DokLocker

Security Overview

How DokLocker protects the passwords, network details and documentation that MSPs and IT teams keep in it, how it supports HIPAA and PCI DSS programs, and where we are on the road to SOC 2.

Version 1.75 · November 2026

A MaxDok product · doklocker.com/security

Contents

1. Summary and certification status
2. How DokLocker is deployed
3. Data protection and encryption
4. Sign-in and identity
5. Access control
6. Audit logging and monitoring
7. Integrations, agents and the AI assistant
8. Secure development and testing
9. HIPAA support
10. PCI DSS support
11. Shared responsibility
12. Roadmap and known gaps
13. Reporting a security problem

1 Summary and certification status

DokLocker is IT documentation for MSPs and internal IT teams. Customers keep their clients' passwords, network layouts, configurations and procedures in it, so we treat everything in DokLocker as highly sensitive.

Certification status. DokLocker does not yet hold a SOC 2 report, ISO 27001 certification or any HIPAA or PCI DSS attestation. We are preparing for SOC 2 (Type I, then Type II) and have built the product with the SOC 2 trust criteria in mind. This document describes what the software does today, based on the shipped code, and lists the gaps we know about. We will update it as each step is completed.

Highlights of what is in place today:

- Every stored password, Wi-Fi key, one-time code secret and integration key is encrypted before it is written to the database. Nightly backups can be encrypted with AES-256-GCM.
- Two-factor sign-in, single sign-on (Google, Microsoft, Okta and other OpenID Connect providers, SAML, Active Directory), lockout, idle sign-out and trusted networks.
- Fine-grained access: roles, a separate permission to see passwords, security groups per client and record, temporary access that ends on its own, and one-step offboarding.
- A tamper-evident audit log of every change, password reveal, sign-in, export and share.
- A one-switch HIPAA and PCI DSS mode that holds sign-in rules to those standards' minimums, plus a card number guard that keeps payment card data out of DokLocker.
- Signed release images with a software bill of materials, vulnerability scanning on every release, and two in-depth security code reviews in November 2026 with every confirmed finding fixed.

2 How DokLocker is deployed

Self-hosted

A single Docker image the customer runs on their own server or NAS (for example Synology), with SQLite or PostgreSQL. All data stays on the customer's hardware. Remote access is through Cloudflare Tunnel, Tailscale or the customer's own reverse proxy, so DokLocker never needs an open inbound port.

Hosted service

The hosted service is being prepared for launch. It is designed to run on a major cloud provider with managed PostgreSQL, encrypted cloud file storage and HTTPS everywhere. Until it is live, this document's hosting statements describe its design, not a running service.

Tenant isolation

Each customer is a separate workspace. Every database query for documentation is scoped to the signed-in person's workspace, and public tokens (share links, API tokens, agent keys, calendar feeds, invitations' links) resolve to exactly one workspace. Our security reviews include cross-tenant access tests of every numbered URL as another tenant.

Container hardening

- The application drops to an unprivileged user as soon as its data folder is prepared.
- Uploaded files are never served as static files: every photo and file goes through a view that checks the person is signed in and allowed to see it.
- On the hosted service, an outbound network guard stops integrations from reaching private or internal addresses (protection against server-side request forgery), and outbound requests only use http and https.

3 Data protection and encryption

Data	Protection
Record passwords, Wi-Fi keys, one-time code secrets, personal vault items, integration keys, BitLocker recovery keys	Encrypted with Fernet (AES-128-CBC with HMAC-SHA256 authentication). The key is derived with HKDF-SHA256 from the install's secret key, which is kept outside the database, so a copy of the database alone can't be read. Earlier values are kept in an encrypted history.
Revealing a password	Passwords are sent to the browser only when someone with permission clicks to reveal one, hide again after 30 seconds, and every reveal is written to the audit log.

Share links, API tokens, partner keys, agent keys, calendar feeds	Only a SHA-256 hash of the token is stored; the full link or key is shown once. Share link passcodes are hashed, links expire, and repeated wrong passcodes lock the link.
Sign-in passwords	Salted, slow hashes (Django's PBKDF2-SHA256 default); never stored or logged in readable form.
Nightly backups	Database and uploaded files in one archive, optionally encrypted with AES-256-GCM in authenticated chunks using a key derived from a passphrase with scrypt, so a cut-off or altered backup is detected.
Data in transit	HTTPS with Secure, HttpOnly cookies and HSTS when DokLocker is marked as running behind HTTPS (always on the hosted service and the public demo).
Uploaded photos and files	Hosted service: encrypted cloud storage. Self-hosted: stored on the customer's disk, so we recommend volume encryption (see section 11).
Payment card data	Subscriptions are paid through Stripe Checkout; card numbers never reach DokLocker's servers. The optional card guard masks card numbers typed into documentation (section 10).

Your data stays yours

- Export everything to Excel or CSV at any time, in a format DokLocker can import again.
- If a trial or plan ends, nothing is deleted: the workspace stays readable so you can look things up and export.
- People can delete their own account, and admins can delete their whole workspace, from settings.

