Runner security

From the SourceLace manual: For your security reviewers and auditors: how a SourceLace Runner is secured, field by field what it sends to SourceLace and what it receives, how it signs in, encrypts and logs, how its image is delivered, what is kept and for how long, and who is responsible for what.

SourceLace version 0.1.0 · Made November 22, 2026
The latest version is at https://sourcelace.ai/docs

Runner security

For your security reviewers and auditors: how a SourceLace Runner is secured, field by field what it sends to SourceLace and what it receives, how it signs in, encrypts and logs, how its image is delivered, what is kept and for how long, and who is responsible for what.

What a runner is and what stays where is in Runners. How to install it, with every network connection, is in Runner requirements and planning and Installing a runner. Questions about SourceLace's cloud are answered in Security questions.

In short

- The runner runs your AI agents inside your network. It connects **out** only, over HTTPS, and **asks** SourceLace for work. Nothing on the internet connects in, and SourceLace has no way to connect to the runner, read its database or run commands on it.
- Your data stays on the runner: the rows and records agents read, run summaries, files, drafts, changes waiting for approval, people's sign-ins to your systems, your sources' secrets and your AI key. SourceLace gets only each run's status, times, one line per step and counts.
- Everything the runner keeps is encrypted with a master key that **you** make and hold. SourceLace never has it.
- You choose when to install a new version. The runner's container runs as an unprivileged user with read-only files.

What the runner sends to SourceLace

Every request goes to SourceLace over HTTPS and carries the runner token (below) and the runner's version. These are all the requests there are:

Asking for work (about every 15 seconds)

Field	What it holds
Version	The runner's version number.
Results page address	The https address of the runner's page, so SourceLace can link "Open result" to it.
How many runs it can take	A number from 1 to 10.
Health	How long it has been running; the **type** of its last error (such as `ControlUnavailable`, never the message); how many runs it holds and reports it has not sent yet; free disk and free memory in MB.
License and build	The identifier of the license it holds, the identifier of its image build, and whether it is asking only for a license. Identifiers only.
Protocol	A version number of the way it talks to SourceLace.

Keeping a run (about every minute while it runs)

The run's identifier and a number SourceLace gave with it. Nothing else. SourceLace answers whether the run must stop.

Reporting a run

Field	What it holds
Status	Running, waiting for approval, done, failed or stopped.
Times	When it started and ended.
Steps (at most 200)	For each step: the tool's name, the source's name, the object's name (such as "Account"), how many rows came back (a number), how long it took, whether it worked, was refused or failed, the **type** of error, and the kind of action.
Counts	How many records it changed, and the step credits it used.
Whether it wrote	Whether the run made a draft, change, post or message, sent before its first write so that a run that wrote is never run twice.
Error	A short message saying why the run failed or stopped, at most 500 characters.
Result address	The address of this run's result on the runner's page.

Reading source settings (every 5 minutes)

The request carries nothing but the token. SourceLace answers with the settings of the sources the runner's agents use, without any secret option.

Completing a sign-in

When someone signs in to the runner's page, the runner sends SourceLace the one-time code the person's browser brought back and a matching proof it made itself, nothing else.

What it never sends

The runner never sends SourceLace: rows or field values from your systems, run summaries, files, email drafts, change previews, people's sign-ins to your systems, your sources' secrets, your AI key, the master key or anything the AI model said. A report has no place for any of these, and **SourceLace refuses any request from a runner that carries a field it does not know**, so a report cannot carry them by mistake. Field sizes are limited too.

The one free text in a report is the short error message. Ask your SourceLace contact if your review needs its exact contents for your agents.

What SourceLace sends to the runner

With each run it hands out:

- **The run**: its identifiers, the agent and its version, the owner, what started it, whether it is a trial run, and when it was due. For a run started by a call to the agent's webhook or by an email to its address, what that call or email sent (an email's attachments only by name and size), which passes through SourceLace on its way.
- **The agent's definition**: its instructions, sources, actions and schedule, as your people built it in SourceLace.
- **The limits and settings** the run must keep, and the **email address of the person** it runs for.
- **Your organization's rules for that run**, for the agent's sources only. These are settings, never data. The runner keeps them in memory only and never runs an agent without them:
  - approval rules and fixed values for changes, and matching rules;
  - data protection rules: what to hide or mask before the AI model sees it;

- for the person the run is for (the `person` part of the run): their read rules (which rows they may read), the identifiers of their groups, and their **access by group** (`person.access`): whether access by group is on and which sources and objects their groups allow them to read or change;
- the **names of that person's groups** (`person.access.group_names`), and nothing else about the groups, so that a custom connector (one your developers built that asks who is asking) is told the same person and groups as on SourceLace's cloud.

