make the impact real. Risk is likelihood times impact, scored against context the spreadsheet cannot supply on its own. The discipline is to supply that context, write the rationale down, and let the evidence set

the rating rather than the optimism.

Tessera's CTO sent a register with no red cells. By the time you have re-rated it, there will be red, and every red will be defensible. That is what risk management looks like when it is real, and it is the foundation the next ten chapters build on. Because once you know what could go wrong, the next question is the one Tessera's CTO has not yet thought to ask: *which rules do we measure ourselves against, and why those ones?*

Chapter 3

Which? Frameworks and Standards

Which yardstick do you measure Tessera against, and why not all of them?

Tessera Case: "Why can't we certify against everything?"

The board meeting runs long. Tessera's CTO has the slide up: ISO/IEC 27001, NIST CSF, the ASD Essential Eight, SOC 2, PCI DSS, a privacy overlay. "We're a SaaS company serving enterprise and government customers," she says. "Surely more badges is more trustworthy. Why not certify against all of them?" It is a reasonable question from a reasonable person, and it is exactly the wrong instinct. Your answer is not "pick one." It is "pick the right one, and use the rest as cross-checks." The whole meeting turns on why that distinction matters, and it sets the criteria every later chapter in this engagement will be measured against.

3.1 Frameworks are tools, not trophies

Before you choose, get clear on what these things actually are. A framework is a structured set of expectations: what good looks like, organised so an organisation can find its gaps and an auditor can test against them. They differ in scope, in origin, and, crucially, in what you can *do* with the result.

Some frameworks are **certifiable**. ISO/IEC 27001 is the canonical example: an accredited body audits you against it and issues a certificate the market recognises. Certifiable frameworks are scarce and valuable

precisely because the opinion behind them is independent and repeatable. Anyone can claim compliance; only a certified body can back the claim.

Others are **assessment or maturity frameworks**. NIST CSF and the ASD Essential Eight live here. You can score yourself against them, an assessor can rate your maturity, and the result is genuinely useful. But there is no certificate to hang on the wall. They describe posture, not proof.

This distinction is the source of the board's confusion, and it is worth naming out loud in the room. "More badges" only counts if the badges are the kind procurement teams and regulators actually ask for. For Tessera, chasing ISO/IEC 27001 is not a stylistic choice; it is the answer to a specific business question, and the other frameworks answer different questions. Your job in that meeting is to choose the framework that answers the question you are actually being asked, then explain why the rest come along for a different reason.

3.2 ISO/IEC 27001:2022: the certifiable yardstick

ISO/IEC 27001 is the standard Tessera will be certified against, and it earns that role for a specific reason: it is the internationally recognised, certifiable ISMS standard. An ISMS, an information security management system, is not a checklist. It is a *management system*: a set of policies, processes, risk decisions, and controls that an organisation runs continuously and that an auditor judges as a living thing. The standard's clauses 4 through 10 lay out how that system is built: understand your context, set leadership accountability, plan around risk, support it with resources and competence, operate it, evaluate its performance, and improve it. Plan, do, check, act, repeated forever. When you audit 27001, you are auditing whether that loop actually turns, not whether a binder of policies exists.

The 2022 edition restructured Annex A, the controls catalogue, decisively. The 2013 edition scattered 114 controls across 14 domains that were easy to read but hard to reason about. The 2022 edition collapses them to **93 controls** organised into **four themes**, and the themes are almost self-explanatory:

- **A.5 Organizational controls** (37): the governance, policy, supplier, and risk-management controls. The "what we have decided" theme.
- **A.6 People controls** (8): the employment-lifecycle controls, screening, terms, training, disciplinary process.

- **A.7 Physical controls** (14): the offices, the server room, the clear-desk rules, the physical entry logs.
- **A.8 Technological controls** (34): access control, cryptography, logging, secure development, the things most people picture when they hear "security."

If you are going to hold one framework in your head as the yardstick, hold this structure. The themes map onto the way a real organisation divides its work: people who decide, people who are employed, places that are secured, and technology that is configured. When you test a control, you immediately know which part of the business it touches and whose evidence you will need.

One pairing worth remembering: 27001 is the certifiable standard, and its companion **ISO/IEC 27002** gives implementation guidance for each control. You audit against 27001; you reach for 27002 when you need to understand what a control is actually asking for. They are a pair, and confusing them is a common rookie mistake. The certificate says 27001. The "how do I do this well?" book is 27002.

3.3 NIST CSF 2.0: the functions lens

The NIST Cybersecurity Framework takes a different cut through the same problem. Where ISO 27001 organises by *who does it*, CSF organises by *what you are trying to achieve*, expressed as six functions:

Govern, Identify, Protect, Detect, Respond, Recover.

The newest of these, **Govern**, is the headline change in CSF 2.0 (released in 2024), and it is the one that matters most to an auditor. The older version treated governance as something that lived outside the framework. Version 2.0 puts it squarely inside, recognising that the technical functions fail when nobody owns the decisions. Governance is where accountability, risk appetite, policy, and oversight live, and pulling it out of the wings and onto the stage is the single most consequential revision in the update.

CSF is not certifiable, and that is a feature rather than a limitation. It is a *communication and orientation* framework. You use it to explain posture to a board, to map what you have, to spot whether your coverage is lopsided (heavy on Protect, thin on Detect, say). For an auditor it is an excellent cross-check because it asks a question ISO 27001's themes do not: *do you cover the full lifecycle, from understanding what you have through to recovering when it breaks?*

One warning that catches people out: CSF is not a control catalogue. It is functions, then categories, then subcategories that *reference* other standards. It points at controls rather than containing them. That is why

it pairs so well with ISO 27001, and why using it *instead of* a control standard leaves you with posture and no proof. Posture tells a story; an audit needs evidence. CSF supplies the story; 27001 is where the evidence is tested.

3.4 The ASD Essential Eight: the baseline

The Essential Eight is the most specifically Australian artefact in the set, and for a Perth company chasing government work, that specificity is its value. Published by the Australian Signals Directorate, it is a **baseline**: eight mitigation strategies the ASD judges most effective at stopping common attacks.

The eight are deliberately concrete and unglamorous: application control, patching applications, configuring Microsoft Office macros, user application hardening, restricting administrative privileges, patching operating systems, multi-factor authentication, and regular backups. Each is scored across **maturity levels 0 to 3**, from "not implemented" through to "fully managed and tested." A government customer asking for your Essential Eight maturity is not asking for a certificate; they are asking a pointed question about how well you do the unglamorous basics.

Two things to keep straight. First, the Essential Eight is a *baseline*, not a complete framework. It will tell you whether your patching and admin-privilege hygiene is sound; it will not tell you whether your supplier risk management is adequate or your incident-response plan works. Treat it as the floor, not the ceiling. Second, it was designed around Microsoft-centric estates, which fits most organisations but not all. For Tessera, built on AWS with a mix of tooling, some strategies map cleanly and some need interpretation. That interpretation is exactly the kind of judgement the standard cannot supply and the auditor must.

I will admit a personal bias here, because it shapes how I use these frameworks. The Essential Eight is the one I reach for first when I want to know whether an organisation can be *hacked tomorrow*. The others tell me whether it is well-governed; the Essential Eight tells me whether the doors are locked. Both questions matter, but on the day of an incident, only one of them was the difference.

3.5 Pick one yardstick, cross-check the rest

Now the actual decision, and the one the board meeting is really about. You do not certify against multiple frameworks because certification is expensive, the audits overlap, and "we are certified against everything" is not a signal of trust but a signal that nobody chose. The mature move

is to pick **one framework as the yardstick**: the thing you measure, certify, and score against. Then use the others as **cross-checks**: lenses that expose gaps the yardstick hides.

For Tessera the choice is almost made for you. The business goal is a certificate that enterprise and government customers will accept. That is ISO/IEC 27001:2022. So 27001 becomes the yardstick. NIST CSF 2.0 becomes the cross-check that asks whether the controls add up to a full Govern-to-Recover lifecycle. The Essential Eight becomes the cross-check that asks whether the baseline technical hygiene actually holds. Three frameworks, one opinion, each doing a job the others cannot.

> ⚠ Watch out: the collect-them-all fallacy
>
> The temptation, especially under a board that has read the marketing, is to chase every badge. "We'll do ISO 27001, SOC 2, PCI, the lot." Each certification is a real engagement with real cost, and the marginal badge is worth less than the marginal effort. Worse, organisations that spread themselves across many frameworks rarely do any one well: they collect mappings instead of building controls. The discipline is to choose the yardstick that answers the business question, certify against it, and let the rest inform rather than multiply the work. A company that certifies against one standard thoroughly beats a company that maps against five superficially, every time.

3.6 Mapping and reconciling

Once you have a yardstick, you map. The point of a cross-mapping is not to prove the frameworks are equivalent; they are not. It is to confirm you have no blind spots, and to translate evidence so a control tested once can satisfy multiple audiences. A government customer wants to see Essential Eight maturity; a board wants CSF posture; your certification auditor wants 27001 evidence. One well-built mapping lets all three read the same controls in their own language.

The mapping is **many-to-many**, and saying so plainly saves you from the most common mapping error. One ISO 27001 control, access reviews, touches Protect and Detect in CSF and several Essential Eight strategies (admin privileges, MFA, application control). One Essential Eight strategy, patching, maps to a technological control *and* to organizational policy. A neat one-to-one table is a lie. An honest many-to-many table is a working paper you can defend.

And here is the auditor's scepticism applied to the frameworks themselves:

a mapping is an *assertion* about correspondence, and assertions get tested. When someone hands you a cross-mapping, your job is to spot the row that claims two controls are equivalent when they are merely adjacent. "Both talk about logging" is not "both require the same evidence." Frameworks overlap; they do not coincide. The discipline of mapping is the discipline of saying exactly how much two controls actually share, and no more.

Now the worked artefact, and the place where AI earns and then loses your trust.

Tessera Case: Choosing the yardstick, and the starter map

You take the board through the decision on one slide. **Yardstick:** ISO/IEC 27001:2022, because certification is the business goal and 27001 is the certifiable answer. **Cross-check 1:** NIST CSF 2.0, to confirm the controls span Govern through Recover. **Cross-check 2:** the ASD Essential Eight, because government customers expect the baseline and Tessera wants the work. The board signs off, not because more is better but because each framework now has a defined job.

You start the cross-mapping as a working paper, not a deliverable, and you keep it honestly many-to-many. A representative slice:

ISO/IEC 27001:2022 (yardstick)	NIST CSF 2.0 (cross-check)	ASD Essential Eight (cross-check)
A.5.15 Access control	PR.AA (Identity, authentication, authorization)	Restrict administrative privileges; MFA
A.8.7 Protection against malware	PR.IR; DE.CM	Application control; user application hardening
A.8.13 Information backup	RC.RP	Regular backups
A.8.19 Installation of software on operational systems	PR.IR (Platform security)	Patch applications; patch operating systems
A.5.7 Threat intelligence	ID.RA; DE.AE	*(no Essential Eight equivalent; the baseline does not cover it)*

The last row is the one to notice. Threat intelligence is a real 2022 control with no Essential Eight home, and a mapping that pretends otherwise is precisely the lie this table refuses to tell. Cross-checks expose gaps as honestly as they expose coverage, and a gap you have named is a gap you can manage.

AI Field Note: Mapping controls across frameworks

This is a task AI is genuinely good at drafting and genuinely dangerous at finishing. Ask an AI to "produce a cross-mapping of ISO/IEC 27001:2022 Annex A to NIST CSF 2.0 and the ASD Essential Eight," and it returns a clean, plausible table in seconds. Most of it is right. That is the trap.

Two confabulations to hunt for. First, **stale control numbers.** Models trained on older corpora quietly cite 2013 Annex A numbers, A.9.x for access control and A.12.x for operations, as if they were current 2022 controls. They are not; the 2022 renumbering retired them. Every control number must be checked against the actual 2022 Annex A before the table leaves your desk. Second, **invented entries.** In one run the AI helpfully added a row for "Configuration baselining" under the Essential Eight column. The Essential Eight has exactly eight strategies, and that is not one of them. The model pattern-matched "configuration management" to something that *should* exist and then asserted it did.

Your correction is the work: count the Essential Eight strategies (you will find eight, never nine), verify each 27001 number against the current standard, and re-classify the many-to-many relationships the AI flattened into tidy one-to-one rows. The AI gave you a draft that saved an hour. The hour you spent correcting it is where the defensible opinion was built. Drafter to evaluator, exactly as the Introduction described.

Auditing AI: Which controls does an AI deployment fall under?

The frameworks were written before generative AI sat inside the estate, so no control is labelled "AI." That does not mean AI is uncovered. Treat an AI deployment as what it is: a cloud-delivered, data-hungry, rapidly-changing software system, and the 2022 controls land on it hard.

The 2022 edition is unusually well-timed here, because several of its *new* controls are the very ones an AI deployment makes load-bearing. **A.5.23 Information security for use of cloud services** applies directly, since most enterprise AI is consumed through a cloud API and Tessera's data leaves its boundary to reach the model. **A.5.7 Threat intelligence** matters more, not less: model-abuse techniques, prompt-injection trends, and data exfiltration via tool calls are exactly the threat intel a current ISMS needs. **A.8.11 Data masking** applies to the training and prompt data that may carry personal information. **A.8.25 Secure development life cycle**

(and its partner A.8.28 Secure coding) applies to the pipelines that build, fine-tune, and deploy models. And **A.5.30 ICT readiness for business continuity** applies the moment a business process depends on an AI service that can degrade silently.
The audit move is to take each AI system in scope, walk it through these controls, and ask the standard question: *what evidence proves this control is present and working for this system?* An AI deployment does not earn an exception. It earns a closer look.

Evidence Locker: lock this in (Week 3)

Add your framework-selection working paper to the locker. Record three things: the **yardstick** and why it was chosen (ISO/IEC 27001:2022, because certification is Tessera's business goal); the **cross-checks** and the question each one answers (CSF 2.0 for lifecycle coverage; the Essential Eight for baseline hygiene); and a **starter cross-mapping table** with at least a dozen controls mapped honestly many-to-many, with gaps explicitly noted rather than papered over. This is the standard against which every later control test in this audit will be judged. Get the yardstick wrong and the whole engagement measures the wrong thing.

3.7 The question, answered

Which yardstick? The one that answers the business question you are actually being asked. For Tessera that is ISO/IEC 27001:2022, chosen not because it is the only framework but because certification is the goal and 27001 is the certifiable answer. NIST CSF 2.0 and the Essential Eight come alongside it as cross-checks, each exposing what the yardstick alone would hide: one the full Govern-to-Recover lifecycle, the other the technical baseline your government customers expect.

And "why not all of them?" Because collecting frameworks is not the same as building controls. The mature organisation picks one yardstick, certifies against it thoroughly, and lets the others inform the work without multiplying it. The discipline of choosing, and then of mapping honestly, is itself an audit skill: it is professional scepticism applied to the standards before you ever apply them to Tessera.

You now have your criteria. The next question is the one that turns criteria into a plan: *what exactly are we auditing, on what terms, and how?* That is scope, and it is where the engagement stops being a reading exercise and becomes a method.

Chapter 4

Scope? Audit Planning and Methodology

An audit that tries to verify everything verifies nothing.

Tessera Case: "Just a quick health check"

Tessera's CTO opens the second conversation with a request that sounds modest. "Before we go deep, can you just do a quick health check? A general sense of where we stand." Reasonable, and dangerous. A "quick health check" has no boundary, no criteria, and no definition of done. It is the kind of engagement that swells until it is neither quick nor a check, and that ends with an opinion so vague no one can rely on it. Your job here is to resist the open-ended ask and replace it with something defensible: an engagement with a stated objective, a drawn boundary, and a moment when you can say, on the evidence, that the work is finished.

4.1 Scope before substance

You cannot audit everything, and you should not try. An information security management system touches every server, every account, every supplier, every line of policy, and every corridor with a lock on it. A certification-readiness audit has weeks, not years, and limited access. The gap between what exists and what you can examine is enormous, and the auditor's first job is to close it intelligently rather than pretend it is not there.

That act of choosing, what you will examine and what you will deliberately leave, is called scope, and it is the first place your judgement becomes visible. Anyone can say "we'll look at everything, to be safe." That

sentence is not diligence. It is the absence of a point of view, dressed up as thoroughness. A scope that contains everything contains no priority, and an engagement with no priority will spend its hours evenly across the trivial and the critical, which is how audits run out of time before they reach the part that matters.

I have inherited engagements whose scope read "all security controls." The first week was not spent auditing. It was spent drawing the boundary that should have existed on day one. Scope is the cheapest mistake to make and the most expensive to fix, because every later stage is built on it.

The discipline is to choose. Some choices will turn out wrong; that is the nature of planning under uncertainty, and it is why scope is revisited, not carved in stone. But a scope that names what is in and what is out is a scope an auditor can defend, and a scope that can be defended is the only kind worth having. The rest of this chapter is the machinery for making those choices well.

4.2 The four things you must fix before you start

Before a single document is reviewed or a single interview booked, four things must be fixed in writing. They are the legs the engagement stands on, and a wobble in any one of them propagates through everything that follows.

The **audit objective** is the question the engagement exists to answer. For Tessera, that question is specific: *is Tessera's information security management system ready for ISO/IEC 27001:2022 certification?* Not "is Tessera secure," which is unanswerable, and not "what is wrong with Tessera," which has no end. A good objective is a question with a finite, opinion-shaped answer.

The **criteria** are the benchmark the subject matter is measured against. You cannot say whether something conforms without first saying what it should conform to. Tessera's criteria are ISO/IEC 27001:2022, supported by the controls of Annex A and the 93 controls across the four themes you met in Chapter 3. The criteria decide what counts as evidence and what counts as a finding.

The **scope** is the boundary: which parts of the subject matter the objective applies to. Which systems, which locations, which business units, which period of time, which controls. Scope turns the objective from an idea into a finite piece of work.

And **materiality** is the threshold that decides what matters. Not every-

thing that is imperfect is a finding; only what is material is. Materiality is what stops the report from being an exhaustive catalogue of minor nicks and makes it a judgement about the things that affect the opinion.

Notice the relationship. The objective is the question; the criteria define "good"; the scope draws the fence; materiality decides what inside the fence is worth reporting. Fix all four, and the engagement has a spine. Leave any vague, and you are back to the health check.

4.3 Materiality is not completeness

Here is the distinction students trip over most, and it is worth slowing down for. Completeness asks: *did we look at everything?* Materiality asks: *did we look at what matters?* These are not the same question, and an auditor who answers the first has not answered the second.

In financial audit, materiality has a number: a percentage of revenue or profit below which an error is too small to affect a reader's decision. Information security is fuzzier. There is no tidy formula that says a control gap of severity 3.2 crosses the line. But the principle transfers exactly. A missing offboarding step that leaves orphaned administrator accounts is material, because it breaks a core control and the evidence of the breakage is exactly what a certification body will probe. A policy typo is not material, because it changes nothing about whether the controls work. The job is to tell them apart.

The temptation is to chase completeness because it feels safe. If you catalogue every imperfection, the reasoning goes, you cannot be accused of missing anything. You can, and you will be, because a report that lists everything lists nothing in priority order, and the reader who relies on it cannot tell the catastrophic from the cosmetic. Materiality is the auditor's permission to be selective, and selectivity, backed by judgement, is what makes the opinion useful.

> ⚠ Watch out: completeness masquerading as rigour
>
> A scope or a report that tries to cover everything is not more thorough. It is less useful. The trap is to treat "we examined all 93 controls" as the deliverable. It is not. The deliverable is an opinion on the controls that matter, supported by evidence you can defend. If your materiality instinct is missing, your engagement will drown in detail and still miss the thing that sinks the certification. Completeness is a checklist. Materiality is a judgement.

4.4 The engagement letter

Scope decisions are only worth something if they are agreed, and agreement that is not written down is an argument waiting to happen. The **engagement letter** is where the four planning primitives, the objective, the criteria, the scope, the materiality threshold, become binding between you and Tessera. It is part contract, part charter. It says what you will do, what you will not do, what you need from Tessera to do it, and what you will deliver at the end.

A good engagement letter is boring on purpose. It names the parties, the criteria, the objective, and the scope boundary in language a board and a certification body would both recognise. It sets out responsibilities: Tessera provides access and accurate information; you provide competence, independence, and an evidence-based opinion. It records the deliverables, the timeline, and the fees. And it is honest about limits: what the audit will not catch, what "snapshot, not guarantee" means, and how findings will be raised if the picture changes.

The letter is also where the hard conversations happen early. If Tessera wants the acquired subsidiary included, that is a scope and a fee decision, made now, not in week three. If the CTO expects a clean opinion regardless of evidence, the letter is where you restate, in writing, that the opinion follows the evidence and not the deadline. A scope that everyone has read and signed is a scope that survives pressure. One that lives in a hallway conversation does not.

