
crashplancli

Release 1.0.4

CrashPlan Group

Aug 19, 2026

CONTENTS

1	User Guides	1
1.1	Get started with the CrashPlancommand-line interface (CLI)	1
1.2	Configure a profile	4
1.3	Manage legal hold custodians	5
1.4	Clean up your environment by deactivating devices	8
1.5	Manage Users	9
1.6	Using Bulk Commands	11
1.7	Ingest Audit Logs into a SIEM tool	12
1.8	Write custom extension scripts using the CrashPlan CLI and pycpg	13
2	Commands	15
2.1	audit-logs	15
2.2	devices	18
2.3	legal-hold	24
2.4	profile	29
2.5	users	32
3	Requirements	43
4	Content	45
	Index	47

USER GUIDES

1.1 Get started with the CrashPlancommand-line interface (CLI)

- *Licensing*
- *Installation*
- *Authentication*
- *Troubleshooting and Support*

1.1.1 Licensing

This project uses the [MIT License](#).

1.1.2 Installation

You can install the CrashPlan CLI from PyPI, from source, or from distribution.

From PyPI

The easiest and most common way is to use `pip`:

```
python3 -m pip install crashplancli
```

To install a previous version of the CrashPlan CLI via `pip`, add the version number. For example, to install version 1.0.0, enter:

```
python3 -m pip install crashplancli==1.0.0
```

Visit the [project history](#) on PyPI to see all published versions.

From source

Alternatively, you can install the CrashPlan CLI directly from [source code](#):

```
git clone https://github.com/CrashPlan-Labs/crashplancli.git
```

When it finishes downloading, from the root project directory, run:

```
python setup.py install
```

From distribution

If you want create a .tar ball for installing elsewhere, run the following command from the project's root directory:

```
python setup.py sdist
```

After it finishes building, the .tar ball will be located in the newly created dist directory. To install it, enter:

```
python3 -m pip install crashplancli-[VERSION].tar.gz
```

1.1.3 Updates

To update the CLI, use the pip --upgrade flag.

```
python3 -m pip install crashplancli --upgrade
```

1.1.4 Authentication

Important

The CrashPlan CLI currently only supports token-based authentication.

Create a user in CrashPlan to authenticate (basic authentication) and access data via the CLI. The CLI returns data based on the roles assigned to this user. To ensure that the user's rights are not too permissive, create a user with the lowest level of privilege necessary. See our [Role assignment use cases](#) for information on recommended roles. We recommend you test to confirm that the user can access the right data.

If you choose not to store your password in the CLI, you must enter it for each command that requires a connection.

The CrashPlan CLI supports local accounts with MFA (multi-factor authentication) enabled. The Time-based One-Time Password (TOTP) must be provided at every invocation of the CLI, either via the --totp option or when prompted.

The CrashPlan CLI currently does **not** support SSO login providers or any other identity providers such as Active Directory or Okta.

1.1.5 Proxy Support

Note

Proxy support was added in crashplancli version 1.16.0

The CrashPlan CLI will attempt to connect through a proxy if the https_proxy/HTTPS_PROXY environment variable is set.

Windows and Mac

For Windows and Mac systems, the CLI uses Keyring when storing passwords.

Red Hat Enterprise Linux

To use Keyring to store the credentials you 2enter in the CrashPlan CLI, enter the following commands before installing.

```
yum -y install python-pip python3 dbus-python gnome-keyring libsecret dbus-x11
pip3 install crashplancli
```

If the following directories do not already exist, create them:

```
mkdir -p ~/.cache
mkdir -p ~/.local/share/keyring
```

In the following commands, replace the example value \n with the Keyring password (if the default Keyring already exists).

```
eval "$(dbus-launch --sh-syntax)"
eval "$(printf '\n' | gnome-keyring-daemon --unlock)"
eval "$(printf '\n' | /usr/bin/gnome-keyring-daemon --start)"
```

Close out your D-bus session and GNOME Keyring:

```
pkill gnome
pkill dbus
```

If you do not use Keyring to store your credentials, the CrashPlan CLI will ask permission to store your credentials in a local flat file with read/write permissions for only the operating system user who set the password. Alternatively, you can enter your password with each command you enter.

Ubuntu

If Keyring doesn't support your Ubuntu system, the CrashPlan CLI will ask permission to store your credentials in a local flat file with read/write permissions for only the operating system user who set the password. Alternatively, you can enter your password with each command you enter.

To learn more about authenticating in the CLI, follow the [Configure profile guide](#).

1.1.6 Troubleshooting and support

CrashPlan command not found

If your python installation has added itself to your environment's PATH variable, then running `crashplan` *should* just work.

However, if after installation the `crashplan` command is not found, the CLI has some helpers for this (added in version 1.10):

You can execute the CLI by calling the python module directly:

```
python3 -m crashplancli
```

And the base `crashplan` command now has a `--script-dir` option that will print out the directory the `crashplan` script was installed into, so you can manually add it to your PATH, enabling the `crashplan` command to work.

On Mac/Linux:

Run the following to make `crashplan` visible in your shell's `PATH` (to persist the change, add it to your shell's configuration file):

```
export PATH=$PATH:$(python3 -m crashplancli --script-dir)
```

On Windows:

```
$env:Path += ";$(python -m crashplancli --script-dir)"
```

To persist the change, add the updated `PATH` to your registry:

```
Set-ItemProperty -Path 'Registry::HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\↪Session Manager\Environment' -Name PATH -Value $env:Path
```

Debug mode

Debug mode may be useful if you are trying to determine if you are experiencing permissions issues. When debug mode is on, the CLI logs HTTP request data to the console. Use the `-d` flag to enable debug mode for a particular command. `-d` can appear anywhere in the command chain:

```
crashplan <command> <subcommand> <args> -d
```

File an issue on GitHub

If you are experiencing an issue with the CrashPlan CLI, select *New issue* at the [project repository](#) to create an issue. See the [Github guide on creating an issue](#) for more information.

Contact CrashPlan Support

If you don't have a GitHub account and are experiencing issues, contact [crashplan support](#).

1.1.7 What's next?

Learn how to *Set up a profile*.

1.2 Configure a profile

Use the *crashplan profile* set of commands to establish the CrashPlan environment you're working within and your user information.

1.2.1 User token authentication

Use the following command to create your profile with user token authentication:

```
crashplan profile create --name MY_FIRST_PROFILE --server example.authority.com --↪username security.admin@example.com
```

Your profile contains the necessary properties for authenticating with crashplan. After running `crashplan profile create`, the program prompts you about storing a password. If you agree, you are then prompted to enter your password.

Your password is not shown when you do `crashplan profile show`. However, `crashplan profile show` will confirm that a password exists for your profile. If you do not set a password, you will be securely prompted to enter a password each time you run a command.

1.2.2 API client authentication

Once you've generated an API Client in your CrashPlan console, use the following command to create your profile with API client authentication:

```
crashplan profile create-api-client --name MY_API_CLIENT_PROFILE --server example.  
→authority.com --api-client-id 'key-42' --secret 'crashplan%api%client%secret'
```

Note

Remember to wrap your API client secret with single quotes to avoid issues with bash expansion and special characters.