Also:

- **Source settings** without any secret, every 5 minutes, and only for systems your organization may use now: a system SourceLace switched off for your organization is left out, and runs that use it are not handed out.
- **Its license**: a short signed note saying that this runner of your organization may take new work until a time (The runner's license).
- **Which runs to stop**, when an agent is paused or stopped or your organization's stop switch is used.
- **For a sign-in**: the person's email address, your organization and their role in SourceLace.

SourceLace cannot reach into the runner. It only answers the runner's own requests, so it cannot open the runner's database, read results, change its settings or run commands on it.

Signing in

The runner token

- An admin creates a runner on the **Runners** page in SourceLace; its token is shown **once**.
- SourceLace keeps only a SHA-256 fingerprint of the token, never the token itself, so a copy of SourceLace's database cannot be used as a runner token.
- On the runner, the token is a setting your IT department holds (in a settings file readable only by its administrators, or a Kubernetes secret). The runner never writes it to its log or shows it on its pages, and its support bundle leaves it out.
- **Revoking** the runner on the **Runners** page stops the token at once: SourceLace refuses everything from it, and nobody can sign in to its page. At its first refused request (within about a minute) the runner stops its runs in progress and writes nothing more. To replace a token, add a new runner, give its token to the runner, move the agents to it and revoke the old one.

The license

The runner takes new work only while it holds a current license from SourceLace, signed by SourceLace. It lasts 30 days and is renewed about once a day during the runner's usual requests while your contract includes the runner. Without a license, the runner takes no new work; runs in progress finish and nothing is deleted.

Getting the runner

The install bundle and the image are downloaded in SourceLace, behind your sign-in, by your organization's admins only (**Runners**, **Get the runner**). The bundle comes with a SHA-256 checksum of each file, and each version with its image digest; an image file link works for 5 minutes and then expires. Pulling uses a registry login issued on the same page: read-only, for your organization's images only, valid one hour. The images are kept in SourceLace's private, encrypted registry on Amazon ECR, one repository per organization. Every link and registry login issued is written to your audit log, without the link or password itself.

People signing in to the runner's page

- People sign in with their usual SourceLace sign-in (including your single sign-on); the runner has no passwords of its own.
- The runner sends the browser to SourceLace, which checks that the person is signed in, is a member of the runner's organization and that the runner is not revoked, and sends back a one-time code that works once, for two minutes, for that runner only. The runner exchanges it, with a proof only it holds, for the person's **email address, organization and role**. SourceLace tells the runner nothing else.
- The runner then keeps the person signed in for **12 hours**, with a cookie that scripts cannot read, that is sent only over https and is signed with a key derived from your master key. Forms on the page are protected against requests from other sites.
- Admins see every run and the **Settings** page; other members see their own agents' runs.
- Someone removed in SourceLace cannot sign in to the runner again; a sign-in already open ends within 12 hours.

Encryption

In transit

- **Runner to SourceLace**: HTTPS only, TLS 1.2 or newer, with SourceLace's certificate checked. The runner refuses to start with an address for SourceLace that is not https.
- **Runner to your AI provider and to your systems**: TLS with certificate checks, as on SourceLace's cloud (Encryption in transit). If your network inspects TLS, you give the runner your own certificate authorities (TLS inspection).
- **People to the runner's page**: https through your own TLS proxy and internal certificate. The runner refuses to start with an `http://` page address.
- **Runner to its database**: TLS when you require it on the connection (Database).

At rest

- In the runner's database, these are encrypted with your organization's own key, which is itself protected by your **master key**: people's sign-ins to your systems, your sources' secret options, the OAuth apps registered for the runner, your AI key, run summaries, files, changes waiting for approval, reports not yet sent, its license and its own audit trail.
- Large results the runner holds while agents work are kept in an encrypted file per result on the runner's disk, deleted when they expire.

