

ADMIN CONSOLE > MANAGE MEMBERS > DIRECTORY CONNECTOR

Directory Connector CLI

View in the help center:

<https://bitwarden.com/help/directory-sync-cli/>

Directory Connector CLI

The Directory Connector CLI is suited toward work in environments where a desktop GUI is unavailable, or if you want to programmatically script directory sync operations using tools provided by the operating system (cron job, scheduled task, and more). The Directory Connector CLI can be used cross-platform on Windows, macOS, and Linux distributions.

Getting started

Tip

The desktop app and CLI [share a database and configurations](#), so **simultaneous** use on a single machine is not recommended. The recommended path is to complete configuration and testing using the [desktop app](#), and subsequently using the CLI to [schedule automatic syncing](#) to your production organization.

To get started using the Bitwarden Directory Connector CLI:

1. Download the CLI from one of the following links:

-  [Windows CLI](#)
-  [macOS CLI](#)
-  [Linux CLI](#)

2. Extract the `.zip` and move the contents (`bwdc` and `keytar.node`) to `/usr/local/bin` or another directory in your `$PATH`. Please note, `keytar.node` **must** be in the same directory as the primary `bwdc` executable.

Linux only: If not already installed, install `libsecret` with your package manager of choice. Note that the package is titled `libsecret-1-0` for Ubuntu and Debian specifically, users should find the equivalent title for their particular distribution:

Bash

```
apt-get install libsecret-1-0  
brew install libsecret
```

Windows only: Windows users can [add](#) .

3. Verify that the `bwdc` command works in your terminal by running the following:

Bash

```
bwdc --help
```

4. Connect Directory Connector to your directory using the `bwdc config <setting> <value>` command (see [here](#)).
5. Configure sync options by editing your `data.json` file (to learn more, see [Directory Connector File Storage](#)). Use the `bwdc data-file` command to obtain the absolute path of your `data.json` file.
Available **sync options** depend on the directory type in use, so refer to one of the following articles for a list of options available to you:
 - [Sync with Active Directory or LDAP](#)
 - [Sync with Azure Active Directory](#)
 - [Sync with G Suite \(Google\)](#)
 - [Sync with Okta](#)
 - [Sync with OneLogin](#)
6. Run the `bwdc test` command to check whether your configuration would sync the expected results.
7. Once your directory and sync options are properly configured, and `bwdc test` yields the expected results, run the `bwdc sync` command to start a live sync operation.

Note

The `--pretty` flag can be included in bwdc commands to modify the output for readability.

Commands reference

login

Use the `login` command to log in to Directory Connector with your [organization API key](#). If you don't have the API key, reach out to an [organization owner](#). There are a few ways to use the `login` command:

- By itself:

Bash

```
bwdc login
```

Passing `bwdc login` by itself will prompt you to subsequently enter `client_id` and `client_secret`.

- With parameters:

Bash

```
bwdc login organization.b5351047-89b6-820f-ad21016b6222 yUMB4trbqV1bavhEHGqbuGpz4AlHm9
```

- With saved environment variables:

Bash

```
BW_CLIENTID="organization.b5351047-89b6-820f-ad21016b6222"
```

```
BW_CLIENTSECRET="yUMB4trbqV1bavhEHGqbuGpz4AlHm9"
```

```
bwdc login
```

Saving the environment variables `BW_CLIENTID` and `BW_CLIENTSECRET` allows you to login to Directory Connector using only `bwdc login`, which will check for those variables and use them if present.

If these environment variables aren't present, you will be prompted to enter your `client_id` and `client_secret`.

logout

Use the `logout` command to logout of the Directory Connector CLI.

Bash

```
bwdc logout
```

help

The Bitwarden Directory Connector CLI is self-documented with `--help` content and examples for every command. List all available commands using the global `--help` option:

Bash

```
bwdc --help
```

Use the `--help` option on any specific command to learn more about that command:

Bash

```
bwdc test --help  
bwdc config --help
```

test

The `test` command queries your directory and prints a JSON formatted array of groups and users that would be synced to your Bitwarden organization whenever you run a real sync operation.

Bash

```
bwdc test
```

Use the `--last` option to test only the changes since the last successful sync.

Bash

```
bwdc test --last
```

sync

The `sync` command runs a live sync operation and pushes data to your Bitwarden organization.

Bash

```
bwdc sync
```

Synced users and groups will be immediately available in your Bitwarden organization. Newly added users will receive an email invite to your organization.