> ⚠ Watch out: scope creep and vanity scope
>
> Scope has two enemies that look like virtues. **Vanity scope** is the impulse to put everything in, because a big scope looks ambitious and "thorough." It is the health check in a suit. **Scope creep** is the slow drift that happens mid-engagement, when a curious finding tempts you to follow it past the boundary, or a stakeholder asks you to "just take a quick look" at something out of scope. Each ask sounds small. Together they turn a finished engagement into one that never ends. The defence is the same for both: every change to scope is a written decision, with a reason, a cost, and a signature. "While we're here" is not a scope.

4.5 The six-stage audit lifecycle

Once the engagement is agreed, the work itself follows a lifecycle that ISO/IEC 19011 sets out and that every competent audit function recognises. ISO/IEC 19011 is the standard for *auditing management systems*:

not what to audit, but how to run an audit. It is the methodology volume of the standards library, and its six stages are the backbone of how an engagement actually unfolds. Learn them, because they are how you will structure your time from here to the opinion.

The six stages, in order:

1. **Initiating.** You establish contact, confirm the engagement is feasible, agree the objective, scope, and criteria, and form the audit team with the competence the engagement needs. The engagement letter is signed here, and the audit programme is begun.
2. **Preparation.** You review the documents Tessera has given you, build the audit plan, and prepare the working papers, checklists, and test procedures that fieldwork will use. This is where the desk audit happens, and where most of an Act I engagement is lived.
3. **Conducting.** The audit itself. The opening meeting, the gathering of evidence through interviews, configuration review, and testing, and the closing meeting where preliminary findings are shared. Evidence is gathered on terms you set, not on terms Tessera chooses.
4. **Reporting.** You prepare the audit report: the findings in a defensible format, the conclusion, and the opinion. The report is drafted, reviewed, and distributed to the agreed recipients.
5. **Completion.** The engagement is formally closed. Working papers are finalised and retained, lessons are captured, and the records that someone may need to reconstruct this audit in three years are put away properly.
6. **Follow-up.** You verify that the corrective actions raised in the report have actually been taken and are working. An audit that reports a finding and never checks the fix is an audit that stops at diagnosis.

Notice where the judgement sits. The stages are linear, but the decisions inside them are not. Initiation fixes scope, but preparation often forces a scope correction when the documents reveal something unexpected, and follow-up can reopen an engagement that completion thought was closed. The lifecycle is a spine, not a script. You move through it with your professional scepticism switched on at every stage.

4.6 The audit programme and risk-based scoping

The engagement letter says *what* you will do and *where the boundary is*. The **audit programme** says *how*, in operational detail: which controls you will test, which evidence you will gather for each, which sample sizes, which interviews, and in what order. If the engagement letter is

the charter, the audit programme is the work order. (ISO/IEC 19011 is precise about terms: it reserves "audit plan" for one audit's schedule and "audit programme" for a coordinated set of audits. In practice, auditors say "audit programme" for the engagement's procedure list, and so will this book.)

You do not write the programme by listing all 93 controls and assigning each a test. That is the completeness trap in a different costume. You write it by ranking. **Risk-based scoping** is the principle that audit effort follows risk: the controls and areas where a failure would most threaten the objective get the most hours, the deepest sampling, and the most experienced eyes. The areas where a failure is unlikely or low-impact get a lighter touch, sometimes just a documented review.

This is where Chapter 2 does real work. The Tessera risk register you helped build, the likelihood-times-impact ratings, the residual risks after treatment, all of it now becomes the input to where this engagement spends its time. A risk that is rated high and is the only treatment for a critical asset is where you dig. A control that backs a risk already accepted at a low residual level gets enough attention to confirm it exists, and no more. The audit programme is the risk register translated into hours.

Risk-based scoping has a courage requirement. It asks you to *not* test something, on purpose, because the risk does not justify it, and to write down why. That "why" is your defence when a reviewer asks why a control was lightly examined. The answer "because the risk was low, and here is the analysis" is a complete answer. The answer "we ran out of time" is not.

Tessera Case: Drawing the Tessera boundary

The artefact for this chapter is a pair: the engagement letter and the audit programme, and the most useful part of both is what they exclude. Tessera's scope, agreed in writing, runs like this.

In scope: the AWS-hosted Tessera SaaS platform, its production and staging environments, the customer data flows, and the information security management system that governs them, audited against ISO/IEC 27001:2022. The ISMS documentation, the risk register, the policies, and the controls that back them are all inside the fence.

Out of scope, with reasons recorded: a recently acquired subsidiary, brought in three months ago, whose systems are not yet integrated into Tessera's ISMS and whose inclusion would require a separate assessment. And the third-party payment processor Tessera routes transactions through, which is covered by its own current SOC 2 Type II report, a form of assurance you will accept and verify rather than duplicate.

Two exclusions, two different logics, both written down. The subsidiary is out because it is not yet part of the system of record; including it would mean auditing an ISMS boundary Tessera has not finished drawing. The processor is out because its assurance already exists; re-auditing it would waste hours Tessera needs elsewhere. These are not evasions. They are documented scope decisions an auditor can defend, and they are exactly the kind of decision the generic "health check" would never have forced. You can run this same exercise against the live Tessera site at tessera.locoensayo.org and see whether your own boundary holds up.

> **AI Field Note: Drafting the audit programme**
>
> A programme is a fine thing to draft with an AI, because its structure is regular and its content is largely standard. Prompt an AI: "Draft a Stage 2 certification-readiness audit programme for a multi-tenant B2B SaaS company hosted on AWS, against ISO/IEC 27001:2022, with controls grouped by the four Annex A themes." In a minute you have a structured programme: objectives, the control families to test, sample procedures, and a schedule. That draft is genuinely useful, and it is where the work begins, not where it ends.
> Because the AI scopes generically, it scopes wrongly for Tessera, and that is the lesson. Read the draft as an evaluator and the gaps are obvious. The draft assumes a single-tenant system and never tests the **multi-tenant boundary**, the control that stops one customer reaching another's data, which is the single highest-risk area in the platform. It schedules a full access-control test of the **payment processor**, duplicating assurance a SOC report already provides and burning hours you do not have. And it silently sweeps the **acquired subsidiary** into scope as if integration were finished, when the subsidiary is not yet on Tessera's ISMS at all. Three Tessera-specific boundaries, three misses, because the AI has no knowledge of the engagement letter and no judgement about where this company's risk actually lives.
> The correction is yours. You add a tenant-isolation test procedure that pulls configuration evidence on the separation logic. You remove the processor from the test plan and replace it with a procedure to review and rely on the SOC report. You add a scope-exclusion note for the subsidiary and a flag to revisit it at the next review. The AI wrote the skeleton. You supplied the judgement that turned a generic programme into Tessera's programme. That is the drafter-to-evaluator shift, on a real artefact, this week.

! Auditing AI: Scoping the AI surface

Tessera runs an AI assistant inside its support portal and an AI-driven anomaly detector in its logging stack. Neither is on the security page, and both are inside the boundary you just drew, so the scoping question lands now, not later. An AI system has a scope of its own: the model, the training and inference data, the prompts and outputs, the human-in-the-loop checks, and the integrations that feed it. Decide which of those the engagement covers, because "we tested the model" is not the same as "we tested the control the model is part of."

The sharper point is that AI changes what "in scope" means over time. A model that behaves one way today can drift after a retrain, so a control that passed in week 2 may not pass in week 8. Scope has to say *when* the AI surface was examined, and the opinion has to honour that snapshot. We will test AI-generated evidence properly in Chapter 7. Here, the lesson is simpler: an AI you cannot see on the security page is still part of the ISMS, and a scope that ignores it is a scope with a hole in it.

Evidence Locker: lock this in (Week 4)

This week's artefact is the planning pair, and the part that earns its place in the locker is the documentation of *why*. Lock in: the **engagement letter** with objective, criteria, scope, and materiality stated; the **audit programme** with the controls and procedures prioritised by risk; and, most importantly, the **recorded scope decisions**, the subsidiary excluded pending integration, the processor covered by its SOC report, the AI surface included with its snapshot date. Explicit exclusions are not footnotes. They are the proof that the scope was a decision, not a default. By Week 12 this pair is the front matter of your working papers, and anyone reading them will see an engagement that knew what it was doing before it began.

4.7 The question, answered

What is in, what is out, and how do we know we are done? *In* is the AWS-hosted Tessera platform and its ISMS, measured against ISO/IEC 27001:2022. *Out*, deliberately and on the record, is the unintegrated subsidiary and the SOC-covered processor. We know we are done when the six-stage lifecycle has run its course, the audit programme's procedures have been executed to the depth the risk justified, the findings that cross the materiality line are raised, and the opinion follows the evidence rather

than the deadline. Scope is not the paperwork before the audit. Scope is the first opinion the auditor forms, and the one every later opinion depends on.

The boundary is drawn and the programme is written. Next week the work turns to what "good" looks like inside that boundary: *what should the technical controls actually do, and how do we recognise it when they do?*

Chapter 5

Expect? Technical Control Objectives

What should be in place, and what does "good" actually look like?

Tessera Case: The diagram is not the territory

Tessera's head of engineering hands you a single-page architecture diagram in your second week. It is beautiful. Boxes for VPCs and subnets, arrows labelled TLS, a padlock on the database, "KMS" stamped on every storage layer, "IAM roles" on every service. He watches you study it, looking for the nod. You give him a different response. You ask, for each box: *what is the objective here, and how would I test whether it is actually met?* The diagram is a claim. Your job, from this chapter on, is to turn each claim into a control objective you can substantiate, or refute, with evidence.

5.1 From control to control objective

The previous chapters gave you the auditor's mindset, the risk picture, the frameworks, and the plan. None of that tells you what to test. This chapter does. It introduces the single tool that turns a list of controls into an audit: the **control objective**.

A control is a safeguard. Encryption is a control. A firewall rule is a control. Multi-factor authentication is a control. A control objective is the *intent* that control is meant to deliver, written so that intent can be tested. "We encrypt the database" is a control. "No party, internal or external, authorised or not, can read customer data without an access decision that is logged and reviewable" is an objective. The first describes

a configuration. The second describes an outcome you can try to disprove. That is the difference between auditing a screenshot and auditing a system.

ISO/IEC 27001:2022 gives you the raw material in Annex A.8, the Technological controls, the largest of the standard's four themes. NIST CSF 2.0's Protect function and the ASD Essential Eight give you the same engineering from different angles. None of them hands you the objective. That is your job, and it is the job this chapter teaches: read the control, ask what "good" looks like, phrase it as a statement you could prove or disprove with evidence, and then go look. Every objective in this chapter is built that way.

Here is a useful test for whether you have written an objective rather than a control. Can you describe how it would *fail silently*? If you cannot, you have written a feature, not a test. The objective is only real once you can picture the version of the world in which the control is present, the dashboard is green, and the objective is still not met. That picture is what you will spend the rest of the engagement trying to find.

5.2 Access control: identity is the new perimeter

For a SaaS company like Tessera there is no perimeter in the old sense. There is no castle with a wall and a moat. There is a set of identities, human and machine, each holding some set of permissions, talking to services that live on someone else's hardware. Get identity right and most of your technical risk falls into line. Get it wrong and nothing else matters, because the attacker walks through the front door with valid credentials.

The first objective is **least privilege**: every identity, human or service, holds only the permissions it needs, and nothing more. That sounds tidy until you ask for the access review and find the founding engineers still hold administrator rights across production, staging, and every customer tenant, because nobody has ever pruned them. I have yet to audit a fast-growing environment where the first access review did not surface a surprise, and Tessera is built to be no exception. Least privilege maps to ISO 27001 A.8.2 (privileged access rights) and A.8.3 (information access restriction), and to the Essential Eight's "restrict administrative privileges" strategy. The objective, though, is the testable version: *any permission not justified by current job function is removed within a defined window.*

The second is **authentication strength**: access requires proof of identity proportionate to the risk, and for anything privileged, that proof resists phishing. Passwords alone fail this objective, and so does a six-digit SMS code relayed to a tired person at midnight. Phishing-resistant MFA,

the kind built on FIDO2 and WebAuthn, is what "good" looks like for administrators and for anyone touching customer data. The Essential Eight pushes you here for a reason.

The third is the **identity lifecycle**: accounts are created when someone joins, changed when their role moves, and removed when they leave, on a clock you can measure. The leaver control is the one that fails quietly, because the person is already gone and nobody is watching the account. Federation through SAML or OIDC, single sign-on, and automated provisioning through SCIM are how mature organisations make the lifecycle a system rather than a memory.

The fourth is **privileged access** specifically. Administrators do not sit in a standing admin role all day. They elevate when they need to, for as long as they need to, and what they do while elevated is recorded. This is privileged access management: just-in-time elevation, break-glass accounts for emergencies, session recording, and time-boxed grants. The objective is not "we have PAM tooling." The objective is *no privileged action occurs without a time-limited, logged, reviewable elevation.*

5.3 Network security: assume the breach

The old network question was "can the attacker get in?" The modern question, the one a zero-trust mindset demands, is "once the attacker is in, how far can they get?" NIST SP 800-207 states the principle: never trust, always verify, on every request, as if the network were already hostile. For a company whose entire estate is API calls to AWS, the network *is* already hostile by design.

Segmentation is the engineering of that mindset. Tessera's VPCs, subnets, security groups, and network ACLs should carve the estate into zones that contain blast radius. Production separate from staging. Customer-facing services separate from internal tooling. Databases reachable only from the application tier, never from the open internet. Each of those is a control; the objective is that *an attacker who compromises one component cannot traverse to another without a fresh access decision.* That is the difference between a breached container and a breached company.

Firewalls and security groups are the mechanism, but they are only as good as their rules, and rule drift is where segmentation goes to die. The rule opened for a vendor demo eighteen months ago and never closed. The `0.0.0.0/0` that someone applied to make a test pass. The database port exposed to the office range because a developer wanted to work from home. The objective is not "we use security groups." It is *every rule is justified, scoped to least access, and reviewed on a schedule*, which is a test you can actually run against the live configuration.

5.4 Cryptography and key management: the control that fails silently

Encryption is where auditors go wrong, because encryption looks finished. The dashboard says "at rest: enabled." The browser shows the padlock. You tick the box and move on. Do not.

Encryption at rest and in transit is the baseline, not the achievement. The achievement is **key management**, and it is where a cryptography control that looks perfect can be substantively broken. AWS Key Management Service (KMS) and hardware security modules (HSMs) exist to keep keys out of the data they protect. Customer-managed keys give Tessera control over who can use a key and when, through policies and grants that are themselves auditable. Key rotation limits the damage of a key that leaks. Separation of duties limits the damage of a rogue insider. ISO 27001 A.8.24 (use of cryptography) points here, and the Essential Eight's expectations on backups and data protection lean on the same foundation.

Now picture the failure. Encryption is enabled on every S3 bucket. The dashboard is green. But the application role that reads customer data holds a KMS grant for a single key used by *every tenant*. One compromised token, one over-permissive policy, and the attacker decrypts everyone's data at once. The control exists. The objective, *ciphertext is useless to an attacker who steals the storage*, is not met. This is the silent failure, and it is exactly the kind of gap a control objective, not a control list, will catch.

5.5 Multi-tenant SaaS: isolation is the whole product

Everything in this chapter converges on one fact about Tessera: it is a multi-tenant SaaS. Customers share the same application, the same databases, the same infrastructure. The product they are paying for is, at root, the promise that they cannot see each other. Get tenant isolation wrong and you do not have a security problem. You have an existential one.

Tenant isolation is enforced through layered controls: per-tenant identity and namespace boundaries in IAM, row-level or logical separation in shared data stores, and, for the most sensitive work, per-tenant or customer-managed encryption keys so that even a database administrator with broad access cannot read another tenant's ciphertext. The classic failure has a name: insecure direct object reference, where a request for `/api/tenant/1043/data` is honoured for an authenticated user who belongs to tenant 1077, because nobody checked the relationship. IDOR

is the silent failure of multi-tenancy, and it is widespread because it fails open: the request succeeds, the log looks normal, nobody is alerted.

The objective for tenant isolation has to be absolute: *under no condition, including error paths, including a compromised application role, can tenant A read tenant B's data.* That is a strong statement, and it is the one your testing has to try to break. Customer-managed keys, sometimes called bring-your-own-key or hold-your-own-key, are how the most regulated tenants raise the bar: Tessera cannot decrypt what it has no key to decrypt, and the customer can prove it.

5.6 The shared-responsibility line: who secures what

Underneath all of this is a line that determines who is responsible for what, and Tessera sits on the wrong side of it more often than its diagram suggests. AWS operates on a shared-responsibility model. AWS secures the cloud itself; Tessera secures what it puts in the cloud. The line is not a footnote. It is the boundary of your entire technical audit, because you can only test what Tessera owns, and a great deal of what looks like "AWS's problem" is actually Tessera's.

AWS secures:

- the physical data centres, hardware, and networking;
- the virtualisation layer, and the regions, Availability Zones, and edge locations;
- the managed services' underlying platforms.

Tessera secures, in the cloud:

- customer data, and its classification and handling;
- IAM users, groups, roles, and policies;
- the application code and its third-party dependencies;
- the guest operating system and AMIs where the service is self-managed;
- network configuration, VPCs, security groups, and routing;
- client- and server-side encryption, key management, and the configuration of every managed service it switches on.

And because Tessera is itself a service provider, there is a third line. Tessera's customers trust Tessera for application-layer security and tenant isolation, while they keep responsibility for their own users, their own tenant configuration, and the data they choose to put in. A multi-tenant SaaS inherits a shared responsibility on both sides.

The shared-responsibility gap is the assumption that AWS covers some-

thing Tessera actually owns. "Encryption is AWS's job." "Patching is managed, so we're fine." "The region is resilient, so we don't need backups of our own." Each of those is a real sentence I have heard, and each is wrong in a way that a misread shared-responsibility line produces. ISO 27001 Annex A.5 covers supplier and cloud-service relationships precisely because this line is where assurance leaks out. Your audit has to draw the line explicitly, on Tessera's own architecture, and test only the Tessera-owned side.

> ⚠ Watch out: 'it exists' is not 'it works'
>
> Two traps run through every technical audit, and they are the same disease: confusing the surface with the substance.
> The first: *a control that exists is not a control that is configured correctly.* Encryption is on, but the keys are shared across tenants. MFA is enforced, but the SMS fallback is enabled. Security groups exist, but the database is open to `0.0.0.0/0`. Each looks compliant on a dashboard. None meets its objective. Always ask what "good" looks like, then test for that.
> The second: *a control someone else owns is not a control you can assume.* AWS secures the hypervisor. It does not secure your IAM policies, your application code, your key grants, or your patch cadence on self-managed components. The shared-responsibility line is where you stop auditing AWS, which you cannot, and start auditing Tessera, which you can. If you cannot point to who owns a control, you have found a gap, not a comfort.

Tessera Case: What "good" looks like for tenant isolation

Take the padlock on the database in Tessera's diagram and turn it into a picture of good. The control objective for isolating one customer tenant from another reads something like this:

> *A request that returns data belonging to tenant X is only ever honoured for an identity that a current, least-privilege policy has authorised for tenant X, on a decision that is logged, with the data decrypted only under a key whose use is restricted to that tenant's context. Under any error, compromise, or misconfiguration, the default behaviour is denial.*

That single sentence is now your test plan. It tells you to examine the authorisation logic for tenant-boundary checks, to confirm IAM policies are scoped per tenant, to verify the KMS key grants do not collapse tenants into one decryptable blob, and to probe the error paths for IDOR. The diagram's padlock was a picture. This is the same padlock, rewritten as something you can break, and therefore something you can audit.

AI Field Note: From a generic list to a testable objective

Ask an AI to "list the technical security controls for a multi-tenant SaaS hosted on AWS." You will get a competent list in seconds: encryption at rest, TLS in transit, MFA, a WAF, logging and monitoring, network segmentation, regular backups. Every item is correct, and the list is useless, because it describes every SaaS on Earth. It is a draft, and a draft of what everyone already knows has no edge.

Push it. Ask, for each control: *what is the objective, in one testable sentence, and how would it fail silently at a fast-moving company where the founders still have god-mode access?* Now the AI has to engage with Tessera's reality, not a textbook. Take encryption. Its first cut was "encrypt data at rest using KMS." Pressed for the objective and the silent failure, a better answer emerges: the objective is that a stolen snapshot reveals no plaintext to anyone outside the access-decision path; the silent failure is that encryption is on but a single over-broad KMS grant lets one compromised application token decrypt every tenant's data at once. The first sentence any AI can write. The second is the variation you supplied. That is the drafter-to-evaluator shift, done on one line of output.

