

A Read-Only Assessment Model for Microsoft 365 Tenants

Most Microsoft 365 assessments answer the wrong question. They ask whether a tenant is compliant, when the prior question — *what can we actually observe, and what could we not see at all* — remains unanswered. This paper describes an assessment model built on the opposite premise: collect read-only evidence, record every failure as loudly as every success, and hand a qualified engineer a defensible artifact rather than a score.

Field	Detail
Subject	M365 Tenant Readiness — an open, read-only tenant discovery and assessment tool
Repository	github.com/Testingdevops22/Tenant-Assessment
Tool version	0.1.1 · snapshot schema 1.0
Author	Priyanshu Upadhyay, Lead Technical Advisor
Audience	CIO, CISO, Microsoft 365 platform owners and security architects
Classification	Public — no tenant data included

This paper describes tooling and method. It is not an audit opinion, a compliance certification, or legal advice on data residency obligations.

Contents

- 1 Executive summary
- 2 The assessment gap
- 3 Design principles
- 4 Architecture
- 5 What the assessment evaluates
- 6 What an assessment produces
- 7 Security and privacy posture
- 8 Operating model
- 9 Interpretation boundaries
- 10 Engineering roadmap
- 11 Conclusion
- A Microsoft Graph permission reference
- B Complete finding catalogue
- C Output schema reference

1 Executive summary

Every Microsoft 365 migration, tenant consolidation, data-residency programme and security uplift begins with the same activity: someone has to establish what the tenant currently looks like. That activity is usually performed badly. It is performed with a mixture of portal screenshots, ad-hoc PowerShell, spreadsheets assembled by hand, and commercial scanners whose scoring logic is opaque and whose agents want write permissions the customer would rather not grant.

The result is an assessment nobody can defend. Six weeks into delivery, when a migration wave stalls, the question "did we know about this before we started?" has no answer, because the original assessment recorded conclusions rather than evidence, and recorded nothing at all about what it had failed to read.

M365 Tenant Readiness is an open, read-only PowerShell assessment that takes the opposite position. It authenticates with delegated read scopes, issues GET requests and `Get-*` cmdlets exclusively, and produces four things: a self-contained HTML report, a machine-readable JSON snapshot of the raw evidence, a findings register, and two operational CSV extracts. Its most important architectural property is that a collection which failed is recorded as **failed**, with the endpoint and the read scope it needed — never silently as an empty result.

WHAT THIS CHANGES

- **The tenant is never modified.** Read-only is structural, not a policy promise — GET is the only HTTP verb in the codebase, and the only service cmdlets called are `Get-OrganizationConfig`, `Get-Mailbox`, `Get-SPOTenant` and `Get-SPOGeoStorageQuota`.
- **Unknown is distinguishable from clean.** A per-collection manifest travels with the data, recording status, endpoint, required permissions, record count, duration and error text for all seventeen possible collections.
- **Findings are evidence, not verdicts.** Twenty-one rules across seven domains produce prioritised prompts for human review across five severity bands — including a *Review* band that exists specifically to refuse a false judgement.
- **The evidence outlives the engagement.** The JSON snapshot can be re-evaluated and re-rendered months later with no tenant access at all, which makes assessments diffable and makes the rules engine deterministically testable.
- **Identifiers can be redacted without breaking the analysis.** Redaction hashes user and mailbox identifiers from the same source value, so cross-dataset correlation survives circulation to a wider audience.

SCOPE The tool does not change the tenant, certify compliance, interpret Microsoft Product Terms, or prove the physical data location of every workload. Findings are evidence-led prompts for a qualified administrator to validate. That constraint is the design, not a disclaimer bolted on afterwards.

2 The assessment gap

2.1 Three failure modes

Tenant assessments fail in three recognisable ways, and each one produces a document that looks complete while being materially wrong.

Silent partial collection

An operator runs a discovery script under an account that lacks `RoleManagement.Read.Directory`. The role-assignment query returns a 403. The script catches the exception, moves on, and the report renders a directory-roles section containing zero rows. A reader cannot distinguish that from a tenant with no privileged role assignments — which is impossible, and therefore should have been obvious, but the report gave no signal at all. This is the single most common defect in home-grown assessment tooling, and it is dangerous precisely because the output looks clean.