1.2.3 View profiles

You can add multiple profiles with different names and then change the default profile with the use command:

```
crashplan profile use MY_SECOND_PROFILE
```

When you use the `--profile` flag with other commands, such as those in `audit-logs`, that profile is used instead of the default profile. For example,

```
crashplan audit-logs search -b 2020-02-02 --profile MY_SECOND_PROFILE
```

To see all your profiles, do:

```
crashplan profile list
```

1.2.4 Profiles with Multi-Factor Authentication

If your CrashPlan user account requires multi-factor authentication, the MFA token can either be passed in with the `--totp` option, or if not passed you will be prompted to enter it before the command executes.

1.3 Manage legal hold custodians

Once you [create a legal hold matter in the CrashPlan console](#), you can use the CrashPlan CLI to add or release custodians from the matter.

To see a list of all the users currently in your organization:

- Export a list from the [Users action menu](#).
- Use the [CLI users commands](#).

Use the `legal-hold` commands to manage legal hold custodians.

- To view a list of legal hold matters for your organization, including the matter ID, use the following command:
`crashplan legal-hold list`
- To see a list of all the custodians currently associated with a legal hold matter, enter `crashplan legal-hold show <matterID>`.

1.3.1 Get CSV template

To add multiple custodians to a legal hold matter:

1. Generate a CSV template. Below is an example command that generates a template to use when bulk adding custodians to legal hold matter. Once generated, the CSV file is saved to your current working directory.

```
crashplan legal-hold bulk generate-template add
```

To generate a template to use when bulk releasing custodians from a legal hold matter:

```
crashplan legal-hold bulk generate-template remove
```

The CSV templates for `add` and `remove` have the same columns, but the commands generate different default filenames.

2. Use the CSV template to enter the matter ID(s) and CrashPlan usernames for the custodians you want to add to the matters. To get the ID for a matter, enter `crashplan legal-hold list`.
3. Save the CSV file.

1.3.2 Add custodians to a legal hold matter

You can add one or more custodians to a legal hold matter using the CrashPlan CLI.

Add multiple custodians

Once you have entered the matter ID and user information in the CSV file, use the `bulk add` command with the CSV file path to add multiple custodians at once. For example:

```
crashplan legal-hold bulk add /Users/admin/add_users_to_legal_hold.csv
```

Add a single custodian

To add a single custodian to a legal hold matter, use the following command as an example:

```
crashplan legal-hold add-user --matter-id 123456789123456789 --username user@example.com
```

Options

- `--matter-id` (required): The identification number of the legal hold matter. To get the ID for a matter, run the command `crashplan legal-hold list`.
- `--username` (required): The CrashPlan username of the custodian to add to the matter.
- `--profile` (optional): The profile to use to execute the command. If not specified, the default profile is used.

1.3.3 Release custodians

You can [release one or more custodians](#) from a legal hold matter using the CrashPlan CLI.

Release multiple custodians

To release multiple custodians at once:

1. Enter the matter ID(s) and CrashPlan usernames to the *CSV file template you generated*.
2. Save the file to your current working directory.

3. Use the bulk `remove` command with the file path of the CSV you created. For example:

```
crashplan legal-hold bulk remove /Users/admin/remove_users_from_legal_hold.csv
```

Release a single custodian

Use `remove-user` to release a single custodian. For example:

```
crashplan legal-hold remove-user --matter-id 123456789123456789 --username user@example.com
```

Options are the same as `add-user` shown above.

1.3.4 View matters and custodians

You can use the CrashPlan CLI to get a list of all the [legal hold matters](#) for your organization, or get full details for a matter.

List legal hold matters

To view a list of legal hold matters for your organization, use the following command:

```
crashplan legal-hold list
```

This command produces the matter ID, name, description, creator, and creation date for the legal hold matters.

View matter details

To view active custodians for a legal hold matter, enter

```
crashplan legal-hold show
```

with the matter ID, for example:

```
crashplan legal-hold show 123456789123456789
```

To view active custodians for a legal hold matter, as well as the details of the preservation policy, enter

```
crashplan legal-hold show <matterID> --include-policy
```

To view all custodians (including inactive) for a legal hold matter, enter

```
crashplan legal-hold show <matterID> --include-inactive
```

List legal hold events

To view a list of legal hold administrative events, use the following command:

```
crashplan legal-hold search-events -m <matterID>
```

This command takes the required filters of a specific matter uid, and optional filters of beginning timestamp, end timestamp, and event type.

Future versions of the crashplancli will allow searching events from all legal holds.

Learn more about the *Legal Hold* commands.

1.4 Clean up your environment by deactivating devices

Your CrashPlan environment may contain many old devices that are no longer active computers and that have not connected to CrashPlan in quite some time. In order to clean up your environment, you can use the CLI to deactivate these devices in bulk.

1.4.1 Generate a list of devices

You can generate a list of devices using `crashplan devices list`. By default, it will display the list of devices at the command line, but you can also output it in a number of file formats. For example, to generate a CSV of active devices in your environment, use this command:

```
crashplan devices list --active -f CSV
```

To save to a file, redirect the output to a file in your shell:

```
crashplan devices list --active -f CSV > output.csv
```

Filter the list

You can filter or edit the list of devices in your spreadsheet or text editor of choice, but the CLI has some parameters built in that can help you to filter the list of devices to just the ones you want to deactivate. To see a full list of available parameters, run `crashplan devices list -h`.

Here are some useful parameters you may wish to leverage when curating a list of devices to deactivate:

- `--last-connected-before DATE|TIMESTAMP|SHORT_TIME` - allows you to only see devices that have not connected since a particular date. You can also use a timestamp or short time format, for example `30d`.
- `--exclude-most-recently-connected INTEGER` - allows you to exclude the most recently connected device (per user) from the results. This allows you to ensure that every user is left with at least N device(s), regardless of how recently they have connected.
- `--created-before DATE|TIMESTAMP|SHORT_TIME` - allows you to only see devices created before a particular date.

1.4.2 Deactivate devices

Once you have a list of devices that you want to remove, you can run the `crashplan devices bulk deactivate` command:

```
crashplan devices bulk deactivate list_of_devices.csv
```

The device list must be a file in CSV format containing a `guid` column with the unique identifier of the devices to be deactivated. The deactivate command can also accept some optional parameters:

- `--change-device-name` - prepends `deactivated_<current_date>` to the beginning of the device name, allowing you to have a record of which devices were deactivated by the CLI and when.
- `--purge-date yyyy-MM-dd` - allows you to change the date on which the deactivated devices' archives will be purged from cold storage.

To see a full list of available options, run `crashplan devices bulk deactivate -h`.

The `crashplan devices bulk deactivate` command will output the `guid` of the device to be deactivated, plus a column indicating the success or failure of the deactivation. To change the format of this output, use the `-f` or `--format` option.

You can also redirect the output to a file, for example:

```
crashplan devices bulk deactivate devices_to_deactivate.csv -f CSV > deactivation_
↪results.csv
```

Deactivation will fail if the user running the command does not have permission to deactivate the device, or if the user owning the device is on legal hold.

Generate the list and deactivate in a single command

You can also pipe the output of `crashplan devices list` directly to `crashplan devices bulk deactivate`. When using a pipe, make sure to use `-` as the input argument for `crashplan devices bulk deactivate` to indicate that it should read from standard input.

Here is an example:

```
crashplan devices list --active \
--last-connected-before 365d \
--exclude-most-recently-connected 1 \
-f CSV \
| CrashPlan devices bulk deactivate - \
-f CSV \
> deactivation_results.csv
```

This lists all devices that have not connected within a year *and* are not a user's most-recently-connected device, and then attempts to deactivate them.

