Runs locally. Your app registration. Your storage.

TenuVault Desktop backs up, compares and restores Intune configuration from the admin's computer. It signs in through an app registration you create in your own tenant and talks directly to Microsoft Graph and, if you choose, your own Azure Storage account. Your Intune data, backups and Microsoft access tokens never pass through a vendor server.

Your own app registration

You create it in your tenant with the included script. No vendor app, no vendor client ID, no secret or certificate, no consent to a third party. Delete it and the app stops working.

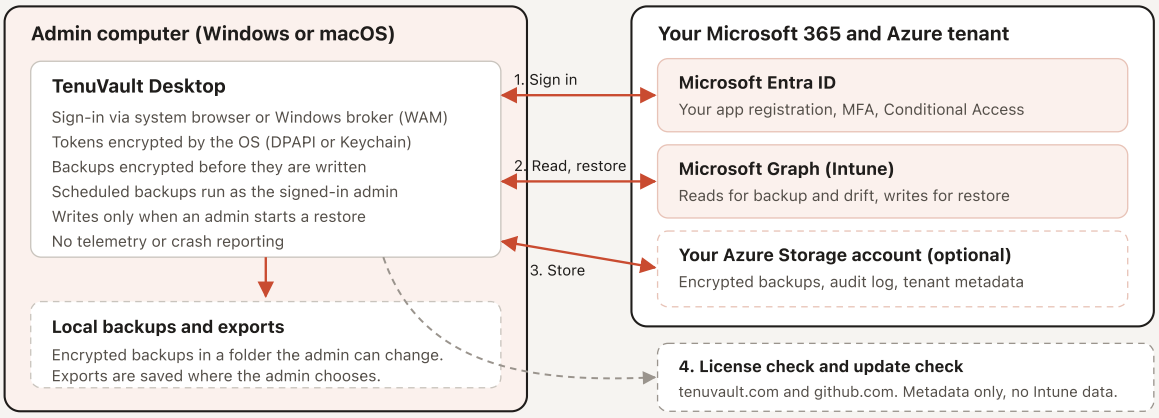
Delegated access, admin in control

Every call runs as the signed-in admin and is limited by their Intune role. Changes to Intune happen only when an admin starts a restore, a drift revert, a baseline deployment or its undo.

Backups stay with you

Backups are encrypted on the computer with AES-256-GCM and kept on local disk or in your own Azure Storage account. No vendor server stores or processes Intune data. No telemetry or crash reporting.

ARCHITECTURE



EVERYTHING THAT LEAVES THE COMPUTER (NO INTUNE DATA GOES TO THE VENDOR)

PURPOSE	DESTINATION	WHAT IS SENT
Sign-in	`login.microsoftonline.com`	Standard OAuth request to your tenant with your app registration
Intune	`graph.microsoft.com`	Requests with the admin's delegated token. Reads for backups and drift; writes only for restores, drift reverts and baseline deployments an admin starts
Storage picker	`management.azure.com`	Optional. Lists the subscriptions and storage accounts the admin can see. Read only
Azure backups	`<account>.blob.core.windows.net`	Optional. Your own storage account: backup content (encrypted before upload) and audit log entries. Not used when backups are stored locally
License check	`tenuvault.com`	License key (if any), tenant ID, random install ID, OS, app version and, when signed in, your client ID and a short-lived Microsoft ID token. The service reads only the tenant and client ID from the token; name and UPN are not stored. Runs at start, every 6 hours, on first use of a tenant and on license changes
Updates	`github.com`, `*.githubusercontent.com`	Version check and signed installer download. Can be turned off in Settings or by policy
Baselines	`api.github.com`, `raw.githubusercontent.com`	Optional. Downloads the public OpenIntuneBaseline repository for Quick Start and Frameworks. Nothing about your tenant is sent

The service also sees the requesting IP address. Sign-in through the browser or the Windows account broker contacts Microsoft services directly, outside TenuVault's control.