Auditing AI: The AI estate is now part of the technical estate

Tessera has shipped AI into its estate in two places. A support assistant reads customer tickets and drafts replies for the helpdesk team. And an AI feature inside the product lets customers summarise and query their own data. These are not experiments on a slide. They are workloads with identities, data access, and failure modes, and they sit squarely inside this chapter.

Treat them like any other component. The support assistant has a service role: what can it read? If it can see every tenant's tickets when it only needs the active one, that is a least-privilege failure identical in shape to any other. The product feature sends customer data to a model: what is sent, what is retained, and does the model provider train on it or promise not to? That is a data-minimisation and key-boundary question, and it foreshadows Chapter 9's privacy work. Both run on a model from a vendor: which model, which version, and what does that vendor's SOC report actually cover? That is supplier assurance, the A.5 cloud-services territory Chapter 3 flagged.

The objective for an AI workload has the same shape as for any other: *the AI system cannot exfiltrate or act on data it has no business*

reason to touch, and every action it takes is attributable. Test it the same way: enumerate the role's permissions, trace what data crosses the model boundary, and check the logs that record model calls. Some of Tessera's controls are now run *by* AI, an assistant triaging alerts or flagging anomalies. When you audit those, you are back in territory Chapter 7 takes up in full: can you trust a control whose evidence an AI produced? For now, the discipline is the one this chapter teaches. State what good looks like, then go and try to break it.

Evidence Locker: lock this in (Week 5)

For each domain below, lock in one technical control objective for Tessera, phrased as a testable "good looks like" statement, plus the single piece of evidence that would prove or disprove it.

- **Access control**: least privilege is in force for every human and service identity, with a documented leaver window, and privileged access is granted just-in-time and logged.
- **Network security**: segmentation contains blast radius, and every security-group or firewall rule is justified and reviewable.
- **Cryptography and key management**: ciphertext is useless to an attacker who steals storage, because keys are isolated, rotated, and bound by grants that respect tenant boundaries.
- **Multi-tenant isolation**: under no condition, including error paths, can one tenant read another's data.
- **Shared-responsibility boundary**: the line between AWS-owned and Tessera-owned controls is drawn on Tessera's own architecture, and only Tessera-owned controls appear in your test plan.

This is the chapter where your Evidence Locker stops being narrative and starts being a test programme. Each objective is a row you will later mark green or red against evidence.

5.7 The question, answered

What should be in place, and what does good look like? In place: a layered set of technical controls across identity, network, cryptography, and the multi-tenant boundary, owned by Tessera on the correct side of the shared-responsibility line. Good looks like a set of *objectives* behind those controls, each phrased so it can be tested, each with a silent failure you can describe, each backed by evidence you can demand. The diagram Tessera's head of engineering handed you was a claim. You now hold the method that turns it into an audit.

There is a catch, and it is the catch that closes the first act of this book. Technical controls are the part of the estate that is easiest to enumerate and easiest to test, and they are still only half the story. A perfectly configured IAM system is worthless if the person holding the admin token was never trained, or never had their access removed when they left, or sits inside an organisation whose security policy is a document nobody reads. Next, we turn from the machine to everything around it: the people, the physical world, and the management system that is supposed to hold it together. *Assess? Physical, people, and ISMS controls.*

Chapter 6

Assess? Physical, People, and ISMS Controls: Preliminary Readiness

How much can you honestly say about a company you have only ever read about?

Tessera Case: Act I closes

Six weeks in, and the document library is exhausted. You have read Tessera's ISMS manual, the information security policy and every standard beneath it, the risk register, the Statement of Applicability, the asset register, the access-control standard, the offboarding checklist, and the supplier list with each master agreement attached. You have interviewed no one, logged into no system, stood in no office. The honest opinion forming in your working papers is the most uncomfortable kind an auditor writes: *looks promising on paper, but…* and the ellipsis is doing real work, because the discipline of this chapter is naming exactly what that limited access could not show.

6.1 What Act I let you do

Chapter 5 taught you the technological control objectives: access, network, cryptography, the AWS shared-responsibility split. Those sit in Theme A.8, the thirty-four technological controls. This chapter assesses the other three families the 2022 standard clusters controls into, plus the management system that holds them together.

The **people** controls (Theme A.6, eight controls) govern the human side:

who you hire, what you make them agree to, how you train them, and what happens when they leave. The **physical** controls (Theme A.7, fourteen controls) govern offices, devices, and the environment they sit in. The **organizational** controls (Theme A.5, thirty-seven controls) are the largest family, and they sweep up everything that is neither purely human nor purely physical: policies, asset management, supplier relationships, classification, and the cloud services Tessera runs on.

Around all ninety-three controls sits the ISMS itself, the management system defined in Clauses 4 through 10 of ISO/IEC 27001: context, leadership, planning, support, operation, performance evaluation, and improvement. The controls are the body. The ISMS is the nervous system that makes them move.

Here is the catch, and it is the whole point of this chapter. Act I gave you documents, not systems. A desk audit can establish that a control is *designed*. It cannot establish that a control is *operating*. The gap between those two words is where every mistake in this chapter hides.

6.2 Physical and environmental security

Tessera runs on AWS, which buys them a tidy advantage: the physical security of the data centre is largely AWS's problem, attested through AWS's own certifications and the shared-responsibility split you learned in Chapter 5. Tessera does not secure the racks their instances live on. That does not leave them with nothing to secure.

They have a Perth office. They have staff laptops and phones that leave the building every evening. They may have a small co-location rack for low-latency work, or a development box under someone's desk that nobody documented. Physical controls (A.7.1 through A.7.14) ask the unglamorous questions that decide whether any of that matters: is the perimeter secured, is access controlled and logged, are visitors managed, is the server cupboard locked, is equipment protected against fire and flood and the cleaner who unplugs the router to plug in a vacuum cleaner.

The desk read gives you Tessera's physical security policy, the badge-access procedure, the visitor-management process, the clear-desk and clear-screen standard, and the equipment-disposal procedure. Read together they describe an office that sounds disciplined. Badge access is "enforced 24/7." Visitors "must sign in and be escorted." Clear desk is "mandatory." Media disposal "follows a certified destruction process."

Notice the verbs. *Enforced, must, mandatory, follows.* Those are promises, and a promise is an assertion. From a desk you cannot confirm the badge reader at the side entrance actually denies a revoked card, that the visitor log carries a real signature for yesterday, that the fire-suppression system

was serviced when the maintenance record claims, or that the back door near the loading bay is not propped open with a recycling bin every afternoon so the smokers can get back in. You have read what should happen. You have not seen what does.

6.3 HR security: the employment lifecycle

People controls are where organisations hurt themselves most predictably, because people are a lifecycle, not a state. A.6 makes that explicit. There is a moment a person joins, a long stretch in the middle where their role and access drift, and a moment they leave. Each transition is a control point, and the controls (A.6.1 screening, A.6.2 terms and conditions, A.6.3 awareness and training, A.6.5 responsibilities after termination or change, A.6.6 confidentiality agreements, and the rest) cluster around the joins.

Onboarding is the first gate. Screening before employment, terms signed before access is granted, confidentiality agreements in place before the new starter sees customer data, and security awareness training before they can do damage. The control objective is simple: nobody gets the keys before they have been checked and have agreed to the rules.

Role change is the silent one. A support engineer becomes an SRE and picks up production access. A developer moves into a team that handles payment data. Each move should trigger an access review, because the permissions that were correct last quarter may now be excessive. Organisations hate doing this, because it is invisible work with no deadline, and so it is exactly the control that quietly stops happening.

Offboarding is the one that ends up in breach reports. When someone leaves, the clock starts on every minute their access stays alive after their last day. Return of assets, revocation of logical and physical access, handover of responsibilities, a final reminder of confidentiality: the offboarding checklist (A.6.5, with A.5.11 on return of assets) is short, well-understood, and routinely incomplete.

The desk read gives you Tessera's offboarding procedure, and it is a good one. Sixteen steps, an owner named against each, a sign-off line for the manager. It is the kind of document that makes you feel reassured, which is precisely when you should lean into the discipline of this chapter. A checklist that exists is not a checklist that runs. You can read every step. You cannot, from a desk, confirm that the last three leavers were walked through it, that their access was revoked inside the policy window, or that anyone signed the sign-off line at all. That confirmation is Act II work, and pretending otherwise is the mistake this chapter exists to prevent.

6.4 Asset management

You cannot protect what you have not inventoried. Asset management is the unsexy foundation under most other controls, and in the 2022 standard it lives largely in the organizational family: the inventory of information and associated assets (A.5.9), acceptable use (A.5.10), return of assets (A.5.11), classification (A.5.12), and labelling (A.5.13).

The desk read gives you Tessera's asset register and their classification scheme. The register should list the systems, the data stores, the devices, and the information that flows between them, each with an owner. The classification scheme should tell you which of that information is public, internal, confidential, or restricted, and what handling each tier demands.

Here is the asset-management reality the desk cannot show. A register is only ever as good as its last update, and registers rot fast in a company growing the way Tessera is. A laptop issued to a contractor and never returned. A database spun up for a feature, abandoned, and forgotten. A customer dataset that nobody ever classified, because the classification scheme arrived six months after the data did. The document tells you the scheme exists and the register is "maintained." It cannot tell you whether either is complete, because completeness is a property of the world, not of the document.

6.5 Suppliers and the cloud Tessera sits on

No modern SaaS company is an island. Tessera sits on AWS, processes payments through a third party, ships with a logistics API, runs analytics on a managed platform, and increasingly bolts an AI service into the product to summarise customer reports. Each of those is a supplier relationship, and the organizational family treats them seriously: information security in supplier relationships (A.5.19), security within supplier agreements (A.5.20), the ICT supply chain (A.5.21), monitoring and reviewing supplier services (A.5.22), and the specific case of cloud services (A.5.23).

The desk read gives you the supplier list, the master service agreements, and the data processing agreements. AWS appears with its SOC 2 and ISO certifications attached. The payment processor's agreement is on file. The AI vendor's terms are there too.

Read those documents as an auditor and you will notice what they are, and are not. A supplier agreement is a statement of *obligation*: what the supplier has promised to do. It is not a statement of *performance*: what the supplier actually does. Due diligence documented in a file is not due diligence performed. A SOC 2 report from a vendor is that vendor's auditor's opinion, at a date, over a window. It is evidence about

the vendor, not evidence about how Tessera uses the vendor. The gap between a signed agreement and an operating control is the same gap as everywhere else in this chapter, just one contract further away.

> **!** Auditing AI: The supplier you cannot inspect
>
> Tessera's product now calls an AI service to summarise customer data. From your desk you can read the vendor's data processing agreement, their security page, and their terms of service. What you cannot read, from documents alone, is what the model actually does with the prompts Tessera sends it. Are customer inputs logged? Used to train the next version? Retained for thirty days or thirty years? Forwarded to a sub-processor in a jurisdiction with different privacy law?
> The agreement will say one thing. The system may do another, and the system is the thing that matters. Verifying it is Act II work: you will need the vendor's actual data-handling configuration, the contract's processing specifics, and Tessera's own logs of what leaves the boundary. For now, note it as a named gap. An AI supplier is a supplier like any other, which is to say: asserted in a document, substantiated nowhere yet.

6.6 The ISMS itself: governance on paper

Everything above is a set of controls. The ISMS is the thing that is supposed to make them work as a system. ISO/IEC 27001 devotes Clauses 4 through 10 to it: the organisation defines its context (Clause 4), leadership commits (Clause 5), it plans the risk treatment that selects controls (Clause 6), it supports the system with resources, competence, awareness, communication, and documented information (Clause 7), it operates it (Clause 8), evaluates its performance (Clause 9), and improves it (Clause 10).

The artefact that ties the controls to the system is the **Statement of Applicability** (SoA), required by Clause 6.1.3. The SoA lists the controls Tessera has chosen, the justification for each inclusion and exclusion, and, crucially, whether each control is implemented. It is the index of the whole control set, and at the end of Act I it is the single document you have spent the most time with.

Here is where the discipline of this chapter has to bite hardest, because the ISMS document set is the most convincing thing you will read in this engagement. It is well-written. It is internally consistent. The policies cross-reference each other correctly. The SoA's status column reads "implemented" down the line. And every one of those qualities is a

reason to be more careful, not less.

> ⚠ Watch out: the document confidence trap
>
> The most dangerous artefact in a desk audit is a complete, well-written ISMS. It creates an illusion: that because the documentation is coherent, the practice is sound. It is the same illusion a polished security webpage creates, and it fails the same way. A policy on the intranet is an assertion. A SoA status of "implemented" is an assertion. An offboarding procedure that reads perfectly is an assertion. None of them is evidence until you have seen the control operate.
> The trap is not that the documents lie. Tessera's may be entirely sincere. The trap is that design quality and operational quality are different properties, and a desk audit can only ever measure the first. Confidence formed from documents alone is confidence in the wrong thing.

I have a small confession, the kind that took a few engagements to learn. The first time I read a genuinely good ISMS, I relaxed. The prose was clear, the scope was honest, the controls mapped cleanly, and it felt like assurance. The fieldwork that followed turned up an offboarding process that had not run correctly in a year and an asset register that missed a quarter of the estate. The documents had been excellent, and they had been wrong about the one thing that mattered. A desk audit earns you the right to a suspicion. It does not earn you the right to a conclusion.

6.7 From documents to a preliminary opinion

So what, honestly, can you say at the end of Act I? More than nothing, and less than you might want to.

You can say what the documents substantiate: that Tessera has *designed* a control set mapped to ISO/IEC 27001:2022 across all four themes; that the policies exist and are coherent; that the risk register and SoA are present and internally consistent; that the supplier agreements are on file; that the offboarding procedure, the access standard, and the classification scheme all read as fit for purpose. That is a real finding. Design maturity matters, and an organisation that cannot produce these documents is not ready, full stop.

You cannot say what the documents do not substantiate, and the discipline of this chapter is to say so out loud. You have not confirmed that a single control is *operating*. You have not watched the badge reader deny anyone,

seen a leaver's access revoked, reconciled the asset register against the live estate, or verified that the supplier due diligence on file was ever performed. The preliminary opinion is calibrated suspicion, not proof, and its value rests entirely on how clearly it names the gap between the two.

> **AI Field Note: Summarising the policy set**
>
> This is a task an AI does fluently, and it is worth seeing both why that helps and why it stops where it does. Hand an AI Tessera's document library and ask it to produce a readiness summary: which control families are documented, where the SoA claims implementation, where the gaps in coverage are. It will return a clean, well-structured brief in minutes. The summary of what the documents *say* will be accurate and genuinely useful, and it will save you hours.
> Then ask it the question that matters: "Which of these controls are actually operating?" It cannot answer. It has read the same documents you have, and documents do not contain operating evidence. Press it, and it will do what large language models do under pressure: it will start to infer, to extrapolate from how thorough the offboarding procedure looks, to slide from "the procedure exists" into "the procedure is followed." That slide is the exact error this chapter is about, and watching a confident machine make it is the clearest demonstration I know of why the evaluator matters more than the drafter.
> The AI's limit is the point. It summarises design beautifully. It cannot substantiate operation at all. Use it for the first. Do not let it pretend to the second.

Tessera Case: Tessera's preliminary readiness opinion

Here is the opinion Act I earns you, written for Tessera's board.

On the basis of a desk review of the documentation provided, Tessera has established an information security management system whose design is substantially aligned with ISO/IEC 27001:2022. The control set is documented across all four themes, the Statement of Applicability is present and internally consistent, and the supporting policies and procedures read as fit for purpose.

This is a preliminary opinion formed from documents alone. It does not constitute assurance that controls are operating effectively. The following remain unverified and are the evidence gaps Act II must close: confirmation that HR offboarding and access-revocation controls operate for every leaver; reconciliation of the asset register against the live estate; verification that supplier due diligence and cloud-service monitoring are performed, not

merely documented; and operational testing of the physical, access, and ISMS controls. The offboarding checklist, in particular, exists, but its execution cannot be substantiated from documents.

Conclusion: promising on paper; not yet proven in practice. Full-access fieldwork recommended.

That last line is the most important sentence in Act I. It is also the only honest one.

> Evidence Locker: lock this in (Week 6)
>
> This is the Act I culmination, and it feeds Assessment 2 directly. Lock your **preliminary readiness opinion** into the Evidence Locker: a one-paragraph opinion on what the documents substantiate (design maturity), followed by a named list of the evidence gaps that limited access could not close. Separate, explicitly, what you *found* from what you *suspect*. Mark each gap with the control it would test and the access you would need to substantiate it.
> This is not a draft to throw away in Act II. It is the agenda for Act II. Every gap you name here becomes a test you run once full access arrives, and the defensible opinion you ultimately stand behind is this opinion plus the evidence that closes those gaps, or fails to. Substantiate what you can. Name what you cannot. That discipline is the whole job.

6.8 The question, answered

How much can you honestly say about a company you have only ever read about? Enough to form a preliminary opinion, never enough to call it proof. You can establish that Tessera has *designed* a control set aligned with the standard, across people, physical, organizational, and the ISMS that binds them. You cannot, from documents alone, establish that any of it *operates*. The preliminary readiness opinion is the disciplined art of saying both, clearly, in the same breath.

That closes Act I. Six weeks of engagement, built on limited access, have produced what limited access can produce: suspicion, calibrated and named, with every evidence gap written down. You have engaged the case. You have not yet proved it.

Act II opens the door. Full access arrives: the interviews, the live configurations, the logs, the incident records, the systems you have so far only read about. The polished policy finally meets its operational reality, and the suspicion you have earned here either hardens into evidence or dissolves. There is only one way to find out which.

Next week the question changes tense, from *what does it look like on paper?* to *can you prove it works?* The question is **Prove?**

Part II

Act II · Prove: From Suspicion to Proof

Chapter 7

Prove? Evidence Collection and Testing

> *You can read your way to a suspicion. You can only evidence your way to an opinion.*

Tessera Case: Full access granted

The email lands on a Monday morning, and it changes the engagement. Tessera's CTO has signed the access schedule. You have credentials for the AWS environment with read-only audit permissions, exported log streams for the last twelve months, the complete HR leavers list, calendar slots with the head of people and the IT operations lead, and a standing invitation to walk the Perth office. For six weeks you have been reading about Tessera. From today you are looking at it. The preliminary opinion you wrote at the close of Act I, *promising on paper, not yet proven in practice*, now meets the only thing that can upgrade or overturn it: the evidence itself.

7.1 The act break: from suspicion to proof

Act I ended with a disciplined confession. You had read the ISMS, the risk register, the Statement of Applicability, and every policy beneath them, and formed an honest opinion: designed well, operating unconfirmed. Every gap you named at the close of Chapter 6 was a control you had read about and never seen run. Full access is the thing that closes those gaps.

Notice the posture change. In Act I you accepted documents because documents were all you had; your scepticism was sharp but pointed at the

page. Now it has somewhere to land. A policy that says access is revoked within twenty-four hours of a leaver's last day is, on the page, a promise. A log entry showing the account disabled at 16:11 that day is the promise kept; one showing it still active a week later is the promise broken. Both are evidence, and only full access could have produced them.

Here is the trap that comes with the door opening. Access feels like proof, and that feeling is the new danger. A dashboard, a log, a configuration export, a confident engineer: each can look like evidence and still fail to substantiate anything. This chapter is the craft that turns raw access into a defensible opinion, which evidence to gather, how much, how to keep it so it survives challenge, and which procedures prove a control operates rather than merely exists.

7.2 Four kinds of evidence

Every conclusion rests on evidence, and evidence is not all the same stuff. The profession sorts it into four kinds, and knowing which kind you are holding tells you how much it can carry.

Observational evidence is what you see yourself: a leaver's badge denied at the side entrance, a privileged elevation granted, used, and revoked beside you. It is the most direct evidence that a control *operates*, with a hard limit: it only proves operation for the moment you watched. A control that ran perfectly under your gaze can fail the day after you leave.

Documentary evidence is the records the organisation and its systems produce: the access-revocation log, the CloudTrail export, the signed offboarding checklist, the supplier's SOC 2 report. This is the bulk of what Act II gathers, and its reliability depends on its source and the controls over how it was made. A system-generated log nobody can alter is strong; a spreadsheet a manager typed last week is weak.

Analytical evidence is your own work: reconciling the asset register against the live estate, re-running the calculation behind a tenant-isolation check, computing a deviation rate from a sample. It is powerful precisely because you produced it on terms you set, independent of the organisation's goodwill.

Oral evidence is what people tell you: the interview, the attestation, the corridor explanation. It is the weakest kind alone, because memory is selective and self-interest is real, but it is also where almost every other kind of evidence begins. A good interview does not conclude anything. It points you at the document, the log, or the observation that will.