Verdicts without evidence

Commercial scanners commonly emit a numeric posture score or a red/amber/green grade. The grade is derived from weightings the customer cannot inspect, applied to evidence the customer never sees. When a finding is challenged — and it will be, because assessments are read by the people whose configuration is being criticised — there is nothing underneath the grade to argue from. The assessment loses the argument and the programme loses momentum.

Write-capable tooling for a read-only question

Discovery is a read activity, but a great deal of tooling asks for write scopes anyway, either because it shares a codebase with a remediation product or because nobody trimmed the manifest. Security teams are right to refuse. The consequence is that assessment gets deferred until permissions are negotiated, or gets performed informally by whoever already holds Global Administrator — which is the least auditable path available.

2.2 Why this matters most in Multi-Geo and migration work

The cost of a weak assessment scales with the irreversibility of what follows it. Preferred Data Location is the clearest example. PDL for a synchronised identity must be corrected in the authoritative on-premises directory, not in the cloud; a mailbox that has already relocated to the wrong geography is moved back only by another service-controlled relocation with its own queue and its own duration. A wrong PDL discovered during discovery is a spreadsheet edit. The same error discovered during wave three is a schedule slip, a data-residency conversation with a client, and occasionally a contractual problem.

The same asymmetry applies to licensing. An assignment error surfaced before a migration wave is a ticket; the same error surfaced when two hundred users cannot access a relocated mailbox is an incident. Assessment quality is not a documentation concern — it is the cheapest risk control available in the entire programme.

FIRST PRINCIPLE A missing or failed collection means **unknown** — not compliant, and not empty. Every design decision in this tool descends from that one sentence.

3 Design principles

Five principles govern the implementation. They are stated here because they explain why the tool refuses to do things that would make its output look more impressive.

3.1 Read-only by construction

The tool cannot modify a tenant, and this is enforced structurally rather than by convention. All Microsoft Graph traffic passes through a single function, `Invoke-MtrGraphRequestWithRetry`, in which the HTTP method is the hard-coded literal `GET`. There is no code path that constructs any other verb. Exchange and SharePoint evidence is limited to four `Get-*` cmdlets. Local writes are confined to the operator-supplied output directory. The tool also never installs PowerShell modules; a missing dependency produces a thrown error naming the module and the install command, leaving the decision with the operator.

3.2 Failed is not empty

Each collection is wrapped so that it cannot throw. It returns a status object either way: on success, `Status = Complete` with a record count; on failure, `Status = Failed` with the exception message, a zero count, and — critically — the endpoint URI and the read scopes that endpoint required. Elapsed duration is recorded in both cases. These rows accumulate into a collection manifest that is persisted in the snapshot, rendered as its own report chapter, and promoted into a finding (COL-001) that names every collection that failed.

The practical effect is that an operator reading the report can tell the difference between "this tenant has no Conditional Access policies" and "we could not read Conditional Access policies because the account was missing `Policy.Read.All`" — and can fix the second case and re-run rather than acting on a false negative.

3.3 Evidence over verdicts

There is no aggregate score, no letter grade, no percentage and no risk index anywhere in the tool. Severity is a fixed label chosen deliberately for each rule, never computed from a weighting the reader cannot see, and never inflated by the number of affected objects. Five severity bands are used — Critical, High, Medium, Review and Info — where **Review** is a first-class band reserved for evidence that genuinely requires human judgement rather than a weak finding.

Six of the twenty-one rules resolve to Review, and they are the rules where a confident verdict would be dishonest: an apparently inactive licensed account may be a shared mailbox, a service identity, a new joiner or someone on extended leave. The tool says so in the recommendation text rather than counting it as waste.

3.4 The snapshot as the architectural seam

Collection and analysis are fully separated by a serialised JSON snapshot. Running the tool with `-SnapshotPath` re-evaluates every rule and re-renders the full report with no authentication, no network access and no tenant contact whatsoever. This one decision buys three properties that are hard to retrofit: the rules engine and renderer become deterministically testable against a fixture; two assessments of the same tenant can be diffed to show configuration drift; and a client can circulate evidence for review long after the collection window has closed.

3.5 Local artifacts, no telemetry

Everything is produced locally in a directory the operator names. The HTML report is a single self-contained file with inline CSS, no external stylesheets, no web fonts, no images and — deliberately — no JavaScript at all. It renders identically offline, survives email and SharePoint preview, prints cleanly to PDF, and is safe in locked-down browser environments. Nothing is transmitted anywhere.