Learn more about *Managing Devices*.

1.5 Manage Users

You can use the CLI to manage user information, update user roles, and move users between organizations.

To view all the users currently in your organization, you can export a list from the [Users list in the CrashPlan console](#) or you can use the `list` command.

You can use optional flags to filter the users you want to view. The following command will print all active users with the Desktop User role who belong to the organization with UID 1234567890:

```
crashplan users list --org-uid 1234567890 --role-name "Desktop User" --active
```

To change the information for one or more users, provide the user UID and updated information with the `update` or `bulk update` commands.

1.5.1 Manage User Roles

Apply [crashplan's user roles](#) to user accounts to provide administrators with the desired set of permissions. Each role has associated permissions, limitations, and recommended use cases.

View User Roles

View a user's current roles and other details with the `show` command:

```
crashplan users show "arthur.pendragon@example.com"
```

Alternatively, pass the `--include-roles` flag to the `list` command. The following command will print a list of all active users and their current roles:

```
crashplan users list --active --include-roles
```

Update User Roles

Use the following command to add a role to a user:

```
crashplan users add-role --username "arthur.pendragon@example.com" --role-name "Desktop_↵  
↵User"
```

Similarly, use the `remove-role` command to remove a role from a user.

1.5.2 Deactivate a User

You can deactivate a user with the following command:

```
crashplan users deactivate arthur.pendragon@example.com
```

To deactivate multiple users at once, enter each username on a new line in a CSV file, then use the `bulk deactivate` command with the CSV file path. For example:

```
crashplan users bulk deactivate users_to_deactivate.csv
```

Similarly, use the `reactivate` and `bulk reactivate` commands to reactivate a user.

1.5.3 Assign an Organization

Use [Organizations](#) to group users together in the CrashPlan environment.

You'll need an organization's unique identifier number (UID) to move a user into it. You can use the `list` command to view a list of all current user organizations, including UIDs:

```
crashplan users orgs list
```

Use the `show` command to view all the details of a user organization. As an example, to print the details of an organization associated with the UID 123456789 in JSON format:

```
crashplan users show 123456789 -f JSON
```

Once you've identified your organizations UID number, use the `move` command to move a user into that organization. In the following example a user is moved into the organization associated with the UID 1234567890:

```
crashplan users move --username arthur.pendragon@example.com --org-id 1234567890
```

Alternatively, to move multiple users between organizations, fill out the move CSV file template, then use the `bulk move` command with the CSV file path.

```
crashplan users bulk move bulk-command.csv
```

1.5.4 Bulk update users

To update a list of users in bulk, for example to update usernames and emails you will need a list of the user's `userId` values and then the new items that you want to update as headers in a provided csv.

A CSV with a `user_uid` column and the new values is required for the `update` command.

The new values that will be updated and their corresponding headers are:

- **username** The new username for the user.
- **email** The new email for the user.
- **password** The new password for the user.
- **first_name** The new first name for the user.
- **last_name** The new last name for the user.
- **notes** Notes about this user.

Once the CSV is created and saved to the working directory the bulk command to update users is:

```
crashplan users bulk update update-users-bulk.csv
```

1.5.5 Get CSV Template for bulk commands

The following command generates a CSV template for each of the available bulk user commands. The CSV file is saved to the current working directory.

```
crashplan users bulk generate-template [update|move|add-alias|remove-alias|update-risk-  
→profile]
```

Once generated, fill out and use each of the CSV templates with their respective bulk commands.

```
crashplan users bulk [update|move|deactivate|reactivate] bulk-command.csv
```

A CSV with a **username** column and a single username on each new line is used for the **reactivate** and **deactivate** bulk commands. These commands are not available as options for **generate-template**.

Learn more about *Managing Users*.

1.6 Using Bulk Commands

Bulk functionality is available for many CrashPlan CLI methods, more details on which commands have bulk capabilities can be found in the [Commands Documentation](#).

All bulk methods take a CSV file as input.

The **generate-template** command can be used to create a CSV file with the necessary headers for a particular command.

For instance, the following command will create a file named `devices_bulk_deactivate.csv` with a single column header row of `guid`.

```
crashplan devices bulk generate-template deactivate
```

The CSV file can contain more columns than are necessary for the command, however then the header row is **required**.

If the CSV file contains the *exact* number of columns that are necessary for the command then the header row is **optional**, but columns are expected to be in the same order as the template.

To run a bulk method, simply pass the CSV file path to the desired command. For example, you would use the following command to deactivate multiple devices within your organization at once:

```
crashplan devices bulk deactivate devices_bulk_deactivate.csv
```

A CSV with a **user_uid** column and the new information is required for the **update** command.

1.7 Ingest Audit Logs into a SIEM tool

This guide provides instructions on using the CLI to ingest CrashPlan Audit logs into a security information and event management (SIEM) tool like LogRhythm, Sumo Logic, or IBM QRadar.

1.7.1 Considerations

To ingest Audit logs into a SIEM tool using the CrashPlan command-line interface, the CrashPlan user account running the integration must be assigned roles that provide the necessary permissions.

1.7.2 Before you begin

First install and configure the CrashPlan CLI following the instructions in *Getting Started*.

1.7.3 Run queries

Audit logs are available in JSON format. You can query the data as a scheduled job or run ad-hoc queries.

Learn more about searching *Audit Logs* using the CLI.

Run a query as a scheduled job

Use your favorite scheduling tool, such as cron or Windows Task Scheduler, to run a query on a regular basis. Specify the profile to use by including `--profile`.

Audit Logs

An example to send to the syslog server only the audit log events that meet the filter criteria from the last 30 days.

```
crashplan audit-logs send-to syslog.example.com:514 -p UDP --profile profile1 --actor-  
↪username 'arthur.pendragon@example.com' -b 30d
```

As a best practice, use a separate profile when executing a scheduled task. Using separate profiles can help prevent accidental updates to your stored checkpoints, for example, by adding `--use-checkpoint` to adhoc queries.

Audit Logs

Print audit log events since June 5 which affected a certain user:

```
crashplan audit-logs search -b 2025-06-05 --affected-username 'arthur.pendragon@examply.  
↪com'
```

Example Outputs

Example output for a single audit log event (in default JSON format):

```
{  
  "type$": "audit_log::logged_in/1",  
  "actorId": "1015070955620029617",  
  "actorName": "arthur.pendragon@example.com",  
  "actorAgent": "pycpg 1.17.0 python 3.7.10",  
  "actorIpAddress": "67.220.16.122",  
  "timestamp": "2021-08-30T16:16:19.165Z",  
  "actorType": "USER"  
}
```

1.8 Write custom extension scripts using the CrashPlan CLI and pycpg

While the CrashPlan CLI aims to provide an easy way to automate many common CrashPlan tasks, there will likely be times when you need to script something the CLI doesn't have out-of-the-box.

To accommodate for those scenarios, the CrashPlan CLI exposes a few helper objects in the `crashplancli.extensions` module that make it easy to write custom scripts with `pycpg` that use features of the CLI (like profiles) to reduce the amount of boilerplate needed to be productive.

1.8.1 Before you begin

The CrashPlan CLI is a python application written using the [click framework](#), and the exposed extension objects are custom `click` classes. A basic knowledge of how to define `click` commands, arguments, and options is required.

The `sdk_options` decorator

The most important extension object is the `sdk_options` decorator. When you decorate a command you've defined in your script with `@sdk_options`, it will automatically add `--profile` and `--debug` options to your command. These work the same as in the main CLI commands.

