

AUDIT • GOVERNANCE • DETECTION

Securing Microsoft Entra ID and Microsoft 365

The CISO's Guide

Frédéric PEREZ

Audit method • Priority workstreams • 12-month roadmap

Securing Microsoft Entra ID and Microsoft 365

The CISO's Guide

Frédéric PEREZ

Securing Microsoft Entra ID and Microsoft 365 — The CISO's Guide

© 2026 Frédéric PEREZ

All rights reserved. No part of this book may be reproduced without the written permission of the author.

Microsoft, Microsoft 365, Entra ID, Azure, Intune, Purview and Defender are trademarks of the Microsoft group.
This book is independent and is neither endorsed nor sponsored by Microsoft.

Information relating to products, licensing and interfaces is subject to change after publication.

Contents

Foreword.....	3
Book outline.....	4
PART I — THE IDENTITY ATTACK SURFACE.....	6
Chapter 1. Why identity became the perimeter.....	7
Chapter 2. Anatomy of an Entra ID and Microsoft 365 tenant.....	10
Chapter 3. The seven attack paths actually observed.....	13
Chapter 4. Shared responsibility: what Microsoft covers, what remains yours.....	16
PART II — ASSESS: THE AUDIT METHOD.....	18
Chapter 5. Scoping an audit: perimeter, stakeholders, duration, deliverables.....	19
Chapter 6. Choosing a framework: CIS, ANSSI, Zero Trust, MITRE ATT&CK.....	22
Chapter 7. Collecting evidence: data sources, reliability, pitfalls.....	25
Chapter 8. The twenty controls that reveal eighty per cent of the risk.....	27
Chapter 9. Scoring and prioritising: criticality, effort, user impact.....	30
Chapter 10. Writing a report the executive committee will actually read.....	32
PART III — REMEDIATE: THE PRIORITY WORKSTREAMS.....	34
Chapter 11. Privileged accounts: roles, time-bound elevation, emergency access.....	35
Chapter 12. Phishing-resistant authentication and the end of legacy protocols.....	38
Chapter 13. Conditional Access: a maintainable, testable rule set.....	40
Chapter 14. Applications and consent: the blind spot.....	42
Chapter 15. Non-human identities: service principals, secrets, certificates.....	44
Chapter 16. Guests, B2B and external sharing.....	46
Chapter 17. Data: classification, loss prevention, labels — the viable minimum.....	48
Chapter 18. Devices: compliance as a condition of access.....	50
Chapter 19. Hybrid: synchronisation, federation, residues.....	52
PART IV — DETECT AND RESPOND.....	54
Chapter 20. Logging: what to keep, for how long, at what cost.....	55
Chapter 21. Detections that matter, noise to eliminate.....	57
Chapter 22. Privileged account compromise: the hour-by-hour sequence.....	59
Chapter 23. Rebuilding: revocation, rotation, identity forensics.....	61
PART V — GOVERN OVER TIME.....	63
Chapter 24. Identity governance: lifecycle, reviews, separation of duties.....	64
Chapter 25. Compliance: NIS2, DORA, ISO 27001, GDPR.....	66
Chapter 26. Steering: metrics and continuous control.....	69

Chapter 27. Budget and twelve-month roadmap.....	71
Chapter 28. The ten most frequent mistakes seen in the field.....	73
APPENDICES — IMPLEMENTATION TOOLS.....	74
Appendix A. Condensed audit checklist.....	75
Appendix B. Licence-to-control matrix.....	77
Appendix C. Report template.....	79
Appendix D. Glossary.....	80
Summary.....	82

Foreword

This is not a Microsoft 365 administration manual. It will not teach you which box to tick in which portal: the official documentation does that better, and it is updated every month.

It answers a different question — the one I have been asked in executive committees across ten years of identity audit and architecture engagements: *where do we actually stand, what exposes us, and in what order should we fix it?*

The distinction matters. A Microsoft 365 tenant carries several hundred security settings, spread across half a dozen administration consoles that change name every eighteen months. No security leader has time to know them all, and none needs to. What a security leader does need to know is which settings actually decide the outcome of a breach, which are merely convenience, which depend on a licence the organisation may not hold, and how to prove to an auditor, an insurer or a regulator that the first category is in place.

The book is therefore built like an audit engagement: understand the attack surface, assess it with a repeatable method, remediate in order of return, detect what gets through anyway, then govern over time. Every chapter closes with a section titled "What you need to decide". These are not technical recommendations; they are the trade-offs that belong to you and that nobody will make on your behalf.

Commands, queries and extraction scripts are deliberately absent from the body of the text. They age quickly and slow the reading down. You will find them in the appendix as a condensed checklist, in a form your technical team can execute.

One final caveat, which applies to any book about the cloud: product names, licensing thresholds and screens change. Attack mechanics and control principles change far more slowly. This book rests on the latter, which is why it should stay useful longer than a screenshot.

Book outline

Part I — The identity attack surface

- Why identity became the perimeter
- Anatomy of an Entra ID and Microsoft 365 tenant: what a CISO strictly needs
- The seven attack paths actually observed
- Shared responsibility: what Microsoft covers, what remains yours

Part II — Assess: the audit method

- Scoping an audit: perimeter, stakeholders, duration, deliverables
- Choosing a framework: CIS, ANSSI, Zero Trust, MITRE ATT&CK
- Collecting evidence: data sources, reliability, pitfalls
- The twenty controls that reveal eighty per cent of the risk
- Scoring and prioritising: criticality, effort, user impact
- Writing a report the executive committee will actually read

Part III — Remediate: the priority workstreams

- Privileged accounts: roles, time-bound elevation, emergency access
- Phishing-resistant authentication and the end of legacy protocols
- Conditional Access: a maintainable, testable rule set
- Applications and consent: the blind spot
- Non-human identities: service principals, secrets, certificates
- Guests, B2B and external sharing
- Data: classification, loss prevention, labels — the viable minimum
- Devices: compliance as a condition of access
- Hybrid: synchronisation, federation, residues

Part IV — Detect and respond

- Logging: what to keep, for how long, at what cost
- Detections that matter, noise to eliminate
- Privileged account compromise: the hour-by-hour sequence
- Rebuilding: revocation, rotation, identity forensics

Part V — Govern over time

- Identity governance: lifecycle, reviews, separation of duties
- Compliance: NIS2, DORA, ISO 27001, GDPR
- Steering: metrics and continuous control
- Budget and twelve-month roadmap

- The ten most frequent mistakes seen in the field

Appendices — A. Condensed audit checklist · B. Licence-to-control matrix · C. Report template · D. Glossary

PART I

The identity attack surface

CHAPTER 1

Why identity became the perimeter

For twenty years, enterprise security was organised around a physical boundary. There was an inside and an outside, a firewall in between, and most of the effort went into controlling what crossed the line. That model had one considerable merit: it was visible. You could draw it on a whiteboard, audit it by following cables, and budget it in appliances.

The model has not disappeared, but it no longer decides very much. When email, files, meetings, the intranet and a growing share of business applications are hosted by a third party and reachable from any network, the question "where is this connection coming from?" loses its primacy. The question that decides becomes: "who is this, on what device, and how far can they go?"

The control point has moved

In a Microsoft 365 architecture, the decision point is no longer the firewall but the Entra ID token service. At every access attempt it evaluates the identity, applies Conditional Access rules, then issues — or refuses — a token that opens the door to Exchange Online, SharePoint, Teams or the Microsoft Graph API.

That shift has three direct consequences for a security leader.

The first is that the configuration of this decision point is, on its own, a major part of the organisation's security posture. A badly written Conditional Access rule does not produce an alert; it produces an authorised access. That is a silent failure mode, and silence is what makes it dangerous.

The second is that the perimeter is now shared with third parties you do not administer. Service providers, partners, SaaS applications connected to your tenant and application identities all hold access that traverses none of your network equipment.

The third, and the most uncomfortable, is that the skill protecting the organisation no longer belongs to the network team. It belongs to the directory administrators, who often still report into IT operations, with operational habits rather than security habits.

What the attacker is really after: a session, not a password

The most common analytical mistake is to reason in terms of passwords. Multi-factor authentication is deployed, the subject is considered closed, and the incident review reveals that the attacker never needed the password at all.

What carries value in a modern architecture is the token: the artefact issued after authentication, which proves to the application that verification has already taken place. A stolen access token gives the attacker exactly what they came for — access to data — without requiring them to authenticate again. A stolen refresh token gives them something better still: the ability to renew that access for days or weeks, without triggering any suspicious sign-in alert.

That is why current phishing campaigns no longer aim to harvest credentials but to intercept a complete session, relaying the legitimate authentication page in real time. The

user sees the genuine Microsoft page, enters their genuine password, approves their genuine strong-authentication prompt — and the attacker walks away with the session cookie.

Hold on to the practical consequence: a control that only protects the moment of authentication is a partial control. The question to put to your teams is not "do we have MFA?" but "what happens if a valid token is used from a device we do not know?".

What public incidents have demonstrated

Three well-documented cases are enough to frame the subject, and they have the advantage of being verifiable rather than theoretical.

In 2023, the actor Microsoft tracks as Storm-0558 used a compromised signing key to forge authentication tokens and reach mailboxes belonging to government organisations. No password was stolen and no anomalous sign-in alert was raised at the victims' end: from the application's point of view, the tokens were perfectly valid. That incident also revealed that the ability to investigate depended on the logging licence tier held — a point that moved regulators enough for Microsoft to subsequently widen access to audit logs.

In January 2024, Microsoft disclosed that a state-sponsored actor it designates Midnight Blizzard had gained initial access through a password-spray attack against an old test account with no multi-factor authentication, in a legacy tenant. Access was then extended through an OAuth application holding elevated permissions on Exchange Online. Two lessons follow: the forgotten account is not a hygiene detail, and the application identity is an escalation vector on the same footing as a human account.

Finally, several criminal groups have demonstrated in recent years that the most profitable weak link remains the human process: a call to the IT service desk, a request to reset an authentication method, and the complete circumvention of the technical apparatus. No tooling budget covers that path. Only an identity verification procedure at the service desk does.

Why this subject cannot be delegated

There is a legitimate temptation, for a security leader, to treat Entra ID configuration as an operational matter for the team that already runs email and workstations. That delegation produces the same audit finding every time.

Operations teams optimise availability and smooth service. They are measured on ticket volume and on the absence of interruption. Faced with a trade-off between a control that will block ten users and a service-level commitment, they will decide in favour of continuity — not out of negligence, but because that is precisely what is asked of them. This is how permanent exclusions appear in Conditional Access rules, how service accounts end up exempt from strong authentication, and how applications get approved "for now" three years ago.

The security leader's role is not to administer the directory. It is to define what is non-negotiable, to require that every exception be dated and traceable, and to hold an independent measurement of the tenant's real state. The next three parts of this book exist for that purpose.

What you need to decide

- **Who owns the identity configuration?** As long as the answer is "operations", you do not have control, you have hope. Name an owner and explicitly separate who configures, who approves and who verifies.
- **What is your level of ambition on session theft?** This is the structuring trade-off of the whole programme: accepting token replay risk costs little today and a great deal on the day of the incident. Addressing it means committing to device compliance and phishing-resistant authentication, with real user impact.
- **How many exceptions are you prepared to tolerate, and for how long?** Set the principle now, in calm conditions: every exception carries an end date and an owner's name. That rule, decided upstream, will prevent most of the drift observed in the field.
- **Is the IT service desk part of your security perimeter?** If an authentication method can be reset over the phone without strong proof of identity, your technical investment is capped by that procedure.

CHAPTER 2

Anatomy of an Entra ID and Microsoft 365 tenant

This chapter is a deliberate shortcut. It does not describe the whole platform, only the subset of objects and consoles a security leader needs in order to ask the right questions, understand a technical answer and recognise an evasive one.

The tenant: a boundary, not a network

The tenant is the directory instance dedicated to your organisation. It is the real security unit of the Microsoft ecosystem: identities, groups, applications, policies and collaboration data all reside in it, and every administrative action attaches to it.

Two consequences deserve to be understood from the outset.

First, the boundary is neither geographic nor network-based. A tenant is reachable from the internet by design. It is not equipment that filters access, but policy.

Second, an organisation rarely owns a single tenant. Mergers, subsidiaries, pilot projects and test environments create them, and these secondary tenants — less monitored, less licensed, often forgotten — are a recurring entry point. The first audit reflex is to establish an inventory of the tenants genuinely attached to the organisation, including those created through self-service sign-up by a user with a corporate email address.

The objects that matter

Users. Member accounts synchronised from an on-premises directory or created in the cloud, guest accounts originating from other organisations, and service accounts held by technical teams. These three populations share neither the same lifecycle nor the same level of control, and must be treated separately.

Groups. They carry access rights. Two natures coexist: security groups, controlled by IT, and Microsoft 365 groups, which users can create themselves and which come with a mailbox, a SharePoint site and a Teams team. A group is also a privileged object as soon as it is used to assign a role: whoever can modify its membership can, indirectly, grant themselves that role.

Directory roles. Around a hundred built-in roles exist, but only a handful carry major risk: Global Administrator, Privileged Role Administrator, Application Administrator, Authentication Administrator, User Administrator, Exchange Administrator. Global Administrator should be treated as the equivalent of a domain administrator account in an on-premises directory: holding it is equivalent to holding the organisation.

Applications and service principals. This is the least understood object and, in practice, the most dangerous. An application registration defines an application identity; the service principal is its active instance in your tenant, with its permissions and secrets. These identities authenticate without human intervention, are subject to no multi-factor authentication, and can hold permissions far greater than a user's — including reading every mailbox in the organisation.

Devices. A device may be registered, joined to Entra ID, or hybrid joined. That state determines whether a compliant device can be required as a condition of access at all.

Without device management, a significant part of the Conditional Access arsenal remains out of reach.