4 Architecture

The implementation is a small PowerShell 7.2+ module — roughly 1,400 lines including tests and fixtures — deliberately kept legible enough that a client security team can read the whole thing before approving a run. That reviewability is a feature: an assessment tool that cannot be audited is asking the customer for exactly the trust the assessment is supposed to establish.

4.1 Component structure

A thin entry script, `Invoke-M365TenantReadiness.ps1`, wraps a module that dot-sources nine numbered part files in deterministic name order and exports precisely three public functions — `Invoke-M365TenantReadiness`, `Get-M365ReadinessFindings` and `New-M365ReadinessReport`. Every other function is module-private.

Part	Responsibility
01-functions	Primitives — null-safe property access, type coercion, the redaction hash, Graph retry, pagination, the generic collector
02-functions	Connection handling and the non-Graph collectors for Exchange Online and SharePoint Online
03-functions	Collection orchestration — the scope list, the request table, profile extras, snapshot assembly
04-functions	Finding constructor and rule-evaluation helpers
05-functions	The rules engine — every finding rule in one auditable file
06-functions	Redaction and the generic HTML table builder
07-functions	The HTML report renderer
08-functions	CSV row projection and writer

Part	Responsibility
09-functions	Public orchestrator — branch, redact, evaluate, emit, return

Table 1 — Module composition. Load order is lexical, which is why the files are zero-padded.

4.2 Execution pipeline

A run proceeds through eight stages. The branch at stage one is what makes offline re-analysis possible; the redaction step at stage five is placed deliberately before rule evaluation so that every downstream artifact is consistently redacted.

#	Stage	Behaviour
1	Branch	Live collection, or load an existing snapshot from disk
2	Connect	Interactive, device-code, certificate, or reuse an existing session
3	Collect	Iterate the request table; append a manifest row per collection
4	Assemble	Build the ordered snapshot object with schema and tool version
5	Redact	Optional — applied before rules, so all five artifacts agree
6	Evaluate	Run the rules engine against the snapshot
7	Emit	Write the HTML report, JSON snapshot and three CSV files
8	Return	Resolved artifact paths plus per-severity finding counts

4.3 Collection layer

Thirteen core collections run in both profiles. The **Full** profile adds two optional Graph collections — user sign-in activity and Microsoft Secure Score — whose availability depends on tenant licensing, directory role and consent. Two further opt-in collectors gather Exchange Online and SharePoint Online evidence through their own PowerShell modules and role models. The manifest therefore contains at most seventeen rows.

Page sizes are set explicitly — `$top=999` for users, groups, applications and service principals; `$top=500` for sign-in activity — and pagination follows `@odata.nextLink` until it is absent. Single-entity responses without a `value` property are normalised into one-element arrays, so every leaf in the snapshot has a uniform shape.

4.4 Reliability engineering

Throttling is the normal condition when reading a large tenant, not an exception. Requests retry up to four times. Retryable conditions are recognised from the response error — HTTP 429, 503 and 504 and their standard reason phrases — and back off on a doubling schedule of two, four and eight seconds, capped at eight. Retry is applied per page rather than per collection, so a large paginated result set does not restart from the beginning after a single throttled page.

Authorisation failures are explicitly *not* retried. A 403 propagates immediately, becomes a Failed manifest row carrying the endpoint and required scope, and the run continues. This is the correct behaviour: retrying a permission error wastes minutes and changes nothing, whereas recording it precisely lets the operator fix consent and re-run.

4.5 Rules engine

All rules live in a single file so that the complete assessment logic can be reviewed in one sitting. Each rule constructs a finding with a stable identifier, a severity from a validated set, an area, a human-readable title, an evidence string derived from the observed data, a recommended action, an affected-object count and one or

more control themes. Findings are sorted by severity rank, then area, then identifier — a deterministic order, so two runs of the same snapshot produce byte-comparable output.

4.6 Rendering

The report is generated as a single interpolated HTML document with an inline stylesheet. Every tenant-supplied string — policy names, display names, error text — is HTML-encoded before interpolation, so a maliciously named object cannot inject markup into the report. The document contains nine chapters: a tenant header, six metric tiles, executive findings, licensing audit, Multi-Geo and PDL evidence, Conditional Access evidence, collection status, interpretation boundaries, and a provenance footer.

5 What the assessment evaluates

Twenty-one rules span seven domains. Two of them are dynamic families that emit one finding per affected licence SKU; the remainder have fixed identifiers. The full catalogue with trigger conditions appears in Appendix B — this section explains the reasoning behind each domain.