There is a persuasiveness hierarchy worth memorising. Evidence from outside the organisation beats evidence from inside it; what you obtain

yourself beats what the client hands you; the original record beats a copy, a copy beats a summary, and a contemporaneous record beats a reconstruction. Keep that ordering in mind and you will feel the right scepticism for each artefact you are handed.

7.3 Sufficiency and appropriateness

Two words do most of the judgement work, and they answer different questions. **Sufficiency** is the *quantity* of evidence: is there enough of it to support the conclusion? **Appropriateness** is the *quality*: is it relevant to the assertion, and is it reliable?

Appropriateness has two halves, and students trip on the second. **Relevance** asks whether the evidence actually tests what you are trying to prove: a clean access-revocation log is irrelevant if your assertion is about badge access. **Reliability** asks whether you can trust the source: was the record produced by a system whose integrity is controlled, at or near the event, by someone without motive to distort it?

The mistake is to chase one axis and ignore the other. A mountain of irrelevant evidence proves nothing, however large the pile; a single relevant item is a start but not a conclusion, because sufficiency is a property of the population, not of one match. High assurance needs both, enough of the right kind, and the craft of this chapter is deciding, control by control, how much of what kind will defend the opinion.

⚠ Watch out: the evidence that isn't

Three artefacts look like evidence and are not, and each sinks an audit that trusted it. A **screenshot** with no timestamp, source, or chain of custody is a picture someone made; it could be staged, old, or edited. A **self-attestation**, the manager's confident "we do that," is oral evidence, the weakest kind, and worthless uncorroborated. An **AI summary** standing in for the underlying logs is a secondary account of a primary record, and summarisation always loses the detail that decides whether a control actually ran.

Then the sufficiency trap, subtler and more common. One corroborating item does not prove a control operates across a population. A single leaver whose access was revoked on time is reassuring; it is not evidence the control works for every leaver. Sufficiency asks how many, across how much, with what confidence. A single match answers none of those questions, and treating it as if it did is the same error as trusting a screenshot: mistaking the appearance of evidence for evidence itself.

7.4 Sampling: why you cannot test everything, and how to size what you do

Chapter 4 decided *where* to dig. Sampling decides *how much*. You cannot test every login, every leaver, every firewall rule across a year, so you test a defensible subset and reason about the whole from it. Done badly, sampling is a licence to miss the thing that matters. Done well, it is the most rigorous thing an auditor does, because the rigour is in the design, not the counting.

The first rule, and the one most often broken, is that a sample proves nothing about a population you have not completely defined. To say something about "every leaver in the period," you need the complete list of every leaver, reconciled to a source you trust, such as payroll. A sample drawn from a convenient subset, the leavers IT happens to remember, proves something about that subset and nothing else. Population completeness comes first, always.

For testing controls, the tool is **attribute sampling**. You are not measuring a sum of money; you are estimating a rate, the proportion of times a control deviated from its objective. The leaver's access was either revoked in the policy window or it was not, each test a yes or a no, and the sample lets you estimate the deviation rate across the whole population.

Here is where the discipline bites. The sample size is not a number you feel; it is derived from three things. The **tolerable deviation rate** is how much failure you can accept and still conclude the control works: tighter tolerance, bigger sample. The **expected deviation rate** is what you think you will find: the worse you expect, the more you test. The **assurance level** is the confidence the engagement requires: certification work demands high assurance, and high assurance means a larger sample. Set those three and the size follows. Pick a round number because it sounds thorough, and you have a sample you cannot defend.

7.5 The five testing procedures

You have your population and your sample. Now you need procedures that actually extract evidence from each item. The audit toolbox has five, rising in persuasive power as you move down the list.

Inquiry is asking. You ask the IT operations lead how offboarding runs. Inquiry is where nearly every test begins, because it tells you where the evidence lives, and where nearly every test must not end, because what someone tells you is oral evidence, the weakest kind. "We disable accounts within twenty-four hours" is a lead, not a finding.

Observation is watching a control run: you attend an offboarding and observe the steps being performed. It proves the control operated at the moment you watched, and only that moment. It is excellent for physical and process controls and useless for anything that has to hold across the year.

Inspection is examining records and assets: the access-revocation log, the signed checklist, the CloudTrail export, the configuration file. Inspection is the workhorse of IS audit, and most of Act II is inspection done well, the right record, the right period, read sceptically.

Re-performance is re-running the control yourself, independently. Rather than trust the reconciliation was done, you redo it; rather than trust the access review caught the orphaned account, you pull the live IAM state and look yourself. It is powerful because it does not depend on the organisation having done anything correctly: you are checking the outcome, not the report of the outcome.

Re-computation is the numeric cousin: you recompute the calculation a control is supposed to perform, whether a deviation rate, a key-rotation interval, or a backup-retention tally. It turns an assertion into a number you can verify.

The order matters, because persuasive value rises as you go. An opinion built on inquiry alone is built on talk; one built on inspection, re-performance, and re-computation is built on what the evidence shows. State the control objective, pick the procedure that tests it at the assurance the engagement demands, and reach for the stronger procedure whenever the control is critical.

> **i** AI Field Note: AI drafts the sampling plan and the test steps
>
> This is the kind of structured work an AI drafts fluently, and so the kind you must evaluate hardest. Ask it to draft the work package for testing Tessera's access-revocation control and it returns a clean document in seconds: sampling plan, test steps, results table. Read it as an evaluator, because it is wrong in two instructive ways.
> First, the sample size. The AI proposes "test 25 leavers." A round, confident number with no basis. It carries no population, because the AI never asked how many leavers there were, and no tolerable deviation rate, expected rate, or assurance level, because it never fixed them. The number was chosen for how it reads, not derived from what the opinion requires. Your correction is to set the parameters the AI skipped: the complete HR leavers list as the population, a tight tolerable deviation rate, a low expected rate, and the high assurance a certification engagement demands. The derivation lands

higher than twenty-five, and against a small population backing a control this critical, the defensible answer may be to test all of them. The AI picked a number. You derived one.

Second, the test steps. The AI drafts three. "Verify that access was revoked within twenty-four hours of the leaver's last day." Performable, if the revocation timestamps exist, and they do. "Confirm the leaver's manager was notified within the policy window." Not performable: Tessera produces no record of that notification, so the step collapses into pure inquiry and proves nothing. "Determine whether the leaver still holds production access." Conflates *now* with *then*. For a departure three months ago, you cannot observe the state on their last day from a current snapshot; you depend on the historical audit trail, not a live check. The AI wrote steps that read as tests. Two of the three cannot be executed as written.

The AI produced the draft quickly and well-structured, and that is the trap. The sample size was a guess dressed as a method; the test steps were plausible sentences that do not survive contact with the evidence Tessera actually keeps. You supplied the parameters, rewrote the steps to what the trail can support, and turned a generic work package into one that will hold up under review. That is the drafter-to-evaluator shift, on the artefact that matters most this week.

7.6 Chain of custody

Evidence you gather has to survive challenge, sometimes years later, in a certification review, a board meeting, or a regulator's inquiry. That survival depends on chain of custody: a complete, defensible record of who collected what, when, from where, by what method, and what happened to it afterwards. An extract with no provenance is just a file someone made, and "someone made a file" is not a basis for an opinion.

The mechanics are unglamorous and non-negotiable. Every extract gets a hash, typically SHA-256, recorded at capture so any later alteration is detectable. Copies are read-only. Each artefact carries its source system, the query or export that produced it, the date and time in a clear timezone, and the name of the auditor who took it. Transfers between people, the log moving from the analyst who pulled it to the lead who filed it, are themselves logged. The whole chain is sealed into the working papers, and another auditor following them should be able to trace every conclusion back to a specific artefact captured on a specific day.

I learned this the hard way. The first time full access landed, I went straight for the logs and forgot to record how I got them. Three weeks

later a reviewer asked where a particular extract had come from, and I could not say. The extract was fine. Its provenance was gone. An evidence extract with no provenance is not evidence; it is an unsupported claim, which is exactly what we refuse to accept from the client. The discipline cuts as deep for us as it does for them, and it should.

7.7 Auditing AI: can you trust a log an AI produced?

This is the question earlier chapters promised to take up in full, and full access is where it lands. Tessera runs AI in two places inside the boundary you drew in Chapter 4: a support assistant that triages alerts, and an anomaly detector in the logging stack. Both were named as evidence gaps at the close of Act I. Both now produce logs, and the question is whether those logs count as evidence you can stand behind.

Take the assistant. When it auto-resolves a low-priority alert, the log entry recording the resolution is, in a real sense, AI-produced. Can you trust it? Treat it as any system log, then add a layer. A log's reliability always depends on the controls over how it was generated, but an AI's record of its own action has a failure mode a deterministic system's log does not: it can confabulate, truncate, or summarise away the detail that would tell you whether the action was correct. The tidy "alert reviewed, no action required" may faithfully record a sound decision, or paper over a misjudgement; you cannot tell from the summary. Corroborate against the underlying system's own raw audit trail, the original alert event and the downstream actions, and treat the AI's log as one account among several, not the account of record.

Now the harder question, the one Chapter 5 raised: how do you test a control that an AI runs? The assistant triaging alerts is itself a detective control, and testing whether it *operates* means measuring its decisions against ground truth. Did it escalate what it should have? Auto-close what it should not have? That needs a labelled set, a sample of alerts where you know independently what the right outcome was, so you can measure the assistant's deviation rate the way you measure any control's. The five procedures apply: inquire how it was tuned, observe it on a live alert, inspect its decision log, then re-perform by re-judging the cases yourself against the labels. Re-performance, for an AI-run control, is exactly the drafter-to-evaluator posture this book argues for: the machine decided, and you are the evaluator who decides whether the decision holds.

The conclusion is steadier than the anxiety around AI usually allows. You can audit a control an AI runs, and use evidence an AI helped produce, on one condition: never let the AI's account of itself be the

only account. Corroborate against the raw trail, test its decisions against ground truth, and treat its summaries as leads to the primary evidence rather than the primary evidence itself. An AI's log is admissible. It is not self-authenticating.

7.8 Interviewing for evidence

Interviews are oral evidence, the weakest kind alone, and also where almost every stronger kind begins. The skill is running them so they open doors instead of closing opinions.

Prepare before you sit down. Know the control objective you are testing and the evidence you seek. Ask open questions: not "do you revoke access within twenty-four hours," which invites the answer you want, but "walk me through what happens when someone leaves." Let silence do the work; people fill it, and what they fill it with is often the detail that matters.

The rule that turns an interview into an audit: never accept "we do that" without the record. Every claim about a process is a hypothesis until you produce the artefact that proves it. "Show me the last three times this ran" is the sentence that converts an attestation into a test. When the head of people says offboarding is always completed, ask for the last three checklists; when the SRE says privilege elevation is time-boxed, ask for the elevation log. Note where the claim and the record diverge: that is where the findings live.

Corroborate across interviewees. The IT operations lead and the head of people should describe the same offboarding in compatible terms. Where their accounts differ, one describes aspiration and the other practice, and the documents will tell you which. A good interview produces leads. The records produce the conclusions.

Tessera Case: A sampling plan and a chain-of-custody log for access revocation

The artefact for this chapter is the test that closes the single most important gap Act I named: confirmation that the access-revocation control operates for every leaver. The control spans A.6.5 (responsibilities after termination or change) and A.8.3 (information access restriction), and its objective is testable: *for every leaver in the period, all logical access to Tessera systems is revoked within twenty-four hours of the last working day, and the revocation is logged.*

The **population** is every leaver in the twelve-month period, sourced from the complete HR list and reconciled against payroll to confirm completeness. There are forty-eight. The **sample** is derived, not chosen: a tight tolerable deviation rate, a low expected rate, and the high assurance

the engagement demands. Against forty-eight leavers backing a control this critical, the defensible answer is to test all of them, because sampling a fraction and extrapolating buys little when the whole population is small enough to test in full.

The **procedure** runs per leaver: inspect the HR termination record for the last working day, inspect the IAM and single-sign-on audit logs for the revocation event, recompute the elapsed time, and flag any deviation over twenty-four hours. For every deviation, inquire for the cause and corroborate against the offboarding checklist.

Every artefact the test touches enters a chain-of-custody log. A representative slice:

Artefact	Description	Source	Captured	By	Hash (prefix)
EVD-07-01	HR leavers list, full period	HR system export	2026-06-15 09:12 AWST	M. Borck	`3f9a…`
EVD-07-02	IAM revocation log, leaver A	AWS CloudTrail	2026-06-15 09:41 AWST	M. Borck	`b1c7…`
EVD-07-03	SSO session log, leaver A	Identity provider	2026-06-15 09:46 AWST	M. Borck	`8e2d…`
EVD-07-04	Offboarding checklist, leaver A	HR document store	2026-06-15 10:03 AWST	M. Borck	`4a55…`

Each row is one extract, hashed at capture and sealed into the working papers. When the test finds forty-five of forty-eight revocations inside the window and three outside, that finding is traceable, leaver by leaver and artefact by artefact, back to the records that produced it. That traceability is what makes the conclusion defensible rather than merely asserted. You can run the same test against the live companion site at tessera.locoensayo.org.

Evidence Locker: lock this in (Week 7)

Act II's locker stops being narrative and becomes working papers in earnest. Lock in three things this week.
First, a **sampling plan** for one Tessera control of your choosing: the population defined and reconciled for completeness, the control objective stated as a testable assertion, the tolerable and expected deviation rates and the assurance level stated explicitly, and the sample size *derived* from them rather than chosen. Second, a **chain-of-custody record** for every artefact the test touches: artefact, source, capture time, extractor, hash. Third, an **interview-notes template** that separates, on every line, what the interviewee *claimed* from what you subsequently *corroborated*, so that aspiration and practice never blur on the page.
These three are not exercises. By Week 12 they are the spine of your Assessment 3 working papers, and the discipline they enforce, sufficiency derived not assumed, custody recorded not implied, claims corroborated not trusted, is what makes an opinion defensible.

7.9 The question, answered

Can we substantiate our suspicions? Yes, when the evidence is sufficient in quantity and appropriate in quality, gathered under a chain of custody that survives challenge, and tested by procedures strong enough to prove operation rather than existence. Act I left you with suspicion, calibrated and named. Act II gives you the means to convert it, item by item, into proof. The access-revocation control is no longer a promise on a page. It is forty-eight tested log entries, each traceable to a hashed extract captured on a known day, and the deviation rate they reveal is a number you can defend.

That is the shift this chapter enacts, and the shift the rest of Act II runs on. The polished policy meets its operational reality, and every suspicion you earned in Act I either hardens into evidence or dissolves. The evidence has started to speak. The question now is whether the policy behind the controls was ever any good, because a control can run perfectly and still serve a policy whose intent has drifted from its operation.

Next week the question turns from *can you prove it works?* to *does the policy itself hold up?* The question is **Comply?**

Chapter 8

Comply? Policy Evaluation: From Published to Proven

A policy that nobody follows is not a control. It is a PDF.

Tessera Case: The library, reopened

Tessera's document library looked finished in Act I. Now you have the keys to the building. Full access means the policy set is no longer something you read; it is something you can test. The information security policy and its supporting standards are still comprehensive on paper. This week's discipline is to find out whether any of that paper survives contact with the people, the systems, and the logs it is supposed to govern.

8.1 What full access changes

Act I closed with calibrated suspicion. Act II opens with the thing that converts suspicion into evidence: access. You can now interview the people named in the policies, log into the systems the policies describe, pull the logs the policies say are kept, and read the meeting minutes where the controls are supposed to be reviewed. Each is a different kind of evidence, and Chapter 7 taught you how to weigh it. This chapter teaches you what to point that evidence at: the policy set itself.

Policies are the soft tissue of an information security management system. They are also the part an organisation finds easiest to produce and hardest to live by, because writing a policy is a writing problem and following one is a behaviour problem. A fast-growing company like Tessera can write

an excellent policy library in a quarter with a good consultant and a long weekend. It cannot build the habits that make those policies operate in anything like the same timeframe. The gap between the two is where this chapter lives.

8.2 Compliance is not effectiveness

Here is the distinction the chapter title is built on, and it is the one auditors get wrong when they are tired. **Compliance** asks whether a control exists and matches the words of a standard. **Effectiveness** asks whether the control actually achieves its objective when it runs. The first is a paperwork question. The second is the only question that matters.

A policy that satisfies every clause of ISO/IEC 27001:2022 A.5.18, that quotes the standard's own language back at it, that carries a version number and an approval signature, is *compliant*. It may also be unenforceable, unowned, or unknown to the people it binds, in which case it is compliant and worthless. Compliance is the floor. Effectiveness is the opinion.

The standard makes this explicit in its own architecture. Clause 9.1 requires the organisation to evaluate the *effectiveness* of its information security management system, not merely confirm it exists. A.5.35 (independent review of information security) and A.5.36 (compliance with policies, rules and standards) exist specifically to make sure the organisation checks whether its own policies are followed, not merely filed. The standard itself separates design from operation. The auditor's job is to make the same separation, with evidence.

So the question for Act II is never "does Tessera have a policy on X?" That was an Act I question, and the answer was almost always yes. The Act II question is sharper: *does the policy on X actually operate, and does the evidence show it?*

8.3 The A.5 backbone

ISO/IEC 27001:2022 clusters its ninety-three controls into four themes, and Theme A.5, the organizational family, is the largest: thirty-seven controls, A.5.1 through A.5.37. Where the people controls (A.6) govern the human lifecycle, the physical controls (A.7) govern offices and devices, and the technological controls (A.8) govern systems, A.5 governs the organisation itself: its policies, its roles, its assets, its suppliers, its information, its incidents, and its compliance.

Think of A.5 as the governance backbone. It opens with the policies themselves (A.5.1, policies for information security) and the roles accountable for them (A.5.2, A.5.3, A.5.4). It sweeps up asset management (A.5.9

through A.5.13), access governance (A.5.15 through A.5.18), supplier and cloud relationships (A.5.19 through A.5.23), incident management (A.5.24 through A.5.28), continuity (A.5.29, A.5.30), legal and privacy obligations (A.5.31 through A.5.34), and the review and compliance machinery meant to hold it all together (A.5.35, A.5.36, A.5.37). Strip out A.5 and the other three themes have nothing to attach to, because A.5 is where the organisation says what it will do. The other themes are where it does it.

That makes A.5 the natural home for the policy-effectiveness question, and it makes the Statement of Applicability's status column the first place an auditor looks for the gap. In Act I you read that column and noted, with discipline, that "implemented" is an assertion. In Act II you can test it. Pick a control, take the policy at its word, and find out whether the word is true.

8.4 From published to operating: how to test a policy

Testing whether a policy operates is a four-step move, and it works on any A.5 control. Learn it once and reuse it.

Read the policy for its operational claims. Most policies mix aspiration with obligation. Your job is to extract the obligations: the things the policy says *will* happen, on a cadence, by a named owner, to a defined standard. "Access rights are reviewed quarterly" is an operational claim. "Security is a core value" is not. Underline the claims. Each one is a test waiting to be designed.

Identify the evidence the claim would produce if it were true. A quarterly access review leaves a footprint: a signed review record, a ticket, a change log, a meeting minute. A 24-hour leaver-revocation control leaves a footprint: a revocation timestamp in the identity provider, correlated against the HR termination date. If the control operates, the world looks a particular way. Decide what that way is before you go looking, so you test the claim rather than your own surprise.

Gather the evidence, then compare it to the claim. This is where full access earns its keep. Pull the population (every leaver, every privileged account, every supplier), not the sample the client offers you. Trace the operational claim through the population. Count the exceptions.

Form a judgement on effectiveness, not existence. The output of a policy test is not "compliant / non-compliant." It is a statement about how reliably the control achieves its objective, on the evidence, across the population you tested. A control that operates for 95 per cent of the population and fails for 5 per cent is not "mostly compliant." It is a

control with a measured exception rate, and the consequence of that rate depends entirely on which 5 per cent.

The tests themselves come in four flavours, and ISO/IEC 19011 and the wider assurance canon all recognise them. **Inquiry** is asking; treat the answer as a lead, never as evidence on its own. **Inspection** is reading the record the control should have produced. **Observation** is watching the control run. **Re-performance** is doing the control's check yourself, on the client's data, and seeing whether you reach the same answer. Inquiry is the weakest; re-performance is the strongest. A policy that survives only inquiry has not really been tested.

Tessera Case: A.5.18 access rights, tested

Tessera's access management standard is two pages and reads cleanly. The operational claims are explicit: access is granted on least privilege (A.5.15); access rights are reviewed at least quarterly (A.5.18); privileged accounts are reviewed monthly; and access is revoked within 24 hours of an employee's last working day. Four claims. You design four tests.

