Security paper

Shared responsibility

Which parts of a self-hosted Pando installation's security Pando handles, and which are yours.

Contents

1 Scope
2 Infrastructure
3 Identity and access
4 Apps
5 Data
6 What Pando doesn't protect against
7 Reporting a vulnerability
Revision history

1 Scope

This covers a self-hosted installation. The customer is whoever runs the installation and deploys apps to it. Pando's column lists controls the software enforces; the customer's column lists what the customer configures, runs or decides.

2 Infrastructure

Area	Pando	Customer
Host and operating system	Runs in containers. Reports whether its container runtime is rootless.	Physical security, operating system hardening and patching, disk encryption, and who has root or SSH access.
Container runtime	Refuses a runtime that won't apply the resource limits it sets.	Installing and patching Docker, Podman or Kubernetes, and running it rootless. On Kubernetes, a network plugin that enforces NetworkPolicy.
Network exposure	Publishes no app ports on the host. Every app is reached through Pando.	Firewall rules and DNS. Not routing traffic to an app around Pando.
TLS	Can run the edge and issue and renew certificates.	Choosing how certificates are issued, or running your own TLS proxy. Setting `PANDO_SERVER_EXTERNAL_URL` and `PANDO_SERVER_TRUSTED_PROXIES` for whichever one terminates TLS.
Pando's database	Generates its database password. No default credential is shipped.	If you run PostgreSQL yourself: patching, backups, network access and who holds superuser.
Pando updates	Releases are signed. In-place upgrades check the signature, back up first and roll back if the new version doesn't start.	Turning on or scheduling upgrades, and patching the operating system, container runtime and PostgreSQL.

Table 1. The host, runtime, network and Pando itself.

3 Identity and access

Area	Pando	Customer
Sign-in to apps	Every request to every app passes through Pando's identity-aware proxy. Apps are private until shared.	Choosing who each app is shared with, and whether public apps are allowed.
MFA and password policy	Local passwords are stored as argon2id hashes. Password sign-in can be turned off once an identity provider is set up.	MFA, password rules and conditional access, in your identity provider. Pando's local accounts have no MFA.
Account lifecycle	Accepts SCIM. Deprovisioned accounts are suspended and their sessions end.	Turning on SCIM or choosing a session lifetime, and offboarding local accounts. Your identity provider is the source of truth.
Permissions	Enforces roles and grants. Using an app and managing it are separate permissions. Built-in roles can't be changed. Denials are audited.	Who holds which role, and access control inside each app beyond opening it.
Identity passed to apps	Replaces forged identity headers, keeps its own cookies and tokens from apps, and signs an assertion on each request.	Verifying the assertion against Pando's JWKS endpoint before an app trusts who the visitor is.
API tokens	Shown once and stored as an HMAC. Delegated tokens follow their owner's access. Policy can limit token lifetime.	Token lifetime policy, reviewing tokens, and storing machine tokens safely.
Terminal access	Its own permission, audited, and can be turned off for the installation.	Whether to allow it and who holds it. Anyone with it can read the app's secrets and data.

Table 2. Who can sign in, and what they can do.

4 Apps

Area	Pando	Customer
App vulnerabilities	Scans each app's image and source and scores it. Can block deploys below a minimum score.	Fixing vulnerabilities in app code, dependencies and base images, and redeploying.
Builds	Builds run rootless in their own container, never on the host, with no access to the container runtime.	What a build runs and downloads.
Isolation between apps	Each app runs on its own private network that only Pando can reach, within its resource limits. Can run apps under gVisor or Kata.	Installing a sandboxed runtime and requiring it in host policy, where stronger isolation is needed.
Outbound network	Enforces egress rules, when set, through its own gateway.	Setting egress rules. Outbound traffic is allowed by default.
AI features	Off unless an AI provider is configured. Every call to a model is audited.	Choosing a provider and accepting its data handling, or using a local model. Repository contents can be sent to it.

Table 3. Building and running the apps you deploy.

5 Data

Area	Pando	Customer
Secrets	Encrypted at rest. Redacted in logs, exports and the audit log. Reading a value is its own permission.	Protecting and backing up the encryption key on the host, and setting and rotating secret values.
App data	Volumes survive redeploys and app deletion.	What apps store, declaring volumes for it, and disk encryption.
Backups	Encrypted backups of each app and of the whole installation, verified before a restore. Written to the host's disk.	Copying backups off the host, keeping the full backup's passphrase, and testing restores.
Audit log	Pando can't change or delete entries. Old months are removed only once archived. Can send events to a SIEM.	Archive storage, connecting a SIEM, and who holds superuser on the database.

Table 4. Secrets, app data, backups and the audit log.

6 What Pando doesn't protect against

- **Someone with root on the host, or superuser on Pando's database.** They can reach any container, secret or record without going through Pando.
- **A compromised host.** The key that encrypts secrets is stored on the host, so encryption at rest protects against a copied database or volume, not against access to the running host.
- **Mutually hostile tenants.** One installation serves one organization. Apps are isolated from each other, but an installation isn't a boundary between separate customers.
- **Vulnerabilities in apps.** Pando scans and scores apps and can block deploys, but it doesn't change an app's code or dependencies.

7 Reporting a vulnerability

Report a vulnerability in Pando through a private security advisory on GitHub. Response times and supported versions are in the security policy.

Report a vulnerability in an app to whoever deployed it.

Revision history

Version	Date	Changes
1.0.2	11 October 2026	Reporting a vulnerability (7) links the security policy on GitHub.
1.0.1	10 October 2026	Corrected TLS (2): `PANDO_SERVER_EXTERNAL_URL` and `PANDO_SERVER_TRUSTED_PROXIES` apply behind Pando's own edge too, not only behind a TLS proxy you run.
1.0	10 October 2026	First published.