5.1 Licensing and cost

Licensing rules are the tool's only source of **Critical** findings, and only in one circumstance: a SKU whose consumed units exceed its enabled units. Over-allocation is treated as critical because it is an active service-continuity risk rather than a hygiene issue — assignment will begin failing, and the failure surfaces to end users. A separate Medium rule warns when remaining capacity falls to the greater of two seats or five percent of the enabled pool, which is early enough to procure without drama.

Three further rules address assignment quality: licence assignment errors, licences still attached to disabled accounts, and — a frequent and expensive migration blocker — licensed users with no `UsageLocation`. The last of these is High severity because it silently prevents service assignment and is trivially fixable before a wave rather than during one.

5.2 Identity controls

Conditional Access checks are deliberately conservative, and the tool says so in the report itself. It detects three structural gaps: no enabled policies at all, no enabled policy that applies multi-factor authentication broadly, and no enabled policy that blocks legacy authentication. A policy in report-only mode is correctly not counted as enabled — a distinction that is verified by the test suite.

BOUNDARY Policy presence is not equivalent to effective coverage. These rules cannot prove that exclusions are safe, that workload identities are covered, or that authentication strengths are appropriate. They exist to catch structural absence, and a human must validate everything else.

5.3 Privileged access

Two rules address Global Administrator distribution from opposite directions: fewer than two active assignments is a resilience risk — a single point of administrative failure — while more than five is flagged for least-privilege review. Both count only standing, active directory role assignments. The tool states inside the finding itself that Privileged Identity Management eligible assignments are not counted, which means a mature tenant running PIM correctly with zero standing administrators will produce this finding. That is a known limitation, disclosed in the output rather than hidden.

5.4 Application credentials

Application registrations and enterprise applications are examined together across both password and certificate credentials. Credentials already past their expiry are High; credentials expiring within a configurable window — ninety days by default — are Medium. Expired application credentials are a recurring cause of silent integration failure, and they are invisible in every portal view that does not specifically look for them.

5.5 Multi-Geo and Preferred Data Location

This domain reflects the tool's origin in cross-tenant and data-residency delivery, and it is where the evidence-first philosophy earns its keep. A blank PDL is normally correct — it means tenant-default placement. Reporting it as a defect in a single-geography tenant would generate hundreds of false findings. The tool therefore gates the entire domain behind a Multi-Geo detection test: more than one allowed Exchange mailbox region, or at least one user carrying a PDL value. When neither holds, a single informational finding records that no Multi-Geo indicators were found, and nothing else fires.

When Multi-Geo is detected, five rules apply: licensed users missing a PDL, PDL values outside the tenant's allowed Exchange regions, mailbox region disagreeing with PDL, mailbox database geography disagreeing with mailbox region, and Microsoft 365 groups with no explicit PDL. Three of the five are Review severity, because disagreement between PDL and mailbox region is frequently a relocation still in progress rather than an error.

5.6 Collection quality

A single rule promotes collection failure into the findings register, naming every collection that did not complete. It is severity Review because the correct response is not remediation of the tenant but re-collection with adequate permissions. Reading this finding first, before interpreting anything else, is the recommended operating discipline.

SEVERITY DISTRIBUTION ACROSS THE CATALOGUE

Severity	Rules	Meaning in this tool
Critical	1 family	Active service-continuity risk — licence pool exhausted
High	8	Structural gap or defect that should be corrected before delivery
Medium	5	Hygiene, cost or forward-looking risk
Review	6	Evidence requires human judgement; a verdict would be dishonest
Info	1	Context — recorded so its absence is not mistaken for a gap

6 What an assessment produces

A run writes five artifacts into the output directory. They are intended for different readers, and the separation is deliberate: the audience that needs a narrative should not be handed a CSV, and the engineer who has to fix something should not have to retype data out of a PDF.

Artifact	Reader	Purpose
tenant-readiness-report.html	Stakeholders, leadership	Self-contained narrative report; print to PDF for circulation
tenant-snapshot.json	The tool itself, future runs	Complete raw evidence with collection manifest; enables offline re-analysis and drift diffing
findings.csv	Delivery and risk owners	Review and remediation register with identifier, severity, evidence and recommendation
license-assignments.csv	Licensing and IT operations	One row per user per assignment state — source, state, error, disabled plan count
pdl-users.csv	Migration and residency engineers	One row per user — identity source, PDL, mailbox region, database geo, placement status

Table 2 — Output artifacts. Column dictionaries appear in Appendix C.