4 Sign-in and identity

- **Two-factor sign-in** with an authenticator app, plus backup codes. Admins can require it for everyone, including client portal logins.
- **Single sign-on** with Google, Microsoft 365, Okta and other OpenID Connect providers, SAML, and Active Directory (LDAP). Admins can turn off passwords for everyone except admins, who keep a password so they can get in if the identity provider is down. A workspace's identity provider can only sign in that workspace's own people.
- **Password rules:** at least 12 characters, checked against a list of common passwords and against the person's own email, not all numbers, and none of the last five passwords can be used again.
- **Lockout** after a set number of wrong passwords (default 5, for 15 minutes), plus sign-in rate limits per visitor.
- **Idle sign-out** after a set number of minutes; sessions otherwise end after 12 hours.
- **Trusted networks:** limit sign-in to office or VPN addresses, with per-person exceptions that require an authenticator app.
- Changing or resetting a password, an admin resetting someone's two-factor, suspending, disabling or offboarding someone all end that person's other sessions.

- Password reset and invitation emails always link to the install's real address, never one taken from the request.

5 Access control

- **Roles:** read-only, contributor and editor, plus workspace admin.
- **Seeing passwords is a separate permission**, off by default, decided in one place in the code for the web, API, AI assistant, share links and the client portal.
- **Security groups** limit which clients' and records' passwords each team can open.
- **Temporary access** for one client or all of them, for a set number of hours, ending on its own.
- **Personal vault** for each person's own logins, which nobody else can see, admins included.
- **Offboarding** in one step: signs the person out, turns off sign-in, revokes their API tokens, share links and open invitations, and lists every password they viewed so it can be changed.
- **Client portal:** each client sees only their own documentation, read-only, and passwords only if the MSP allows it.
- **API tokens** are read-only unless made otherwise, can expire, and can be limited from reading passwords.

6 Audit logging and monitoring

- Every create, change and delete is recorded automatically, wherever it happens (web, import, API, AI assistant, integrations), with who, what, when, from which address and the old and new values. Passwords appear only as "(password changed)".
- Password reveals, sign-ins, failed sign-ins, lockouts, exports, shares, share link opens, token changes, resets and offboarding are all logged.
- **Tamper-evident:** each event is chained to the one before it with a keyed HMAC-SHA256. Admins can check the whole chain at any time; any changed, deleted or cut-off event is reported.
- Events are kept for the life of the workspace, with no automatic deletion. Admins can download the log as CSV, or stream every event to a SIEM or log store with signed webhooks.
- Alerts by email, Teams, Slack and other channels for offboarding, expiring items and other events.

7 Integrations, agents and the AI assistant

- Integration keys are encrypted and never shown again after saving. Changing an integration's server address requires entering the key again, so a saved key can't be redirected to another server.
- Responses from outside services are size-limited, and on the hosted service integrations can only reach the public internet.
- Each client has its own agent key, which only works for its own report type and its own client.
- Signed webhooks (HMAC-SHA256) for outgoing events; inbound mail webhooks use secret per-integration addresses, and unknown senders or mail failing SPF or DMARC go to quarantine.

- **DokBot, the AI assistant,** only acts when asked, sees only what the person asking can see, never reads passwords back, and labels its changes in the audit log. Each workspace chooses its AI provider (including a local model) or turns DokBot off. With the card guard on, card numbers are removed before a message reaches the provider.

8 Secure development and testing

- Every change runs a suite of more than 600 automated tests before a release is built.
- Release images are signed with Sigstore cosign (keyless, tied to our build workflow), carry an SPDX software bill of materials, and are scanned for known vulnerabilities with Trivy. Self-hosted customers can verify the signature before running an update.
- **Security code reviews:** in November 2026 we ran two in-depth reviews of the whole codebase using a structured, multi-phase audit method in which every finding is checked by an independent verifier. They confirmed 36 issues (16 in the first review and 20 in the second; 3 rated high, the rest medium or low). All 36 are fixed, each with a regression test that keeps it fixed. A full review now runs every month.
- Web protections: cross-site request forgery tokens, clickjacking protection (frames denied), content type sniffing off, a content security policy that blocks plugins and framing, and sanitized Markdown.
- These reviews were performed by our own team with automated tooling. They are not a substitute for an independent penetration test, which is planned (section 12).

9 HIPAA support

DokLocker is documentation software, not a health record system, but MSPs serving healthcare clients can end up with electronic protected health information (ePHI) in notes, photos or files. DokLocker is **designed to support** a HIPAA compliance program with the technical safeguards of the Security Rule (45 CFR 164.312). Turning on *HIPAA and PCI DSS safeguards* under Admin › Security enforces the settings below, and a built-in checklist shows what is in place for each workspace.

