Shell

- Default startup command
- Base image

{{% include "sandboxes-local-scope.md" %}}

`sbx run shell` drops you into a Bash login shell inside a sandbox with no pre-installed agent binary. It's useful for installing and configuring agents manually, testing custom implementations, or inspecting a running environment.

```
$ sbx run shell ~/my-project
```

`sbx run` defaults the workspace to the current directory. To run a one-off command instead of an interactive shell, pass it after `--`:

```
$ sbx run shell -- -c "echo 'Hello from sandbox'"
```

To create a mountless sandbox, use `sbx create` without a workspace path, then attach by name:

```
$ sbx create --name scratch shell
$ sbx run --name scratch
```

Default startup command

Without extra args, the sandbox runs `bash -l`. When the first argument after `--` is a flag (begins with `-`), it's added after `-l`, so login-shell behavior is preserved:

```
$ sbx run shell -- -c "echo hi"   # runs bash -l -c "echo hi"
```

When the first argument is a bare word, it replaces `-l` instead.

Store credentials using stored secrets before running the sandbox. The proxy injects them into outbound API requests; credentials are never stored inside the VM:

```
$ sbx secret set anthropic
$ sbx secret set openai
```

Once inside the shell, you can install agents using their standard methods, for example `npm install -g @continuedev/cli`. For complex setups, build a workload kit instead of installing interactively each time.

Base image

The shell sandbox uses the `shell` base image — the common base environment without a pre-installed agent.

□□□ #2
□ □□□ □□□ 2026-10-31 18:58:03 UTC
□ □□□ □□□ 2026-10-31 18:58:04 UTC