Decorating a command with `@sdk_options` also causes the first argument to your command function to be the `state` object, which contains the initialized `pycpg` sdk. There's no need to handle user credentials or login, the `sdk_options` does all that for you using the CLI profiles.

The script group

The `script` object exposed in the `extensions` module is a `click.Group` subclass, which allows you to add multiple sub-commands and group functionality together. While not explicitly required when writing custom scripts, the `script` group has logic to help handle and log any uncaught exceptions to the `~/ .crashplancli/log/crashplan_errors.log` file.

If only a single command is added to the `script` group, the group will default to that command, so you don't need to explicitly provide the sub-command name.

An example command that just prints the username and ID that the sdk is authenticated with:

```
import click
from crashplancli.extensions import script, sdk_options

@click.command()
@sdk_options
def my_command(state):
    user = state.sdk.users.get_current()
    print(user["username"], user["userId"])

if __name__ == "__main__":
    script.add_command(my_command)
    script()
```

1.8.2 Ensuring your script runs in the CrashPlan CLI python environment

The above example works as a standalone script, if it were named `my_script.py` you could execute it by running:

```
python3 my_script.py
```

However, if the CrashPlan CLI is installed in a different python environment than your `python3` command, it might fail to import the extensions.

To workaround environment and path issues, the CLI has a `--python` option that prints out the path to the python executable the CLI uses, so you can execute your script with `$(crashplan --python) script.py` on Mac/Linux or `&&$(crashplan --python) script.py` on Windows to ensure it always uses the correct python path for the extension script to work.

1.8.3 Installing your extension script as a CrashPlan CLI plugin

The above example works as a standalone script, but it's also possible to install that same script as a plugin into the main CLI itself.

Assuming the above example code is in a file called `my_script.py`, just add a file `setup.py` in the same directory with the following:

```
from distutils.core import setup

setup(
    name="my_script",
    version="0.1",
    py_modules=["my_script"],
    install_requires=["crashplancli"],
    entry_points="""
        [crashplancli.plugins]
        my_command=my_script:my_command
    """,
)
```

The `entry_points` section tells the CrashPlan CLI where to look for the commands to add to its main group. If you have multiple commands defined in your script you can add one per line in the `entry_points` and they'll all get installed into the CrashPlan CLI.

Once your `setup.py` is ready, install it with `pip` while in the directory of `setup.py`:

```
$(crashplan --python) -m pip install .
```

Then running `crashplan -h` should show `my-command` as one of the available commands to run!

- *Get started with the CrashPlan command-line interface (CLI)*
- *Configure a profile*
- *Manage legal hold users*
- *Clean up your environment by deactivating devices*
- *Manage users*
- *Perform bulk actions*
- *Send audit logs to a SIEM*
- *Write custom extension scripts using the CrashPlan CLI and Pycpg*

COMMANDS

2.1 audit-logs

Get and send audit log event data.

Usage

```
audit-logs [OPTIONS] COMMAND [ARGS]...
```

2.1.1 clear-checkpoint

Remove the saved audit log checkpoint from *-use-checkpoint/-c* mode.

Usage

```
audit-logs clear-checkpoint [OPTIONS] CHECKPOINT_NAME
```

Options

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CHECKPOINT_NAME

Required argument

2.1.2 search

Search audit log events.

Usage

```
audit-logs search [OPTIONS]
```

Options

-b, --begin <begin>

The beginning of the date range in which to look for audit-logs. Accepts a date/time in yyyy-MM-dd (UTC) or yyyy-MM-dd HH:MM:SS (UTC+24-hr time) format where the 'time' portion of the string can be partial (e.g. '2020-01-01 12' or '2020-01-01 01:15') or a 'short time' value representing days (30d), hours (24h) or minutes (15m) from the current time. [required unless `--use-checkpoint` option used]

-e, --end <end>

The end of the date range in which to look for audit-logs, argument format options are the same as `--begin`.

--affected-username <affected_username>

Filter results by affected usernames.

--affected-user-id <affected_user_id>

Filter results by affected user IDs.

--actor-ip <actor_ip>

Filter results by user IP addresses.

--actor-user-id <actor_user_id>

Filter results by actor user IDs.

--actor-username <actor_username>

Filter results by actor usernames.

--event-type <event_type>

Filter results by event types (e.g. `search_issued`, `user_registered`, `user_deactivated`).

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-c, --use-checkpoint <use_checkpoint>

Use a checkpoint with the given name to only get audit-logs that were not previously retrieved. If a checkpoint for audit-logs with the given name doesn't exist, it will be created on the first run. Subsequent CLI runs with this flag and the same name will use the stored checkpoint to modify the search query and then update the stored checkpoint

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.1.3 send-to

Send audit log events to the given server address in JSON format.

HOSTNAME format: address:port where port is optional and defaults to 514.

Usage

```
audit-logs send-to [OPTIONS] HOSTNAME
```

Options

-b, --begin <begin>

The beginning of the date range in which to look for audit-logs. Accepts a date/time in yyyy-MM-dd (UTC) or yyyy-MM-dd HH:MM:SS (UTC+24-hr time) format where the 'time' portion of the string can be partial (e.g. '2020-01-01 12' or '2020-01-01 01:15') or a 'short time' value representing days (30d), hours (24h) or minutes (15m) from the current time. [required unless `--use-checkpoint` option used]

-e, --end <end>

The end of the date range in which to look for audit-logs, argument format options are the same as `--begin`.

--affected-username <affected_username>

Filter results by affected usernames.

--affected-user-id <affected_user_id>

Filter results by affected user IDs.

--actor-ip <actor_ip>

Filter results by user IP addresses.

--actor-user-id <actor_user_id>

Filter results by actor user IDs.

--actor-username <actor_username>

Filter results by actor usernames.

--event-type <event_type>

Filter results by event types (e.g. `search_issued`, `user_registered`, `user_deactivated`).

-c, --use-checkpoint <use_checkpoint>

Use a checkpoint with the given name to only get audit-logs that were not previously retrieved. If a checkpoint for audit-logs with the given name doesn't exist, it will be created on the first run. Subsequent CLI runs with this flag and the same name will use the stored checkpoint to modify the search query and then update the stored checkpoint

--ignore-cert-validation

Set to skip CA certificate validation. Incompatible with the 'certs' option.

--certs <certs>

A CA certificates-chain file for the TCP-TLS protocol.

-p, --protocol <protocol>

Protocol used to send logs to server. Use TCP-TLS for additional security. Defaults to UDP.

Options

TCP | UDP | TLS-TCP

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

HOSTNAME

Required argument

2.2 devices

Manage devices within your CrashPlan environment.

Usage

```
devices [OPTIONS] COMMAND [ARGS]...
```

2.2.1 bulk

Tools for managing devices in bulk.

Usage

```
devices bulk [OPTIONS] COMMAND [ARGS]...
```

deactivate

Deactivate all devices from the provided CSV containing a 'guid' column.

Usage

```
devices bulk deactivate [OPTIONS] CSV_FILE
```

Options

--change-device-name

Prepend 'deactivated_<current_date>' to the name of any successfully deactivated devices.

--purge-date <purge_date>

The date on which the archive should be purged from cold storage in yyyy-MM-dd format. If not provided, the date will be set according to the appropriate organization settings.

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments**CSV_FILE**

Required argument

generate-template

Generate the CSV template needed for bulk device commands.