Safeguard	How DokLocker supports it	Status
Unique user identification 164.312(a)(2)(i)	Every person has their own login; shared logins aren't needed. Every action is attributed in the audit log.	In place
Emergency access 164.312(a)(2)(ii)	Admins keep a password sign-in when single sign-on is required; temporary access can be granted for a set time and is logged.	In place
Automatic logoff 164.312(a)(2)(iii)	Idle sign-out, held to 15 minutes or less in HIPAA/PCI mode, including the client portal.	In place
Encryption and decryption 164.312(a)(2)(iv)	Passwords and secrets encrypted by DokLocker; backups encrypted with a passphrase. Uploaded files rely on encrypted storage (hosted) or the customer's disk encryption (self-hosted).	Partly
Audit controls 164.312(b)	Automatic audit log of changes, password reveals, sign-ins and exports, kept for the life of the workspace, exportable and streamable to a SIEM.	In place
Integrity 164.312(c)(1)	Tamper-evident audit chain; version history for records and passwords; authenticated encryption detects altered secrets and backups.	In place
Person or entity authentication 164.312(d)	Two-factor sign-in required for everyone in HIPAA/PCI mode; single sign-on; lockout.	In place
Transmission security 164.312(e)	HTTPS with HSTS (hosted, demo); self-hosted behind Cloudflare Tunnel, Tailscale or HTTPS proxy.	In place
Business associate agreement 164.308(b)	A BAA for the hosted service is being prepared and is not yet available. Self-hosted customers keep ePHI on their own systems, so DokLocker isn't a business associate for that data.	Not yet
Administrative and physical safeguards 164.308, 164.310	Risk analysis, policies, workforce training, facility controls. The customer's responsibility; the checklist lists them.	Customer

HIPAA compliance is a property of an organization's whole program, not of a piece of software. Using DokLocker does not by itself make an organization HIPAA compliant.

10 PCI DSS support

DokLocker is not meant to store, process or transmit cardholder data, and the best way to keep it out of PCI DSS scope is to keep card numbers out of it. For MSPs whose clients take payments, DokLocker may document systems in or connected to a cardholder data environment, so it supports the relevant PCI DSS v4.0.1 requirements:

Requirement	How DokLocker supports it	Status
3.2, 3.3 Don't keep cardholder data you don't need	The card guard masks card numbers (checked by brand prefix, length and check digit) in notes, articles, comments, checklists and DokBot chats to the last four digits before saving, and logs each removal.	In place
3.5 Protect stored secrets	Passwords and keys encrypted with keys kept outside the database.	In place
4.2 Strong cryptography in transit	HTTPS with HSTS.	In place
7 Restrict access by need to know	Roles, password permission, security groups, temporary access, per-client portal.	In place
8.2.6 Inactive accounts	In HIPAA/PCI mode, members who haven't signed in for 90 days are turned off nightly; admins are alerted.	In place
8.2.8 Idle session timeout	15 minutes or less in HIPAA/PCI mode.	In place
8.3.4 Lockout	10 attempts or fewer, at least 30 minutes, in HIPAA/PCI mode.	In place
8.3.6, 8.3.7 Password length and reuse	12 characters minimum; last five can't be reused.	In place
8.4 Multi-factor authentication	Required for everyone in HIPAA/PCI mode.	In place
10.2, 10.3 Audit logs and their protection	Automatic audit log; tamper-evident chain; read-only to everyone.	In place
10.5.1 Keep logs 12 months	Kept for the life of the workspace; CSV export and SIEM webhooks for an off-system copy.	In place
11.4 Penetration testing	Independent penetration test planned.	Planned
12 Policies, risk assessment, training	The customer's responsibility; listed on the in-app checklist.	Customer

11 Shared responsibility

DokLocker provides

- The application safeguards in this document.
- Signed, scanned releases and security fixes.
- On the hosted service (when live): hosting, patching, encrypted storage and backups.

The customer is responsible for

- Turning on two-factor sign-in (or HIPAA/PCI mode) and choosing who can see what.
- Self-hosted: HTTPS (`MAXDOK_HTTPS=1` behind a tunnel or proxy), encrypted backups (`MAXDOK_BACKUP_PASSPHRASE`) kept off the server, disk or volume encryption, keeping `MAXDOK_SECRET_KEY` safe and separate from backups, and applying updates.
- Risk analysis, policies, training, BAAs with their own clients, and reviewing the audit log.
- Keeping card numbers and unnecessary health data out of documentation.

12 Roadmap and known gaps

We would rather be clear than impressive. These are the items we know are not done yet:

Item	Status
SOC 2 Type I report, then SOC 2 Type II	Preparing
Independent third-party penetration test	Planned
Business associate agreement for the hosted service	Planned
Hosted service launch on a major cloud provider with encrypted storage and backups	Planned
Application-level encryption of uploaded files for self-hosted installs (today: disk encryption)	Planned
Separate encryption keys per workspace (today: one key per install)	Planned
A stricter content security policy without inline scripts	Planned
Confirmation before the AI assistant changes a different client than the one being discussed	Planned
Written security policies, incident response plan and vendor reviews for SOC 2	Preparing

13 Reporting a security problem

Please tell us privately at **security@doklocker.com**. We'll reply within two business days, keep you updated while we fix it, and credit you if you'd like. Please don't access other customers' data or disrupt the service while testing.

This overview describes DokLocker version 1.75 as of November 1, 2026. It is provided for information and is not a contract or a warranty. The latest version is at doklocker.com/security.html.