6.1 The report

The HTML report opens with the tenant name and six headline metrics — users collected, licensed users, groups, subscribed SKUs, users carrying a PDL, and failed collections. Placing failed collections in the headline metrics rather than in a footnote is a deliberate choice: it is the number that determines how much of the rest of the report can be trusted.

6.2 The snapshot

The snapshot is a versioned, ordered JSON document carrying schema version, tool version, generation timestamp, collection profile, Graph context, and four evidence sections — tenant, licensing, identity and Multi-Geo — followed by the collection manifest. Every leaf is an array, including single-object responses, so consumers can iterate uniformly. Absent optional collectors produce present-but-empty structures rather than missing keys, which keeps downstream parsing simple.

HANDLING A snapshot contains user principal names, display names, domain names, application identifiers and tenant configuration. Treat it as sensitive tenant data: store it accordingly, and use identifier redaction before circulating it beyond the engagement team.

6.3 The operational extracts

The two CSV extracts exist because findings alone do not let anyone do the work. `license-assignments.csv` resolves each assignment to a readable SKU part number and records whether it arrived directly or through group-based licensing — the single most useful fact when untangling an assignment error. `pdl-users.csv` joins directory and mailbox evidence per user and resolves each to one of four placement states: tenant default, evidence not collected, aligned, or mismatch requiring review. That column alone is usually the migration team's work queue.

7 Security and privacy posture

7.1 Least privilege

The Standard profile requests nine delegated Graph read scopes. The Full profile adds exactly two more. No write scope is requested under any configuration, and none would be usable if granted. Exchange and SharePoint evidence uses those services' own role models, where the guidance is unambiguous: prefer Global Reader or the narrowest service reader role that exposes the required configuration, and never grant a write role solely to produce a report. The complete scope matrix is in Appendix A.

Microsoft may additionally require a supported directory role for certain delegated calls. Where that requirement is not met, the collection fails visibly rather than returning empty — which is the entire point of the manifest.

7.2 Authentication modes

Four authentication paths are supported, and the choice has real governance consequences. Interactive sign-in is the default and the right choice for a first assessment. Device-code sign-in suits restricted or headless workstations. Certificate-based application authentication enables unattended, scheduled collection — appropriate for drift monitoring, and requiring an app registration with pre-consented application permissions. Finally, an existing session can be reused, which lets an operator authenticate through their own established process and hand the tool a context it did not create.

7.3 Identifier redaction

Redaction replaces user display names and principal names, application display names, mailbox identifiers, the signed-in account and tenant technical notification addresses with deterministic labels derived from a truncated SHA-256 hash of the lower-cased source value. Because user and mailbox labels derive from the same source,

the user-to-mailbox join used by the PDL analysis survives redaction intact — a redacted snapshot remains fully analysable.

Two properties of this mechanism must be understood before relying on it. First, redaction is applied before rules run, so every artifact — report, snapshot, findings and both CSVs — is consistently redacted; there is no path that emits a redacted report alongside an unredacted extract. Second, the hash is unsalted and deterministic. That is what makes snapshots diffable across runs, but it also means labels are reversible by an adversary who holds a list of candidate user principal names. Redaction reduces incidental exposure when circulating evidence; it is not anonymisation, and it should not be presented to a client as such.

IMPORTANT Redaction covers identifiers, not structure. Tenant identifier, verified domain names, group names, Conditional Access policy names, application identifiers, usage locations and PDL values are all retained by design, because the analysis depends on them. A redacted snapshot is still tenant-attributable.

8 Operating model

The following sequence reflects how the tool is intended to be used on a client engagement. It is deliberately incremental: each step either establishes confidence or reveals a permission gap cheaply, before the next step depends on it.