Usage

```
devices bulk generate-template [OPTIONS] {reactivate|deactivate|rename}
```

Options**-p, --path <path>**

Write template file to specific file path/name.

Arguments**CMD**

Required argument

reactivate

Reactivate all devices from the provided CSV containing a 'guid' column.

Usage

```
devices bulk reactivate [OPTIONS] CSV_FILE
```

Options**-f, --format <format>**

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

rename

Rename all devices from the provided CSV containing a 'guid' and a 'name' column.

Usage

```
devices bulk rename [OPTIONS] CSV_FILE
```

Options

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

2.2.2 deactivate

Deactivate a device within crashplan. Requires the device GUID to deactivate.

Usage

```
devices deactivate [OPTIONS] DEVICE_GUID
```

Options

--change-device-name

Prepend 'deactivated_<current_date>' to the name of the device if deactivation is successful.

--purge-date <purge_date>

The date on which the archive should be purged from cold storage in yyyy-MM-dd format. If not provided, the date will be set according to the appropriate organization settings.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

DEVICE_GUID

Required argument

2.2.3 list

Get information about many devices.

Usage

```
devices list [OPTIONS]
```

Options

--active

Limits results to only active devices.

--inactive

Limits results to only deactivated devices.

--org-uid <org_uid>

Limit devices to only those in the organization you specify. Note that child organizations will be included.

--include-backup-usage

Return backup usage information for each device (may significantly lengthen the size of the return).

--include-usernames

Add the username associated with a device to the output.

--include-settings

Include device settings in output.

--include-legal-hold-membership

Include legal hold membership in output.

--include-total-storage

Include backup archive count and total storage in output.

--exclude-most-recently-connected <exclude_most_recently_connected>

Filter out the N most recently connected devices per user. Useful for identifying duplicate and/or replaced devices that are no longer needed across an environment. If a user has 2 devices and N=1, the one device with the most recent 'lastConnected' date will not show up in the result list.

--last-connected-before <last_connected_before>

Include devices only when the 'lastConnected' field is after the provided value. Accepts a date/time in yyyy-MM-dd (UTC) or yyyy-MM-dd HH:MM:SS (UTC+24-hr time) format where the 'time' portion of the string can be partial (e.g. '2020-01-01 12' or '2020-01-01 01:15') or a 'short time' value representing days (30d), hours (24h) or minutes (15m) from the current time.

--last-connected-after <last_connected_after>

Include devices only when 'lastConnected' field is after the provided value. Argument format options are the same as --last-connected-before.

--created-before <created_before>

Include devices only when 'creationDate' field is less than the provided value. Argument format options are the same as --last-connected-before.

--created-after <created_after>

Include devices only when 'creationDate' field is greater than the provided value. Argument format options are the same as --last-connected-before.

--page-size <page_size>

Number of devices to retrieve per API call. Lower this value if you are getting timeouts when retrieving devices with backup info. Default: 100

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.2.4 list-backup-sets

Get information about many devices and their backup sets.

Usage

```
devices list-backup-sets [OPTIONS]
```

Options

--active

Limits results to only active devices.

--inactive

Limits results to only deactivated devices.

--org-uid <org_uid>

Limit devices to only those in the organization you specify. Note that child organizations will be included.

--include-usernames

Add the username associated with a device to the output.

--page-size <page_size>

Number of devices to retrieve per API call. Lower this value if you are getting timeouts when retrieving devices with backup info. Default: 100

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.2.5 reactivate

Reactivate a device within crashplan. Requires the device GUID to reactivate.

Usage

```
devices reactivate [OPTIONS] DEVICE_GUID
```

Options

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

DEVICE_GUID

Required argument

2.2.6 rename

Rename a device within crashplan. Requires the device GUID to rename.

Usage

```
devices rename [OPTIONS] DEVICE_GUID
```

Options

-n, --new-device-name <new_device_name>

Required The new name for the device.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

DEVICE_GUID

Required argument

2.2.7 show

Print individual device details. Requires device GUID.

Usage

```
devices show [OPTIONS] DEVICE_GUID
```

Options

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

DEVICE_GUID

Required argument

2.3 legal-hold

Add and remove custodians from legal hold matters.

Usage

```
legal-hold [OPTIONS] COMMAND [ARGS]...
```

2.3.1 add-user

Add a custodian to a legal hold matter.

Usage

```
legal-hold add-user [OPTIONS]
```

Options

-m, --matter-id <matter_id>

Required Identification number of the legal hold matter the custodian will be added to.

-u, --username <username>

Required The username of the custodian to add to the matter.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.3.2 bulk

Tools for executing bulk legal hold actions.

Usage

```
legal-hold bulk [OPTIONS] COMMAND [ARGS]...
```

add

Bulk add custodians to legal hold matters using a CSV file. CSV file format: matter_id,username

Usage

```
legal-hold bulk add [OPTIONS] CSV_FILE
```

Options

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

generate-template

Generate the CSV template needed for bulk adding/removing users.

Usage

```
legal-hold bulk generate-template [OPTIONS] {add|remove}
```

Options

-p, --path <path>

Write template file to specific file path/name.

Arguments

CMD

Required argument

remove

Bulk release custodians from legal hold matters using a CSV file. CSV file format: matter_id,username

Usage

```
legal-hold bulk remove [OPTIONS] CSV_FILE
```

Options

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

2.3.3 list

Fetch existing legal hold matters.

Usage

```
legal-hold list [OPTIONS]
```

Options

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.3.4 remove-user

Release a custodian from a legal hold matter.

Usage

```
legal-hold remove-user [OPTIONS]
```

Options

-m, --matter-id <matter_id>

Required Identification number of the legal hold matter the custodian will be removed from.

-u, --username <username>

Required The username of the custodian to add to the matter.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.3.5 search-events

Tools for getting legal hold event data.

Usage

```
legal-hold search-events [OPTIONS]
```

Options

-m, --matter-id <matter_id>

Required Filter results by legal hold UID.

--event-type <event_type>

Filter results by event types.

Options

```
MEMBERSHIP_CREATED | MEMBERSHIP_REACTIVATED | MEMBER-
SHIP_DEACTIVATED | HOLD_CREATED | HOLD_DEACTIVATED |
HOLD_REACTIVATED | RESTORE
```

--begin <begin>

The beginning of the date range in which to look for legal hold events. Accepts a date/time in yyyy-MM-dd (UTC) or yyyy-MM-dd HH:MM:SS (UTC+24-hr time) format where the 'time' portion of the string can be partial (e.g. '2020-01-01 12' or '2020-01-01 01:15') or a 'short time' value representing days (30d), hours (24h) or minutes (15m) from the current time.

--end <end>

The end of the date range in which to look for legal hold events, argument format options are the same as *--begin*.

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.3.6 show

Display details of a given legal hold matter.

Usage

```
legal-hold show [OPTIONS] MATTER_ID
```

Options

--include-inactive

View all custodians associated with the legal hold matter, including inactive custodians.

--include-policy

View details of the preservation policy associated with the legal hold matter.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

MATTER_ID

Required argument

2.4 profile

Manage CrashPlan connection settings.

Usage

```
profile [OPTIONS] COMMAND [ARGS]...
```

2.4.1 create

Create a profile with username/password authentication. The first profile created will be the default.

Usage

```
profile create [OPTIONS]
```

Options

-n, --name <name>

Required The name of the CrashPlan CLI profile to use when executing this command.