The leaver test. HR gives you nine terminations in the last twelve months. You pull the identity provider's revocation log and the AWS IAM credential last-used data, and you correlate. Three of the nine were revoked inside the 24-hour window. The median was six days. One contractor's static AWS keys stayed active for forty-seven days after the contract ended, and they held production read access to the customer database. The policy says 24 hours. The evidence says six days, sometimes seven weeks.

The quarterly review test. The policy says every system owner runs a quarterly access review. You ask for the review records. One exists: a spreadsheet, signed off once, covering fourteen of the forty-odd systems in the asset register. The other systems have no review record, because nobody asked for one. "Quarterly" is operating for a third of the estate and not at all for the rest.

The privileged review test. No evidence. In interview, the CTO is candid: "We check privileged access when someone joins or changes role. We don't have a standing monthly review." The policy was written aspirationally and never built into a calendar.

The least-privilege test. You re-perform it. You take the ten most privileged IAM principals in the AWS estate and compare their attached policies to their stated job function. Three have broader access than their role requires, including one that can delete production databases.

The policy is comprehensive and largely accurate. The control is partially operating, with a measured exception rate that is unacceptable for a company about to handle regulated customer data. You map each gap to

A.5.18 and A.5.15, and you note the deeper failure: A.5.36 (compliance with policies) and A.5.35 (independent review) are the two controls meant to catch exactly this drift, and neither has caught it. The library was compliant on paper. The estate was not.

This is what Act II looks like when it lands. A document that read as assurance in Act I now reads as a finding, and the only thing that changed is that you have the access to test it. Notice the shape of the judgement. You are not calling Tessera dishonest; the policy was written in good faith. You are reporting that a control's design and its operation have diverged, by how much, and for which people and systems. That precision is the difference between an opinion and an impression.

> ⚠ Watch out: the compliant-on-paper trap
>
> The most expensive mistake in policy evaluation is grading the document instead of the behaviour. A policy can satisfy every clause of the standard and still fail three ways that no document review will ever catch. It can be **unenforceable**, with no mechanism behind it: a policy that says "review quarterly" but wires no review into any system or calendar. It can be **unowned**, where the name in the policy's approval block left the company eighteen months ago and nobody inherited the obligation. Or it can be **unknown**, where the people the policy binds have never read it, never acknowledged it, and could not describe what it asks of them.
> All three read as fully compliant to anyone who only opens the file. None of them is a control. Treat "we have a policy" and "the control operates" as different sentences, and refuse to let the first stand in for the second until the evidence says it can.

I will tell you the confession that goes with this chapter, because it took me an embarrassing number of engagements to learn it. I used to be quietly pleased when a client handed me a thick, well-organised policy library. It felt like half the work was done. Then I started testing the policies against the estate, and I noticed a pattern I should have predicted: the organisations with the most impressive policy libraries were often the ones whose controls operated worst, because the library was where the effort went, and the operation was where it didn't. A thin policy that someone actually follows beats a beautiful one that nobody does. I stopped being impressed by policy libraries. I started being impressed by review records.

AI Field Note: The policy-gap analyst

This is one of the tasks where an AI genuinely earns its place, and it is worth being precise about where it helps and where it lies. Hand the model Tessera's access management standard, the information security policy, and the Statement of Applicability, and ask it to flag where the policy text is thin, ambiguous, or inconsistent with A.5. It is good at this. It will catch that the policy never defines what counts as a "privileged" account, that the contractor clause contradicts the leaver-revocation window, and that no retention period is specified for access logs. Those are real findings, surfaced in minutes, and they save you an afternoon.

Then ask the question that matters: "Is Tessera compliant with this policy?" Watch what happens. The model has read the same documents you have, and documents contain no operational evidence. Pressed for an answer, it will do what these models do: it will infer compliance from design quality. It will tell you, with the confidence of a system that has never met a human, that "quarterly reviews appear to be conducted, as required by the policy." That sentence is a confabulation. The policy *requires* the reviews; the model has mistaken the requirement for the performance. It has slid from "the policy says this happens" into "this happens."

Your override is the whole job. Mark every such claim "unverified, pending operational evidence." The AI drafted the gap analysis; you supplied the one judgement it cannot, which is the difference between a promise on paper and a footprint in a log. That is the drafter-to-evaluator shift, demonstrated on a single page.

Auditing AI: The policy that forgot the AI

There is a policy-effectiveness gap that is specific to this moment, and Tessera has it. The acceptable use policy (A.5.10) was written before the product's summarisation feature shipped and before staff began pasting customer data into consumer AI tools to draft replies. It says nothing about either. So an organisation that is "compliant" with its acceptable use policy is, in the same breath, silently permitting exactly the data movement the policy was written to control, because the policy predates the behaviour.

Auditing this means asking not whether the policy exists but whether it covers the AI usage the estate actually contains. Look for a clause on approved AI services, on what data may and may not be entered, on retention by the AI vendor, and on disclosure to customers. If those clauses are absent, the control is designed for last year's

company. Policy effectiveness is a moving target, and AI moved it faster than most policy libraries update.

Evidence Locker: lock this in (Week 8)

This week's artefact is the policy-effectiveness test, and it is the one most directly analogous to real working papers. For each A.5 control you tested, record: the **operational claim** you extracted from the policy (the sentence, quoted), the **test** you designed (type: inquiry, inspection, observation, re-performance), the **population** you drew, the **evidence** you gathered, and the **result**: the exception rate, the specific failures, and your effectiveness judgement.
End with a **gap map**: every control where design and operation diverged, listed by A.5 clause, with the consequence of each gap named in one sentence. By Week 11 this gap map becomes your findings, and by Week 12 it becomes the evidence behind your opinion. A finding that says "A.5.18 partially operates, 9 leavers tested, 3 within window, median 6 days, 1 at 47 days" is defensible in any room. A finding that says "access control needs improvement" is not. Lock in the first kind.

8.5 The question, answered

Do the policies hold up, and do they match reality? The policies hold up. Tessera has built a policy library that is comprehensive, coherent, and well-mapped to the standard; that is a real achievement, and an organisation that cannot produce it is not ready. But the policies do not, not yet, match the reality of the estate they govern. Tested against access, the control operates inconsistently, with exception rates a certification body will not accept and a meta-control (A.5.36) that should have caught the drift and did not. The library is compliant on paper. The operation is the work that remains.

That is the Act II posture, stated for policy. Documents asserted; evidence now decides. The polished standard that read as assurance in Act I has met its logs, its meeting minutes, and its missing review records, and the meeting has been honest in a way the desk review could never be.

Next week the question turns to the thing those policies are ultimately written to protect. The policies are the promise; the data is the asset. *Whose data is Tessera holding, what does the law require of it, and does the estate actually protect it?* The question is **Protect?**

Chapter 9

Protect? Data Protection and Privacy

Security asks whether the wrong hands reached the data. Privacy asks whether you had the right to hold it at all.

Tessera Case: the crown jewel

Full access landed on Monday, and the first thing you traced was the data itself. Tessera's crown jewel is not its source code or its brand. It is its customers' data: contract terms, account configuration, usage telemetry, and the operational reports its customers' own staff upload for Tessera's new AI feature to summarise. In Act I you could read the privacy policy and the data classification scheme and nothing more. Now you can follow the bytes. The question that opened the engagement, *is it actually defended?*, stops being rhetorical this week. It becomes a thing you can answer with a data-flow diagram, a contract clause, and a deletion log, or fail to.

9.1 The crown jewel, and why privacy is not security

Every audit has a thing it is really about, and for Tessera that thing is personal information. Customer contact records. The named individuals inside customer organisations who log in. The contents of the reports those individuals upload. Protect that, and the certification means something. Fail to, and the rest of the control set is decoration.

Here is the distinction this chapter turns on, and it is the one people get

wrong first. Security and privacy are related but they are not the same property. Security is about protecting information: keeping it confidential, keeping it intact, keeping it available. It asks *is the data safe?* Privacy is about the rights of the people the data describes: whether it was collected lawfully, used for the purpose it was collected for, disclosed only with basis, and held no longer than needed. It asks *is the data yours to hold, and are you handling it as you promised?*

A company can have excellent security and still breach privacy law. A perfectly encrypted database of customer records, sent to a third party overseas without a lawful basis for the transfer, is a privacy breach that no cipher will cure. The reverse holds too: a company scrupulous about collection notices can still leak the lot through a misconfigured storage bucket. The two disciplines protect the same asset from different directions, and an auditor who conflates them tests the wrong thing.

The 2022 standard recognises this. ISO/IEC 27001's organisational control A.5.34, *Privacy and protection of personal information*, is explicit that an organisation's privacy obligations sit alongside its security obligations, and that meeting one does not discharge the other. Audit data protection and you are auditing both, against two rulebooks: the technical control objectives from Chapter 5, and the legal regime that governs personal information. For Tessera, that regime is the Australian *Privacy Act 1988* and its Australian Privacy Principles.

I learned the cost of the conflation the hard way, on an engagement I would rather not describe in detail. The client answered every privacy question I asked with a cipher specification, and I wrote it down and moved on. Three weeks later the cross-border transfer I had not chased became the actual finding, and the finding was a notifiable breach. Encryption had been perfect. The legal basis had been absent. That is the gap this chapter exists to close.

> ⚠ Watch out: 'we encrypt it' is not a privacy answer
>
> The first answer most engineers give to a privacy question is an encryption answer. Asked whether customer data is protected in transit to the AI vendor, they will say "TLS 1.3, end to end." Asked whether it is protected at rest, they will cite AES-256 and the key-management configuration. Both answers are correct. Neither is a privacy answer.
> Encryption addresses APP 11, the security principle. It does not address APP 3 (was the data collected lawfully?), APP 6 (is it being used only for the purpose disclosed?), or APP 8 (is the cross-border disclosure to the vendor lawful?). A data set can be encrypted to

> military grade and still be a notifiable breach waiting to happen, because the breach is in the handling, not the cipher. When a client answers a privacy question with a security control, write it down, and then ask the question again.

9.2 The Australian Privacy Principles

The *Privacy Act* regulates how APP entities handle personal information, and the thirteen Australian Privacy Principles are its operative rules. An APP entity is, broadly, an organisation with an annual turnover above $3 million, plus a list of smaller entities caught regardless of size (health service providers, traders in personal information, and others). Tessera's revenue sailed past the threshold a funding round ago. It is an APP entity, and the APPs bind it.

You do not need to memorise all thirteen to audit Tessera. You need to know which ones do the load-bearing work, and why.

APP 1 requires open and transparent management of personal information, manifested in a clear, current privacy policy. The Act I read tested whether that policy exists. The Act II test is whether practice matches it, and the gap is usually where the policy has not kept up with a feature the product team shipped.

APP 3 governs collection. An entity may collect only personal information that is reasonably necessary for its functions, by lawful and fair means. Sensitive information (health information, biometric data, political opinion, and the rest of the statutory list) attracts a higher bar, generally requiring consent. The audit question is whether Tessera collects what it actually needs, or hoovers up what is convenient.

APP 6 governs use and disclosure. Personal information collected for one purpose may be used or disclosed only for that purpose, for a secondary purpose the individual would reasonably expect, or with consent. This is the principle that decides whether Tessera may feed customer content through its summarisation feature at all, and it is the one most likely to surprise a product team.

APP 8 governs cross-border disclosure, and for a company that calls an overseas AI service it is the principle that will either save or sink the engagement. More on it below.

APP 11 requires an entity to take such steps as are reasonable in the circumstances to protect personal information from misuse, interference, loss, and unauthorised access, modification, or disclosure, and to destroy

or de-identify it when no longer needed. APP 11 is where privacy and security meet, and where secure deletion lives.

Read together, the APPs describe a lifecycle: collect lawfully, tell the subject, use it for the purpose you stated, protect it while you hold it, disclose it only with basis, and let it go when you are done. Auditing Tessera's privacy is auditing that lifecycle end to end, against the legal text rather than against a technical checklist.

9.3 The Notifiable Data Breaches scheme

The *Privacy Act* also runs the Notifiable Data Breaches scheme, and it is the scheme that turns a bad weekend into a regulated event. Under Part IIIC, an APP entity that becomes aware of an *eligible data breach* must notify the Office of the Australian Information Commissioner and the affected individuals as soon as practicable.

An eligible data breach is a three-part test, and the auditor should know it cold. First, there is unauthorised access to, unauthorised disclosure of, or loss of, personal information. Second, this is likely to result in *serious harm* to any of the individuals to whom the information relates. Third, the entity has not been able to prevent the risk of serious harm with remedial action. All three, and the notification obligation bites.

The phrase that does the work is *serious harm*, and it is deliberately not a number. Serious harm is serious physical, psychological, emotional, financial, or reputational harm. Assessing it demands judgement about the kind of information, its sensitivity, who holds it now, and what they could do with it. A breach reaching a list of public business contact names may be low harm; the same breach reaching a file of health-related reports is catastrophic. The scheme forces an entity to think, on the worst day of its year, about consequences it would rather not.

The scheme also has a clock. Once an entity suspects an eligible breach, it has a reasonable period to assess, and the Commissioner's guidance treats around thirty days as a reasonable upper bound. An auditor tests NDB readiness the way it tests every other readiness: does a playbook exist, who owns it, has it been rehearsed, and can the organisation answer the three-part test under pressure with its data inventory in front of it?

That last clause hides the trap. You cannot assess serious harm for data you cannot locate or describe. The NDB scheme silently assumes a working data inventory, and a company whose asset register is rotting, the Chapter 6 finding, discovers the cost of that rot in the middle of a breach assessment, when the thirty-day clock is already running.

9.4 Cross-border disclosure and data sovereignty

APP 8 is where modern SaaS architectures collide with a law written before they existed. Before an APP entity discloses personal information to an overseas recipient, it must take such steps as are reasonable in the circumstances to ensure the overseas recipient does not breach the APPs in relation to that information.

Read that sentence twice. It is an accountability transfer, not a liability transfer. Since the 2014 amendments, section 16C of the *Privacy Act* provides that, where an entity discloses personal information to an overseas recipient, the entity is taken to have breached the APPs if the recipient does an act that would breach them. Tessera cannot outsource the obligation. If the overseas AI vendor mishandles the data, it is Tessera's breach, on Tessera's record, notified under Tessera's name.

What counts as *reasonable steps* is the auditor's question, and the answer is unhappily open-ended. Reasonable steps may be contractual (a data-processing agreement that imposes APP-equivalent obligations and a right to audit), may be due diligence (confirming the vendor's certifications, sub-processor list, and data-handling configuration), and may be technical (encryption in transit, tokenisation of identifiers, minimisation of what is sent). The Commissioner expects an entity to have considered the steps and documented its reasoning, and an absence of either is the finding that writes itself.

Note what is absent. Australian privacy law has no equivalent of the European Union's Standard Contractual Clauses, the prescribed contractual instrument the GDPR treats as an adequate transfer mechanism. In Australia a contract is one reasonable step among several; it is not a safe harbour. If you have audited a GDPR compliance programme, unlearn that reflex here, and watch for the people who have not.

Data sovereignty is the adjacent concern, and it is not the same as residency. Where the data physically sits is one question. Which government can compel access to it, and under what law, is another. Tessera's core data lives in the AWS Sydney region (ap-southeast-2), which is the right answer for an Australian customer base that cares about residency. But residency is not the whole of sovereignty, and a single API call can undo it.

> ⚠ Watch out: the hidden cross-border transfer inside an AI API call
>
> A developer adds a feature. The feature sends a customer's report to a model vendor's endpoint to be summarised. From the application's

perspective it is a single function call. From a privacy perspective it is a cross-border disclosure of personal information to an overseas recipient under APP 8, and it triggers the full accountability regime of section 16C.
These transfers hide inside features, not contracts. The procurement team may have signed the vendor's master agreement and reviewed its data-processing terms. The engineering team that wired up the endpoint may never have been told that the payload contains personal information, or that the vendor processes it in a jurisdiction the privacy team has not assessed. The transfer exists in the architecture. It does not exist in the contract file until someone traces the data flow and names it. That tracing is Act II work, and it is the single most common place a privacy audit finds a problem the organisation did not know it had.

9.5 The Privacy Impact Assessment

The instrument that should have caught that hidden transfer before the feature shipped is the Privacy Impact Assessment. A PIA is a structured analysis of how a project, product, or change affects the privacy of the individuals whose information it touches. The Office of the Australian Information Commissioner publishes detailed guidance on conducting one, and it follows a recognisable shape.

A PIA starts with a threshold assessment: is this project likely to have privacy impacts significant enough to warrant a full assessment? If yes, the analyst describes the project, maps the personal information flows (what is collected, from whom, where it goes, who sees it, how long it is kept), tests them against the APPs, and identifies the risks to individuals and their treatments. The output is a living document, not a one-off compliance artefact, and it is only as good as the data flow map it rests on.

The data flow map is the heart of the exercise, and it is where the AI feature should have been caught. A correct map of Tessera's summarisation feature shows customer report content leaving Tessera's boundary, transiting to the model vendor, being processed in a location that may or may not be Australia, and being retained for a period the vendor's terms specify. Each arrow on that map is a question. Each question is an APP. Drawn accurately, the map is the audit.

The Tessera Case below is a fragment of the PIA that should have existed for this feature. It did not exist when you arrived. The Act II finding is that it now needs to.

> **AI Field Note: drafting the PIA data map**
>
> This is a task an AI accelerates dramatically, and it is worth using it, with the evaluator's brakes on. Hand an AI the feature spec, the privacy policy, and the vendor's terms, and ask for a first-cut data flow map with an APP-by-APP impact analysis. It will return a structured draft in minutes: the data elements, the recipients, the retention assumptions, the principles engaged. Most of it will be useful.
>
> Then read it for the invented clauses. Large language models are trained heavily on European material, and they reach for GDPR instruments reflexively. In the transfer-mechanism row, the AI will likely insert *Standard Contractual Clauses* as the safeguard that makes the cross-border disclosure lawful. That instrument exists in EU law. It does not exist in Australian law as a named, prescribed mechanism. Under APP 8 the safeguard is *reasonable steps*, and the accountability sits with Tessera regardless. Accept the clause the AI invented and you have imported a legal instrument that does not govern this engagement, and your PIA now asserts compliance it does not have.
>
> The correction is the audit. Walk every data flow the AI drew against the actual architecture: the endpoint's region, the vendor's documented retention, the contract's actual processing terms, the prompt-logging configuration. The AI produced the draft. You produced the truth. That is the drafter-to-evaluator shift, performed on a document whose only value is accuracy.

Tessera Case: a PIA fragment for the summarisation feature

Here is the fragment you drafted back to Tessera after tracing the flow, abridged.

Project: in-product summarisation of customer-uploaded reports via a third-party model vendor.

Personal information flows: - Report content uploaded by named customer users; may contain personal information of the customer's own staff or third parties named within the reports. - Content transmitted to the model vendor's inference endpoint over TLS 1.3. - Vendor processes in [region not pinned to a single location in the contract; routing across multiple regions permitted]. - Prompt-logging confirmed enabled on the account; vendor retention per terms [30 days for abuse monitoring]. - Training on customer inputs: data-processing agreement states the opt-out is configured; configuration to be verified against the account, not the marketing page.

APP analysis: - APP 3 (collection): collected for the summarisation

function the user requested. Likely necessary. Sensitive-information risk: reports may incidentally contain health or other sensitive data. Mitigation: pre-upload notice, or content classification. - APP 5 (notification): the current privacy policy does not disclose the AI processing or the vendor. Gap. - APP 6 (use and disclosure): summarisation is arguably within the purpose the user would reasonably expect. Secondary use of the content for vendor model training is not, and must be contractually excluded and technically verified. - APP 8 (cross-border): disclosure to an overseas recipient. Reasonable steps: the data-processing agreement is in place, but the processing region is unconfirmed and the sub-processor list is not attached. Accountability retained by Tessera under s.16C. Gap. - APP 11 (security): TLS in transit and encryption at rest confirmed. Tokenisation of customer identifiers before the payload leaves the boundary recommended.

NDB exposure: a breach at the vendor that exposes report content would be an unauthorised disclosure by an overseas recipient for which Tessera is accountable under s.16C. Where the content includes sensitive information, serious harm is likely, and notification to the Commissioner and to affected individuals would be required. Tessera's NDB playbook does not currently include a vendor-side breach path. Gap.

Recommendation: pause auto-enablement of the feature pending closure of APP 5, APP 8 region confirmation, training-opt-out verification against the live account configuration, and extension of the NDB playbook to the vendor-side path.

Each *Gap* in that fragment is an audit finding waiting to be written. Notice how many of them the desk read in Act I could not have produced.