Administrative units. They allow an administrator's scope to be restricted to a given population. Used properly, they are one of the few tools for structural privilege reduction in Entra ID.

Control planes: who administers what

A recurring source of confusion in meetings comes from the sheer number of consoles. Hold on to the following split.

The Entra ID admin centre covers identities, roles, applications and Conditional Access. The Microsoft 365 admin centre covers users, licences and collaboration services. The Microsoft Purview portal covers compliance, data classification, loss prevention and part of the audit logging. Microsoft Intune covers devices and their compliance. The Microsoft Defender portal covers detection and response. The Azure portal, finally, covers subscriptions and infrastructure resources, which form a distinct security domain but depend on the same directory.

The practical consequence is that no single screen gives the security state of the tenant. Any statement along the lines of "the dashboard is green" must be qualified: which dashboard, which scope, as of when. This is also the rationale for tool-assisted audit work: consolidating a view the platform does not provide natively.

The real entry points

A tenant exposes far more than its administration portals. The workloads — Exchange Online, SharePoint, OneDrive, Teams — are reachable by browser, rich client and mobile. Above all, the Microsoft Graph API offers programmatic access to the entire directory and its data, and the administration PowerShell modules constitute a complete administration channel, independent of the graphical interfaces.

From this follows a rule that often surprises management: restricting access to admin portals protects nothing if the same actions remain possible through Graph or PowerShell. Controls must apply to the identity and the session, not to the interface.

Minimum vocabulary

A few terms will recur constantly, and using them correctly changes the quality of exchanges with your technical teams.

Authentication establishes who you are; **authorisation** determines what you may do. The **token** embodies the result of both. **Consent** is the act by which a user or an administrator grants permissions to an application. **Delegated permissions** are exercised on behalf of a signed-in user and remain bounded by that user's rights; **application permissions** are exercised with no user present and know no such bound. That last distinction is probably the most profitable in this chapter: it separates a harmless application from one capable of reading your entire corporate mail.

What you need to decide

- **Do you hold an inventory of tenants?** Until one exists, your audit scope is declarative. Decide who establishes it and how often it is reviewed.
- **Who is allowed to create applications and consent on their behalf?** This is a governance decision, not a technical setting, and it conditions Chapter 14.
- **Do you accept that administrators hold global scope?** If not, administrative units and scoped delegation must enter the roadmap in the first year.
- **What level of knowledge do you require of your own security team?** A security leader who cannot distinguish a delegated permission from an application permission cannot arbitrate application risk. That is a training need to be budgeted, on the same footing as a tool.

CHAPTER 3

The seven attack paths actually observed

Threat catalogues are long and rarely actionable. This chapter takes the opposite approach: seven paths, the ones found in the large majority of Microsoft 365 tenant compromises documented publicly or handled in the field. For each, only three elements: how it works, what it looks like, and which control cuts it.

1. Adversary-in-the-middle phishing and session theft

The attacker hosts a site that relays the legitimate Microsoft authentication page in real time. The victim authenticates normally, approves the strong-authentication prompt, and the attacker captures the session cookie issued at the end of the process. They then replay that cookie from their own machine and reach the mailbox without ever authenticating.

Signal: a single session used from two very different network addresses or browser fingerprints, the appearance of mailbox rules that automatically move or delete messages, registration of a new authentication method immediately after sign-in.

Control that cuts the path: phishing-resistant authentication — FIDO2 keys, Windows Hello for Business, certificate-based authentication — because the relay cannot replay a proof bound to the site's origin. Failing that, requiring a compliant or domain-joined device sharply reduces the exploitability of a stolen token.

2. Illicit application consent

The attacker sends the victim an invitation to authorise an innocuous-looking application. In one click, the application obtains durable permissions on mail or files. No password is exchanged, no token is stolen, and changing the victim's password revokes nothing.

Signal: consent granted to an application whose publisher is unverified, recent creation of a service principal, Graph API activity from an unknown application.

Control that cuts the path: disabling unrestricted user consent, implementing an admin consent request workflow, and periodically reviewing consented applications. This is a low-cost, high-return control, and it is almost always absent in audited tenants.

3. Abuse of alternative authentication flows

Some authentication flows, notably the device code flow designed for input-constrained equipment, allow an attacker to have the victim approve an authentication the attacker initiated themselves. The victim believes they are approving a legitimate action.

Signal: sign-ins using an authentication flow unusual for your estate, particularly from addresses or countries inconsistent with the user's profile.

Control that cuts the path: explicitly blocking unnecessary authentication flows through Conditional Access. Most organisations have no functional need for them and can prohibit them without side effects.

4. Privileged account takeover

Three variants dominate: password spraying against old or test accounts left unprotected, repeated authentication prompts until the user approves out of fatigue, and calling the IT service desk to obtain the reset of an authentication method.

Signal: massive distributed authentication failures, a burst of denied prompts followed by an approval, authentication method resets outside procedure.

Control that cuts the path: no single technical control suffices. It takes the combination of phishing-resistant strong authentication for administrators, number matching on prompts, time-bound privilege elevation rather than permanent assignment, and an identity verification procedure at the service desk. Chapter 11 returns to this in detail.

5. Non-human identities

Service principals, service accounts and application secrets constitute the most discreet path. They escape multi-factor authentication by nature, their secrets are often stored in code repositories or configuration files, and their permissions have rarely been reviewed since creation. An application identity with full access to Exchange Online amounts to access to the organisation's entire mail.

Signal: long-lived secrets, or expired secrets renewed without review, tenant-wide application permissions, service principal authentication from an unusual address.

Control that cuts the path: an inventory with a named owner for every application identity, replacement of secrets with certificates or managed identities, permissions restricted to the strict minimum, and Conditional Access applied to workload identities where licensing allows.

6. The hybrid path, from on-premises directory to cloud

When an on-premises Active Directory is synchronised with Entra ID, compromise of the on-premises domain becomes compromise of the tenant. Synchronisation components hold elevated privileges, residual federation servers allow authentication assertions to be forged, and certain single sign-on mechanisms rely on directory accounts whose secret is never rotated.

Signal: federated authentications inconsistent with real activity, changes to federation configuration, synchronisation accounts used from an unexpected server.

Control that cuts the path: treating synchronisation and federation servers as tier-zero systems, on the same footing as a domain controller; removing residual federation when it is no longer needed; rotating the secrets of legacy technical accounts. Chapter 19 covers this workstream.

7. External collaboration and guests

External sharing, anonymous links and guest accounts create access that escapes conventional reviews. A forgotten guest keeps their access for years; an "anyone with the link" share outlives the end of the project; a shared channel opens cross-organisation access.

Signal: guests with no recent activity, anonymous links on sites holding sensitive data, invitations issued by users not authorised to do so.

Control that cuts the path: cross-tenant access settings, automatic link expiry, and regular access reviews over the guest population.

What these seven paths have in common

None of them relies on a software vulnerability to be closed by a patch. All of them exploit a configuration, a permission or a procedure. That is good budget news — remediation rarely means buying something — and bad organisational news: remediation means trade-offs, governance and sustained discipline.

What you need to decide

- **Do you treat these seven paths as formalised risk scenarios?** If they do not appear in your risk analysis, your action plan cannot be prioritised rationally.
- **Which path do you consciously accept not to cover this year?** Declining is legitimate, provided the decision is explicit, documented and carried in the risk register — rather than discovered in a post-mortem.
- **Do your detection capabilities cover the signals listed above?** Chapter 21 returns to this, but the question arises now: without adequate logging, these seven paths are invisible.

CHAPTER 4

Shared responsibility: what Microsoft covers, what remains yours

The shared responsibility model is quoted in every presentation and understood in few organisations. The usual formula — "the provider is responsible for the security *of* the cloud, the customer for security *in* the cloud" — is accurate but too abstract to arbitrate a budget. This chapter translates it into operational terms.

What the provider actually delivers

Microsoft operates the physical infrastructure, the hypervisor, the service network, platform availability, patching of its own components and data resilience at service level. It also supplies the security tooling you enable: Conditional Access, threat protection, logging, classification.

The verb deserves attention: the provider *supplies* these capabilities. It does not configure them for you, does not decide your exceptions, and does not carry responsibility for access granted by your own configuration. A tenant compromised through illicit application consent is a customer incident, not a provider incident.

What remains entirely yours

Identities and their lifecycle. Privileges and their assignment. Access policy configuration. The applications authorised in your tenant and the permissions granted to them. Classification and sharing of your data. Device compliance. Detection and response across your perimeter. And, systematically underestimated, retention of your data beyond the service's native windows.

The costliest misunderstanding: backup and retention

Two confusions recur in audits and deserve to be settled once and for all.

The first concerns backup. The replication performed by the provider protects against failure of its infrastructure. It does not protect against mass deletion by a privileged attacker, nor against content encryption, nor against an operational error detected three months later. Native retention and hold mechanisms offer limited recovery windows, governed by settings a privileged attacker can themselves modify. If your continuity plan rests on the assumption that mail and files are backed up, that assumption must be explicitly verified and documented.

The second concerns logs. Native retention of directory sign-in and audit logs is measured in days or weeks depending on licence tier, and collaboration activity logs in months. Yet the average interval between initial compromise and its detection frequently exceeds those windows. An organisation that discovers an incident after four months without having exported its logs will not be able to establish the sequence of events: it will know neither when access began nor what was consulted. That inability has direct regulatory consequences, particularly for notification obligations. Chapter 20 addresses the sizing of this retention.

Licensing as the limiting factor

This is the constraint security leaders discover latest, and often at the worst moment. In the Microsoft ecosystem, a significant share of security controls is conditioned on the licence tier held, and that dependency covers structuring functions.

Conditional Access rules, the foundation of the entire access policy, sit at the first premium directory tier. Time-bound privilege elevation, access reviews and risk-based protection sit at the second. Identity governance — entitlement management, lifecycle workflows — sits in a dedicated offering. Conditional Access applied to application identities sits in an add-on module. On the data side, automatic classification and part of loss prevention sit at the upper tier of the Microsoft 365 suites.

Two symmetrical mistakes follow. The first is to design a target architecture that assumes licences the organisation does not hold and will not buy: the plan is unachievable and loses credibility at the first review. The second is to leave the gap uncoded: executive management then decides without knowing that forgoing a licence tier means no access reviews, or no ability to elevate privileges temporarily.

The professional position is to produce two versions of the roadmap as a matter of course: the one achievable on the current licence base, and the one requiring an upgrade, with its cost and the risks it covers. That is the only way to turn a commercial constraint into an informed decision.

Supplement with a third party, or move up a tier?

The trade-off arises in almost every engagement. Three criteria settle it without dogma.

Integration first: a native control applies at the authentication decision point, where a third-party tool often acts downstream, in detection rather than in blocking. Total cost next: a solution cheaper to buy consumes integration time, maintenance and scarce skills. Reversibility finally: single-vendor adherence is a genuine concentration risk, one that some regulators are beginning to examine explicitly.

In practice, two domains frequently justify a third-party supplement regardless of licence tier: backup of collaboration data, and long-term log retention at controlled cost.

What you need to decide

- **Do you hold a written position on Microsoft 365 backup?** If the answer is "we assume it's covered", you carry an untreated risk that will materialise at the worst moment.
- **What log retention period do you commit to?** Set it from your regulatory obligations and a realistic detection delay, not from the service default.
- **Is the cost of security licensing carried by the security budget or the productivity budget?** The question looks administrative; it in fact determines who arbitrates security trade-offs.
- **Does your roadmap exist in two versions?** One at constant licensing, one costed target. Without that, no informed decision is possible at executive level.

PART II

Assess: the audit method

CHAPTER 5

Scoping an audit: perimeter, stakeholders, duration, deliverables

The quality of an audit is decided before the first extraction. Loose scoping produces a report nobody contests and nobody applies: too broad to be actionable, too technical to be arbitrated, too vague to be held against anyone. This chapter sets out the minimum scoping for an assessment of an Entra ID and Microsoft 365 environment.

What the audit must settle

Start by writing down the question the report will have to answer. It is never "are we secure". It takes one of three forms, and they do not lead to the same work.

What is our level of exposure? This is a posture audit. It measures the gap between the actual configuration and a reference state, prioritises, and feeds a roadmap.

Are we compliant with a given framework? This is a compliance audit. It produces a control-by-control answer, evidenced, and defensible before an external auditor, an insurer or a regulator.

Have we been compromised? That is an investigation, not an audit. The methods, the skills and the urgency differ radically. If the doubt exists, address it first, separately.

The most frequent confusion in organisations is to commission the first and expect the third. Write the question into the engagement letter.

Three perimeters to distinguish

The identity perimeter covers the tenant, human and application identities, roles, access policies and authentication methods. This is the core.

The collaboration perimeter covers Exchange Online, SharePoint, OneDrive and Teams: external sharing, anonymous links, transport rules, retention.

The infrastructure perimeter covers Azure subscriptions and their resources. It depends on the same directory but constitutes a distinct technical domain, with its own frameworks. Including it doubles the workload. Deciding it explicitly prevents scope drift mid-engagement.

To these three a prior question must be added, and it is not optional: **how many tenants?** Until the inventory is established, the stated perimeter is a hypothesis. Always plan a discovery phase upstream, including for tenants created through self-service sign-up on a corporate domain.

The access you need, and the access you must refuse

A posture audit is conducted entirely read-only. Global reader and security reader roles, supplemented as needed with read access to compliance logs and device management, are enough to collect all the evidence.

Refuse an administrative role, even offered in good faith to "move faster". An auditor holding a Global Administrator role contaminates their own evidence: it becomes impossible to assert afterwards that an observed change did not come from them. Separating who measures from who configures is the first rule of the discipline.

A practical point often overlooked: read access must be requested at least a week before the start. In most engagements, the time taken to grant permissions is the leading cause of delay.

Stakeholders

Four roles must be named in the engagement letter, with a person's name rather than a team's.

