Authenticate cloud agents

- Use the cloud secret store
- Choose an authentication method
- Keep credentials out of the sandbox filesystem
- Set credential scope
- Service identifiers
- Configure a custom secret
- List and remove credentials

Cloud agents authenticate with credentials from the cloud secret store. Set up these credentials before launching an agent so authentication doesn't depend on files stored inside the sandbox.

Use the cloud secret store

The following secret interfaces are separate:

- `sbx secret` manages secrets for local sandboxes
- `sbx --cloud secret` manages secrets for cloud sandboxes created by the CLI
- Docker Agentic Platform manages its own secret names through its web interface

A credential created through one interface isn't available through the others. Configure a cloud credential using one of the following methods before starting a cloud sandbox.

Choose an authentication method

For Claude Code, use an Anthropic API key. For Codex, use an OpenAI API key or account-scoped OpenAI OAuth credentials.

Configure your agent with the corresponding command:

Agent or provider	Recommended command	Authentication

Claude Code	`sbx --cloud secret set anthropic`	Prompts for an Anthropic API key
Codex with OpenAI OAuth	`sbx --cloud secret set openai --oauth`	Opens the OpenAI OAuth flow and stores the resulting credential at account scope
Service API key	`sbx --cloud secret set <service>`	Prompts for an API key or token

For Codex, you can use OpenAI OAuth at account scope or store an OpenAI API key with `sbx --cloud secret set openai`.

After storing an Anthropic API key, launch Claude Code:

```
$ sbx --cloud run claude --name cloud-project
```

Keep credentials out of the sandbox filesystem

Credentials configured with `sbx --cloud secret` stay in the cloud secret store, outside the sandbox filesystem.

An agent's interactive sign-in can write credentials inside the sandbox. Those files can be included in templates and `sbx move` snapshots. If you signed in inside an agent, follow the provider's sign-out guidance and remove those credentials before capturing or moving the sandbox.

Set credential scope

Credentials use account scope by default. An account-scoped credential is available to cloud sandboxes in the Docker account:

```
$ sbx --cloud secret set github
```

Scope an API key or token to one sandbox by name:

```
$ sbx --cloud secret set openai --sandbox cloud-project
```

A sandbox-scoped secret takes precedence over an account-scoped secret for the same service. Set it before creating the sandbox so the CLI can include it when the sandbox starts. OAuth credentials can't use sandbox scope.

Service identifiers

The following table shows cloud secret support for the built-in services documented for local sandboxes:

Service	Cloud secret authentication
`anthropic`	API key
`cursor`	API key
`droid`	API key; cloud-managed OAuth is not supported
`github`	Token
`google`	API key
`groq`	API key
`mistral`	API key
`nebius`	API key
`openai`	API key or OAuth
`openrouter`	Use a custom secret
`xai`	API key

To configure a supported service, run `sbx --cloud secret set <service>`. For OpenAI OAuth, add `--oauth` and use account scope.

For other services, configure a custom secret.

The cloud secret commands don't support registry credentials or dynamic `--ref` and `--command` resolvers. An environment file can resolve a host command or vault reference once with `snapshot: true` and upload the result as a literal cloud secret.

Configure a custom secret

Use `set-custom` for an API that isn't a built-in service:

```
$ sbx --cloud secret set-custom --name project-api \
    --host api.example.com --env PROJECT_API_TOKEN
```

The command prompts for the value. Sandboxes using this secret receive a placeholder in `PROJECT_API_TOKEN`. The cloud proxy sets `Authorization: Bearer <secret>` on requests to

`api.example.com`. Unlike local custom secrets, cloud injection sets the header based on the destination host without requiring a matching placeholder in the request.

Use `--header` to select another header. When you specify a header, the default value is the raw secret. Use `--format` with one `%s` placeholder to add a prefix, for example `--header Authorization --format 'Bearer %s'`.

Each `--host` must be an exact DNS name without a scheme, port, or path. Repeat the flag for multiple hosts. IP addresses and wildcards aren't supported. Add `--sandbox cloud-project` to scope the secret to that sandbox, and set it before creating the sandbox. A custom secret name can't match a built-in service name.

List and remove credentials

List cloud secret metadata without revealing values:

```
$ sbx --cloud secret ls
```

Remove a credential from the matching scope:

```
$ sbx --cloud secret rm github
$ sbx --cloud secret rm openai --sandbox cloud-project
$ sbx --cloud secret rm project-api
```

Custom secrets can also be removed by a host they serve, using `sbx --cloud secret rm --host api.example.com`. Removal asks for confirmation. Use `--force` in scripts. Cloud mode doesn't support removing every secret in one operation.

□□□ #2
□ □□□ □□□ 2026-10-31 18:58:09 UTC
□ □□□ □□□ 2026-10-31 18:58:10 UTC