9.6 Data masking, minimisation, and secure deletion

If the APPs describe the legal lifecycle, the 2022 control set describes how to engineer it. Four controls do most of the privacy work, and all four were added in the 2022 edition, which is why a privacy audit against the 2013 standard felt so thinly tooled.

A.5.34, *Privacy and protection of personal information*, is the organisational anchor. It requires the organisation to identify and meet its privacy obligations, which in practice means knowing which data is personal, who is responsible for it, and how the APPs are satisfied. It is the control under which the PIA lives.

A.8.10, *Information deletion*, requires secure deletion of information no longer needed, in line with the retention schedule and using techniques appropriate to the storage medium. Deletion is not `rm`. A record deleted

from a database may persist in backups, snapshots, replicas, and logs for weeks or months. Secure deletion means removal across every copy, backed by a retention schedule that decides when deletion is due. The auditor asks for the retention schedule and the deletion evidence, and reconciles the two.

A.8.11, *Data masking*, requires the use of masking, pseudonymisation, or other techniques to limit the exposure of personal information where the full data is not needed. This is the control that should govern every development database, every analytics warehouse, and every test fixture built from production data. Real customer records do not belong in a staging environment; masked or synthetic records do. Unmasked production data in a non-production environment is one of the most common findings an auditor writes, because it is one of the most common shortcuts an engineering team takes.

A.8.12, *Data leakage prevention*, closes the perimeter, requiring controls to detect and prevent the unauthorised exfiltration of information.

Underlying all of them is data minimisation, which is less a control than a principle: collect the least personal information consistent with the function, hold it for the shortest time consistent with the purpose, and expose it in the narrowest form consistent with the use. Minimisation is the privacy control that pays back in every other dimension. Less data stored is less data to breach, less data to mask, less data to delete, and less data to map in the PIA. A company that minimises well audits well, because there is simply less of it to get wrong.

9.7 The privacy of the AI you bolt on

The chapter's hardest question is also the most contemporary. Tessera's summarisation feature is not a privacy question only about the transfer. It is a privacy question about what the model itself does to personal information, and the 2022 control set does not yet carry a control labelled "AI." You audit it under the controls that exist, and you apply the scepticism the topic demands.

> **!** Auditing AI: privacy risks of the model itself
>
> An AI system introduces privacy risks that a conventional data store does not, and three of them belong in any privacy audit of an AI feature.
> *Training data retention and reuse.* A model vendor that ingests customer prompts as training data is folding that information into a statistical object it will keep indefinitely and expose, in fragmentary

form, through the model's future outputs. A data-processing clause that says "we do not train on your inputs" is the right clause. Verify it against the configuration that governs the account, not the marketing page. The configuration is the control; the clause is the assertion.

Model inversion and membership inference. These are the classes of attack that recover information about a model's training data from its behaviour. Model inversion reconstructs approximations of inputs from outputs; membership inference determines whether a specific record was in the training set. For a model trained on Tessera's customer data, these are privacy risks to Tessera's customers, and the risk scales with how much of their data is in the model. A vendor that does not train on your inputs removes the vector. A vendor that does, retains the risk permanently.

Prompts containing personal data. Every call Tessera's feature makes to the vendor is a payload, and the payload will contain personal information whenever the summarised content does. Each call is therefore a fresh APP 8 disclosure, a fresh entry on the data flow map, and a fresh item in the breach surface. The privacy of the feature is not a single assessment done at procurement; it is a property re-asserted on every request. Tokenising customer identifiers before the payload leaves the boundary, and minimising what is sent, are the engineering controls that keep it defensible.

Audit all three against evidence, not against the vendor's security page. The vendor's security page is the polished policy. The configuration, the data-processing agreement, the prompt-logging settings, and the architecture are the server behind it.

Evidence Locker: lock this in (Week 9)

This week's artefact is the privacy half of your working papers, and it has two parts. Lock in the **Privacy Impact Assessment fragment** for the summarisation feature: the data flow map, the APP-by-APP analysis, and each named gap. Then lock in the **data-flow-to-AI-vendor assessment**: the endpoint, the confirmed processing region (or the gap where it is unconfirmed), the prompt-logging configuration, the training-opt-out verification status, and the NDB-relevant gaps, with the control each gap maps to and the access you would need to close it.

Separate, as always, what you *found* from what you *suspect*. The cross-border transfer to the vendor is found; the lawful basis for it is the gap. The NDB playbook's missing vendor-side path is found; whether it would hold under a real breach is the gap. By the end

of Act II these become the privacy findings in your certification-readiness opinion, and the locker is where they earn the right to be there.

9.8 The question, answered

Is the crown jewel actually defended? On the evidence Act II has now let you gather, the honest answer for Tessera is: *partly, and the part that is missing is the part that matters most.*

The security of customer data is largely in place. Encryption in transit and at rest is configured. Access is controlled. The AWS residency keeps the core data in Australia. On the security axis, Tessera can defend the crown jewel.

On the privacy axis, the picture is rougher. The cross-border transfer to the AI vendor is a real APP 8 disclosure, made under a data-processing agreement whose region is unconfirmed and whose training opt-out is asserted but not yet verified against the live account. The privacy policy does not disclose the AI processing. The NDB playbook has no vendor-side path. The feature shipped without a PIA, and the fragment you drafted back lists four named gaps. Each gap is a place where the privacy lifecycle is broken, and a broken lifecycle is a notifiable breach waiting for its trigger.

The crown jewel is encrypted. It is not yet, fully, protected. The distinction is the work of this chapter, and it is the distinction your opinion now has to carry.

Next week the question changes shape. A defended crown jewel is still a crown jewel at risk from fire, flood, ransomware, and the region failure that takes a whole multi-tenant SaaS down with it. When the worst happens, the question is no longer whether you protected the data, but whether you can get it back. The question is **Recover?**

Chapter 10

Recover? Business Continuity and Incident Management

When (not if) the region goes dark, the runbook you never tested is the plan you actually have.

Tessera Case: When the region fails

Full access has changed the kinds of question you can ask. In week one you would have been grateful for a continuity policy to read; now you ask to see the failover runbook and the result of the last time it ran. Tessera's CTO points you to a 47-page business-continuity plan and a diagram showing a primary region, a standby region, and an RTO of four hours. It looks exactly right. So you ask the only question that matters: *when did you last fail over, and what happened?* The room goes quiet. That silence is your second piece of evidence, and it is the one this chapter is built around.

10.1 The break that is coming

Every control in the nine chapters behind you exists to stop something going wrong. This one is about what happens when they do not. Continuity and incident management are the controls that assume failure, and a mature organisation treats the question as *when*, not *if*. AWS regions fail. Databases corrupt. Ransomware encrypts. A contractor clicks the link. The discipline here is not to prevent those events, because that was the last nine chapters. It is to survive them, and to prove you can.

ISO/IEC 27001:2022 treats this seriously, and mostly inside the large organizational family. Information security during disruption (A.5.29) and ICT readiness for business continuity (A.5.30) are the two continuity controls, the latter new in the 2022 edition and worth a section of its own. Incident management is a five-control cluster: planning and preparation (A.5.24), assessment and decision on security events (A.5.25), response (A.5.26), learning (A.5.27), and collection of evidence (A.5.28). NIST CSF 2.0 gives the work two whole functions of its own: Respond (RS) and Recover (RC). The ASD Essential Eight, for its part, folds the same intent into its eighth and most underestimated mitigation: regular backups.

The posture shift of Act II matters here more than anywhere. In Act I you read the continuity policy and noted, correctly, that its design looked complete and that you could not tell whether any of it operated. Now you hold the runbook, the test schedule, the incident log, the backup configuration, and the people who would actually run it at 2am on a Sunday. The polished plan finally meets its evidence. What follows is how you make them meet, and how you form an opinion that survives the question every certification body eventually asks: *have you ever actually done this?*

10.2 RTO and RPO: derived, not picked

Two acronyms carry almost all the weight in continuity, and most of the mistakes. The Recovery Time Objective (RTO) is the maximum acceptable time to restore a service after a disruption: how long the business can be down before the damage becomes intolerable. The Recovery Point Objective (RPO) is the maximum acceptable loss of data, measured in time: how stale the restored data is allowed to be, which in practice dictates how often you back up or replicate. If the RPO is fifteen minutes and the last good backup is six hours old, the plan and the reality are two different documents.

Here is the part the textbooks underplay and the machine gets quietly wrong. RTO and RPO are not numbers you pick from a menu of industry defaults. They are *derived*, from a business-impact analysis (BIA) that asks, for each critical process, what the disruption actually costs hour by hour: revenue lost, customers breached, SLAs broken, regulators notified, reputation damaged, and, for some industries, people harmed. The BIA traces each process down to the systems and data it depends on, and the RTO is set at the point where the cumulative cost of downtime crosses from tolerable to unacceptable. It has to sit underneath a ceiling called the Maximum Tolerable Downtime (MTD): the absolute limit the organisation can survive, within which the RTO plus the work of making the recovered system usable must fit. RPO is set the same way, against the cost of

recreating or losing data.

This is why an RTO of "four hours" scrawled into a plan with no analysis behind it is not a control. It is a wish, and it is the single most common finding in continuity auditing, because the number looks responsible and proves nothing. The audit test is simple and uncompromising: show me the analysis that produced the number, and show me it is consistent with the promises you have already made. For a SaaS like Tessera that sells on uptime, those promises live in customer contracts. A 99.9% monthly uptime SLA allows roughly forty-three minutes of downtime a month; a stated RTO of four hours is not consistent with that SLA, because a single qualifying incident consumes the entire allowance five times over. The RTO either comes down, the SLA comes honest, or the gap becomes a finding.

i AI Field Note: Drafting the BCDR test plan

Ask an AI to "draft a BCDR test plan for a multi-tenant SaaS on AWS, including RTO and RPO targets," and you will have a fluent, well-structured document in under a minute. The testing methods will be right (tabletop, simulation, failover), the RACI will be sensible, and the schedule will look professional. It will also, almost certainly, have copied its RTO and RPO from whatever SaaS continuity plans it was trained on. You will see "RTO 4 hours, RPO 1 hour" presented as if those were universal constants, with no derivation, no BIA, and no mention of what Tessera has actually promised its customers.

This is the exact point where the drafter hands back to the evaluator. The AI's defaults are not wrong because the numbers are impossible; they are wrong because they are unattributed. Your correction is to send the numbers back to their source: tie the RTO to the 99.9% uptime clause in the enterprise contract (which it contradicts), tie the RPO to the volume of transactions Tessera processes per hour (which the AI never asked about), and require a per-service BIA with an MTD ceiling. Then ask the question the AI cannot answer: *was this test ever actually run?* The plan it drafted is a competent first cut. The derivation is your judgement, and the proof of execution is your evidence. Use it for the first; never let it pretend to the other two.

10.3 How you test a plan: tabletop, simulation, failover

A plan that has never been exercised is a draft. ISO/IEC 27001 expects continuity controls to be tested, and the discipline in Act II is to establish not just that a test happened but that it tested the right thing at the right depth. Testing falls along a spectrum, and the deeper you go, the more it costs in time and risk, and the more it proves.

Tabletop is a paper walkthrough. The team sits in a room, or on a call, and talks through a scenario: "Sydney is down, here is the runbook, what do you do?" It tests understanding, roles, decision rights, and whether the runbook reads the way someone under stress would need it to. It costs almost nothing, and it reliably surfaces the first category of problem: the plan nobody has read, the contact list with last year's phone numbers, the approval step that names a person who left the company. Tabletop proves the team knows the plan. It proves nothing about whether the plan works.

Simulation raises the stakes. A scenario is run against a realistic but non-production environment, with some actual mobilisation: pages go out, an incident channel opens, people log in to the recovery systems. Simulation tests the *process* under load: how fast the team assembles, how the runbook holds when the steps are real, where the confusion lives. It still does not touch production, which is what makes it the workhorse test for most organisations, run once or twice a year.

Failover is the real thing. The service is actually moved to the recovery environment, either in parallel with production or, in the most demanding form, by interrupting production to force the switch. Failover is the only test that proves the technical claim: that the standby database really has the data, that DNS really reroutes traffic, that the application really starts in the standby region, that the RTO is achievable and the RPO holds. It is also the test organisations avoid, because it is expensive, it is risky, and it can break the very thing it is meant to protect. So they do not run it, and then they tell you it would work.

My own rule, earned the hard way, is to ask for the last failover test before I ask for the runbook. Organisations that have actually done one will show you before you finish the sentence, usually with a date and a post-mortem. Organisations that have not will change the subject. The speed of the answer is the evidence; the runbook is just the paperwork.

10.4 The region-failure audit

Tessera Case: The failover that has never failed over

You take Tessera's continuity plan at its word and work backwards. The plan promises a four-hour RTO and a fifteen-minute RPO against an AWS region failure, achieved by failing over from the primary (Sydney, ap-southeast-2) to a standby region. You ask for three things: the BIA that set those numbers, the runbook an on-call engineer would follow, and the result of the last failover test.

The BIA is a spreadsheet labelled "v0.3 (draft)". It lists RTO and RPO columns with the numbers already filled in and a comment in one cell that reads "industry standard for SaaS". That is your first finding: the objectives were picked, not derived. There is no per-process impact curve, no link to the customer contracts that set the uptime SLAs, and no MTD ceiling anywhere in the document. The four-hour RTO is inconsistent with the 99.9% SLA Tessera's largest customer negotiated, which the BIA does not even mention.

The runbook is better. It is detailed, step-numbered, with a named owner. But step seven, "promote the Aurora Global cluster to primary", assumes a configuration you cannot find in the Terraform, and step twelve, "switch Route 53 to the standby region", has no health-check policy attached. You ask to see the standby region's actual state. The primary database is Multi-AZ within Sydney (good: it survives an availability-zone failure), but there is no cross-region read replica and no global database. The "standby region" in the diagram is a drawing, not infrastructure. On the present estate a region failure is unrecoverable, and the fifteen-minute RPO is a fiction: with no cross-region replication at all, the real RPO is "since the last backup", which is hours.

The test record is the last piece. The most recent entry is a tabletop, eighteen months ago, whose action items (build the standby, document the failover, run a simulation) are all still open. No simulation has been run. No failover has ever been performed. The plan is an artefact of aspiration, and every number in it is untested. When you put it to the CTO that a Sydney outage would take Tessera's multi-tenant service offline for an unbounded period, the honest answer is yes. That is the finding, and it is a serious one.

> ⚠ Watch out: the untested plan, and the single-zone SaaS
>
> Two traps recur in continuity auditing, and both look responsible from the outside. The first is the beautiful runbook that has never been exercised. It reads perfectly, it maps cleanly to the standard, it names every owner, and it has never been run. A plan untested is a plan unproven, and an RTO written without a BIA is an aspiration, not an analysis. Insist on the test record, and read the action items.

> Items left open year after year tell you the testing is theatre.
> The second is more technical and easier to miss: the "highly available" SaaS whose HA lives in a single availability zone. Multi-AZ resilience survives the loss of a zone; it does not survive the loss of a region. If the only replication is within one region, a region outage takes the service down, and "high availability" was a marketing term. Check the actual topology: where do the primary and its replica live, and what is the replication lag? The architecture diagram says one thing; the configuration usually says another. Verify the configuration.

10.5 Incident response: the lifecycle you audit

Continuity is the slow failure: the region is down, we recover. Incident response is the fast one: we are under attack, or something is wrong, right now. The two share people and overlap in practice, because every incident has a recovery phase, but they answer different questions and they audit differently.

NIST SP 800-61 lays out the incident lifecycle in four stages, and it is the spine you audit against: preparation; detection and analysis; containment, eradication, and recovery; and post-incident activity. The ISO 27001:2022 controls map onto it almost one-to-one, which makes them a useful cross-check: A.5.24 is preparation, A.5.25 is the assessment that decides an event is an incident, A.5.26 is the response, A.5.27 is learning, and A.5.28 is the evidence you collect along the way, increasingly load-bearing when the incident touches personal data and the Notifiable Data Breaches clock from chapter nine starts running.

The desk audit in Act I could confirm the IR plan existed and named a team. Act II asks whether it operates, and the test is the same as everywhere else: trace real events through the lifecycle. Ask for the last two or three security incidents and walk each one forward. Was it detected, by what, and how fast? Was it triaged and severity-assigned against the defined scale? Was the plan invoked, and is there a record of who did what and when: the incident channel, the paging history, the incident bridge notes? Was containment applied, and was the eradication complete or merely cosmetic? Was recovery validated before the incident was closed? And the step organisations skip most: was there a post-incident review, and were its action items tracked to closure, or did they die in a spreadsheet?

The single most revealing test is the gap between events and incidents.

Pull the security event log over the same window as the incident register and compare them. If the alerts fired and nothing was triaged, detection exists but response does not, and the control is broken in the place nobody looked. A plan that handles declared incidents beautifully while the queue of undeclared ones piles up is a control that works on paper and fails in practice.

> **!** Auditing AI: when the detector decides for you
>
> Tessera's detection is partly automated, and increasingly so. AWS GuardDuty raises findings from machine-learning models, and Tessera has wired some of them to auto-remediate through Lambda. This is the modern shape of incident response, and it raises a question the lifecycle does not answer on its own: when a machine decides an event is benign, or fires a response on one, can you trust it?
> The audit moves in two directions. First, tune and coverage: what is the false-positive rate, what classes of threat are in scope and (crucially) out of scope, and who reviews the detection logic and how often? An untuned detector drowns the team in noise until they mute it, which is functionally identical to having no detector at all. Second, the human-in-the-loop boundary: which automated actions run without approval, and is that list defensible? Auto-quarantining an endpoint is low-risk and high-value; auto-shutting down a production database on a model's judgement is not. Ask to see the action that fired, the model's confidence, and the human review that followed. If the response was fully automated with no record, the control ran, but nobody can tell you whether it ran correctly. An AI that reports "all clear" is still a judgement, and a judgement with no reviewer is an unattended control.

10.6 ICT readiness for business continuity

A.5.30, ICT readiness for business continuity, is a 2022 addition, and it deserves to be named because it is easy to read past and because it does work the older "information security during disruption" control does not. A.5.29 says the organisation must keep information security working during a disruption: controls do not switch off because you are in a crisis. A.5.30 says something more demanding: the ICT services themselves must be ready to meet the continuity objectives the business has set. It forces the join between the BIA's numbers and the technology that has to hit them, and that join is exactly where Tessera's plan falls apart.

Readiness is a property you can test. It asks whether the recovery

environment exists and is kept current, not spun up on the day from a cold backup; whether the data replication actually runs and within the RPO; whether the runbook matches the infrastructure that is really there; and whether the people who would execute it have rehearsed. NIST SP 800-34 and ISO 22301 give the broader contingency and continuity disciplines their proper home, and ISO/IEC 27031 is the one aimed squarely at ICT readiness. For an auditor, A.5.30 collapses all of that into a single testable question: if the disruption happened today, on the estate as it actually stands, would the ICT services meet the objectives? For Tessera, as you have just seen, the honest answer is no, because the standby does not exist and the objectives were never derived.

10.7 The question, answered

When it breaks, can Tessera survive? On the evidence you now hold, not by the plan they have written. The continuity policy is well-structured and the runbook reads well, but the RTO and RPO were picked from industry defaults rather than derived from a business-impact analysis, the standby region the plan depends on is a diagram rather than running infrastructure, and no failover has ever been tested. A region failure would take the multi-tenant service offline for an unbounded period, and the fifteen-minute RPO is a fiction the present estate cannot meet. Incident response is in better shape: the plan exists, the team is named, and the events you traced through the lifecycle mostly made it from detection to closure, with the predictable gap in post-incident learning. Continuity is the finding; incident management is the partial.

That is a defensible opinion because every line of it is tied to evidence: the draft BIA, the Terraform missing the replica, the eighteen-month-old tabletop with its open action items, and the incident register you reconciled against the alert log. None of it was visible from the desk. All of it became visible the moment full access turned the polished plan into a testable claim. *Substantiate, don't assume* has never been more practical than it is here: the plan said they could recover in four hours, and the estate says they cannot.

Evidence Locker: lock this in (Week 10)

Lock two artefacts this week. First, the **BCDR test evidence**: the BIA (or its absence), the RTO and RPO derivation traced to the business impact and to the customer SLAs, the runbook, and the record of the most recent test at each depth you can find (tabletop, simulation, failover), with every open action item flagged. State explicitly, for each critical service, whether the stated objectives are

> achievable on the present estate. Second, the **incident-response readiness assessment**: the IR plan, the lifecycle test (two or three real incidents traced from detection to post-incident review), and the gap analysis between security events raised and incidents declared. By the end of the week your locker holds the proof, not the policy: what the organisation claims it would survive, and what the evidence says it actually would.