The sponsor, who will arbitrate the recommendations — usually the security leader or the IT director. The identity technical lead, who answers configuration questions and prepares access. The compliance or legal contact, wherever the audit touches personal data or a regulatory obligation. And the operations representative, because operations will carry the remediation load, and their absence from scoping guarantees an unrealistic action plan.

Realistic duration

For a single tenant, no on-premises directory, no formal compliance requirement: five to eight working days, of which two for collection and three for analysis and writing.

For a hybrid environment with directory synchronisation, several domains and the collaboration perimeter included: ten to fifteen days.

For a multi-tenant group, with regulatory requirements and a board-level readout: beyond twenty days, at which point it becomes preferable to split into batches rather than produce a single three-hundred-page report nobody will read.

Always add calendar time between collection and readout: clarification questions to technical teams take days, not hours.

Deliverables

Three deliverables, no more, and they address three different audiences.

An executive summary of two to four pages: exposure verdict, three major risk scenarios, decisions requested, indicative budget.

A detailed report: findings by domain, evidence, criticality, recommendation, estimated effort.

A sequenced action plan, usable as a roadmap as it stands, with a candidate owner and a time wave for each action.

Chapter 10 returns to the drafting of these documents in detail. Retain only this here: a single hundred-page deliverable is not a deliverable, it is an archive.

Scoping pitfalls

The first is the audit based on assertion. An answer from a technical team is not evidence. Every finding must rest on a dated extraction.

The second is scope drift. Once the engagement starts, every stakeholder wants their subject added. Set the rule: any extension requires a written amendment.

The third is confusion with penetration testing. A configuration audit measures exposure; a penetration test demonstrates exploitability. The two are complementary, neither replaces the other, and commissioning them as a single lot generally produces a mediocre result on both counts.

What you need to decide

- **What question must the report answer?** Write it in one sentence in the engagement letter. Without it you will get a list of gaps, not decision support.
- **Is the Azure infrastructure perimeter included?** The answer changes the workload, the skill profile required and the framework used.
- **Who will carry the remediation?** If that person is not present at scoping, the action plan will be unrealistic and the report will stay on the shelf.
- **Is the report meant to be defensible?** If so, evidence retention, timestamping and integrity must be organised from collection onwards, not reconstructed afterwards.

CHAPTER 6

Choosing a framework: CIS, ANSSI, Zero Trust, MITRE ATT&CK

An audit without a framework is only an opinion. The choice of framework determines what will be examined, what counts as a gap, and how much weight the finding carries against a technical team that will contest it. This chapter presents the four useful bodies of work and, above all, how to combine them without simply stacking them.

CIS Microsoft 365 Foundations Benchmark

This is the most directly usable configuration framework. It enumerates precise controls, each with a rationale, an audit procedure and a remediation procedure. It comes in two levels: level 1 gathers measures applicable without noticeable effect on users, level 2 the hardening measures whose adoption implies a usability trade-off.

Its strengths: granularity, neutrality, regular updates, and above all the ability to produce a finding that cannot be argued with — the control is applied or it is not.

Its limits: it does not rank controls by actual risk. Two hundred level 1 gaps handled without prioritisation will produce an unmanageable action plan. It also assumes a licence tier that is not always held. Finally, it evaluates a configuration, never a usage: a control correctly configured but carrying thirty exclusions passes the test.

The CIS benchmark is therefore an excellent measuring instrument and a poor prioritisation instrument. Use it for the first function only.

ANSSI publications

The French national cybersecurity agency does not publish a Microsoft 365 configuration framework comparable to CIS, but its work provides three things CIS does not.

The tiered administration model, drawn from its Active Directory hardening guidance, remains the best analytical grid for handling the hybrid path and administration workstations. It transposes directly to the cloud: synchronisation components and directory administrators belong to the most sensitive tier and must be administered from dedicated workstations.

Its doctrine on the use of cloud computing services provides the framework for analysing the sensitivity of data entrusted to a provider — a question CIS does not address and that your executive management will ask.

Finally, its qualification schemes for providers and services offer a shared language with authorities and public-sector customers. In a French compliance file, a recommendation anchored in an ANSSI publication carries more weight than one anchored in a foreign framework.

Zero Trust

Zero Trust is not an audit framework: it is a target architecture model. Its three principles — verify explicitly, apply least privilege, assume breach — are not ticks, they orient.

Its usefulness lies elsewhere: it gives the report its narrative backbone. It allows you to explain to executive management why device compliance and phishing-resistant authentication are not two more measures in a list of two hundred, but the two pillars of a coherent model. An audit resting only on a control list produces a pile of fixes; an audit anchored in a model produces a direction.

Its limitation is symmetrical: used alone, it licenses any discourse. "We are on a Zero Trust journey" means nothing measurable. Never accept it as an answer to a finding.

MITRE ATT&CK

ATT&CK describes the techniques attackers use, including in identity environments and software-as-a-service platforms. It is neither a configuration framework nor an architecture model: it is a coverage grid for detection.

Its use in an audit is precise and limited, which is what makes it valuable: confronting the list of techniques applicable to your environment with the list of your effective detection rules, and producing a map of blind spots. It is the only one of the four that answers the question "what would we see if this happened?". Chapter 21 builds directly on it.

The special case of the native score

The security score provided by the platform is useful as a trend indicator and useless as a measure of exposure. Three reasons.

It awards points for measures whose real impact varies considerably, without weighting for your organisation's risk. It includes recommendations tied to products you do not own, which mechanically caps the score and makes comparison between organisations meaningless. And it rewards enabling a capability more than the quality of its application.

A score of seventy-five per cent therefore does not mean three quarters of the risk is covered. It can perfectly well coexist with a complete, untreated attack path. Use it to track a trend over time; never use it as an audit conclusion, and least of all in front of an executive committee, where it will create a false sense of control that is very hard to correct afterwards.

How to combine them

The combination that works in the field comes down to four lines.

CIS is the measuring instrument: it delivers the control-by-control finding. Zero Trust is the target model: it provides coherence and trajectory. ATT&CK is the detection grid: it reveals the blind spots. ANSSI publications and regulatory texts provide defensibility: they justify the requirement to executive management and to third parties.

What not to do: add all four into a single table of several hundred rows. The reader gives up, and the audit loses its prioritising function.

What you need to decide

- **Which framework prevails in case of disagreement?** Name one, once and for all. Without a designated arbiter, every finding gets renegotiated with the technical team.

- **Are you targeting level 1 or level 2 of the benchmark?** Level 2 means accepting user impact. That is a management decision, not a technical choice.
- **Is the native score an official steering indicator in your organisation?** If so, frame its interpretation immediately, before it is used as proof of control in committee.
- **Is your detection coverage assessed, or only your configuration?** Most audits stop at the latter and leave the former entirely unknown.

CHAPTER 7

Collecting evidence: data sources, reliability, pitfalls

An audit finding is worth what its evidence is worth. This chapter covers collection: where the data comes from, what it actually says, and the pitfalls that produce false conclusions with an appearance of rigour.

The principle: dated, reproducible, independent extraction

Audit evidence must satisfy three conditions. It is dated: a tenant's configuration changes daily, and a finding without a timestamp is not defensible. It is reproducible: a third party must be able to replay the extraction and obtain the same result. And it is independent of assertion: it comes from reading the actual state, not from an interview answer.

The requirement looks obvious. In practice, a significant share of the audit reports I review in second reading rest on undated screenshots and on team statements, which makes them unusable at the first challenge.

The sources

The administration interface suits spot checks and illustration, never bulk collection: it is slow, not reproducible, and shows only a partial view of objects.

The directory programming interface — the Microsoft Graph API — is the reference source. It exposes objects as they actually exist, including properties the portals do not display. It is the only source allowing exhaustive, replayable collection.

Command-line administration modules cover the collaboration workloads, some of whose configuration remains inaccessible otherwise, particularly on the mail side.

Sign-in logs, directory audit logs and collaboration activity logs form the fourth source. They do not establish configuration; they measure actual usage — an essential distinction on which the next section rests.

Pitfall number one: confusing capability with application

This is the error that invalidates the largest number of reports.

A Conditional Access policy can exist, be correctly written, and apply to nobody: because it targets an empty group, because it is in report-only mode, because its exclusions cover the entire population, or because a second, less restrictive policy grants the same access by another route.

Likewise, a strong authentication method can be *registered* by a user without being *required* for their access. A tenant where ninety-five per cent of users have registered a strong method, but where no policy enforces it, is a tenant without strong authentication.

The collection rule is therefore: never conclude from configuration alone. Systematically confront configuration with actual sign-in logs. The question to answer is not "does the rule exist?" but "how many accesses took place without it over the last thirty days?". That figure changes the conversation with the technical team, because it cannot be argued with.

The other classic pitfalls

Insufficient permissions produce silent false negatives. An extraction run with an incomplete read role returns an empty list rather than an error. Always verify that a known control set comes back before concluding that an object is absent.

Pagination truncates. Programming interfaces return results in batches. Collection that ignores pagination produces a partial inventory, and the risky object is statistically as likely to sit in the second batch as in the first.

Some properties are not returned by default. Last sign-in dates, extension properties and certain security attributes must be requested explicitly. Their absence from an extraction does not mean their absence from the directory.

Account states look alike. A disabled account, an unlicensed account, an account blocked from signing in and an account in the deleted-items retention do not carry the same risk. Confusing them inflates findings and discredits the report at the first technical review.

Legacy mechanisms overlap. Strong authentication may be enforced by a legacy per-user setting, by security defaults, or by a Conditional Access policy. These mechanisms coexist, sometimes exclude one another, and their accumulation creates blind spots. Establish which one is actually in force before making any finding on the subject.

Synchronised objects are not fixed where they are observed. An inconsistent attribute seen in the cloud often originates on premises. The finding is valid; the recommendation must target the right source.

Personal data and evidence retention

Audit collection brings back personal data: identities, addresses, sign-in logs, network addresses, sometimes location data. Three rules suffice.

Extract the strict minimum: the identifiers and attributes needed for the finding, never mailbox or file content. Store extractions in a controlled, encrypted location, with a retention period defined in advance and aligned with the period during which the report may be challenged. Document that retention in the engagement letter — it is also what you will require of your own providers.

What you need to decide

- **Do your findings rest on extractions or on interviews?** Require dated evidence for every finding in any report handed to you, including from a reputable firm.
- **Do you measure usage or only configuration?** Without confrontation with the logs, your audit describes an intention, not a reality.
- **Where are extractions stored, and for how long?** This is both a security and a data protection question; it must be settled before collection, never after.
- **Are your audits replayable identically?** A non-reproducible audit cannot measure progress, and Chapter 26 depends on it entirely.

CHAPTER 8

The twenty controls that reveal eighty per cent of the risk

A complete benchmark holds several hundred control points. This list keeps twenty. They are not the twenty most important in absolute terms: they are the ones whose verification, at the start of an engagement, most effectively separates a well-run tenant from an exposed one. In practice, they allow a first diagnosis in a single day.

Each is phrased as a question that an extraction answers with a fact, never with a judgement.

Privilege

- 1. How many accounts hold the Global Administrator role, and since when?** Beyond five in a mid-sized organisation, the role's informational value disappears. Check permanent assignments against eligible ones as well.
- 2. Are those accounts dedicated to administration?** A privileged account that receives mail, browses the internet and opens documents pools productivity risk with administration risk.
- 3. Do emergency access accounts exist, and have they been tested?** Two accounts, cloud-only, with strong authentication, monitored, and a test procedure dated less than a year ago.
- 4. Is privilege elevation time-bound?** Permanent assignment of a sensitive role turns every compromise of an administrator's workstation into a compromise of the tenant.
- 5. Who can modify membership of the groups used to assign roles?** This is the most frequently missed escalation path in audits.

Authentication

- 6. What proportion of successful sign-ins over the last thirty days used a strong method?** The question is about sign-ins, not about method registration.
- 7. Do administrators use a phishing-resistant method?** Strong authentication by push notification or code remains defeatable by relay. For sensitive roles it is not a sufficient answer.
- 8. Are legacy authentication protocols still in use, and by whom?** The finding is not made on configuration but on logs: the protocol may be blocked in theory and remain live for an exempted population.
- 9. What is the service desk procedure for resetting an authentication method?** An organisational control, but it caps the effectiveness of the eight above.

Conditional Access

10. How many policies are enabled, and how many have been in report-only mode for more than three months? A policy in report-only mode for a year is an intention, not a control.

11. Who is excluded, from which policies, and since when? Consolidate every exclusion into a single list: it is the most revealing document of an audit, and it exists in no portal.

12. Is there a fallback policy blocking access not explicitly authorised? Without one, the whole arrangement operates as a list of exceptions rather than a list of authorisations.

Applications and non-human identities

13. Can users consent freely to third-party applications? A single setting, a major effect, and an almost zero remediation cost.

14. Which applications hold broad application permissions over mail, files or the directory? This list is short and must be justifiable line by line.

15. How many application secrets expire in less than ninety days, and how many have a validity longer than two years? Long-lived secrets signal the absence of application lifecycle management.

External collaboration

16. How many guest accounts have not authenticated in ninety days? The ratio of active to inactive guests is a direct indicator of governance quality.

17. Is anonymous link sharing allowed, and on which sites? To be cross-referenced with data classification where it exists.

Data, devices, logging

18. Does access to sensitive resources require a compliant or joined device? This is the control that limits the exploitability of a stolen token, and therefore the one that addresses the most frequent attack path.

19. What is the effective retention of sign-in and audit logs, and are they exported off the platform? Compare that period with your realistic detection delay, not with a theoretical value.

20. Are the on-premises synchronisation components administered as tier-zero systems? A single poorly segmented server of this kind nullifies all nineteen preceding controls.

How to use the list

These twenty questions do not replace a full audit; they prepare it. Ask them at the start of an engagement, and the distribution of answers will tell you where to focus detailed analysis. They are also an excellent entry grid for assessing a tenant you inherit — after an acquisition, a change of provider, or on taking up a new post.