PERMISSIONS (DELEGATED ONLY, GRANTED BY YOUR ADMIN CONSENT)

PERMISSION	ACCESS	USED FOR
`User.Read`	Read	Sign-in
`Organization.Read.All`	Read	Tenant name and domains
`DeviceManagementConfiguration.ReadWrite.All`	Read, write	Configuration, compliance and endpoint security policies
`DeviceManagementApps.ReadWrite.All`	Read, write	Apps, app configuration and app protection policies
`DeviceManagementServiceConfig.ReadWrite.All`	Read, write	Enrollment, Autopilot, terms and conditions, branding
`DeviceManagementScripts.ReadWrite.All`	Read, write	Scripts and remediations
`DeviceManagementRBAC.ReadWrite.All`	Read, write	Intune roles, role assignments, scope tags, multi admin approval
`DeviceManagementManagedDevices.Read.All`	Read	Device counts and compliance rate, device categories, clean-up rules. Device records are not stored
`Policy.Read.All`	Read	Requested by the setup script, not used by this version
`Azure Service Management: user_impersonation`	Read	Storage account picker (GET requests only)
`Azure Storage: user_impersonation`	Read, write	Backups and audit log in your storage account

Plus openid, profile and offline_access. No application permissions, no client secret, no certificate. Consent is tenant-wide and tokens carry all consented scopes. Intune calls use Graph beta; the tenant overview also uses v1.0. Write scopes exist for restore; Microsoft enforces the admin's Intune role and scope tags on every call.

DATA ON THE COMPUTER AND AT REST

- **Tokens, settings and backup keys** are encrypted with DPAPI or a macOS Keychain key, bound to the OS user. With the Windows account broker, Windows holds the tokens. The app will not start without OS encryption.
- **Local backups** use AES-256-GCM per file, with keyed-hash file names so policy names are not visible on disk. Default folder: Documents/TenuVault Backups.
- **Azure backups** are encrypted on the computer (AES-256-GCM) before upload. Blob names (policy display names), audit log and tenant metadata rely on Azure's encryption at rest.
- **Recovery key.** Azure uploads are blocked until the admin saves it. Keep it in a password manager; without it, backups cannot be read on another computer or after a reinstall.
- **Exports** (backup ZIP, reports) are unencrypted, saved where the admin chooses. Backups can hold secret values such as OMA-URI settings, so treat exports and the recovery key as sensitive.
- **Retention.** Backups older than the retention setting (default 30 days) and audit entries older than 90 days are deleted.

CHANGES TO YOUR TENANT

- **Admin-started only.** Scheduled jobs read Intune and write to backup storage, never to Intune. Restores, reverts and deployments need an admin to start them.
- **Preview and confirm.** The live tenant is compared with the backup, and overwriting requires an explicit confirmation. The default restore creates copies, with assignments off.
- **Cross-tenant copies** are always created unassigned; referenced apps need reviewed mappings.
- **Write journal.** Restore, revert and Quick Start writes are journaled locally; an unconfirmed write blocks retries until reconciled.
- **Audit trail** under the admin's name, in your Azure storage or, for local backups, on the computer. Intune also logs every change.

Vendor. Ugurlabs UG (haftungsbeschränkt), Düsseldorf, Germany. Licensing at tenuvault.com; keys and payments by Polar, sharing records in Supabase. DPA available on request. Security contact: security@ugurlabs.com

BACKGROUND BACKUPS

- TenuVault runs backups itself, as the signed-in admin, while it runs in the tray. An optional per-user Task Scheduler task or LaunchAgent starts it at logon. No service account, no stored password, no app-only identity.
- If the session expires, the backup stops and asks the admin to sign in again.

APPLICATION HARDENING

- Released Windows builds are signed with Azure Trusted Signing; macOS builds are signed and notarized.
- Electron sandbox, context isolation, no Node.js in the interface, a Content Security Policy limiting scripts and connections to the app, and navigation to other sites blocked.
- Electron fuses disable run-as-node, NODE_OPTIONS and inspector flags and enforce an integrity-checked app archive, verified in CI.
- The license service issues an Ed25519-signed entitlement, valid 14 days and verified offline.

YOUR CONTROLS

- ✓ **Assignment required.** The setup script can limit sign-in to one admin group. This also decides who can use a license shared with the tenant.
- ✓ **Conditional Access.** Require MFA and a compliant device for this app. Sign-ins appear in your Entra logs.
- ✓ **Least privilege.** Intune roles and scope tags limit what TenuVault sees and changes. With a read-only role, Microsoft rejects restores.
- ✓ **Storage access.** Grant Storage Blob Data Contributor on the backup account only, or keep backups local.
- ✓ **Update control.** Updates download automatically and install on quit. Turn this off in Settings or with the `DisableAutoUpdate` policy (HKLM or HKCU `\SOFTWARE\Policies\TenuVault`, or managed preference `com.tenuvault.desktop`).
- ✓ **Verify it yourself.** Intune audit logs show every change. With Graph activity logs enabled, filter by your client ID to see every call.
- ✓ **Revoke anytime.** Disable or delete the app registration.