Step	Action	Why this order
1	Run the Standard profile interactively	Establishes a baseline with the smallest permission surface and no scheduling or app-registration dependency
2	Read the collection status chapter first	Determines how much of the report can be trusted. Fix permissions and re-run before interpreting any finding
3	Add the Full profile only when needed	Sign-in activity and Secure Score require two further scopes and depend on tenant licensing and consent
4	Add Exchange and SharePoint collectors for residency work	Multi-Geo rules become materially more useful once mailbox region and geo quota evidence are present
5	Retain the snapshot as engagement evidence	Enables offline re-analysis, drift comparison and defensible answers months later
6	Validate every high-impact finding with the service owner	The tool prioritises evidence for human review; it does not authorise change

8.1 Assessment cadence

Because analysis is separated from collection, the snapshot supports a monitoring pattern that a one-shot report cannot. Unattended certificate-based collection on a schedule produces a series of snapshots; comparing consecutive snapshots surfaces configuration drift — a Conditional Access policy disabled, a Global Administrator added, a licence pool approaching exhaustion, an application credential entering its expiry window — as a change rather than as a standing finding. For most enterprises a monthly cadence is proportionate, moving to weekly during active migration.

8.2 Where this fits in a delivery method

In a migration or residency engagement, this assessment belongs in the discovery and readiness phase, before the baseline is agreed and before any user list is frozen for fixed-price control. Its outputs feed directly into the readiness report, the user mapping document and the risk register. Re-running it at closure produces an as-built record from the same evidence pipeline that produced the baseline — which makes the before-and-after comparison genuinely like-for-like rather than two differently assembled documents.

8.3 Verification

The project ships a Pester suite that runs the rules engine and full artifact generation against a sanitised offline fixture, asserting that specific findings are produced and that all five artifacts are written. Because the snapshot seam makes collection unnecessary for these tests, they run in seconds with no tenant, no credentials and no network — which is exactly what makes them worth running on every change.

9 Interpretation boundaries

The report states its own limits in a dedicated chapter, and they are repeated here because a white paper that omits them would misrepresent the tool. This assessment is a point-in-time configuration snapshot. It does **not** test end-user connectivity or DNS propagation, prove data-at-rest location for every workload, establish legal compliance, interpret contractual licence rights, or measure the effectiveness of operational processes.

9.1 Specific things a reader must not conclude

- **Conditional Access findings do not measure coverage.** The rules examine whether a policy structurally targets all users; they do not evaluate exclusion groups, targeted applications, workload identities or authentication strengths. A tenant with no CA findings may still have ineffective policy.
- **Privileged access findings do not account for PIM.** Only standing active role assignments are counted. A tenant using eligible-only assignment correctly will produce a finding that is, in context, a false positive — and the tool says so in the finding text.
- **Inactivity is a signal, not waste.** Shared, service, newly created and leave-of-absence accounts are legitimate exceptions, and the rule is Review severity for exactly that reason. It also requires the Full profile; under Standard, sign-in activity is not collected and inactivity is simply not evaluated.
- **Mailbox database geography is inferred, not asserted.** It is derived from a database naming prefix, which is a useful corroborating signal and not an API-backed fact. Treat any finding based on it as a prompt to verify, never as proof.
- **A clean report is not a compliant tenant.** The tool evaluates twenty-one specific conditions. It is silent on everything else, including entire security domains it does not collect.

OPERATING RULE Validate high-impact findings with service owners before remediation. The assessment produces evidence and priorities; the decision to change a production tenant belongs to the people accountable for it, through their change-control process.

10 Engineering roadmap

The tool is at version 0.1.1 and is honest about being an MVP. The following items are the substantive gaps identified by a full source review, listed because a white paper that presented an early-stage tool as finished would be the same kind of overclaim the tool itself is designed to avoid.

10.1 Coverage

- **PIM-aware privileged access.** Reading eligible role assignments alongside active ones would remove the most significant false positive in the catalogue and would let the tool reward good practice rather than penalise it.
- **Rules for already-collected evidence.** Named locations, the authentication methods policy, the authorization policy, Microsoft Secure Score and SharePoint tenant sharing configuration are all collected and persisted but not yet evaluated or displayed. These represent immediate analytical value at zero additional permission cost.
- **Scope trimming.** Until Secure Score is consumed by a rule or a report chapter, the scope that enables it should arguably not be requested. Least privilege applies to the assessment tool before it applies to the tenant.

10.2 Robustness

- **Honour Retry-After.** Graph returns an explicit backoff hint under throttling; using it is both faster and better-behaved than a fixed doubling schedule. Adding jitter would prevent concurrent runs from resynchronising.
- **Status-code-based retry classification.** Recognising retryable conditions from the structured response rather than from exception message text would remove an incidental-match risk.
- **Broaden test coverage.** The current suite asserts six behaviours against one fixture. Redaction, CSV schemas, severity assignment and the collection layer are untested; mocked Graph responses would close most of that gap.
- **Ship a module manifest.** A .psd1 declaring the PowerShell version floor, module dependencies and a version would let the requirements be resolved by the platform rather than discovered at runtime.