What you need to decide

- **Do these twenty answers exist today in your organisation?** If obtaining the consolidated exclusion list takes three weeks, that is already a finding in itself.
- **How often do you recompute them?** An annual measurement says nothing about the tenant's state for the rest of the year; Chapter 26 covers continuous control.
- **Are you prepared to publish these figures to your management?** Indicators that never surface fund no remediation.

CHAPTER 9

Scoring and prioritising: criticality, effort, user impact

A list of gaps is not an action plan. Prioritisation is the part of the work that turns a finding into a decision, and it is also the part most reports handle worst, settling for a severity rating inherited from the framework.

Why a single score is not enough

Criticality alone does not support arbitration, because it ignores what actually blocks remediation. In the field, the measures that are not applied are almost never the ones nobody knows about: they are the ones whose effects the organisation fears. A prioritisation that does not model that brake produces a plan operations will politely refuse and never execute.

Three axes suffice, and they must be kept separate.

Axis 1 — Criticality

It combines two factors: probability of exploitation and impact if exploited. A four-level scale is enough, provided each level is anchored on an observable fact rather than an impression.

The highest level is reserved for what opens a complete path to tenant compromise: a Global Administrator role without phishing-resistant authentication, an application identity with broad mail access, an unsegmented synchronisation server.

The level immediately below covers what strongly facilitates an attack without achieving it alone: active legacy protocols, unrestricted application consent, absence of a device requirement.

Anchor the scoring on the seven paths from Chapter 3. A gap that belongs to none of those paths cannot be rated at the top level, whatever score the framework assigns it.

Axis 2 — Effort

Effort is not only technical. It adds three components: implementation load, validation and testing load, and organisational coordination load. The last is almost always the heaviest and almost always missing from estimates.

Removing unrestricted application consent takes one minute of configuration and three weeks of preparing an approval workflow with the service desk. Rating that measure as low effort guarantees an action plan that is late in its first month.

Axis 3 — User impact

This is the determining axis, and the one reports most often omit. It measures what the change means for people: an extra step at sign-in, a personal device that stops working, an external share that must now go through a request.

A measure with high user impact is not a bad measure. But it demands an explicit decision at the appropriate level, prepared communication and often a staged rollout by

population. Treating it as one more technical measure is the most frequent cause of failed deployment, and of rollback — rollback being more costly than never deploying, because it discredits the entire programme.

The decision matrix

Cross criticality with effort, using user impact as a sequencing criterion rather than a selection one.

High criticality, low effort: immediate action, with no committee in between. These are the quick wins, and they fund the credibility of the rest of the plan.

High criticality, high effort: these are the workstreams. They need a budget, an owner and a milestone. Their number must stay limited — three a year at most in a mid-sized organisation.

Low criticality, low effort: handle in batches, in a quarterly hygiene campaign, without making them a committee subject.

Low criticality, high effort: document as accepted risk, with a review date. That is a legitimate decision, provided it is written down.

Sequencing in waves

Three waves work better than a detailed calendar, which goes stale in two months.

At thirty days, the measures with no user impact and no dependency: they demonstrate execution.

At ninety days, measures with limited impact and short preparation, notably on privileged populations, which are small and therefore quick to address.

At one hundred and eighty days, the structural workstreams: devices, phishing-resistant authentication across the whole population, application governance.

This sequence has a useful property: it places the most sensitive measures after a period in which the team has already delivered, which makes the trade-off easier to obtain.

What you need to decide

- **Is your criticality scale anchored on attack paths or on a rating inherited from a framework?** Without anchoring, prioritisation is arbitrary and will be renegotiated at every committee.
- **Who arbitrates measures with high user impact?** If the answer is "the technical team", the arbitration will not happen and the measure will not be deployed.
- **How many structural workstreams can your organisation genuinely absorb this year?** Answering three rather than ten is an act of steering, not a retreat.
- **Are accepted risks written down, dated and attributed?** An accepted risk with no name and no date is not a decision, it is an oversight.

CHAPTER 10

Writing a report the executive committee will actually read

An audit report has one function: to provoke decisions. A document that is accurate, exhaustive and without effect is not a good report; it is wasted work. This chapter describes the structure that, in practice, obtains the decision.

What the reader is looking for

The decision-maker opening your report asks three questions, in this order: are we in danger, how badly, and what must I decide today. They are not looking for the list of gaps. If they do not find those answers in the first two pages, they will delegate the reading, and the arbitration will not take place.

That leads to a simple rule: the summary is not a summary of the report, it is the report. The rest is justification.

The five-block structure

The verdict, one page. One sentence concluding on the level of exposure, three facts supporting it, one deadline. The verdict is expressed in exposure, not in compliance: "a compromised administrator account would today allow access to every mailbox without detection" is a verdict. "Forty-seven CIS controls are in breach" is not.

The scenarios, two pages. Three scenarios at most, drawn from the paths in Chapter 3 and instantiated on your environment, with the audit facts that make them plausible. This is the part that makes the risk understood, because it tells a sequence rather than a state.

The plan in waves, one or two pages. The three waves from Chapter 9, with a candidate owner, an estimated effort and the risk covered for each action. No dates to the day.

The decisions requested, one page. This is the block most often missing, and it is the one that turns a report into a useful meeting. Three to five decisions, phrased so they can receive a yes or a no: accept a given user impact, fund a given licence change, appoint a given owner, formally accept a given residual risk.

The technical appendix, no size limit. Detailed findings, evidence, extractions, framework references. It serves the technical team and defensibility. Nobody else will read it, and that is not a problem.

What to avoid

The single score on the cover. It will be remembered, quoted out of context, and become the objective in place of the risk.

Portal screenshots. They age within weeks, prove nothing without a timestamp, and inflate the document.

Untranslated jargon. Executive management does not have to know the vendor's technical vocabulary. Writing "an application approved by a user can read every mailbox" rather than the technical equivalent is not dumbing down: it is the condition for the decision to be taken.

Uncosted recommendations. A recommendation with no estimated effort cannot be arbitrated and will be deferred.

The exhaustive catalogue in the body of the report. Two hundred lines of findings produce paralysis, never action.

The readout

Plan two distinct meetings, never one.

The technical review, with the teams concerned, before the executive readout. Its purpose is to correct factual errors — there are always some — and to secure agreement on estimated effort. A finding contested live in front of executive management destroys the credibility of the entire report, even when you are right.

The executive readout, short, centred on the verdict, the scenarios and the decisions requested. An hour is enough if the document is well built. Close on the decisions, not on thanks: the meeting must produce names and commitments, recorded in writing within twenty-four hours.

What comes next

A report with no follow-up mechanism reverts to being a document within six weeks. Set, at the readout itself, the date of the progress review, the tracking format for the three waves, and the indicator that will be presented at the following committee. Chapter 26 covers the durable steering arrangement; it starts here.

What you need to decide

- **Does your report contain a decisions page?** Without it you will get comments, not decisions.
- **Is the verdict expressed in exposure or in compliance?** The second formulation obtains compliance; the first obtains budget.
- **Who validates the facts before the executive readout?** Skipping this step is the surest way to lose a technically flawless audit.
- **What is the date of the first progress review?** Set it in the readout meeting, or it will never be set.

PART III

Remediate: the priority workstreams

CHAPTER 11

Privileged accounts: roles, time-bound elevation, emergency access

If your programme could contain only one workstream, this would be it. Privileged accounts carry the most favourable effort-to-benefit ratio in the entire ecosystem: the population is small, the measures are well known, and user impact is confined to people who understand why it is imposed on them.

Reduce before protecting

The usual reflex is to harden the protection of existing accounts. The correct order is the reverse: reduce first, protect second. Protecting twenty Global Administrator accounts costs more and works less well than bringing the number down to four.

Reduction proceeds in three passes. The first removes historical assignments, those whose justification nobody remembers. The second replaces the broadest role with a specialised one: most day-to-day administration tasks do not require Global Administrator but a narrower role. The third restricts scope itself, through administrative units, for administrators whose remit corresponds to an entity, a country or a subsidiary.

The exit objective is a number, and it belongs in your roadmap: fewer than five permanent assignments of the highest role, whatever the size of the organisation.

Separate the administrative identity from the productivity identity

A privileged account must not receive mail, must not browse, must not open documents. The rule is old, it has held in on-premises directories for twenty years, and it applies identically in the cloud: it prevents a weaponised document or a phishing link from opening a privileged session directly.

In practice this means a separate account, with no mail licence, and an identifier that does not follow the corporate address format — a predictable format makes enumeration and password spraying easier.

The objection is always the same: convenience. It is admissible for occasional administrators, and that is precisely what time-bound elevation is for.

Time-bound elevation

A permanent assignment means the privilege exists twenty-four hours a day, including while the administrator reads their email. An eligible assignment, activated on demand for a limited period, with justification and approval, reduces the exploitation window considerably.

Three settings separate a real arrangement from a cosmetic one: activation requires strong authentication at the moment of elevation, not the mere replay of the session opened that morning; the activation window is short, two to four hours; and activations of the most sensitive roles trigger a notification and, for some, approval by a third party.

The classic trap: configuring time-bound elevation while retaining permanent assignments "for emergencies". Those residual assignments cancel the benefit, because an attacker will target precisely those. The right answer to emergencies is not a permanent assignment; it is emergency access accounts.

Emergency access accounts

Two accounts, cloud-only, never synchronised, with an identifier on a dedicated domain, protected by hardware-based strong authentication, excluded from the policies liable to lock them out if the authentication service fails, and monitored: any sign-in by these accounts must raise an immediate alert to the security team.

Two mistakes to avoid: excluding them from every control without monitoring them — they then become the best target in the tenant — and never testing them. A half-yearly test procedure, with a dated record, is part of the control. An untested emergency account is not an emergency account.

Administration workstations

This is the most effective and least applied measure. As long as administration is performed from the ordinary productivity workstation, the tenant's security is capped by that workstation's: malicious code residing there will see privileged sessions pass through.

The tiering principle from directory hardening guidance applies without transposition: tenant administration tasks are performed from a dedicated, hardened workstation, with no mail and no general browsing, administered separately. That workstation belongs to the highest sensitivity tier, alongside the synchronisation servers.

This is a workstream with real organisational impact, to be placed in the third wave, but it must be there. An organisation applying every measure in the previous sections while administering from a productivity workstation retains a complete compromise path.

The service desk and resets

The last link, often outside the technical perimeter and yet decisive. If an authentication method can be reset on a simple phone call, the entire arrangement is bypassed by telephone. Three measures suffice: identity verification that does not rest on public or guessable information, traceability of every reset, and a separate, reinforced procedure for privileged accounts, with second-person approval.

What you need to decide

- **What is your numerical target for permanent assignments of the highest role?** Without a written target, reduction will not happen.
- **Do you accept separate administrative accounts for your teams?** That is a convenience-versus-risk trade-off, and it belongs to you, not to operations.
- **Are dedicated administration workstations in this year's roadmap?** If not, write down the residual risk explicitly: it cancels part of the benefit of the other measures.

- **Is the service desk reset procedure reinforced for privileged accounts?** It is the most used and least costly bypass available to an attacker.

CHAPTER 12

Phishing-resistant authentication and the end of legacy protocols

This chapter covers two distinct workstreams run in parallel because they condition each other: raising the level of authentication methods, and removing the paths that allow them to be avoided.

The hierarchy of methods

Not all strong authentication methods are equal, and the gap between them is larger than the gap between password-only and strong authentication.

Codes sent by text message are the weakest level: interception through number takeover, transmission over a network outside your control. They remain preferable to no method at all, and nothing more.

Approval notifications and application-generated codes represent the intermediate level. They resist password theft but not real-time relay: the victim approves, the attacker takes the session. Number matching and context display reduce fatigue-driven approval without addressing the relay.

Phishing-resistant methods — security keys, certificate-based authentication, platform-bound authentication — are the only level that genuinely addresses attack path number one from Chapter 3, because the proof produced is bound to the site's origin and cannot be replayed elsewhere.

The deployment sequence

Full simultaneous deployment almost always fails. The sequence that works proceeds by population, from the most sensitive to the broadest.

Privileged roles first: a small population, high motivation, no admissible cost argument. It must be handled within ninety days.

Then exposed populations: executive management, finance, human resources, executive assistants — the real targets of payment fraud and espionage.

Finally the rest of the organisation, on an annual calendar, with a coexistence phase during which older methods remain accepted for non-sensitive uses.

Three conditions determine success: having a fallback method for loss or forgetting, having addressed the service desk reset procedure before deployment, and providing support for populations without a corporate smartphone — this is the most frequent social blocker, and it is solved with a hardware security key.

The password question

Once a phishing-resistant method is deployed, the password ceases to be the determining factor. Two practical consequences.

Complexity rules and periodic expiry today produce more reuse and more sticky notes than security. The position recommended by the main standards bodies converges: lengthen, block known-compromised passwords, and stop systematic expiry in the absence of any indication of compromise.

The medium-term objective is removing the password as an authentication factor for equipped populations. This is no longer an avant-garde position; it is the platform's default trajectory.

Legacy protocols

An older authentication protocol cannot present a second factor. As long as it remains accepted, it constitutes a complete bypass of your authentication policy, however carefully your access policies are written.

Blocking always runs into the same uses: multifunction printers sending mail, older line-of-business applications, monitoring scripts, a few unmaintained mail clients. The effective approach works from the logs: identify what actually uses these protocols over thirty days, handle each case by name, then block.

Once blocked, the protocol must stay blocked. Every exemption granted "while we migrate the application" must carry a real end date, failing which it becomes permanent — this is the mechanism by which organisations that have officially removed legacy authentication retain it for a population of a few dozen accounts, several of them privileged.