- The database's own disk encryption is yours to choose, as for any database you run.
- The algorithms and how keys are kept are in the security review pack, which you can request from security@sourcelace.com.

Who holds the keys

Key	Who makes it	Where it is kept	Does SourceLace have it?
The master key	Your IT department, when installing	Your secret store and the runner's settings	**No.** Without it, the runner's data cannot be read, by anyone.
Your organization's AI key	Your admin	The runner's database, encrypted	No
Your sources' secrets and OAuth apps	Your admin	The runner's database, encrypted	No
People's sign-ins to your systems	Each person, on the runner's page	The runner's database, encrypted	No
The runner token	SourceLace, shown once to your admin	The runner's settings	Only its fingerprint
A registry login (to download the image)	SourceLace, from **Get the runner**, for an admin; recorded in your audit log	Not kept: used once with `docker login`	Expires by itself after one hour

You can change the master key yourself (Changing the master key).

The runner image

- **Made for your organization**: SourceLace builds a runner image for your organization and puts it in a private registry. Your admins download it in SourceLace, behind sign-in (Getting the runner), as a file or by pulling it with a registry login that can read only your organization's images and expires after one hour. Every login issued is recorded in your audit log.
- **Versioned, never "latest"**: every image has an exact version number. You choose when to install a new one; nothing updates by itself. Each version is supported until 6 months after the next one is released (How long each version is supported).
- **Checked before release**: before an image is released, SourceLace's release checks run the runner's tests against the image's contents and check that it holds no readable source code, no shell, package manager or compiler, that it does not run as the administrator (root), and that it starts and answers its health address with read-only files.
- **Small and locked down**: the image holds only what the runner needs to run. It runs as an unprivileged user (number 10001), its own files are read-only, it writes only to a temporary folder and the folder for large results, and it needs no extra privileges. The Helm values also drop every Linux capability, forbid privilege escalation and use the default seccomp profile.
- **Traceable**: each image carries an identifier of its build, which the runner reports to SourceLace with its requests for work, so an image can be traced to the organization it was made for.
- **What you can check yourself**: `docker image inspect <image> | grep '"User"'` shows the user it runs as (`10001:10001`); `docker image ls --digests <image name>` shows the digest of each version you downloaded, which you can compare with the digest shown for that version under **Get the runner** (Installing a runner, step 3). Images are not signed today.

Logs and audit trail

On the runner

- The runner writes its log to the container's standard output, which Docker or Kubernetes keeps and your log collection can pick up. It holds the runner's own events: starting, contact with SourceLace, runs that could not be run, by identifier. It never writes rows, values, summaries, tokens, keys or passwords on purpose, and every log line, including error details, is cleaned of anything that looks like a token, key, password, signed address or sign-in code before it is written.
- The runner also keeps its own audit trail in its database: sign-ins to its page, admins' changes to its settings, connectivity checks, support bundle downloads, and each agent run's steps and approvals, as SourceLace does. It is not shown on the runner's page yet.
- The **support bundle** an admin can download on the **Settings** page holds the version, settings with every secret left out, the types of recent errors and how many of each, and the connectivity check, and no tokens, keys, passwords, data or results. You can open it and check it before sending it to support.

On SourceLace

- SourceLace records in your organization's audit trail every request from a runner that hands out, keeps, stops or reports a run, every request it refused, every admin action on runners (adding, renaming, revoking), and every download link and registry sign-in issued under **Get the runner**. Each entry names the runner as `runner:<runner id>` and holds identifiers and counts only. A request for work that finds nothing to do is not recorded, but updates the runner's "last seen".
- Signing in to a runner's page, and refused sign-ins, are recorded too.
- When a runner with agents goes quiet, and when it is back, it is recorded once.
- Your admins read, filter and export the trail on the **Audit log** page (Can we see and export it?).