10.3 Reporting

- **Snapshot diffing as a first-class mode.** The architecture already supports it; a comparison renderer would turn a point-in-time report into drift monitoring.
- **Provenance accuracy when re-rendering.** A report generated from an older snapshot should carry that snapshot's tool version, so a re-rendered artifact is not mislabelled.

11 Conclusion

The value of a tenant assessment is not the number of checks it performs. It is whether, six weeks later, anyone can defend what it said — and whether it was honest about what it could not see.

M365 Tenant Readiness is built around that standard. It reads and never writes. It records failure as prominently as success, and refuses to let an unreadable section masquerade as a clean one. It declines to issue a score it cannot justify, and reserves an entire severity band for evidence that needs a human. It separates collection from analysis so that the evidence outlives the engagement and two assessments can be compared rather than merely re-read. And it is small enough that a client security team can read the whole implementation before granting it a single scope.

None of that makes an assessment easy. It makes it defensible — which, in migration and data-residency work where the cost of a wrong assumption is measured in stalled waves and residency conversations, is the property that actually matters.

FURTHER READING — PRIMARY MICROSOFT REFERENCES

- Microsoft Graph — list subscribed SKUs, list users, and the licence assignment state resource
- Configure Preferred Data Location for Microsoft 365 resources with Microsoft Entra Connect
- Exchange Online data residency and Multi-Geo mailbox properties
- SharePoint Multi-Geo storage quota administration
- Conditional Access policy API reference

Priyanshu Upadhyay

Lead Technical Advisor · InnovaticT Technologies

Microsoft 365 solution architecture, tenant migration and data residency delivery

Appendix A • Microsoft Graph permission reference

Delegated read scopes requested by each profile. No write scope is requested under any configuration. Exchange and SharePoint use their own role models, shown separately.

STANDARD PROFILE — NINE DELEGATED SCOPES

Area	Scope	Enables
Tenant	Organization.Read.All	Tenant properties, sync state, notification contacts
Tenant	Domain.Read.All	Verified and unverified domain inventory
Licensing	LicenseAssignment.Read.All	Subscribed SKUs and per-user assignment state
Licensing	User.Read.All	User inventory, usage location, PDL, assigned licences
Licensing	Group.Read.All	Group inventory and group-based licensing
Identity	Policy.Read.All	Conditional Access policies, named locations, authorization policy
Identity	Policy.Read.Authentication Method	Authentication methods policy
Privileged	RoleManagement.Read.Directory	Directory role definitions and active assignments
Applications	Application.Read.All	App registrations and enterprise applications with credentials

FULL PROFILE — TWO ADDITIONAL SCOPES

Area	Scope	Enables
Activity	AuditLog.Read.All	User sign-in activity for inactivity review
Security	SecurityEvents.Read.All	Microsoft Secure Score (collected; not yet evaluated)

SERVICE COLLECTORS — OPT-IN, SEPARATE ROLE MODELS

Service	Access required	Evidence collected
Exchange Online	View-Only Recipients, View-Only Configuration	Default and allowed mailbox regions; per-mailbox region, database and last update
SharePoint Online	SharePoint Administrator or Global Reader	Tenant storage and sharing configuration; per-geo storage quota allocation

NOTE Microsoft may require a supported directory role in addition to the scope for certain delegated calls. Where that is not satisfied, the collection is recorded as failed with the endpoint and required scope preserved — it is never reported as an empty result.

Appendix B • Complete finding catalogue

All twenty-one rules, grouped by domain. Severity is fixed per rule and is never derived from the number of affected objects. Two identifiers are dynamic families emitting one finding per affected SKU.