Do not forget the second path

Blocking legacy protocols and requiring a strong method is not enough if an alternative authentication flow remains open, or if an older application holds an exclusion. The final check consists of verifying, in the logs, that no successful sign-in over the last thirty days escaped the arrangement. That verification is the only finding that counts; the rest is configuration intent.

What you need to decide

- **Which population moves to a phishing-resistant method this year?** Administrators at minimum, and the decision is taken now, not after the incident.
- **Are you funding hardware keys?** It is the answer for populations without a corporate phone and the project's most common social blocker.
- **Do you still enforce periodic password expiry?** If so, the decision is yours: the practice degrades security and costs service desk time.
- **What is the real end date of the last legacy protocol exemptions?** With no date, they are permanent.

CHAPTER 13

Conditional Access: a maintainable, testable rule set

Conditional Access is the central decision point of the architecture. It is also the component that degrades fastest: policies accumulate, exclusions sediment, and nobody dares modify a set nobody understands any more. This chapter aims at maintainability rather than sophistication.

The guiding principle

A policy set must be readable in under an hour by someone who did not write it. That constraint, apparently aesthetic, is in fact the first security control: an unreadable set is never audited, never tested, and ends up permitting what it believes it forbids.

From it follows a recommended structure: few policies, each with a single objective, named according to an explicit convention that states the target population, the condition and the effect.

The minimum architecture

Six policies cover most needs of a mid-sized organisation.

A policy blocking unnecessary protocols and authentication flows. A policy requiring a phishing-resistant method for privileged roles. A policy requiring strong authentication for all users. A policy requiring a compliant or joined device for access to sensitive resources. A policy framing guest account access. And a fallback policy blocking anything not explicitly authorised.

The last is the most important and the most frequently absent. Without it, the arrangement operates as a list of prohibitions: anything unforeseen is allowed. With it, it operates as a list of authorisations, which is the only defensible posture.

Exclusions

Exclusions are the systematic point of failure. They are added one at a time, each justified in the moment, and nobody ever holds the consolidated view.

Three rules contain the drift. Every exclusion carries an owner's name and an end date, without exception, including for emergency access accounts — whose end date is simply the next review. Every exclusion is carried by a dedicated, explicitly named group, never by an account hard-coded into the policy, so that the consolidated list can be extracted. And the consolidated exclusion list is reviewed quarterly, in a meeting, by the security leader.

That last point looks bureaucratic. In practice it is the control that separates a maintained arrangement from one that erodes.

Test before enforcing

Every change goes through report-only mode, for a defined period, with analysis of the accesses that would have been blocked. That mode is a step, not a state: a policy remaining there for more than a few weeks must be enforced or deleted.

Simulation tools allow the effect of a policy on a given user and context to be verified before enforcement. Their use must be systematic for any change affecting a broad population, and it must be documented in the change record.

What breaks production

Three mistakes cause most access incidents in production.

Full lockdown: a policy applied to all users, including emergency accounts, with a condition impossible to satisfy. This is precisely the scenario emergency access accounts address, provided they have been excluded and tested.

Forgetting non-human identities: service accounts and application principals cannot satisfy a strong authentication requirement. Including them in a general policy breaks technical chains, often at night. They must be handled by dedicated policies, as set out in Chapter 15.

Uncontrolled overlap: two policies overlap, the more restrictive applies, and the observed effect matches neither intent. This is the main argument for a small number of well-named policies.

Documentation

Every policy should carry three lines: what it protects, why it exists, and who approved it. That documentation fits in a table and is maintained in a few minutes per change. Its absence is the best predictor of degradation at eighteen months, once the team has changed.

What you need to decide

- **Does a default-block policy exist?** If not, your access policy operates as a list of exceptions, and completeness is impossible.
- **Who reviews the consolidated exclusion list, and how often?** Name the person and the frequency, or the list will never be reviewed.
- **Is report-only mode a step or a refuge?** Set a maximum duration and hold to it.
- **How many policies can your organisation genuinely maintain?** Answering six and keeping to it beats thirty policies nobody rereads.

CHAPTER 14

Applications and consent: the blind spot

There is no subject where the gap between actual risk and attention paid is as wide. In most audited tenants, nobody knows how many third-party applications hold access to corporate data, nor who granted it to them.

The mechanism

A third-party application requests permissions; a user or an administrator grants them; the application obtains durable access, exercised through its own tokens. Three properties make this mechanism dangerous.

The access survives a password change and the revocation of the user's sessions: only revoking the consent removes it.

It produces no authentication event identifiable as suspicious, since the application authenticates legitimately.

And the user granting consent does not gauge the scope of what they are granting. A screen asking permission to "read your mail" is accepted without thought in ordinary working conditions.

The distinction that decides everything

Delegated permissions are exercised on behalf of a signed-in user: the application can only reach what that user can reach. Application permissions are exercised with no user, at the scope granted, potentially the whole tenant.

An application holding an application-level read permission on mail can read every mailbox in the organisation, including executive management's. This is the vector used in several documented major compromises. That list, in your tenant, must be short, known and justifiable line by line.

The three measures that address the subject

Close unrestricted consent. The historical default allowed users to consent themselves. Restricting it is a measure with no technical cost and immediate effect. The restriction must come with a request workflow, failing which users will work around it with personal accounts — which makes the problem worse rather than solving it.

Assess the requests. An approval workflow only has value if the approver has what they need to decide: verified publisher or not, permissions requested, scope, existing alternative. Three questions suffice in practice: who publishes the application, what data it reaches, and what happens if its publisher is compromised.

Review what exists. The initial inventory is almost always instructive: test applications consented years ago, tools from a completed project, connectors from a provider whose contract has ended. The review targets, in order, broad application permissions, then applications with an unverified publisher, then those with no recent activity.

Internal applications

Internally developed applications fall under the same regime, with one added difficulty: they are created by development teams, often with broader permissions than necessary, and nobody comes back to narrow them.

Two requirements: a named owner for every application registration, and a justification of permissions at creation time. This control belongs to application governance rather than security operations, but security must require it.

Detection

A granted consent is a logged event. Three useful alerts: consent granted to an unverified application, addition of a broad application permission, and creation of a service principal outside a known workflow. These three alerts are low-noise and cover attack path number two from Chapter 3.

What you need to decide

- **Is unrestricted user consent still open in your tenant?** If so, this is the highest-return measure in the whole book, and it closes in a day.
- **Who approves application requests, and within what response time?** A workflow taking more than a few days will be bypassed.
- **Do you hold the list of applications with broad mail permissions?** If obtaining that list takes more than an hour, the subject is not governed.
- **Do internal applications have a named owner?** With no owner, no later review is possible.

CHAPTER 15

Non-human identities: service principals, secrets, certificates

Non-human identities often outnumber human ones, and they benefit from none of the controls designed for the latter: no strong authentication, no leaving date, no manager review. This is a structural blind spot.

What we are talking about

Three families, to be handled differently.

Service principals corresponding to third-party or internal applications, authenticating with a secret or a certificate. Managed identities, whose secret is handled by the platform and never exposed. And user accounts repurposed as service accounts — a human account used by a script or an application, often with a shared password and an exemption from strong authentication.

The third family is the most dangerous and the most widespread. It combines every drawback: it consumes a licence, it is excluded from controls, it nominally belongs to someone who may have left the organisation, and its password circulates in configuration files.

The workstream, in order

Inventory and assign. Every non-human identity must have a named owner, a justification and a review date. That inventory exists in no portal: it is built and maintained, and it is the prerequisite for everything else.

Eliminate human service accounts. Each case is handled individually: replace with a service principal or a managed identity. It is long work, to be run in waves, starting with accounts holding privileges or data access.

Replace secrets with certificates or managed identities. A shared secret always ends up in a code repository, a ticket or a configuration file. Where replacement is not possible, imposing a short validity forces rotation and reveals abandoned identities.

Narrow permissions. Permissions granted at creation are almost always excessive. The review targets broad application permissions first, as in the previous chapter.

Apply access control. Where licensing allows, workload identities can be subjected to network origin restrictions: an application identity authenticating from an unexpected address is then blocked, not merely logged. It is the only control that genuinely addresses secret theft.

Rotation

Secret rotation is an operations subject before it is a security subject. It fails for one reason: nobody knows what breaks when a secret changes.

The answer lies in the inventory and the owner. A planned rotation, announced to the owner, with an overlap window during which the old and new secrets coexist, passes without incident. An improvised rotation after an incident breaks production and feeds the argument that "security stops us working".

Monitoring

Three signals deserve an alert: a service principal authenticating from an unusual origin, the addition of a secret or certificate to an existing application — a classic persistence technique after compromise — and the use of an application identity outside its usual activity window.

The second signal matters particularly. An attacker who has obtained privileges frequently adds a secret to an existing legitimate application: this gives durable access that survives password resets and session revocation.

What you need to decide

- **Does an inventory of non-human identities with owners exist?** Without it, none of the other measures is executable.
- **Do you still allow the creation of human service accounts?** Prohibiting the practice going forward is simpler than clearing the backlog, and it is the starting point.
- **What maximum validity do you impose on secrets?** A short value forces rotation and reveals abandoned identities.
- **What happens today if an application secret leaks?** If the answer is "we would not know", monitoring this family of identities belongs in the roadmap.

CHAPTER 16

Guests, B2B and external sharing

External collaboration is the domain where security conflicts most directly with usage. The measures are technically simple and politically difficult, because they touch how teams work with their partners. They are therefore addressed through governance rather than blocking.

What creates external access

Four mechanisms, often conflated, with distinct risks.

Inviting a guest account, which creates an identity in your directory and gives it access to the resources shared with it. Link sharing, which may be restricted to named people, extended to anyone in the organisation, or anonymous. Shared channels, which open access between two organisations without creating a guest account. And third-party application access, covered in Chapter 14.

Anonymous sharing is the most problematic: the link circulates, outlives the departure of the people concerned, and appears in no access review based on identities.

Baseline settings

Three decisions structure the domain.

Who can invite? Allowing all users to invite is a default rarely chosen consciously. Restricting invitation to certain roles, or subjecting it to approval, sharply reduces the guest population without blocking collaboration.

With which organisations? Cross-tenant access settings allow differentiated rules: known partners with lighter conditions, the rest of the world with strict conditions or a block. It is the most useful mechanism in the domain, and the least used.

Which sharing links? Disabling anonymous sharing, or failing that limiting its lifetime and restricting it to read-only, addresses most of the document leakage risk.

The question of guest authentication

A poorly understood point: your organisation's access policies apply to guests reaching your resources, but authentication takes place in their home organisation. You therefore inherit their security level.

Two consequences. First, requiring strong authentication from guests is legitimate and technically possible, including by accepting the verification performed by their organisation where that organisation is trusted. Second, for critical partners — managed service providers, firms with access to sensitive data — the security level of their own tenant becomes your problem, and it belongs in the contract.

Lifecycle

This is the heart of the subject. A guest is created for a project and almost never removed at the end of it, because nobody owns the ending.

Three measures address it. Assigning an internal sponsor to every guest, responsible for their justification. Automatic access expiry, with explicit renewal rather than tacit continuation. And periodic access review over the guest population, which relies on the governance capabilities described in Chapter 24.

Failing those mechanisms, a manual half-yearly review based on an extraction — guests with no sign-in for ninety days — already delivers most of the benefit at low cost.

Shared channels and multiple tenants

Cross-organisation shared channels and multi-tenant arrangements create access that does not appear in conventional reviews of guest accounts. They must be inventoried separately.

The frequent case: a subsidiary with its own tenant, connected to the group tenant through access agreements. The subsidiary's security posture then becomes a component of the group's, without any reporting line necessarily reflecting it.

What you need to decide

- **Can all your users invite external people?** That is a default; confirming or restricting it is a decision, not a setting.
- **Is anonymous link sharing acceptable in your organisation, and on which spaces?** Answer in writing, otherwise the default answer is "everywhere".
- **Who sponsors a guest?** With no named sponsor, no review can reach a conclusion.
- **Does your critical partners' security level appear in your contracts?** You inherit their authentication; it is a procurement subject as much as a security one.

CHAPTER 17

Data: classification, loss prevention, labels — the viable minimum

Data protection programmes fail more often than others, for one constant reason: they begin with an ambitious taxonomy and end with a deployed tool nobody uses. This chapter proposes the reverse: the minimum that produces an effect, and the order in which to reach it.

Why these programmes fail

Three causes recur.

Classification is designed by a working group that produces six levels, with detailed usage rules, before any contact with reality. Users cannot choose between "internal" and "restricted", so they choose at random or not at all.

Loss prevention is switched on in blocking mode across a broad perimeter. False positives interrupt legitimate business processes within days, management demands a rollback, and the subject becomes politically dead for two years.

Finally, the programme is carried by security alone, with no business owner for the data. Yet the sensitivity of data is a business question; security can tool it, not decide it.

The viable minimum

Three levels, not six. Public, internal, confidential. A fourth level, reserved for a precise and rare category — merger files, health data, industrial secrets — may be added where a real need exists. Beyond that, users no longer choose correctly, and wrong classification is worse than none.

A default label. Unclassified content must automatically receive the internal level. This removes the often-majority population of unlabelled documents and makes the statistics usable.

Encryption at one level only. Reserve document-attached encryption for the highest level. It is the most protective measure — it survives exfiltration of the file — and the most constraining in use, particularly for external sharing and automated processing.

Loss prevention in observation mode first. Enable rules in logging mode for two to three months, measure the triggers, adjust, and switch to blocking only for rules whose false-positive rate is under control. This sequence is not excessive caution: it is the difference between an arrangement that holds and one that is abandoned.

Where to start

Not with the entire document estate, but with flows of proven risk: regulated data sent externally, anonymous sharing of documents carrying a confidential label, automatic mail forwarding to personal addresses.

That last case deserves a specific mention: an auto-forwarding rule created in a compromised mailbox is both a common exfiltration mechanism and a signal of compromise. Addressing it yields an immediate benefit in detection as well as protection.