-s, --server <server>

Required The URL you use to sign into crashplan.

-u, --username <username>

Required The username of the CrashPlan API user.

--password <password>

The password for the CrashPlan API user. If this option is omitted, interactive prompts will be used to obtain the password.

--totp <totp>

TOTP token for multi-factor authentication.

--disable-ssl-errors <disable_ssl_errors>

For development purposes, do not validate the SSL certificates of CrashPlan servers. This is not recommended, except for specific scenarios like testing. Attach this flag to the update command to toggle the setting.

-d, --debug

Turn on debug logging.

2.4.2 create-api-client

Create a profile with CrashPlan API client authentication. The first profile created will be the default.

Usage

```
profile create-api-client [OPTIONS]
```

Options

-n, --name <name>

Required The name of the CrashPlan CLI profile to use when executing this command.

-s, --server <server>

Required The URL you use to sign into crashplan.

--api-client-id <api_client_id>

Required The API client key for API client authentication. Used with the *--secret* option.

--secret <secret>

Required The API secret for API client authentication. Used with the *--api-client* option.

--disable-ssl-errors <disable_ssl_errors>

For development purposes, do not validate the SSL certificates of CrashPlan servers. This is not recommended, except for specific scenarios like testing. Attach this flag to the update command to toggle the setting.

-d, --debug

Turn on debug logging.

2.4.3 delete

Deletes a profile and its stored password (if any).

Usage

```
profile delete [OPTIONS] PROFILE_NAME
```

Options

-y, --assume-yes

Assume “yes” as the answer to all prompts and run non-interactively.

Arguments

PROFILE_NAME

Required argument

2.4.4 delete-all

Deletes all profiles and saved passwords (if any).

Usage

```
profile delete-all [OPTIONS]
```

Options

-y, --assume-yes

Assume “yes” as the answer to all prompts and run non-interactively.

2.4.5 list

Show all existing stored profiles.

Usage

```
profile list [OPTIONS]
```

2.4.6 reset-pw

Change the stored password for a profile. Only affects what's stored in the local profile, does not make any changes to the CrashPlan user account.

Usage

```
profile reset-pw [OPTIONS] [PROFILE_NAME]
```

Options

-d, --debug

Turn on debug logging.

Arguments

PROFILE_NAME

Optional argument

2.4.7 show

Print the details of a profile.

Usage

```
profile show [OPTIONS] [PROFILE_NAME]
```

Arguments

PROFILE_NAME

Optional argument

2.4.8 update

Update an existing profile.

Usage

```
profile update [OPTIONS]
```

Options

-n, --name <name>

The name of the CrashPlan CLI profile to use when executing this command.

-s, --server <server>

The URL you use to sign into crashplan.

--api-client-id <api_client_id>

The API client key for API client authentication. Used with the *--secret* option.

--secret <secret>

The API secret for API client authentication. Used with the *--api-client* option.

-u, --username <username>

The username of the CrashPlan API user.

--password <password>

The password for the CrashPlan API user. If this option is omitted, interactive prompts will be used to obtain the password.

--totp <totp>

TOTP token for multi-factor authentication.

--disable-ssl-errors <disable_ssl_errors>

For development purposes, do not validate the SSL certificates of CrashPlan servers. This is not recommended, except for specific scenarios like testing. Attach this flag to the update command to toggle the setting.

-d, --debug

Turn on debug logging.

2.4.9 use

Set a profile as the default. If not providing a profile-name, prompts for a choice from a list of all profiles.

Usage

```
profile use [OPTIONS] [PROFILE_NAME]
```

Arguments

PROFILE_NAME

Optional argument

2.5 users

Manage users within your CrashPlan environment.

Usage

```
users [OPTIONS] COMMAND [ARGS]...
```

2.5.1 add-role

Add the specified role to the user with the specified username.

Usage

```
users add-role [OPTIONS]
```

Options

- username** <username>
Username of the target user.
- role-name** <role_name>
Name of role to add.
- d, --debug**
Turn on debug logging.
- totp** <totp>
TOTP token for multi-factor authentication.
- profile** <profile>
The name of the CrashPlan CLI profile to use when executing this command.

2.5.2 bulk

Tools for managing users in bulk.

Usage

```
users bulk [OPTIONS] COMMAND [ARGS]...
```

add-roles

Add roles to a list of users from the provided CSV in format: username,role_name

Usage

```
users bulk add-roles [OPTIONS] CSV_FILE
```

Options

- f, --format** <format>
The output format of the result. Defaults to table format.
- Options**
TABLE | CSV | JSON | RAW-JSON
- d, --debug**
Turn on debug logging.
- totp** <totp>
TOTP token for multi-factor authentication.
- profile** <profile>
The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

deactivate

Deactivate a list of users from the provided CSV in format: username

Usage

```
users bulk deactivate [OPTIONS] CSV_FILE
```

Options

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

--purge-date <purge_date>

The date on which all a users archives should be purged from cold storage in yyyy-MM-dd format. If not provided, the date will be set according to the appropriate organization settings.

Arguments

CSV_FILE

Required argument

generate-template

Generate the CSV template needed for bulk user commands.

Usage

```
users bulk generate-template [OPTIONS] {update|move}
```

Options

-p, --path <path>

Write template file to specific file path/name.

Arguments

CMD

Required argument

move

Change the organization of the list of users from the provided CSV in format: username,org_id

Usage

```
users bulk move [OPTIONS] CSV_FILE
```

Options

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

reactivate

Reactivate a list of users from the provided CSV in format: username

Usage

```
users bulk reactivate [OPTIONS] CSV_FILE
```

Options

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

remove-roles

Remove roles from a list of users from the provided CSV in format: username,role_name

Usage

```
users bulk remove-roles [OPTIONS] CSV_FILE
```

Options

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

CSV_FILE

Required argument

update

Update a list of users from the provided CSV in format: user_uid,username,email,password,first_name,last_name,notes,archive_size_quote

Usage

```
users bulk update [OPTIONS] CSV_FILE
```

Options

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments**CSV_FILE**

Required argument

2.5.3 deactivate

Deactivate a user.

Usage

```
users deactivate [OPTIONS] USERNAME
```

Options**--purge-date <purge_date>**

The date on which all a users archives should be purged from cold storage in yyyy-MM-dd format. If not provided, the date will be set according to the appropriate organization settings.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments**USERNAME**

Required argument

2.5.4 list

List users in your CrashPlan environment.

Usage

```
users list [OPTIONS]
```

Options**--org-uid <org_uid>**

Limit users to only those in the organization you specify. Note that child orgs are included.

--role-name <role_name>

Limit results to only users having the specified role.

--active

Limits results to only active users.

--inactive

Limits results to only deactivated users.

--include-legal-hold-membership

Include legal hold membership in output.

--include-roles

Include user roles.

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.5.5 move

Change the organization of the user with the given username to the org with the given org_id, or org_uid.

Usage

```
users move [OPTIONS]
```

Options

--username <username>

Required The username of the user to move.

--org-id <org_id>

Required The unique identifier (UID) for the organization to which the user will be moved.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.5.6 orgs

Tools for viewing user orgs.

Usage

```
users orgs [OPTIONS] COMMAND [ARGS]...
```

list

List all orgs.

Usage

```
users orgs list [OPTIONS]
```

Options

-f, --format <format>
The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug
Turn on debug logging.

--totp <totp>
TOTP token for multi-factor authentication.