Note

If you're on the [Teams Starter](#) plan, you are limited to 10 members. Directory Connector will display an error and stop syncing if you try to sync more than 10 members.

This plan is no longer available for purchase. This error does not apply to Teams plans.

last-sync

The `last-sync` command returns an ISO 8601 timestamp for the last sync operation that was performed for users or groups. You must specify either `users` or `groups` as an `<object>` to run the command against:

Bash

```
bwdc last-sync <object>
```

Returns an empty response if no sync has been performed for the given object.

config

The `config` command allows you to specify your directory settings:

Bash

```
bwdc config <setting> <value>
```

Available options are:

Option	Description
<code>server <server-url></code>	URL of your self-hosted installation (e.g. <code>https://business.bitwarden.com</code>) or EU server (<code>https://vault.bitwarden.eu</code>).
<code>directory <directory-type></code>	Type of directory to use. See the following table for enumerated values.
<code>ldap.password <password></code>	Password for connection to the LDAP server.
<code>azure.key <key></code>	Azure AD secret key.
<code>gsuite.key <key></code>	Google Workspace/GSuite private key.

Option	Description
<code>okta.token <token></code>	Okta token.
<code>onelogin.secret <secret></code>	OneLogin client secret.

`directory-type` values

Source directory	Value
Active Directory/LDAP	0
Azure Active Directory	1
Google Workspace/GSuite	2
Okta	3
OneLogin	4

data-file

The `data-file` command returns an absolute path to the `data.json` configuration file used by the Directory Connector CLI:

Bash

```
bwdc data-file
```

Some configuration settings can be modified for the Directory Connector CLI by editing the `data.json` configuration file directly in your favorite text editor, however `ldap.password`, `azure.key`, `gsuite.key`, `okta.token`, and `onelogin.secret` can **only** be

modified from the CLI using `config`, or from the [desktop app](#).

clear-cache

The `clear-cache` command allows you to clear cached data that the application stores while performing sync operations. For more information, see [Clear Sync Cache](#).

Bash

```
bwdc clear-cache
```

update

The `update` command allows you to check if your Directory Connector CLI is up-to-date:

Bash

```
bwdc update
```

If a newer version is found, the command will return a URL to download a new version. **The Directory Connector CLI will not automatically update.** You will need to use this URL download the new version manually.

Warning

If you using the CLI and desktop app together, it is important to ensure their versions match whenever in use. Running two different versions may cause unexpected issues.

Check the version of the Directory Connector CLI using the `--version` global option.

Troubleshooting

libsecret missing

If you receive an error message referring to the libsecret shared object `Error: libsecret-1.so.0: cannot open shared object file: No such file or directory`, you may need to install libsecret which is required to store things securely on the host.

dbus Errors

If you receive an error message referring to the dbus when using `bwdc config`, for example `Failed to execute child process "dbus-launch" (No such file or directory)` or `Cannot autolaunch D-Bus without X11`, assign the following environment variable to allow plaintext storage of secrets in `data.json`:

Bash

```
export BITWARDENCLI_CONNECTOR_PLAINTEXT_SECRETS=true
```

Debug

The debug environment variable can be added for troubleshooting information.

Plain Text

```
export BITWARDENCLI_CONNECTOR_DEBUG=true
```

Unable to get local issuer certificate

If you receive an error message that states `unable to get local issuer certificate`, set the `NODE_EXTRA_CA_CERTS` variable to your `root.pem`, for example:

Bash

```
export NODE_EXTRA_CA_CERTS="/absolute/path/to/your/certificates.pem"
```

If you're using the desktop app, this may also manifest as the following error: `Warning: Setting the NODE_TLS_REJECT_UNAUTHORIZED environment variable to '0' makes TLS connections and HTTPS requests insecure by disabling certificate verification.`

Failing to set private key

If you are receiving error `Object does not exist at path "/org/freedesktop/secrets/collection/login"` while configuring your private key, see the following steps to correct the issue.

The Bitwarden Directory Connector uses Linux's keyring, check that the following dependencies have been installed:

Bash

```
sudo apt install dbus-x11 gnome-keyring
```

Next, run the following commands to start the daemon:

Bash

```
export $(dbus-launch)
dbus-launch
gnome-keyring-daemon --start --daemonize --components=secrets
echo '<RANDOM-PASSPHRASE>' | gnome-keyring-daemon -r -d --unlock
```

Following these commands, try to run the key again, for example:

Bash

```
bwdc config gsuite.key /path/to/key/
```