Data governance

Two elements are almost always missing and cost little to put in place.

An owner for each major data domain — finance, human resources, legal, research — who decides sensitivity and arbitrates exceptions. Security provides the tooling; the business decides.

A retention policy. Data you do not keep cannot leak. Scheduled deletion of content past its useful life reduces both the exposure surface and the regulatory burden, and it is often the only workstream in this domain whose return on investment is beyond dispute.

What you need to decide

- **How many classification levels?** Three is almost always the right answer; six is almost always the founding error.
- **Who decides data sensitivity?** If the answer is "security", the programme will fail for lack of business legitimacy.
- **Do you accept an observation phase before blocking?** Refusing it means accepting the rollback a few weeks later.
- **Is a retention policy actually applied?** It is the most profitable surface-reduction measure and the most often forgotten.

CHAPTER 18

Devices: compliance as a condition of access

This is the workstream that genuinely addresses session theft, and therefore the most frequent attack path. It is also the heaviest, because it touches the entire estate and every user's daily experience. It is not decided at technical level.

The reasoning

A token stolen by relay is used from the attacker's machine. If access to resources requires, in addition to identity, a known and compliant device, the token becomes unusable outside the estate. It is the only measure that cuts the path without depending on user behaviour.

It brings a second benefit, often decisive in the budget discussion: it restores the notion of an inventory. An organisation requiring a compliant device knows which devices reach its data — information most organisations believe they hold and almost never do.

Prerequisites

A device requirement assumes a managed estate: devices registered or joined to the directory, enrolled in device management, with defined compliance rules.

Compliance rules must stay frugal. Disk encryption, up-to-date operating system, active malware protection, screen lock after inactivity: four criteria cover the essentials. A policy with thirty criteria produces a population of devices non-compliant for minor reasons, hence exclusions, hence the failure of the measure.

The hard cases

Three populations systematically block deployment, and they must be handled before, not during.

Personal devices. Two options: exclude them from sensitive resources by allowing only browser access without download, or bring them in through application-level management that protects corporate data without administering the device. The choice is a matter of organisational culture and applicable employment law, not technology.

Contractors and temporary staff. They often work on equipment you do not own. The contractual answer — providing a workstation, or requiring compliance of the contractor's own — must be settled upstream.

Shared devices and production workstations. Workshops, points of sale, laboratories: they require specific configurations and often permanent exceptions, which must then be offset by other controls.

Deployment sequence

By resource rather than by population. Start with administration applications and the most sensitive data, where the population concerned is small and motivated. Then extend to line-of-business applications, then to mail and files.

This progression demonstrates value before confronting volume, and handles edge cases as they arise rather than in a single wave of exceptions.

The link with revocation

One last point, often ignored: a compliant-device requirement only takes full effect if loss of compliance has a rapid effect on sessions in progress. Continuous evaluation mechanisms allow a change of context to be acted on without waiting for the token to expire. Check that they are active; without them, a device that becomes non-compliant retains its access for the lifetime of the session.

What you need to decide

- **Do personal devices access corporate data, and on what conditions?** This is a management decision, with social and legal implications.
- **How many compliance criteria?** Four well maintained beat thirty that generate exclusions.
- **Which resources require a compliant device this year?** Starting with administration and sensitive data makes the workstream achievable.
- **What happens when a device becomes non-compliant?** If the answer is "nothing until the session ends", the control is partial.

CHAPTER 19

Hybrid: synchronisation, federation, residues

When an on-premises directory is synchronised with the tenant, the two environments form a single security perimeter. Compromise of the on-premises domain becomes compromise of the tenant, and the reverse is sometimes true. This chapter covers the components that form the junction.

The components that matter

The synchronisation server holds a privileged account in both directories. It writes into the tenant, reads the entire on-premises directory, and depending on configuration holds the means to validate authentications. Its compromise is equivalent to the compromise of both environments.

Authentication agents deployed for direct password validation receive, by design, the credentials presented. They must be treated with the same level of protection.

Federation servers, where they remain, hold the keys allowing authentication assertions to be issued. A stolen key allows the authentication of any user to be forged, including an administrator, with no password compromised and no failed sign-in trace. This is the technique used in several major compromises.

Finally, some single sign-on mechanisms rely on a directory account whose secret has never been rotated since creation. That account is a known and documented objective; its secret must be rotated regularly.

The single rule

These components belong to the most sensitive administration tier, exactly like domain controllers. That means: administration from dedicated workstations, separate administrative accounts, no third-party applications installed, exported logging, priority patching, and restricted inbound connections.

The rule is simple to state and rarely applied, because these servers were often installed by the mail or directory team in a project context, with no particular classification. The first action of the workstream is therefore to identify them — their inventory is not always current — and to reclassify them.

Removing residual federation

Many organisations have migrated to direct cloud authentication while keeping the federation infrastructure "just in case". That infrastructure remains a complete attack path while providing no service.

Where federation is no longer needed, it must be removed, not switched off. A server that is powered down but still declared as a trusted issuer in the tenant configuration remains exploitable.

Where it remains necessary — smart card authentication requirements, a regulatory constraint, an older application — regular rotation of signing keys and monitoring of the federation configuration become mandatory controls.

The direction of risk

The usual path runs from on-premises to cloud. The reverse exists too: depending on configuration, a tenant administrator may hold means of acting on the on-premises directory, notably through write-back mechanisms or through the installed agents. That possibility must be checked explicitly, because it contradicts the common assumption that on-premises is protected from the cloud.

What to check first

Which on-premises accounts are synchronised and hold roles in the tenant: a privileged cloud account whose identity is managed on premises depends on the security of the on-premises directory. Tenant administration accounts should be cloud-only.

Which objects are excluded from synchronisation: privileged on-premises accounts and service accounts should not be synchronised without reason.

The state of the secrets and keys of the junction components, and how long since they were rotated.

What you need to decide

- **Are synchronisation and federation servers classified at the highest tier?** If not, every preceding chapter is capped by their security.
- **Can residual federation be removed this year?** If so, it is one of the best risk-to-effort ratios in the programme.
- **Are your tenant privileged accounts cloud-only?** A Global Administrator synchronised from the on-premises directory inherits on-premises risk.
- **How long since the junction components' secrets were rotated?** If the answer is "since installation", the action is immediate and inexpensive.

PART IV

Detect and respond

CHAPTER 20

Logging: what to keep, for how long, at what cost

Logging is the only workstream whose absence produces no symptom until the day it becomes indispensable. That is why it is systematically undersized: it costs every month and is used for a few hours once every three years — at precisely the moment the organisation cannot afford not to know.

The three families

Sign-in logs record every access attempt: identity, target application, protocol, origin, device, outcome, and the policies evaluated. It is the richest and the bulkiest source.

Directory audit logs record changes: account creation, role assignment, addition of a secret to an application, consent granted, modification of an access policy. This is the source that allows an attacker's actions after entry to be reconstructed.

Collaboration service activity logs record usage: mailbox access, file downloads, mail rule creation, external sharing. This is the source that answers the question deciding regulatory notification — which data was accessed.

These three families share neither the same native retention, nor the same volume, nor the same investigative value. Treating them as a single set leads either to overpaying or to losing what matters.

The reasoning on duration

The rule is simple to state: retention must exceed the realistic interval between compromise and detection, plus investigation time.

That interval is measured in months, not days, even in well-equipped organisations, and particularly for discreet compromises — token theft, application consent, application-level mail access. Native platform retention, by contrast, is measured in days or weeks for the directory and in months for collaboration.

The gap between the two defines exactly what you must export and keep elsewhere. There is no other reasoning to apply, and in almost every case it leads to a target of at least twelve months, often eighteen or twenty-four depending on the sector and applicable obligations.

Hot and cold

The classic budget error is keeping everything online and immediately queryable, at a cost that makes the arrangement unsustainable after two years.

The model that works distinguishes two tiers. A hot tier, thirty to ninety days, holding what feeds detections and the security team's daily work. A cold tier, up to the target duration, in low-cost archive, queryable with delay — a few hours of latency is acceptable in an investigation that will run for weeks.

The discipline lies in deciding what goes hot rather than sending everything there by default. Directory audit logs and privileged sign-ins deserve the hot tier; the detailed document activity of the entire user population rarely does.

What to keep first

If budget forces choices, the order of priority is as follows.

Directory audit logs, in full: low volume, maximum investigative value. They contain the traces of the attacker's persistence.

Sign-ins of privileged accounts and application identities, in full, with no sampling.

Sign-ins of the whole user population, with origin and device properties, which allow session theft to be characterised.

Mail and sharing activity, indispensable for establishing the scope of data accessed — this is the source that decides a regulatory notification, and its absence leads to notifying at maximum scope as a precaution.

Integrity

A point most organisations discover too late: an attacker holding privileges in the tenant can alter logging configuration, disable collection, or shorten retention. Logs that never leave the compromised environment are not evidence.

Export to a storage space under separate control, with write permissions distinct from tenant administration rights, is therefore a requirement, not a convenience. That separation is also what an insurer or a regulator will ask you for.

Cost

Cost is steered through three levers: the volume ingested into the hot tier, the retention period per tier, and the selection of log categories. Correct sizing always begins with a thirty-day measurement of actual volume by category — theoretical estimates are wrong by a factor of three in both directions.

A governance note: this budget belongs to security, not to operations. Handed to operations, it will be the first cut, because it has no visible effect on service delivery.

What you need to decide

- **What retention do you commit to, by log family?** Write it down; it is the only way to defend it in budget arbitration.
- **Do logs leave the tenant, to a space under separate control?** If not, your evidence depends on the environment you are auditing.
- **Can you answer "which data was accessed" today for a six-month-old incident?** If not, the consequence is regulatory before it is technical.
- **Who carries the logging budget?** If operations carries it, it will be cut in the first constrained year.

CHAPTER 21

Detections that matter, noise to eliminate

An organisation receiving two hundred alerts a day handles none of them. The quality of a detection capability is not measured by the number of rules enabled but by the number of alerts actually investigated and by the handling time. This chapter proposes a short baseline.

Start from the paths, not the catalogue

Platforms offer dozens of ready-made detections. Enabling them all produces an unmanageable flow and rapid desensitisation of the team.

The method consists of returning to the seven paths from Chapter 3 and asking, for each: what would we see if this happened? The answer gives a short list, ranked by risk, and it is defensible to management because it attaches to scenarios rather than to a product catalogue.

The useful baseline

Session theft. The same session or token used from two incompatible origins; a successful sign-in immediately followed by registration of an authentication method; the appearance of a mailbox rule that moves or deletes incoming messages. This last detection is among the highest-return of all: it is low-noise and almost always signals a genuine compromise.

Application consent. Consent granted to an application whose publisher is unverified; addition of a broad application permission; creation of a service principal outside a known workflow.

Post-compromise persistence. Addition of a secret or certificate to an existing application; creation of a privileged account; addition of a member to a role-assignable group; modification of a Conditional Access policy or addition of an exclusion.

Privileged accounts. Any sign-in by an emergency access account; activation of a sensitive role outside usual hours; a burst of failed sign-ins followed by a success on the same account.

Application identities. Service principal authentication from an unusual origin; use outside its normal activity window.

Hybrid path. Modification of the federation configuration; synchronisation account authentication from an unexpected server.

Around fifteen detections, correctly calibrated, cover the essentials. That is an achievable objective; two hundred rules are not.

The noise to eliminate

Three categories produce most false positives.

Poorly calibrated impossible-travel detections, which fire on corporate virtual private networks, outbound proxies and travelling staff. They require a learning phase and a list of known origins, failing which they should be turned off.

Service accounts and application identities, whose behaviour structurally differs from human behaviour. They must be covered by separate rules with their own thresholds, never by general rules.

Low-severity alerts from the default catalogue, which inform without requiring action. They should not reach a human queue; their place is in a weekly dashboard.

Triage

A detection with no recipient and no action deadline is not a detection. For each rule, three things must be written before it is enabled: who receives it, within what time it must be investigated, and what the first expected action is.

That last point is the most neglected. An alert about a privileged account compromise at three in the morning must come with an instruction the on-call person can execute — revoke sessions, disable the account, call this number — not with an invitation to analyse.

Verify that it works

An enabled rule is not an operational rule. Three checks, at least annually.

Deliberately generate the expected events in a controlled setting and verify that the alert fires, reaches the right recipient, and does so within the expected time. Most arrangements fail at this step, usually on the notification path.

Verify coverage using the technique grid from Chapter 6, and explicitly document the blind spots you accept.

Measure the real interval between the event and its investigation, not the theoretical one. That is the indicator to present in committee, the one from Chapter 26.

What you need to decide

- **How many alerts can your organisation genuinely investigate per day?** Size the capability on that capacity, not on the available catalogue.
- **Does every detection have a recipient and a written first action?** Without that, it produces noise and false assurance.
- **Have your detections been tested end to end?** A rule never triggered in real conditions has never proved it works.
- **Which blind spots do you accept?** Writing them down beats believing in complete coverage.

CHAPTER 22

Privileged account compromise: the hour-by-hour sequence

This chapter describes the conduct of the first seventy-two hours. It does not replace a formal response plan, but it provides the sequence and, above all, the order of actions — because the most frequent error is not inaction, it is the hasty action that destroys evidence or alerts the attacker before the scope is known.

The opening mistake

The natural reflex, on being told an account is compromised, is to reset the password and consider the incident handled.

In a modern identity environment, that action alone addresses almost nothing. It does not revoke sessions in progress, does not remove authentication methods added by the attacker, does not revoke application consents granted, does not remove secrets added to an application, nor mail rules created. It does, however, warn the attacker that they have been spotted.

Hour 0 to hour 1 — Confirm and preserve

Confirm the signal before acting: identify the suspicious sign-in, its origin, the target application, the method used. One alert in five is a false positive, and a response triggered in error has a real cost.