Next week the evidence stops accumulating and starts to speak. You have findings; now they have to become an opinion someone will stand behind. The question is **Report?**

Chapter 11

Report? Findings and the Audit Opinion

What did we find, and how do we say it so the board acts?

Tessera Case: Twelve weeks of evidence, one page of findings

The fieldwork is done. Your Evidence Locker holds the re-rated risk matrix, the audit programme, the interview notes, the configuration exports, the log samples, the policy-gap analysis, the privacy walkthrough, and the incident-response test. Spread across a desk it is a small mountain of paper, and not one sheet of it is a finding yet. The CTO is expecting a report the board can act on by Friday. Your job this week is the hardest compression in the engagement: turn that mountain into a handful of statements so specific, so evidenced, and so honestly rated that a busy board knows exactly what to fix and why. Nothing you found matters until it survives this translation.

11.1 From evidence to a finding

There is a moment every audit reaches where the evidence stops accumulating and the judgement begins. You have spent four chapters gathering proof: the configuration that contradicts the policy, the log that shows the control never fired, the interview where the operations lead admits the offboarding checklist "sometimes slips." Each is a fact. None of them, yet, is a finding.

A finding is a structured claim. It takes a fact and binds it to a standard, a cause, a consequence, and a recommendation, in a form another auditor could reproduce and a board could act on. The discipline of this chapter

is that binding. Evidence without a finding is noise. A finding without evidence is an opinion. The 5C+R format is how you make neither mistake.

The compression is brutal, and it should be. A board will give your report twenty minutes. Out of a twelve-week engagement, twenty minutes is what you actually have to change how Tessera is run, so every sentence of the findings has to earn its place. The format exists because unstructured findings get ignored: too vague to act on, too soft to prioritise, too padded to read. A 5C+R finding cannot be any of those things and still call itself a finding.

11.2 The 5C+R format

Five elements do the work, plus a sixth that ties the bow. Memorise the names. The order matters less than the completeness, but the names are how you will be marked, by certification bodies and by your own conscience.

Criteria is the standard you are measuring against. A clause of ISO/IEC 27001:2022, a control objective, a Tessera policy. It is the *should*: what good looks like. Without a named criterion a finding is just a preference, and preferences do not drive remediation budgets. "A.6.5 requires that the access rights of personnel leaving or changing role be removed or revised" is a criterion. "People should lose access when they leave" is a gripe.

Condition is what you actually found. The *is*. This is where your evidence lands, cited specifically enough to be defensible. Not "offboarding is weak" but "of twelve leavers tested in the period, five retained active AWS IAM or Google Workspace access beyond the 24-hour window defined in Tessera's own offboarding procedure; the longest, a senior engineer, retained production database access for 41 days." The condition is the fact, bound to the sample, stated without softening.

Cause is why the condition exists. This is the one auditors under-invest in and boards need most, because a finding with no cause cannot be fixed, only patched. Causes live at the level of process and ownership, not individual mistake. "An engineer forgot" is not a cause. "The offboarding checklist is owned by HR but executes on systems HR cannot touch, with no automated trigger between the HR system and the identity provider, and no single accountable owner with cross-system authority" is a cause. This is where root-cause analysis earns its keep, and we will spend a section on it.

Consequence is what the condition means for Tessera's objectives. This is the *so what*, and it is where the finding earns its risk rating. A retained

access account is, in itself, a line in a log. The consequence is that an ex-employee with a grievance could reach customer data, that the same gap is a major nonconformity against A.6.5 that can block certification, and that any misuse is a notifiable data breach under the APPs and the NDB scheme, with contractual exposure to the enterprise customers whose data is involved. The consequence is measured against objectives and obligations, the same way impact was in Chapter 2.

Corrective action is the immediate fix. Tight, time-bound, owned. Revoke the outstanding accounts this week. Reconcile the leavers list against live access today. Corrective action is the bandage: it stops the bleeding but says nothing about the cause.

Recommendation (the +R) is the durable fix that addresses the cause. Automate the deprovisioning trigger, assign a single cross-system owner, build the leavers population into the quarterly access review, enforce the sign-off line with a system that will not let the checklist close otherwise. A recommendation maps to a cause the way a corrective action maps to a condition. Get this pairing right and your findings become a remediation plan. Get it wrong and the same finding reappears next year.

11.3 Root-cause analysis: the finding's centre of gravity

Here is a confession that took me a few engagements to learn. I used to write the Cause last, as an afterthought, because the Condition felt like the real finding and the Recommendation felt like the deliverable. Boards taught me otherwise. A board given a finding with a patchy cause will fix the symptom, declare victory, and hand you the identical finding at re-audit. A board given a finding whose cause is named precisely will fix the system, and the finding stays fixed.

Root-cause analysis is the discipline of refusing to stop at the first plausible explanation. The technique is older than information security. Ask *why*, repeatedly, until the answer stops being a person and starts being a process. The offboarding checklist did not run. Why? Because no one triggered it. Why? Because HR raises it, but the revocations live in systems HR cannot reach, and the handoff depends on an email that may or may not arrive. Why? Because there is no automated link between the HR system of record and the identity provider. Now you have a cause worth writing: a broken handoff and a missing integration, not a forgetful engineer.

The "five whys" is a heuristic, not a religion. Sometimes the cause is obvious after two. Sometimes you need six and the sixth is "the organisation has never treated leavers as a security event, only an HR

one." The point is to keep digging past the convenient answer, because the convenient answer is almost always a person, and blaming a person fixes nothing. A good root cause names a control gap, an ownership gap, or a design gap. It rarely names a name.

This is also where professional scepticism turns inward. You have been sceptical of Tessera's claims all engagement. Now be sceptical of your own first draft of the cause. If your cause reads "human error," you have not finished. If your cause reads "insufficient training," ask why the training was insufficient and keep going. The cause you write is the cause that gets fixed. Settle for a shallow one and you have lent your authority to a remediation that will not hold.

11.4 Rating the finding: from guess to evidence

Chapter 2 taught you likelihood times impact, and warned you that the matrix is a hypothesis. Here is what changes in Act II. You now have evidence, and evidence turns a hypothesis into an observation.

The likelihood of the offboarding gap is not "medium" any more. You tested twelve leavers and five failed. That is not a likelihood estimate; it is a measured 42 per cent failure rate over the sample. Rate it on the observed frequency, and write the sample down so the rating is reproducible. This is the difference between a risk in a register, which is a forecast, and a risk in a finding, which is a fact.

Impact is where judgement still rules, because impact is always counterfactual. The gap did not, as far as anyone knows, cause a breach. It could have. Rating the impact means reasoning about what a 41-day window of ex-employee production access would cost Tessera if it were misused: the NDB notification, the contractual breach with enterprise customers whose data-isolation clauses are implicated, the certification major nonconformity, the reputational drag. None of that happened. All of it was possible, and the possibility is what impact measures.

The rating is the finding's priority, and it is where you will be pressured. Clients want findings down-rated so the report reads better and the remediation budget shrinks. Your defence is the same defence every chapter has built: the rating follows the evidence and the consequence, written down, reproducible by another auditor. If you can defend the 42 per cent sample and the contractual exposure, the rating defends itself. If you cannot, you should not have written it.

11.5 Forming a defensible audit opinion

The findings are the building blocks. The opinion is the building. After eleven weeks you are finally doing the thing the whole engagement exists to do: standing behind a judgement about whether Tessera's controls do what they claim.

An audit opinion is not a list of findings. It is a synthesised conclusion, defensible and repeatable, that a stakeholder can rely on. A defensible opinion has three properties, and they are the same three the Introduction named: it is **fair** (it presents the picture honestly, including what works, not just what fails), **repeatable** (another auditor with your working papers would reach it), and **ethical** (formed without pressure, on the evidence, regardless of what the client hoped to hear). Strip any of the three and the opinion is not worth the paper it is written on.

The shape of the opinion follows the evidence. An **unqualified** opinion says controls are designed and operating effectively. You do not hand these out lightly, and you almost never hand them out on a first certification engagement, because the standard of proof is high and the tolerance for untested assertions is zero. A **qualified** opinion says controls are largely effective, with named exceptions. Most real opinions are qualified, because most real organisations have a mix of working and broken controls, and honesty demands you name both. An **adverse** opinion says controls are not effective, full stop. It is rare, it is serious, and it ends the certification conversation.

The opinion you form for Tessera this week is built directly from the findings register. Each material finding presses on the opinion. A single high-rated finding on a load-bearing control may be enough to qualify it. A pattern of medium findings across a control family may amount to the same thing. Your job is to weigh the findings as a system, not count them, because five minor findings in an otherwise sound ISMS are a different opinion from one major nonconformity in the one control the certification depends on.

I will not give you Tessera's verdict here. That is the work of the next chapter, which takes this opinion and converts it into a certification-readiness judgement, and then asks you to defend it under board questioning. What this chapter delivers is the opinion's raw material: a findings register so honestly rated that the verdict is, to a careful reader, already implied. Substantiate the findings, and the opinion writes itself.

! Auditing AI: who writes the opinion?

An AI can draft a finding. It cannot sign an opinion, and the distinction is the whole argument of this book in one sentence. An opinion is an act of accountability. It attaches a name, a reputation, and a professional licence to a judgement about someone else's security. The machine has none of those things to attach.

This matters now because AI-assisted controls and AI-generated evidence have been flowing into your findings all engagement: logs summarised by a model, policies analysed by a model, findings drafted by a model. Each is input to the opinion. Your job, at the moment of signing, is to have substantiated every one of them, because the opinion is yours, not the model's, and you cannot defer accountability to a system that has none. We return to the two faces, AI in audit and auditing AI, in the final chapter. For now, hold the line: the opinion is the one artefact that is irreducibly human.

i AI Field Note: AI drafts the finding; you supply the opinion

This is where the drafter-to-evaluator arc lands, and it lands hard. Hand an AI the raw evidence for the offboarding finding, the relevant clauses of ISO/IEC 27001:2022, and a one-line prompt: "write this up as a 5C+R finding." It returns a perfectly structured finding in seconds. Criteria, Condition, Cause, Consequence, Corrective action, Recommendation, all in the right order, all grammatically immaculate. On structure alone it is indistinguishable from a senior auditor's first draft.

Now read it as the evaluator, because this is where the work actually is. The Criteria and Condition the AI got right: it copied the clause and restated your evidence faithfully. The Cause it got wrong, in the precise way machines get causes wrong. Its draft read "offboarding is not consistently followed due to insufficient process adherence." That is a symptom dressed as a cause, and it is the shallow answer root-cause analysis exists to refuse. It tells the board to enforce the process harder, which is exactly the remediation that fails, because the process is not the problem; the broken handoff between HR and the identity provider is. You rewrite the Cause with the actual root cause.

The Consequence the AI under-weighted. It listed "potential unauthorised access" and stopped there, because the generic register has no notion of Tessera's enterprise contracts, the APPs, the NDB scheme, or the certification exposure. You supply the consequence

that matters: the contractual and regulatory exposure that turns a latent access gap into a board-level risk, and you rate it accordingly. The Recommendation the AI wrote was generic ("improve the offboarding process"). You replace it with the durable fix that maps to your named cause.
What the machine produced in seconds was the scaffold. What you supplied in an hour was the judgement: the root cause, the consequence weighted to Tessera's real obligations, the rating, and, ultimately, the opinion. That is the entire arc of this book compressed into one artefact. The AI drafted the finding. The auditor supplied everything that makes the finding worth acting on.

11.6 Executive reporting: the board-facing version

The findings register is for the auditors, the risk owners, and the remediation programme. The executive summary is for the board, and it is a different document written from the same evidence.

A board does not want the 5C+R structure on the first page. A board wants, in one page, three things: what you found, in order of how much it matters; what the opinion is, in plain language; and what the organisation must do first. Lead with the opinion, because that is what the board is accountable for. Follow with the top three to five findings, each a sentence, each carrying its rating. Close with the remediation priorities, sequenced. Everything else is an appendix the board can ignore and the risk owners cannot.

The trap in executive reporting is the softening that happens when a finding travels up the chain. An engineer says "production access retained for 41 days." A manager writes "minor delay in deprovisioning." A director reads "access management requires enhancement." By the boardroom the finding has evaporated into a platitude, and nobody acts because nobody was told what was actually wrong. Your job is to write the summary so tightly that softening has nowhere to hide. State the condition. State the rating. State the risk to objectives. Let the discomfort do the work, because discomfort is the point. A board that is comfortable with your findings has been given the wrong findings.

Tessera Case: The offboarding finding, in full

Here is one finding from Tessera's register, the one Act I named as a gap and Act II closed with evidence. It is the artefact this chapter has been building toward.

Criteria. ISO/IEC 27001:2022 A.6.5 (responsibilities after or during termination or change) requires that the access rights of personnel leaving or changing role be removed or revised. A.5.18 (access rights) requires formal authorisation and timely removal of access. Tessera's offboarding procedure (ISMS-PR-014, v3.1) defines a 24-hour window from last working day to full revocation across AWS IAM, Google Workspace, and Slack.

Condition. Testing of twelve leavers in the twelve months to fieldwork found five (42 per cent) retained active access beyond the 24-hour window. The longest case retained production database access for 41 days. The Statement of Applicability records A.6.5 and A.5.18 as "implemented." Sample: 12 of 14 leavers (2 records incomplete and excluded, itself a control observation).

Cause. The offboarding checklist is owned by HR but its revocation steps execute on systems HR cannot access. There is no automated trigger between the HR system of record and the identity provider; the handoff depends on email to IT, which went unacknowledged in three of the five failures. The checklist sign-off line exists but is not enforced by any system that can prevent checklist closure without revocation evidence. Root cause: a broken cross-functional handoff with no accountable single owner and no automation, not individual error.

Consequence. A departing employee with retained production access represents a latent insider threat outside Tessera's awareness. Misuse would constitute a notifiable data breach under the APPs and the NDB scheme, breach data-isolation clauses in three enterprise contracts, and constitute a major nonconformity against A.6.5 and A.5.18 capable of blocking ISO/IEC 27001 certification.

Rating: High. Likelihood established by observed 42 per cent sample failure (not estimated). Impact established by regulatory, contractual, and certification exposure.

Corrective action. Revoke all outstanding leaver access within five working days; reconcile the full leavers population against current access; report results to the audit team. Owner: Head of IT. Due: fieldwork close plus two weeks.

Recommendation. Implement automated deprovisioning triggered by HR status change into the identity provider, with revocation evidence required to close the checklist. Assign a single accountable owner with cross-system authority over the full offboarding workflow. Include the leavers population in the quarterly access review. Re-test at the surveillance audit.

That is a finding a board can act on, a risk owner can remediate, and

another auditor can reproduce. Notice what is absent: blame, softening, and any sentence that does not earn its place.

⚠ Watch out: the finding that isn't

Three counterfeits pass for findings, and each wastes the board's twenty minutes.

The **recommendation dressed as a finding**. "Tessera should implement automated deprovisioning" is not a finding; it is the +R with the other five elements missing. A board cannot rate it, cannot trace it to a clause, and cannot tell whether it matters. If your finding reads as advice, rewrite it from the Criteria down.

The **symptom without a root cause**. "Access revocation is inconsistent" names the Condition and stops. Without a Cause it tells the board to try harder at a process that is already failing for a reason it has not been given. The remediation will patch, not fix, and you will write it again next year. Every finding without a named root cause is a finding you have deferred, not delivered.

The **consequence bent to please**. The client who wants the rating down will argue the impact is theoretical, that nothing happened. The client who wants a rival team's budget cut will argue the impact is catastrophic. Both are pressure on the Consequence, and both are the Introduction's "pressure is the auditor's real test" arriving on schedule. Rate the consequence against the objectives and the obligations, write the rationale, and refuse to move it for either flavour of comfort. A finding whose consequence has been massaged is not a softer finding. It is a wrong one with your name on it.

There is a fourth pressure that does not counterfeit a single finding so much as the whole opinion: the urge to soften the verdict so the deal closes, the certification lands, the executive is spared. Hold the line the Introduction drew. An opinion given under pressure, without the evidence to back it, is not an efficiency. It is a liability you have signed.

💡 Evidence Locker: lock this in (Week 11)

This is the second-to-last entry, and it is the one that turns eleven weeks of work into an audit. Lock in two artefacts.

First, your **findings register**: every finding in 5C+R form, each with a named criterion, a cited condition bound to its sample, a root cause, a consequence weighted to Tessera's objectives and obligations, a rating with written rationale, and a corrective action plus recommendation mapped to the cause. Order them by rating.

This is the evidence base for your opinion, and by next week it is the body of your report.
Second, your **draft defensible opinion**: one paragraph that synthesises the register into a fair, repeatable, ethical conclusion on whether Tessera's controls are designed and operating effectively. State the opinion type you are leaning toward and why, and name the one to three findings that press hardest on it. This is a draft. Chapter 12 will sharpen it into the certification-readiness verdict and make you defend it. But a draft opinion you can already stand behind is the measure that eleven chapters of substantiation have done their job.

11.7 The question, answered

What did we find? A set of findings so specific, so evidenced, and so honestly rated that each one is reproducible by another auditor and actionable by a risk owner. The offboarding gap that Act I only suspected is now a measured 42 per cent failure rate with a named root cause and a durable fix. The polished policy has finally met its operational reality, and the reality is written down in a form no amount of softening can disguise.

How do we say it so the board acts? In a 5C+R structure that leaves no finding vague, a rating that follows the evidence rather than the optimism, a root cause that points at the system rather than a scapegoat, and an executive summary tight enough that the discomfort has nowhere to hide. We say it fairly, naming what works alongside what fails. We say it ethically, refusing to bend the consequence or the opinion for the deal on the table. And we sign it, because an opinion is an act of accountability, and accountability is the one thing the machine cannot supply.

Eleven weeks have run their course. The evidence is gathered, the findings are written, the opinion is drafted. What remains is the verdict that everything has been building toward, and the moment you have to stand up and defend it. Tessera's board is waiting. So is the certification body. The question, finally, is **Conclude?**

Chapter 12

Conclude? Certification, Defence, and the Future of Audit

An audit ends where it began: with a judgement, stated plainly, and defended under fire.

Tessera Case: The boardroom

Twelve weeks of fieldwork collapse into forty minutes in a glass-walled room on St Georges Terrace. Tessera's board has cleared the agenda for your readout. The CTO, who opened the engagement hoping for "pretty much there," is watching you with the particular stillness of someone about to hear a number. The chair cuts the small talk: "So. Are we ready, or aren't we?" That question is the whole engagement, distilled, and the honest answer is not the one the CTO wants. It is also the one you came to give.

12.1 The certification-readiness opinion

Chapter 6 ended Act I with a preliminary opinion: *promising on paper; not yet proven in practice.* Act II has done the proving. You have interviewed the people, walked the office, pulled the configurations, read the logs, traced the incidents, and tested the controls you could only read about. The discipline now is to turn that evidence into a single, defensible statement: a **certification-readiness opinion**.

Before you write a word, get one thing straight, because it is the misconception that ends careers. You are not the body that grants ISO/IEC 27001

certification. No one grants it to themselves. Certification is awarded by an **accredited certification body**, an independent organisation itself audited by a national accreditation authority, which sends its own auditors to run a two-stage certification audit against Clauses 4 through 10 and the Annex A controls. Your engagement with Tessera is a **readiness assessment** (sometimes called an internal audit, a gap assessment, or a Stage 1 dry run). Your opinion answers a narrower, honest question: *on the evidence we gathered, would Tessera pass a certification audit, and what stands between them and one?*

The distinction matters twice over. It protects you from overclaiming. You are not certifying Tessera; you are giving them and their board the most credible possible read on whether they are ready to be certified, and what to fix first. And it protects Tessera from a false sense of safety, the feeling that the polished security page can now add a badge because you said something nice. It cannot. What you hand over is judgement, scoped and dated, and the named conditions under which it holds.

A readiness opinion has four moving parts, and each has been built over the preceding chapters. The **criteria** are ISO/IEC 27001:2022 (the four themes, ninety-three controls, the management system clauses) reconciled with NIST CSF 2.0 and the ASD Essential Eight where Tessera has adopted them. The **condition** is what your Act II testing actually found, written up in the 5C+R format from Chapter 11. The **cause** is the root of each gap, because a board that does not understand the cause cannot fix the control. And the **conclusion** is the opinion itself: ready, ready with conditions, or not ready, each one defensible only because the evidence behind it is documented and reproducible.

There is a particular honesty to the middle option, and it is usually the right one. A clean "ready, no exceptions" is rare in a company growing as fast as Tessera, and a board that hears it should be more suspicious than relieved. A flat "not ready" wastes the nuance the fieldwork earned. The defensible opinion names what works, names what does not, and tells the board exactly what must happen for the gap between the two to close. That is the opinion you are about to defend.