TENANT READINESS

ID	Severity	Finding	Trigger condition
TEN-001	High	No verified custom domain discovered	Domains were collected, but none is both verified and a non-onmicrosoft.com domain

LICENSING

ID	Severity	Finding	Trigger condition
LIC-CAP-*	Critical	SKU is over allocated	Consumed units exceed enabled units for a subscribed SKU
LIC-LOW-*	Medium	SKU has low remaining capacity	Available units at or below the greater of two seats or five percent of the pool
LIC-001	High	Licence assignment errors detected	Any assignment state reports an error condition or carries error detail
LIC-002	Medium	Disabled accounts retain licences	Account is disabled but still holds one or more assigned licences
LIC-003	High	Licensed users missing UsageLocation	Enabled, licensed user has no usage location set
LIC-004	Review	Licensed accounts appear inactive	Sign-in activity present and older than the stale threshold (Full profile only)

IDENTITY CONTROLS

ID	Severity	Finding	Trigger condition
CMP-001	High	No enabled Conditional Access policies	Policies exist but none is in the enabled state; report-only does not count
CMP-002	High	No broad enabled MFA policy detected	No enabled policy applies an MFA grant control to all users
CMP-003	Medium	No enabled legacy-authentication block	No enabled policy blocks legacy client app types

PRIVILEGED ACCESS

ID	Severity	Finding	Trigger condition
TEN-002	High	Fewer than two active Global Administrators	Under two standing assignments — PIM-eligible assignments are not counted
TEN-003	Review	Large number of active Global Administrators	More than five standing assignments; review against least privilege

APPLICATIONS

ID	Severity	Finding	Trigger condition
APP-001	High	Expired application credentials	Password or certificate credential end date has already passed
APP-002	Medium	Application credentials nearing expiry	Credential expires within the warning window, ninety days by default

MULTI-GEO AND PDL

ID	Severity	Finding	Trigger condition
GE0-000	Info	No Multi-Geo indicators discovered	Emitted when the Multi-Geo gate is not met; suppresses all other GEO rules
GE0-001	Medium	Licensed users missing PreferredDataLocation	Multi-Geo detected and licensed users carry no PDL value
GE0-002	High	PDL outside allowed Exchange mailbox regions	A user PDL value is not among the tenant allowed mailbox regions
GE0-003	Review	MailboxRegion and PDL differ	Both values present for a user and they disagree
GE0-004	Review	Microsoft 365 groups use no explicit PDL	Unified groups exist with no PDL value set
GE0-005	Review	Mailbox database geo and MailboxRegion differ	Database name prefix disagrees with reported mailbox region

COLLECTION QUALITY

ID	Severity	Finding	Trigger condition
COL-001	Review	One or more evidence collections failed	Any collection recorded a failed status; names every failure

Appendix C • Output schema reference

C.1 Snapshot structure

Key	Contents
schemaVersion, toolVersion	Snapshot schema and tool version at generation time
generatedAtUtc	ISO-8601 round-trip UTC timestamp
collectionProfile	Standard or Full
graphContext	Tenant identifier, signed-in account, authentication type
tenant	Organization properties and domain inventory
licensing	Subscribed SKUs, users with assignment states, groups
identity	Conditional Access, named locations, authentication and authorization policy, role definitions and assignments, applications, service principals, Secure Score
multiGeo	Exchange organization and mailbox evidence; SharePoint tenant and geo quotas
collection	Manifest — name, area, endpoint, permissions, optional, status, count, duration, error

C.2 findings.csv

Eight columns. One row per finding, ordered by severity, then area, then identifier.

Column	Contents
id	Stable rule identifier
severity	Critical, High, Medium, Review or Info
area	Domain grouping
title	Human-readable finding name
evidence	Observed data supporting the finding
recommendation	Recommended action for human validation
affectedCount	Number of affected objects
controlThemes	Semicolon-separated control themes

C.3 license-assignments.csv

Eleven columns. One row per user per licence assignment state; users with no licences produce no rows.

Column	Contents
userPrincipalName, displayName	Identity, redacted when requested

Column	Contents
accountEnabled	Account state
usageLocation, preferredDataLocation	Directory placement attributes
sku	SKU part number, resolved from the SKU identifier
assignmentSource	Direct, Group:<id>, or Unknown
state, error	Assignment state and error detail
disabledPlanCount	Number of disabled service plans in the assignment
lastSuccessfulSignIn	Sign-in evidence; empty under the Standard profile

C.4 pdl-users.csv

Ten columns. One row per user, licensed or not, joined to mailbox evidence where collected.

Column	Contents
userPrincipalName, displayName	Identity, redacted when requested
accountEnabled	Account state
identitySource	On-premises synchronized or Cloud managed — determines where PDL must be corrected
usageLocation, preferredDataLocation	Directory placement attributes
mailboxRegion, mailboxRegionLastUpdateTime	Exchange mailbox placement evidence
databaseGeo	Geography inferred from the mailbox database name prefix
placementStatus	Tenant default, evidence not collected, aligned, or mismatch requiring review