How long data is kept

What	Where	How long
Run results, files and decided approvals	The runner's database	Your retention period, set on the runner's **Settings** page (30 days unless you change it); then deleted. The runner checks every 10 minutes.
Changes waiting for approval	The runner's database	Until decided, or until they expire undecided.
Large results while agents work	The runner's disk, encrypted	Deleted when they expire.
Reports not yet sent to SourceLace	The runner's database, encrypted	Until SourceLace takes them (at most 30 days).
People's sign-ins and secrets	The runner's database, encrypted	Until the person disconnects or an admin removes them.
Each run's status, steps and counts	SourceLace	As other agent runs (How long are chats, agent runs and the audit trail kept?).

What	Where	How long
The runner's log	Your Docker or Kubernetes log storage	As you set it.

When a runner is revoked, SourceLace refuses its token, nobody can sign in to its page, and runs queued for it end as failed. A run already going on the runner is stopped at the runner's next contact with SourceLace (usually within about 15 seconds, at most about a minute), marked as failed, and writes nothing more to your systems; the runner then applies no change waiting for approval, and its **Settings** page and log say that SourceLace refused its token. Stop the runner's containers to end everything. Nothing on the runner is deleted by revoking: its database still holds its results and sign-ins, encrypted, until you delete it (Revoking and removing a runner).

Who is responsible for what

Area	SourceLace	Your organization
The runner software	Builds, tests and releases it; fixes problems in it; makes each version available as a private image	Installs each version, following the upgrade plan agreed with SourceLace
The machine or cluster		Provides it, patches its operating system, Docker or Kubernetes, and watches it
The database		Provides PostgreSQL, its backups, restores and upgrades, and its disk encryption
The master key	Never has it	Makes it, keeps it safe, backs it up apart from the database
The runner token and registry logins	Issues them; can revoke a token; logins expire after one hour	Keeps them safe; revokes a runner that is no longer used
The network	Keeps SourceLace's own service reachable over HTTPS	Opens the outbound connections, keeps the runner's page internal, runs the DNS name, certificate and TLS proxy
Who signs in to the runner	Checks each sign-in: member of your organization, runner not revoked	Manages your people in SourceLace and your single sign-on; decides who is an admin
Your AI key and source secrets		Enters and rotates them on the runner's page
Rules for agents	Sends your rules with each run; the runner applies them	Your admins set them in SourceLace
Monitoring	Shows each runner's status, health and license, and alerts your admins when it goes quiet	Watches the runner's health address, log, disk and backups
Incidents	Tells you about problems in the runner software and SourceLace's service, as for the rest of SourceLace	Tells SourceLace about problems on the runner, with the support bundle

For your security review

SourceLace does not hold a SOC 2 report or any other attestation today; see Organization and certifications. The table below maps the runner's controls to the areas of the Trust Services Criteria your reviewers use, so you can assess them yourselves.

Area	The runner's controls
Access control (CC6.1 to CC6.3)	A runner token per runner, shown once, stored by SourceLace only as a fingerprint, revocable at once; people sign in through SourceLace (with your single sign-on), members of your organization only, for 12 hours; roles: admins and members; secrets entered on the runner are never shown again.
Network boundaries (CC6.6)	Outbound connections only, over HTTPS; no inbound connection from the internet; the runner's page reachable only inside your network; SourceLace cannot connect to the runner.
Data in transit and at rest (CC6.1, CC6.7)	TLS 1.2 or newer with certificate checks; everything the runner keeps encrypted with keys protected by your master key, which SourceLace never has.
Change management (CC8.1)	Versioned images, released after SourceLace's tests and image checks; you decide when to install a version; nothing updates by itself.
Monitoring (CC7.1, CC7.2)	Status, health and license on the **Runners** page; alerts when a runner goes quiet; a health address for your monitoring; SourceLace's audit trail of every runner request; the runner's own audit trail and log.
Data boundaries (C1.1)	Reports carry metadata only, and SourceLace refuses anything else; your data protection rules, read rules and access by group apply on the runner.
Retention and disposal (C1.2)	Run results deleted after your retention period; large results deleted when they expire; you delete the runner's database when you remove it.
Backup and recovery (A1.2)	Your database backups plus the master key kept apart; a restarted runner sends any report it had not sent and never runs a run twice.

For more, ask security@sourcelace.com for the security review pack. The onboarding included with a runner has a security review with you.