--profile <profile>
The name of the CrashPlan CLI profile to use when executing this command.

show

Show org details.

Usage

```
users orgs show [OPTIONS] ORG_UID
```

Options

-f, --format <format>
The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug
Turn on debug logging.

--totp <totp>
TOTP token for multi-factor authentication.

--profile <profile>
The name of the CrashPlan CLI profile to use when executing this command.

Arguments

ORG_UID

Required argument

2.5.7 reactivate

Reactivate a user.

Usage

```
users reactivate [OPTIONS] USERNAME
```

Options

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

USERNAME

Required argument

2.5.8 remove-role

Remove the specified role to the user with the specified username.

Usage

```
users remove-role [OPTIONS]
```

Options

--role-name <role_name>

Name of role to remove.

--username <username>

Username of the target user.

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

2.5.9 show

Show user details.

Usage

```
users show [OPTIONS] USERNAME
```

Options

--include-legal-hold-membership

Include legal hold membership in output.

-f, --format <format>

The output format of the result. Defaults to table format.

Options

TABLE | CSV | JSON | RAW-JSON

-d, --debug

Turn on debug logging.

--totp <totp>

TOTP token for multi-factor authentication.

--profile <profile>

The name of the CrashPlan CLI profile to use when executing this command.

Arguments

USERNAME

Required argument

2.5.10 update

Update a user with the specified unique identifier.

Usage

```
users update [OPTIONS]
```

Options

--user-uid <user_uid>

Required The unique identifier of the user to be modified.

--username <username>

The new username for the user.

--password <password>

The new password for the user.

--email <email>

The new email for the user.

--first-name <first_name>
The new first name for the user.

--last-name <last_name>
The new last name for the user.

--notes <notes>
Notes about this user.

--archive-size-quota <archive_size_quota>
The total size (in bytes) allowed for this user.

-d, --debug
Turn on debug logging.

--totp <totp>
TOTP token for multi-factor authentication.

--profile <profile>
The name of the CrashPlan CLI profile to use when executing this command.

- *Audit Logs*
- *Devices*
- *Legal Hold*
- *Profile*
- *Users*

The CrashPlan command-line interface (CLI) tool offers a way to interact with your CrashPlan environment without using the CrashPlan console or making API calls directly. For example, you can use it to extract CrashPlan audit log data or to manage users, devices, and legal holds.

REQUIREMENTS

To use the CrashPlan CLI, you must have:

- A [CrashPlan product plan](#) that supports the feature or functionality for your use case
- Python version 3.11 and later installed

CONTENT

- *User Guides*
 - *Get started with the CrashPlan command-line interface (CLI)*
 - *Configure a profile*
 - *Manage legal hold users*
 - *Clean up your environment by deactivating devices*
 - *Manage users*
 - *Perform bulk actions*
 - *Send audit logs to a SIEM*
 - *Write custom extension scripts using the CrashPlan CLI and Pycpg*
- *Commands*
 - *Audit Logs*
 - *Devices*
 - *Legal Hold*
 - *Profile*
 - *Users*

Symbols

- active
 - devices-list command line option, [21](#)
 - devices-list-backup-sets command line option, [22](#)
 - users-list command line option, [38](#)
- actor-ip
 - audit-logs-search command line option, [16](#)
 - audit-logs-send-to command line option, [17](#)
- actor-user-id
 - audit-logs-search command line option, [16](#)
 - audit-logs-send-to command line option, [17](#)
- actor-username
 - audit-logs-search command line option, [16](#)
 - audit-logs-send-to command line option, [17](#)
- affected-user-id
 - audit-logs-search command line option, [16](#)
 - audit-logs-send-to command line option, [17](#)
- affected-username
 - audit-logs-search command line option, [16](#)
 - audit-logs-send-to command line option, [17](#)
- api-client-id
 - profile-create-api-client command line option, [30](#)
 - profile-update command line option, [31](#)
- archive-size-quota
 - users-update command line option, [42](#)
- assume-yes
 - profile-delete command line option, [30](#)
 - profile-delete-all command line option, [30](#)
- begin
 - audit-logs-search command line option, [16](#)
 - audit-logs-send-to command line option, [17](#)
 - legal-hold-search-events command line option, [27](#)
- certs
 - audit-logs-send-to command line option, [17](#)
- change-device-name
 - devices-bulk-deactivate command line option, [18](#)
 - devices-deactivate command line option, [20](#)
- created-after
 - devices-list command line option, [22](#)
- created-before
 - devices-list command line option, [22](#)
- debug
 - audit-logs-clear-checkpoint command line option, [15](#)
 - audit-logs-search command line option, [16](#)
 - audit-logs-send-to command line option, [17](#)
 - devices-bulk-deactivate command line option, [18](#)
 - devices-bulk-reactivate command line option, [19](#)
 - devices-bulk-rename command line option, [20](#)
 - devices-deactivate command line option, [20](#)
 - devices-list command line option, [22](#)
 - devices-list-backup-sets command line option, [23](#)
 - devices-reactivate command line option, [23](#)
 - devices-rename command line option, [23](#)
 - devices-show command line option, [24](#)
 - legal-hold-add-user command line option, [25](#)
 - legal-hold-bulk-add command line option, [25](#)
 - legal-hold-bulk-remove command line option, [26](#)
 - legal-hold-list command line option, [26](#)
 - legal-hold-remove-user command line option, [27](#)

```

legal-hold-search-events command line
    option, 28
legal-hold-show command line option, 28
profile-create command line option, 29
profile-create-api-client command line
    option, 30
profile-reset-pw command line option, 31
profile-update command line option, 32
users-add-role command line option, 33
users-bulk-add-roles command line
    option, 33
users-bulk-deactivate command line
    option, 34
users-bulk-move command line option, 35
users-bulk-reactivate command line
    option, 35
users-bulk-remove-roles command line
    option, 36
users-bulk-update command line option, 36
users-deactivate command line option, 37
users-list command line option, 38
users-move command line option, 38
users-orgs-list command line option, 39
users-orgs-show command line option, 39
users-reactivate command line option, 40
users-remove-role command line option, 40
users-show command line option, 41
users-update command line option, 42
--disable-ssl-errors
    profile-create command line option, 29
    profile-create-api-client command line
        option, 30
    profile-update command line option, 32
--email
    users-update command line option, 41
--end
    audit-logs-search command line option, 16
    audit-logs-send-to command line option,
        17
    legal-hold-search-events command line
        option, 28
--event-type
    audit-logs-search command line option, 16
    audit-logs-send-to command line option,
        17
    legal-hold-search-events command line
        option, 27
--exclude-most-recently-connected
    devices-list command line option, 21
--first-name
    users-update command line option, 41
--format
    audit-logs-search command line option, 16
    devices-bulk-deactivate command line
        option, 18
    devices-bulk-reactivate command line
        option, 19
    devices-bulk-rename command line option,
        20
    devices-list command line option, 22
    devices-list-backup-sets command line
        option, 22
    legal-hold-list command line option, 26
    legal-hold-search-events command line
        option, 28
    users-bulk-add-roles command line
        option, 33
    users-bulk-deactivate command line
        option, 34
    users-bulk-move command line option, 35
    users-bulk-reactivate command line
        option, 35
    users-bulk-remove-roles command line
        option, 36
    users-bulk-update command line option, 36
    users-list command line option, 38
    users-orgs-list command line option, 39
    users-orgs-show command line option, 39
    users-show command line option, 41
--ignore-cert-validation
    audit-logs-send-to command line option,
        17
--inactive
    devices-list command line option, 21
    devices-list-backup-sets command line
        option, 22
    users-list command line option, 38
--include-backup-usage
    devices-list command line option, 21
--include-inactive
    legal-hold-show command line option, 28
--include-legal-hold-membership
    devices-list command line option, 21
    users-list command line option, 38
    users-show command line option, 41
--include-policy
    legal-hold-show command line option, 28
--include-roles
    users-list command line option, 38
--include-settings
    devices-list command line option, 21
--include-total-storage
    devices-list command line option, 21
--include-usernames
    devices-list command line option, 21
    devices-list-backup-sets command line
        option, 22