12.2 Defending the opinion

An opinion you cannot defend is not an opinion. It is a guess with a letterhead. The boardroom is where this gets tested in real time, and it is the moment the whole engagement turns on the one quality no machine and no amount of drafting can supply: your willingness to stand behind your judgement when it is inconvenient.

The pressures arrive dressed as reasonableness. The enterprise deal that

needs the certification badge before quarter-end. The CTO's reputation, riding on "we were ready." A friendly offer to "tidy up" the language so the major finding reads as a minor one. Each of these sounds like collaboration. Each is a test of independence, the same independence Chapter 1 made the foundation of the mandate, and each requires the same answer: the opinion is the evidence, restated. Change the evidence and the opinion moves. Pressure the opinion and nothing moves at all, because the evidence does not bend to deadlines.

The defence has a shape, and it is worth learning it cold. You state the conclusion. You point to the evidence that supports it. You name the criterion the evidence was tested against. You acknowledge the limitation honestly, because every audit has one (the period, the sample, the controls in scope) and conceding it is a strength, not a weakness. And you refuse, politely and absolutely, to upgrade the rating to suit the room. A board that presses you and finds you unmoved learns something more valuable than the finding: that the opinion is worth paying for, because it does not move when it is inconvenient. That is the whole commercial case for the profession, demonstrated in a single meeting. I learned this the hard way, the way most auditors do. Early on I rounded a finding down for a client I liked, told myself the gap was small and the deadline was real, and waited six months for the control I had discounted to fail. It did not fail, which is the most dangerous outcome there is, because it taught me how easily the line moves once you cross it the first time. I have not rounded a rating since.

> ⚠ Watch out: the clean opinion at all costs
>
> Two pressures converge in the final week, and both are traps. The first is the **certification deadline pressure**: the deal, the board commitment, the CTO's bonus, all riding on a "ready." The second is **polish pressure**: the quiet temptation to round a finding down because the client is likeable, the fieldwork was hard, and a clean report feels like a kinder way to end. Both ask you to substitute a nicer conclusion for an evidenced one. The professional answer is identical: the rating is what the evidence supports, and an opinion given under deadline or social pressure is a liability you have signed your name to. Certification is not a reward for effort. It is a statement about controls. If the evidence says "ready with conditions," that is the opinion, and the conditions are the favour you do the client, not the obstacle.

> **AI Field Note: Rehearsing the sceptical board**
>
> Here is a use of AI that genuinely earns its place in the final week. Ask it to play a sceptical non-executive director: "You are a board chair with a finance background and no patience for jargon. Read this opinion and probe it hard. Ask the sharpest, fairest questions a board would put to it." Then give it the opinion and the findings. It will come back with a string of sharp, fair questions: *Why is this a major and not a minor? How did you sample? What is your confidence interval? Why should we trust the AWS SOC 2 over your own eyes?*
>
> This is good practice, and it works because the AI is relentless in a way a willing colleague is not. Treat its questions as a dress rehearsal. Answer each one, out loud or in writing, and anywhere you stumble, that is a sentence in your opinion that is not yet defensible. Fix it before you walk into the room.
>
> But notice the line you do not cross. The AI can stress-test the opinion. It cannot deliver it. When the real chair leans forward and asks the question the AI predicted, it is your name, your registration, and your judgement on the answer. The machine rehearsed you. You own every word under questioning, and that ownership is the entire point of the opinion. Rehearse with the tool. Defend the judgement yourself.

12.3 Cloud and third-party assurance

No modern SaaS company is an island, and Tessera certainly is not. AWS hosts the estate. A payment processor moves the money. A logistics API ships the product. An analytics platform chews the data. An AI service summarises customer reports. Each of those relationships is a control Tessera relies on but cannot directly inspect, and your readiness opinion has to say something credible about each one without you having audited the vendor yourself.

This is where **SOC reports** earn their keep, and where they most often mislead the people who cite them. The System and Organization Controls reports issued under the AICPA framework are a service organisation's auditor's opinion over its controls. A **SOC 2 Type II** covers the Trust Services Criteria (security, and any of availability, processing integrity, confidentiality, and privacy the vendor chose to include) over a defined period, and it is the report you reach for when a vendor holds Tessera's customer data. A Type I describes design at a point in time and says nothing about whether the controls actually ran. A SOC 1 concerns financial reporting controls, not security, and is the wrong report to cite

for a security assurance point, however often it gets cited.

Reading a SOC report well is an audit skill in itself, and the discipline is to read the parts everyone skips. The **scope** tells you which systems and which criteria the opinion actually covers, and a report that covers Security but not Privacy tells you nothing about the vendor's privacy controls, however reassuring the cover page looks. The **period** tells you the window the testing covered, and a report that ended nine months ago describes a control environment you can no longer assume is current; check the **bridge letter** that covers the gap. The **qualified or excepted opinions** hide in the body, where the service auditor declined to opine cleanly on a control, and they are the sentences that decide whether the report actually supports the assurance Tessera's board is about to rely on.

> ⚠ Watch out: the SOC report you cited but never read
>
> A vendor hands over a SOC 2 with a clean opinion and it gets filed as proof the vendor is secure. Then someone asks which criteria it covered, over what period, and whether the system Tessera actually uses was in scope. If you cannot answer from having read the report, you have relied on its cover page, not its opinion. The classic third-party assurance failure is a clean SOC 2 cited for a control or criterion it never opined on, by an auditor who stopped at page one.

The trap that catches even careful auditors is the **complementary user entity controls**. These are the controls the service organisation assumes *you*, the customer, have in place. AWS secures the hypervisor and the physical data centre, but the IAM configuration, the encryption keys, the public S3 buckets, and the network exposure are Tessera's responsibility under the shared-responsibility split from Chapter 5. A clean AWS SOC 2 is evidence that AWS did its half. It is not evidence that Tessera did hers. Treating the vendor's clean report as proof of Tessera's security is the same error as treating a polished policy as proof of an operating control: it confuses one party's assertion for another party's reality.

The disciplined position for your opinion is layered. You cite the SOC 2 Type II as evidence about the vendor. You point to the shared-responsibility controls as the evidence you tested yourself. And you name, honestly, the residual gap where neither reaches: the sub-processor the vendor added since the report's period, the criterion the report did not cover, the configuration that is technically Tessera's but operationally opaque. Read scopes, current periods, tested user controls, and named residuals: that is third-party assurance you can defend.

12.4 The two faces of the next decade

Stand back from Tessera for a moment, because this is the last chapter and the view from here is the point. Two forces are reshaping the profession at once, and they look alike but pull in opposite directions. Both involve the letters AI. Only the discipline of this book tells them apart.

The first face is **AI in audit**: the machine on your side of the desk. Drafting the control matrix in seconds. Mapping a finding to the right control. Summarising a hundred pages of evidence into a brief. Sampling a population, flagging the anomalies, generating the first cut of the report. This is the commoditisation the Introduction warned about, and it is genuinely useful, the same way a calculator is useful to an accountant. Used as a thinking partner, the way the companion book *Conversation, Not Delegation* teaches, it makes you faster and often sharper. Used as a delegate, it quietly hollows out exactly the judgement the profession sells, and you wake up one morning to find you have outsourced the part that was worth paying for.

The second face is **auditing AI**: the machine on the other side of the desk, inside the estate you must now opine on. The AI service Tessera bolts into its product. The AI-assisted controls that approve transactions, triage incidents, and revoke access. The large language model that writes the security summary a customer reads. Each of these is a new control surface with new failure modes: training-data leakage, prompt injection, drift after retraining, opaque decision logic, evidence you cannot reproduce. The frameworks you already hold are where this lands. ISO/IEC 27001:2022's cloud-services control (A.5.23) and threat-intelligence control (A.5.7) become load-bearing. The supplier controls (A.5.19 through A.5.23) govern an AI vendor the same way they govern any other, with the difference that what the vendor *does* with the data is harder to inspect. NIST CSF 2.0's Govern function demands accountability for decisions an algorithm now makes on Tessera's behalf.

> **!** Auditing AI: The two faces, made one
>
> This is the synthesis the whole book has been building towards. AI in audit and auditing AI are not two topics. They are the same coin, and the discipline that handles one handles the other.
> The thread that connects them is the drafter-to-evaluator shift. When the machine drafts *your* work (AI in audit), your job is to evaluate it: catch the confabulated compliance, the generic example, the smoothed-over nuance, the rating that drifted to please. When the machine runs *the client's* controls (auditing AI), your job is exactly the same: evaluate the output, test the evidence it produces,

refuse to assume that because the model said "compliant" the control is operating. In both cases the work that matters is the human judgement layered on top of the machine's first draft, and in both cases the failure mode is the same: trusting the polished output instead of substantiating it.
So the future is not auditors replaced by AI, nor auditors ignoring it. It is auditors who can do both at once: use the tool to do the technical work faster, and then turn the same sceptical eye on the AI systems inside the estate, because those systems are now part of what the opinion must cover. The auditor who can hold both faces in view, who drafts with the machine and audits the machine with equal fluency, is the auditor the next decade rewards. The opinion is still human. The estate now includes the machine. Substantiate both.

12.5 Where the engagement lands

Twelve weeks ago Tessera's CTO believed the company was "pretty much there." The discipline of the engagement has tested that belief against evidence, one question at a time, and the answer is now ready to be delivered. Here it is, in the form the board will receive it.

Tessera Case: Tessera's certification-readiness opinion and board Q&A

Certification-readiness opinion, full-access fieldwork, ISO/IEC 27001:2022.

On the basis of full-access fieldwork conducted over the engagement period, Tessera Limited has established and is operating an information security management system substantially aligned with ISO/IEC 27001:2022 across all four control themes and Clauses 4 through 10. The control set is designed and, in the majority of controls tested, operating effectively. Third-party assurance has been obtained and read for AWS, the payment processor, and the AI summarisation service, with complementary user entity controls tested by us directly.

The following were identified during testing and condition this opinion:

1. (Major nonconformity, A.5.18 / A.8.2) The quarterly privileged-access recertification for production and customer-data systems had not been performed for two consecutive quarters. Role-change reviews are not operating. This is systemic, not isolated.

2. (Minor nonconformity, A.6.5) Of eight sampled leavers, two retained production access beyond the policy window (nine and twenty-three days respectively). Root cause addressed; control now operating for current

leavers.

3. (Observation, A.5.9) Asset-register reconciliation identified six unregistered assets, now added. Register currency to be monitored.

4. (Observation, A.5.23 / APP 11) The AI summarisation service's prompt-logging is enabled; no evidence of training on customer data, but logging retention should be reviewed against APP 11's security and destruction obligations.

Conclusion: Tessera is ready to proceed to a Stage 1 certification audit with an accredited body, conditional on remediation of the major nonconformity (item 1) and its verification. The minor and the observations should be progressed through the corrective action programme and do not, individually or collectively, preclude certification.

That is the opinion. Now the room tests it.

Chair: *"You're telling me the whole certification rides on access reviews? We do access reviews."* **Auditor:** The evidence says they stopped two quarters ago. I can show you the recertification log, the gap, and the three production accounts that were not reviewed. This is the one control whose absence I cannot discount, because it is exactly the gap an attacker looks for. Fix it and verify it, and the opinion moves to unconditional.

CTO: *"Can't we just call the major a minor? It's one control."* **Auditor:** I cannot. The rating follows the evidence, not the deadline. A systemic, undetected gap in privileged-access governance is a major by any defensible standard. Downgrading it to ease the deal would make the opinion worthless to the very board and customers it is meant to protect. The favour I owe you is the honest rating and a clear path to fix it, which is what you have.

Non-exec: *"And you're comfortable relying on AWS's SOC 2 for the cloud? You didn't fly to Sydney and inspect a rack."* **Auditor:** I read the scope, the period, and the exceptions. AWS's SOC 2 Type II covers the hypervisor and physical layers that are their responsibility under the shared-responsibility split. The IAM, the keys, the encryption, the public exposure, those are Tessera's half, and I tested them directly. The opinion cites the SOC 2 for AWS's controls and my own testing for Tessera's. That is the honest layering.

The room goes quiet. The chair nods. The opinion stands, unchanged, because it was always the evidence restated, and the evidence did not move.

Evidence Locker: lock this in (Week 12)

This is the capstone. By now your Evidence Locker is not a study aid; it *is* the audit. Lock in your **final certification-readiness opinion**: the conclusion, the criteria, the named findings with their 5C+R write-ups and ratings, the limitations of the engagement (period, sample, scope), and the conditions under which the opinion holds. Attach the third-party-assurance summary: each SOC report cited, its scope read, its period checked, and the complementary user controls you tested yourself. Add the board Q&A you rehearsed, with your defensible answer to each hostile question.
Do this and the locker becomes the working papers another auditor could pick up, follow, and reach the same conclusion. That reproducibility is the standard a defensible opinion is held to, and it is the standard your Assessment 3 will be marked against. The opinion is the artefact. The locker is the proof the opinion was earned.

12.6 Where to go from here

You have audited Tessera. The engagement is closed. Three next steps are worth naming, because this is a discipline you build by doing, not by finishing.

Practise on the real thing. The companion site at tessera.locoensayo.org stays open, and the cast is AI and the judgement is yours. Run the engagement again with different assumptions, interview the staff you skipped, chase the red herrings you fell for. Every pass sharpens the judgement the next engagement will pay for. The site is a rehearsal room, and auditing is a rehearsal art.

Earn the credential. ISACA's **CISA** (Certified Information Systems Auditor) is the professional registration this discipline is measured against, and it is built on the standards this book has taught: the audit lifecycle, evidence and sampling, the control frameworks, the code of ethics. CISM, CGEIT, and CRISC extend the path into management and risk. The credential is not the judgement, but it is the profession's promise to the people who rely on your opinion that you have been held to a standard.

Read the companion. *Conversation, Not Delegation* (https://michael-borck.github.io/conversation-not-delegation/) is the method book to this book's discipline book. Where these chapters taught you what to verify, that one teaches how to work with the machine well: the conversation loop, staying in the loop, compensating for AI's predictable weaknesses. The drafter-to-evaluator shift lives in the practice of the conversation, and the next decade of audit belongs to the people who can hold both.

12.7 The question, answered

Are you ready to conclude? You are, and the conclusion is this. Tessera is ready for certification conditional on one named fix, and you can defend that judgement in a boardroom because every word of it traces back to evidence you gathered on terms you set. That is what twelve weeks of *substantiate, don't assume* produces: not a guess, not a vibe, not a polished page full of badges, but an opinion a board can rely on because you proved it, named what you could not, and refused to soften it when the room pushed back.

And here is the premise, returned to, one last time. The Introduction asked what a human auditor is for when a machine can draft the work. Twelve chapters have answered it. The technical craft is commoditised, and the machine does it faster than you. The opinion is not commoditised, because an opinion is a piece of judgement someone stands behind, and a machine cannot stand behind anything. In a world where the drafting is free and the controls include the machine itself, the defensible human opinion is not a leftover from an older profession. It is the whole point.

Every chapter in this book has been one rehearsal of the same habit. The polished policy is an assertion. The green dashboard is an assertion. The vendor's clean report is an assertion. The AI's confident summary is an assertion. The CTO's "we're pretty much there" is an assertion. None of them is evidence until you have seen it work. You substantiate what you can. You name what you cannot. You form an opinion you can defend, and you defend it. That is the audit mindset, and it is the only mindset worth having now that the machine can draft and the estate is half algorithm.

Substantiate, don't assume. The cast is AI. The judgement is yours.

Appendix A

The Drafter-to-Evaluator Loop

This is the one-page reference for the habit the AI Field Notes build, chapter by chapter. It is not a prompting manual. For the craft of conversing with AI in general, read the companion, *Conversation, Not Delegation*. This is the audit-specific loop: how you use AI to draft the technical work, and then supply the judgement the machine cannot.

A.1 The loop, in five moves

1. **Set the specific context.** Give the model Tessera's reality, not a generic one: a multi-tenant SaaS on AWS, the customer contract, the regulatory regime, the population. A generic prompt earns a generic answer, and a generic answer has no edge.
2. **Take the draft, not the answer.** Expect a first cut that is confident, well-structured, plausible, and probably wrong in the places that matter. It is the start of the work, not the end of it.
3. **Probe the silent failure and the unattributed number.** Ask: *how would this control fail quietly? where did this rating, sample size, or RTO come from?* The model reaches for defaults that read well. Make it show its working, then check that working.
4. **Hunt the confabulation.** Test every control number, legal clause, framework mapping, and compliance claim against the actual standard, the actual law, and the actual population. The model is fluent, not careful.
5. **Supply the judgement.** The rating, the root cause, the consequence, and the opinion are yours. The machine drafts; you decide, you evidence, and you sign.

Run this every chapter and AI stops being a shortcut and becomes a force multiplier. That is the whole argument of the book, in working form.

A.2 Failure modes to hunt for

The same handful of errors recur across the engagement. When the AI drafts, read specifically for these.

Failure mode	What it looks like	First seen
Generic examples	Controls and risks that describe every company	Ch 1
Stale standards	2013 Annex A control numbers cited as 2022	Ch 3
Invented entries	A mapping or control that exists in neither framework	Ch 3
Round-number samples	"Test 25 leavers" with no derivation	Ch 7
Compliance from design	"Reviews appear to be conducted," inferred from policy quality	Ch 6
Wrong-jurisdiction law	EU Standard Contractual Clauses offered for an Australian context	Ch 9
Symptom as cause	"Insufficient process adherence" reported as the root cause	Ch 11
Unattributed defaults	RTO and RPO copied from training data as if universal	Ch 10

None of these announce themselves. They look like good work, and that is exactly why the evaluator, not the drafter, signs the opinion.

> **Try this: run the loop on your own draft**
>
> Take any AI-generated item in your Evidence Locker and put it through the five moves. Anywhere you cannot defend a line, that line is not yet yours. Fix it before it reaches the opinion.

Acknowledgments

This book grew out of a frustration and an opportunity. The frustration: information security auditing is taught from standards — ISO/IEC 27001, NIST CSF, the Privacy Act — which are authoritative but not a story a student can read end to end. The opportunity: AI is commoditising the technical craft of auditing, which forces the question of what the human auditor is actually *for*. This book is an attempt to answer both at once — to give ISYS6018 a readable spine and to take a clear stance on judgement in a world where machines can draft.

The students of ISYS6018 — Information Security Audit and Control at Curtin University — are the reason this book exists. Their questions, their confusion, and their moments of "oh, *that's* what an auditor does" shaped every chapter. The Tessera engagement is written for them, and the discipline it teaches — substantiate, don't assume — is the one they carry into the profession.

Colleagues who reviewed early drafts and pushed back on arguments that were not yet strong enough made the book sharper. You know who you are, and I am grateful.

The open-source community behind Quarto, Pandoc, and GitHub made it possible to write, build, and publish this book with tools that are free and transparent.

This book was written using the methodology its companion, *Conversation, Not Delegation*, describes. AI was involved at every stage — as a drafting partner, a sounding board, and a source of plausible-sounding mistakes that needed catching. That last role is the point. An auditor's relationship to AI-generated work is exactly what this book teaches: the machine produces the draft; the human supplies the judgement, the evidence, and the defensible opinion. Every sentence here reflects the author's judgement. The AI made the work faster. It did not do the thinking.

About the Author

Figure A.1: Photo of Michael Borck

Michael Borck is a software developer, educator, and auditor-by-training working at the intersection of human expertise and artificial intelligence. He teaches information security audit and control at Curtin University, where the question that started this book — *what is a human auditor actually for when a machine can draft the work?* — is asked every week.

His earlier book, *Conversation, Not Delegation*, argues that AI is most valuable not as a tool you delegate to, but as a thinking partner you converse with. *Substantiate, Don't Assume* is the discipline that pairs with that craft: it is about what an auditor must *verify* once the drafting is done. The two books are companions — one on how to work with the machine, one on what to check.

Michael builds educational software and resources, and writes about the 80/20 principle in learning and professional judgement.

Connect

- michaelborck.dev — Professional work and projects
- michaelborck.education — Educational software and resources
- 8020workshop.com — Passion projects and workshops
- LinkedIn

Other Books in This Series

AI-Collaboration Track:

- *Conversation, Not Delegation: How to Think With AI, Not Just Use It* — the companion to this book (how to use AI well)
- *Partner, Don't Police: AI in the Business Classroom*

Python Track:

- *Think Python, Direct AI: Computational Thinking for Beginners*
- *Code Python, Consult AI: Python Fundamentals for the AI Era*
- *Ship Python, Orchestrate AI: Professional Python in the AI Era*
- *Converse Python, Partner AI: The Intentional Prompting Methodology*

Web Track:

- *Build Web, Guide AI: Business Web Development with AI*