Preserve immediately: export sign-in and audit logs for the period concerned, before any change. This is the one action to take before everything else, because retention windows are running.

Form the team: an incident lead, an operator authorised to act on the tenant, a legal or compliance contact on standby. Open a timestamped incident log, maintained continuously — it will serve the regulatory notification and, potentially, the insurance claim.

Hour 1 to hour 4 — Contain

Order matters. Revoke the account's sessions and tokens, then disable the account or remove its roles depending on the strategy chosen. Revoking refresh tokens is essential: without it, the attacker retains access despite the password reset.

Then search for persistence, in this order of likelihood: authentication methods added or modified on the account; mailbox rules created; mailbox access delegations; recent application consents; secrets or certificates added to existing applications; accounts created or roles assigned since the presumed compromise; changes to Conditional Access policies and their exclusions.

This list must be walked in full, even when the first anomaly is found. An experienced attacker plants several persistence points, and finding the first creates a false sense of resolution.

Finally, decide a strategic question: contain immediately or observe. Observation allows the true extent to be measured but leaves the attacker active. That choice belongs to the incident lead and must be written down, with its time and its justification.

Hour 4 to hour 24 — Eradicate

Remove every identified persistence. Revoke suspicious application consents. Rotate the secrets of the applications concerned.

Widen the scope: was the compromised account used to reach other accounts? Look for sign-ins from the same origins, other accounts showing the same anomalies, and administrative actions performed since the compromise.

If a highly privileged account is involved, treat the whole tenant as potentially affected and start the rotation described in the next chapter.

Hour 24 to hour 72 — Establish impact

This is the phase that determines legal obligations. The question is no longer "how did they get in" but "what data did they reach".

It is handled with collaboration activity logs: mailboxes accessed, files downloaded, shares created, exports performed. The quality of the answer depends entirely on the decisions taken in Chapter 20. Without logs, the organisation is forced to notify at maximum scope, with everything that entails.

In parallel, start the legal analysis of notification obligations, whose deadlines are short and run from knowledge of the incident, not from its resolution.

Afterwards

Two deliverables, not to be confused. The incident report, factual and timestamped, for regulatory and contractual obligations. And the lessons-learned review, aimed at remediation, which must identify not only the flaw exploited but also the controls that should have detected or blocked it and did not.

The second must produce actions written into the roadmap, with an owner. A lessons-learned review with no actions is a compliance document, not an improvement.

What you need to decide

- **Who is authorised to revoke and disable, at three in the morning, without management approval?** If nobody is, your reaction time is that of a call chain.
- **Contain immediately or observe: who decides?** That choice must be assigned before the incident, not debated during it.
- **Do your logs allow the data accessed to be established?** That is what determines the scope of your regulatory notification.
- **Does the lessons-learned review produce tracked actions?** If not, the same incident will recur with another account.

CHAPTER 23

Rebuilding: revocation, rotation, identity forensics

After a compromise involving privileges, one question dominates all others: from what point can the tenant be trusted again? Answering it requires methodical revocation, rotation and verification.

Revoking for real

Revoking access in a token-based environment is neither instantaneous nor limited to the password.

Three distinct actions are required: revoke refresh tokens, which allow access to be renewed; force re-evaluation of sessions in progress, failing which a valid access token keeps working until it expires; and remove or reset registered authentication methods, because an attacker who has enrolled their own method retains a legitimate means of authentication after the password reset.

Continuous evaluation mechanisms sharply reduce the delay before revocation takes effect. Enabling them, decided in calm conditions, is what separates revocation effective in minutes from revocation effective in hours.

Rotation: the real scope

Rotation is almost always undersized. Following the compromise of a highly privileged account, the scope to handle includes, at minimum: secrets and certificates of applications that received recent changes; credentials of emergency access accounts; secrets of the synchronisation and federation components; federation signing keys where federation remains; and credentials of any service account that passed through a potentially compromised system.

That rotation must be planned, with the owners identified in Chapter 15. It is precisely in this situation that the absence of a non-human identity inventory becomes expensive: the organisation does not know what to rotate or what will break.

Identity forensics

It answers three questions, in order.

When did the access begin? The date of the first anomalous sign-in determines the period to examine. It frequently precedes detection by several weeks.

What did the attacker do? Directory audit logs provide the answer: roles assigned, applications modified, policies changed, accounts created. It is the densest and fastest source to exploit.

What did they reach? Collaboration service activity logs, the only source able to bound data exposure.

The limit of the exercise is always the same: retention. An investigation cannot go back beyond the available logs, and an organisation discovering that constraint during an incident can no longer do anything about it.

The persistence that gets forgotten

Six mechanisms routinely escape cleanup and deserve explicit checking: mailbox access delegations; organisation-level transport rules; consented applications and their secrets; eligible role assignments, which do not appear in a review of active roles; administrative units and scoped delegations; and guest accounts created during the compromise period.

To that list, in a hybrid environment, add verification of synchronised objects: a change made in the on-premises directory propagates to the tenant and will survive a cleanup performed on the cloud side alone.

Clean up or rebuild

Full reconstruction of a tenant is in practice unachievable for an operating organisation. The question is framed differently: it is the junction components and administration workstations that must be rebuilt, not the directory.

A synchronisation or federation server suspected of compromise is reinstalled; it is not cleaned. The same holds for an administration workstation. That rule is costly and it is the only one that allows trust to be declared restored afterwards.

Restoring trust

The return to normal must be declared explicitly, on written criteria: persistence verified and removed, rotations performed, sensitive components rebuilt, heightened monitoring maintained for a defined period, and no new indicator during that period.

That declaration, dated and signed by the incident lead, closes the episode. Without it, the organisation stays in an intermediate state where nobody knows whether the subject is handled.

What you need to decide

- **Is continuous session evaluation active?** It determines whether your revocation takes effect in minutes or in hours.
- **Do you know what would need rotating if an administrator were compromised?** If the list does not exist, it will be built under pressure and will be incomplete.
- **Do you accept rebuilding sensitive components rather than cleaning them?** That is a cost-versus-certainty trade-off, and it is prepared before the incident.
- **Who declares the return to normal, and on what criteria?** Without written criteria, that declaration will never be made.

PART V

Govern over time

CHAPTER 24

Identity governance: lifecycle, reviews, separation of duties

The previous four parts deal with a state: measure, remediate, detect. This one deals with movement. An organisation that does not govern its identities returns, eighteen months after a successful remediation programme, to an exposure level comparable to where it started. Access accumulates because nothing organises its removal.

The lifecycle

Three moments decide everything, and two of them are handled badly everywhere.

Joining is the best covered: there is almost always a process, because missing access blocks work and is reported immediately.

Changing role is the most destructive moment. The previous role's access is not removed, the new role's is added, and someone who has moved three times accumulates three perimeters. This is the main origin of the privilege accumulation seen in audits, and the corresponding control — removing the previous role's access within a defined period — almost never exists.

Leaving is handled for employees and forgotten for everyone else: contractors, interns, partner accounts, guests. These populations have no centralised exit process, since their departure does not pass through human resources.

The control measure is simple and rarely tracked: the interval between the effective leaving date and account deactivation, measured and presented each quarter.

Access reviews

An access review has one function: to remove what is no longer justified. It fails when it becomes a compliance formality.

Three causes of failure recur. The review covers too large a volume: a manager receiving two hundred lines approves the lot. It is assigned to the wrong person: security does not know whether an access is justified by the business, only the line manager or the application owner does. And it carries no consequence: unapproved access is not removed, and the next exercise receives the seriousness it deserves.

What works: targeted, frequent reviews rather than exhaustive annual ones. In priority order: privileged roles, guests, applications with broad permissions, and role-assignable groups. These four populations are small, and reviewing them quarterly produces more effect than a general annual review of all access.

One non-negotiable rule: automatic removal in the absence of a response. A review where no response means access is retained serves no purpose.

Separation of duties

The principle is old and transposes directly: whoever configures should not be whoever verifies, and whoever grants an access should not be its beneficiary.

Three concrete applications in an Entra ID environment.

The administrator who modifies access policies should not be the one reviewing the exclusion list. An application owner should not approve alone the permissions they request. And the team administering the tenant should not be the sole recipient of the logs recording its own actions — hence the export to a separately controlled space described in Chapter 20.

These separations are difficult in small organisations, where one person holds everything. The answer there is not separation but reinforced traceability and a periodic external review.

Access by need rather than by request

The next maturity step is to attach access to a business role or to a request catalogue rather than to individual assignments. Access then becomes a consequence of a situation — membership of a team, participation in a project — rather than the result of a one-off request that is never cancelled.

It is a long workstream, requiring reliable reference data about people and the organisation. It is not launched in the first year. But its absence explains why access reviews remain necessary: they compensate for the lack of automatic removal.

What you need to decide

- **Does a change of role trigger access removal?** If not, privilege accumulation is structural and no one-off remediation will address it.
- **Who offboards contractors and guests?** If the answer is "human resources", it is wrong: those populations do not appear there.
- **Does no response to a review mean retention or removal?** That decision determines whether your reviews serve any purpose.
- **How many targeted reviews per year, over which populations?** Four targeted quarterly reviews beat one general annual review.

CHAPTER 25

Compliance: NIS2, DORA, ISO 27001, GDPR

Compliance does not improve security, but it funds measures that do. That is its real function in a programme, and it is how it should be used: as a lever for decisions, not as an objective.

This chapter sets out what each of these texts concretely requires of an Entra ID and Microsoft 365 environment. Regimes evolve and national transposition varies; always confirm what applies to your organisation with your legal function.

GDPR

The European regulation does not mandate named technical measures, but it does impose security appropriate to the risk, notification of a breach within a short deadline, and data minimisation.

Three direct consequences for your environment. The ability to establish which personal data was accessed during an incident, failing which notification must cover the maximum scope — this is the reasoning of Chapter 20. Control over transfers and third-party access, which links this text to Chapter 14 on applications. And the retention policy, which serves both minimisation and reduction of the exposure surface.

NIS2

The European directive on network and information system security considerably widens the range of entities concerned and introduces two novelties that change the internal conversation.

The first is the explicit responsibility of management bodies, which must approve risk management measures and may be held liable for failures. This is the most effective argument available to you for obtaining a decision on measures with user impact: choosing to forgo them is no longer a technical decision.

The second is the multi-stage notification regime — an early warning within twenty-four hours, a substantiated notification within seventy-two hours, and a final report later. That timetable is incompatible with an organisation that cannot establish, within three days, what was accessed. It directly validates Chapters 20 and 22.

The directive also imposes requirements on supply chain security, which includes your providers holding access to your tenant, and on multi-factor authentication.

DORA

The regulation on digital operational resilience for the financial sector is more prescriptive and directly applicable to the entities concerned.

Four requirements touch identity. Technology risk management, explicitly including access and privilege management. The register of information on technology service providers, which must list your critical suppliers — your mail and identity provider is one

of them. Major incident reporting on a strict timetable. And resilience testing, based on realistic scenarios.

For a Microsoft 365 environment, that translates into concrete questions: what happens if the identity provider is unavailable, do you have an exit strategy, and do your emergency access accounts work. Chapter 11 answers part of this, but single-provider dependency is a matter for executive management.

ISO 27001

The standard mandates not a configuration but a management system and a set of measures, several of which bear directly on identity: identity management, authentication information, access rights, privileged access management, secure authentication, logging and monitoring.

Its practical value in a programme is twofold. It provides a recognised frame for structuring the governance described in Chapter 24. And its continual improvement logic, with management review, offers the simplest vehicle for institutionalising the steering of Chapter 26 without creating another committee.

Insurers

A fifth framework has established itself without being a legal text: cyber insurers' questionnaires. They now almost systematically require multi-factor authentication on remote access and privileged accounts, offline backup, privileged access management, and logging.

Their value lies in their immediate effect: an inaccurate declaration can ground a refusal of cover. It is often the argument that obtains a budget refused for two years through the risk route.

How to use all this

The mapping between your measures and the applicable requirements should be built once, then maintained. It serves three purposes: justifying a budget, answering a customer or insurer questionnaire without remobilising the teams, and avoiding handling the same requirement twice under two different names.

What not to do: structure the security programme around the texts. You would obtain a compliant and incoherent programme. Structure it around attack paths, then map the measures to the requirements.

What you need to decide

- **Which regimes actually apply to your organisation?** The answer belongs to legal, and it conditions the whole trajectory.
- **Does your executive management know it carries personal responsibility?** It is the most powerful lever for decisions, and it is rarely stated.
- **Is the measure-to-requirement mapping maintained?** Without it, every questionnaire remobilises the teams for a week.
- **Are your declarations to insurers accurate?** An inaccuracy discovered at claim time voids the cover.

CHAPTER 26

Steering: metrics and continuous control

An annual audit describes the state of the tenant on one day a year. On the other three hundred and sixty-four, the configuration drifts: exclusions added, applications consented, accounts created, policies modified. Steering means reducing that blind spot.

The principle

Continuous control does not mean monitoring more; it means periodically recomputing a small number of measures and tracking the trend. The relevant question is not "are we compliant" but "has the drift been positive or negative since the last point".

That implies a methodological requirement: measures must be computed the same way every time. An indicator whose method changes is no longer an indicator. This is why Chapter 7 insists on the reproducibility of extractions.

The indicators worth tracking

A dozen suffice, and all of them must come from an extraction, never from a declaration.

Number of permanent assignments of the most sensitive roles. Proportion of administrators holding a phishing-resistant method. Number of active exclusions in access policies, and number whose end date has passed. Proportion of successful sign-ins covered by a strong method. Number of applications with broad mail permissions. Number of application secrets expiring within ninety days. Number of guests with no sign-in for ninety days. Average interval between a person's effective departure and account deactivation. Median interval between a security alert and its investigation. Effective log retention period.

These ten measures can be computed automatically and, together, tell the real state of governance.

What not to track

The platform's native score as a primary indicator, for the reasons in Chapter 6. The number of alerts handled, which rewards noise. The number of compliant controls against a framework, which encourages closing easy gaps rather than important ones. And the number of completed roadmap actions, which measures activity rather than risk.

A badly chosen indicator does not merely inform poorly: it steers the team's work towards what it measures. That is why the choice of indicators is a management decision.

Frequency and audience

Monthly for the security team, across all measures, with analysis of variation.

Quarterly for management, on three to five measures only, with an interpretation. Executive management does not need ten indicators; it needs to know whether the trajectory is holding and what is slowing it.

Annually for the full review, which replays the audit and verifies that the automated measures have left no blind spot.

Drift control

Three drifts are so frequent that they deserve dedicated monitoring, independently of the indicators.

The addition of an exclusion to an access policy. The creation of a privileged account or the assignment of a sensitive role. Consent to a new application with broad permissions.

These three events should produce a notification to the security leader, not as an incident alert but as governance information. They are few in a healthy tenant, and their volume is itself an indicator.

What you need to decide

- **Which three indicators do you present to your management?** Choose them yourself, or the tool will choose them for you.
- **Are your measures computed automatically?** An indicator costing two days of work a month will not be maintained.
- **Who is notified when an exclusion or a sensitive role is added?** It is the highest-return drift control in this chapter.
- **How often is the full recomputation performed?** Annually is the minimum; quarterly is the target.

CHAPTER 27

Budget and twelve-month roadmap

This chapter proposes a one-year trajectory from a typical starting state: strong authentication deployed but incomplete, Conditional Access present but accumulated, applications ungoverned, logging at default values.

What a programme actually costs

Four cost lines, only one of which is visible.

Licensing, where the coverage gap justifies it. It is the most debated line and rarely the most decisive: most of the measures in Chapters 11 to 19 apply on a common licence base.

Tooling: logging and retention, possibly backup of collaboration data, possibly an audit and continuous control tool. This is a recurring cost, to be defended as such.

External services, for the initial audit and for workstreams requiring scarce, one-off expertise.

Internal time, which is the heaviest line and the one that appears in no budget. A phishing-resistant authentication workstream across several thousand people mobilises the service desk, internal communications and business relays. Costing it explicitly prevents failure through underestimation.

The three waves

Days 1 to 30 — no user impact. Close unrestricted application consent and put the request workflow in place. Build and publish the consolidated exclusion list. Create and test emergency access accounts. Enable export of audit logs to a separately controlled space. Establish the inventory of non-human identities and of applications with broad permissions.

These are low-cost, high-return measures, and they establish the programme's credibility with both management and the teams.

Days 31 to 90 — small populations. Reduce permanent assignments of sensitive roles and introduce time-bound elevation. Deploy a phishing-resistant method for administrators and exposed populations. Reinforce the service desk reset procedure. Rationalise the Conditional Access policy set and introduce the default-block policy. Put the fifteen detections from Chapter 21 in place with their handling workflow.

Days 91 to 365 — structural workstreams. Compliant-device requirement, by resource then by population. Reclassification and hardening of the hybrid junction components, removal of residual federation. Identity governance: lifecycle, targeted quarterly reviews. Extension of phishing-resistant authentication to the whole organisation. Dedicated administration workstations.

What blocks, and how to unblock it

The blocker is almost never technical. Three obstacles recur.

The absence of an owner for the identity configuration. It is addressed by a governance decision, not by a project.

Refusal of user impact. It is addressed by an explicit decision at the right level, supported by the scenarios in Chapter 3 and, where applicable, by the management responsibility discussed in Chapter 25.

Competition with business projects. It is addressed by sequencing: the first wave consumes no project capacity, and delivering it quickly lends credibility to the request for capacity thereafter.

How to defend the budget

Three elements, in this order. A costed scenario: what the compromise of a privileged account would cost, in downtime, notification, insurance and reputation. A comparison of the two trajectories — at constant licensing and at target licensing. And a commitment to measurement: the three indicators you will present quarterly to demonstrate effect.

A security budget obtained without a commitment to measurement will not be renewed.

What you need to decide

- **What is your first wave, and its end date?** Thirty days is a keepable commitment and it changes how the programme is perceived.
- **Is internal time costed in your budget?** If it is not, the third wave will not be executed.
- **Which indicators do you commit to?** The next budget depends on them more than on the content of the plan.
- **What are you choosing not to do this year?** A plan with no written renunciation is a plan that will drift.

CHAPTER 28

The ten most frequent mistakes seen in the field

These ten findings recur from one organisation to the next, regardless of size and sector. They form a quick review grid for your own programme.

1. Believing that strong authentication settles the subject. It addresses password theft, not session theft. As long as the question "what happens if a valid token is used from an unknown device" goes unanswered, most of the risk remains.

2. Leaving application consent open. A one-day measure, a major effect, almost always missing. It is the best effort-to-benefit ratio in the whole ecosystem.

3. Confusing configuration with application. A policy in report-only mode for a year, an empty target group, exclusions covering half the organisation: the rule exists and protects nobody. Verification is done on the logs, never on the configuration screen.

4. Not maintaining the exclusion list. Exclusions are added one at a time, each justified, and nobody holds the overall view. It is the most revealing document of an audit, and it exists in no portal.

5. Ignoring non-human identities. They often outnumber human identities, escape strong authentication, and their secrets circulate. An inventory with a named owner is the prerequisite, and it is missing almost everywhere.

6. Administering from the productivity workstation. Every measure protecting privileged accounts is capped by the security of the machine used to administer.

7. Sizing logging on default values. Native retention is shorter than the real detection interval. The organisation discovers the gap during the incident, at precisely the moment it can no longer be closed.

8. Treating the IT service desk as out of scope. An authentication method reset obtained by telephone bypasses the entire technical arrangement.

9. Forgetting the hybrid path. Synchronisation and federation servers are tier-zero systems. Installed in project mode and rarely reclassified, they cancel the benefit of everything else.

10. Producing a report with no decisions page. An accurate audit that asks for nothing obtains nothing. A list of gaps is not a plan, and a plan without explicit decisions is not executed.

The common thread

Nine of these ten are not technical errors. They are governance failures: no owner, no end date, no inventory, no written decision. That is consistent with the finding in Chapter 3 — none of the observed attack paths rests on a software vulnerability.

The consequence for a security leader is encouraging: most of the journey does not depend on a purchasing budget. It depends on decisions you are in a position to take, and on the discipline to hold them over time.

APPENDICES

Implementation tools

APPENDIX A

Condensed audit checklist

Each line is verified by a dated extraction. The implicit column to fill in against each is the evidence: source, date, method.

Privilege

- Number of permanent and eligible assignments of each sensitive role, with age
- Privileged accounts holding a mailbox or a productivity licence
- Existence, configuration and last test date of emergency access accounts
- Role-assignable groups, and who can modify their membership
- Administrator scope: global or restricted by administrative unit
- Privileged accounts whose identity is managed in the on-premises directory

Authentication

- Distribution of registered authentication methods, and of methods actually used over thirty days
- Successful sign-ins without a strong method, by application and by population
- Phishing-resistant methods on sensitive roles
- Residual use of legacy protocols, by account and by application
- Which strong authentication mechanism is actually in force: per-user, security defaults, or access policy
- Service desk reset procedure and traceability of resets

Conditional Access

- List of policies, state, last modification date, author
- Policies in report-only mode for more than three months
- Consolidated exclusion list, by policy, with owner and end date
- Existence of a default-block policy
- Specific handling of non-human identities

Applications and non-human identities

- User consent settings and existence of a request workflow
- Applications holding broad application permissions, with owner and justification
- Applications with unverified publishers, and applications with no recent activity
- Inventory of service principals and of human service accounts
- Validity periods of secrets and certificates, expired secrets renewed without review

External collaboration

- Guest population, including guests with no sign-in for ninety days
- Invitation settings and cross-tenant access settings
- Anonymous link sharing: permitted, lifetime, scope
- Shared channels and multi-tenant arrangements

Data and devices

- Classification labels in use and share of unlabelled content
- Loss prevention rules: mode, trigger rate, false-positive rate
- Automatic mail forwarding rules to external addresses
- Managed estate: registered, joined and compliant devices, and compliance criteria applied
- Resources requiring a compliant device

Logging and hybrid

- Effective retention by log family, and export destination
- Distinct access control over the retention space
- Inventory and classification of synchronisation and federation components
- Residual federation declared in the tenant configuration
- Age of the secrets and keys of the junction components

APPENDIX B

Licence-to-control matrix

This appendix gives the structure of the reasoning, not the commercial state of the moment: offerings, names and thresholds change. Always confirm what applies to your contract before committing to a budget decision.

Base, without premium licensing

Blocking legacy protocols through service configuration. Application consent settings. Invitation and external sharing management. Security defaults for small organisations. Logging at native retention. Emergency access accounts. Separation of administrative accounts. Dedicated administration workstations.

In other words: a significant part of Chapters 11, 12, 14 and 16 depends on no additional licence.

First premium directory tier

Conditional Access, including compliant-device and phishing-resistant method requirements. Write-back of certain operations in hybrid environments. Extended retention of directory logs.

This is the structuring threshold: without Conditional Access, the access policy of Chapter 13 does not exist.

Second premium directory tier

Time-bound privilege elevation with approval and notification. Access reviews. Risk-based protection and its associated policies.

This threshold conditions Chapters 11 and 24 in their automated form. Without it, the same objectives are pursued manually, at a higher running cost and with lower coverage.

Identity governance

Entitlement management, lifecycle workflows, advanced reviews. Corresponds to Chapter 24 in its industrialised form.

Workload identities

Applying access conditions to service principals. Corresponds to the last point of Chapter 15, and there is no manual equivalent.

Upper tier of the collaboration suites

Automatic classification, extended loss prevention, advanced detection and investigation capabilities, extended audit retention. Corresponds to Chapters 17, 20 and 21.

How to present the decision

For each threshold, three elements: the measures it unlocks, the risk they cover, and the manual alternative with its annual running cost. It is the only presentation that allows executive management to decide knowingly rather than compare two prices.

APPENDIX C

Report template

A structure to reuse as it stands. The lengths given are maxima.

Page 1 — Verdict

One sentence concluding on the level of exposure, expressed in consequence rather than compliance. Three dated audit facts supporting it. One deadline: the date by which the situation will have changed if the decisions requested are taken.

Pages 2 to 3 — Scenarios

Three scenarios at most, each on a third of a page: the attack sequence, the audit facts that make it possible in your environment, the expected impact, and the control that would cut it.

Pages 4 to 5 — Plan in waves

Thirty days, ninety days, one hundred and eighty days and beyond. For each action: a one-line title, a candidate owner, effort estimated in person-days, and the risk covered. No dates to the day, no detailed dependencies.

Page 6 — Decisions requested

Three to five decisions phrased to receive a yes or a no, each with the default option that applies in the absence of a decision. Example phrasing: accept the user impact of the compliant-device requirement on this population; fund this licence tier to cover this risk; appoint this owner for the identity configuration; formally accept the residual risk on this perimeter until this date.

Technical appendix

Detailed findings by domain, in the order of the checklist in Appendix A. For each finding: the statement, the evidence with source and date, the criticality and its rationale, the recommendation, and the effort. No size limit.

What does not appear in the report

The platform score on the cover. Portal screenshots. Explanations of how the products work. The exhaustive list of gaps in the body. Uncosted recommendations.

APPENDIX D

Glossary

Access token. Artefact issued after authentication and presented to the application to obtain access.

Administrative unit. Object allowing an administrator's scope to be restricted to a given population.

Application permission. Permission exercised with no user present, at the scope granted, potentially across the whole tenant.

Adversary-in-the-middle phishing. Technique intercepting a complete session by relaying the legitimate authentication page in real time, allowing the session cookie to be stolen despite strong authentication.

Compliant device. Managed device satisfying the compliance rules defined by the organisation, and which can be required as a condition of access.

Conditional Access. Mechanism evaluating, at every access attempt, a set of conditions — identity, device, application, origin, risk — in order to grant, deny or strengthen access.

Consent. Act by which a user or an administrator grants an application permissions over data or functions.

Continuous access evaluation. Mechanism re-evaluating a session in progress when a significant event occurs, without waiting for the token to expire.

Delegated permission. Permission exercised on behalf of a signed-in user, bounded by that user's rights.

Emergency access accounts. Privileged accounts reserved for lockout situations, excluded from the policies liable to prevent them working, and closely monitored.

Federation. Delegation of authentication to a third-party identity provider, which issues assertions attesting to the user's identity.

Managed identity. Application identity whose secret is handled by the platform and never exposed.

Phishing-resistant method. Authentication method whose proof is bound to the site's origin and therefore cannot be replayed by a relay.

Refresh token. Artefact allowing new access tokens to be obtained without re-authentication; revoking it is necessary to genuinely cut off access.

Service principal. Instance of an application in a tenant, with its permissions and its means of authentication.

Tenant. Directory instance dedicated to an organisation, and the security unit of the ecosystem.

Time-bound privilege elevation. Role assignment activated on demand for a limited period, rather than held permanently.

Zero Trust. Architecture model resting on explicit verification, least privilege and the assumption of breach.

Summary

A Microsoft 365 tenant carries several hundred security settings spread across half a dozen consoles. No security leader can know them all — and none needs to.

This guide answers the only question that matters in the boardroom: where do we actually stand, what exposes us, and in what order should we fix it?

Built like an audit engagement, it takes the reader from understanding the identity attack surface to governing it over time, by way of a repeatable assessment method, a remediation hierarchy based on the real return of each control, and a detection capability sized for investigation. Every chapter closes with the trade-offs that belong to the security leader and that nobody else will make.

Frédéric PEREZ is an identity and access management architect and an independent consultant in Microsoft environment security. He runs identity audit and governance engagements in France, Belgium and Luxembourg.