```

--last-connected-after
 devices-list command line option, 21
 --last-connected-before
 devices-list command line option, 21
 --last-name
 users-update command line option, 42
 --matter-id
 legal-hold-add-user command line option, 25
 legal-hold-remove-user command line option, 27
 legal-hold-search-events command line option, 27
 --name
 profile-create command line option, 29
 profile-create-api-client command line option, 29
 profile-update command line option, 31
 --new-device-name
 devices-rename command line option, 23
 --notes
 users-update command line option, 42
 --org-id
 users-move command line option, 38
 --org-uid
 devices-list command line option, 21
 devices-list-backup-sets command line option, 22
 users-list command line option, 37
 --page-size
 devices-list command line option, 22
 devices-list-backup-sets command line option, 22
 --password
 profile-create command line option, 29
 profile-update command line option, 32
 users-update command line option, 41
 --path
 devices-bulk-generate-template command line option, 19
 legal-hold-bulk-generate-template command line option, 26
 users-bulk-generate-template command line option, 34
 --profile
 audit-logs-clear-checkpoint command line option, 15
 audit-logs-search command line option, 16
 audit-logs-send-to command line option, 18
 devices-bulk-deactivate command line option, 19
 devices-bulk-reactivate command line option, 19
 devices-bulk-rename command line option, 20
 devices-deactivate command line option, 21
 devices-list command line option, 22
 devices-list-backup-sets command line option, 23
 devices-reactivate command line option, 23
 devices-rename command line option, 24
 devices-show command line option, 24
 legal-hold-add-user command line option, 25
 legal-hold-bulk-add command line option, 25
 legal-hold-bulk-remove command line option, 26
 legal-hold-list command line option, 27
 legal-hold-remove-user command line option, 27
 legal-hold-search-events command line option, 28
 legal-hold-show command line option, 28
 users-add-role command line option, 33
 users-bulk-add-roles command line option, 33
 users-bulk-deactivate command line option, 34
 users-bulk-move command line option, 35
 users-bulk-reactivate command line option, 35
 users-bulk-remove-roles command line option, 36
 users-bulk-update command line option, 37
 users-deactivate command line option, 37
 users-list command line option, 38
 users-move command line option, 38
 users-orgs-list command line option, 39
 users-orgs-show command line option, 39
 users-reactivate command line option, 40
 users-remove-role command line option, 40
 users-show command line option, 41
 users-update command line option, 42
 --protocol
 audit-logs-send-to command line option, 17
 --purge-date
 devices-bulk-deactivate command line option, 18
 devices-deactivate command line option, 20
 users-bulk-deactivate command line option, 34
 users-deactivate command line option, 37

```

--role-name
    users-add-role command line option, 33
    users-list command line option, 37
    users-remove-role command line option, 40
--secret
    profile-create-api-client command line
        option, 30
    profile-update command line option, 32
--server
    profile-create command line option, 29
    profile-create-api-client command line
        option, 29
    profile-update command line option, 31
--totp
    audit-logs-clear-checkpoint command
        line option, 15
    audit-logs-search command line option, 16
    audit-logs-send-to command line option,
        18
    devices-bulk-deactivate command line
        option, 19
    devices-bulk-reactivate command line
        option, 19
    devices-bulk-rename command line option,
        20
    devices-deactivate command line option,
        20
    devices-list command line option, 22
    devices-list-backup-sets command line
        option, 23
    devices-reactivate command line option,
        23
    devices-rename command line option, 23
    devices-show command line option, 24
    legal-hold-add-user command line option,
        25
    legal-hold-bulk-add command line option,
        25
    legal-hold-bulk-remove command line
        option, 26
    legal-hold-list command line option, 27
    legal-hold-remove-user command line
        option, 27
    legal-hold-search-events command line
        option, 28
    legal-hold-show command line option, 28
    profile-create command line option, 29
    profile-update command line option, 32
    users-add-role command line option, 33
    users-bulk-add-roles command line
        option, 33
    users-bulk-deactivate command line
        option, 34
    users-bulk-move command line option, 35
    users-bulk-reactivate command line
        option, 35
    users-bulk-remove-roles command line
        option, 36
    users-bulk-update command line option, 37
    users-deactivate command line option, 37
    users-list command line option, 38
    users-move command line option, 38
    users-orgs-list command line option, 39
    users-orgs-show command line option, 39
    users-reactivate command line option, 40
    users-remove-role command line option, 40
    users-show command line option, 41
    users-update command line option, 42
--use-checkpoint
    audit-logs-search command line option, 16
    audit-logs-send-to command line option,
        17
--user-uid
    users-update command line option, 41
--username
    legal-hold-add-user command line option,
        25
    legal-hold-remove-user command line
        option, 27
    profile-create command line option, 29
    profile-update command line option, 32
    users-add-role command line option, 33
    users-move command line option, 38
    users-remove-role command line option, 40
    users-update command line option, 41
-b
    audit-logs-search command line option, 16
    audit-logs-send-to command line option,
        17
-c
    audit-logs-search command line option, 16
    audit-logs-send-to command line option,
        17
-d
    audit-logs-clear-checkpoint command
        line option, 15
    audit-logs-search command line option, 16
    audit-logs-send-to command line option,
        17
    devices-bulk-deactivate command line
        option, 18
    devices-bulk-reactivate command line
        option, 19
    devices-bulk-rename command line option,
        20
    devices-deactivate command line option,
        20
    devices-list command line option, 22

```

- devices-list-backup-sets command line option, 23
- devices-reactivate command line option, 23
- devices-rename command line option, 23
- devices-show command line option, 24
- legal-hold-add-user command line option, 25
- legal-hold-bulk-add command line option, 25
- legal-hold-bulk-remove command line option, 26
- legal-hold-list command line option, 26
- legal-hold-remove-user command line option, 27
- legal-hold-search-events command line option, 28
- legal-hold-show command line option, 28
- profile-create command line option, 29
- profile-create-api-client command line option, 30
- profile-reset-pw command line option, 31
- profile-update command line option, 32
- users-add-role command line option, 33
- users-bulk-add-roles command line option, 33
- users-bulk-deactivate command line option, 34
- users-bulk-move command line option, 35
- users-bulk-reactivate command line option, 35
- users-bulk-remove-roles command line option, 36
- users-bulk-update command line option, 36
- users-deactivate command line option, 37
- users-list command line option, 38
- users-move command line option, 38
- users-orgs-list command line option, 39
- users-orgs-show command line option, 39
- users-reactivate command line option, 40
- users-remove-role command line option, 40
- users-show command line option, 41
- users-update command line option, 42
- e
 - audit-logs-search command line option, 16
 - audit-logs-send-to command line option, 17
- f
 - audit-logs-search command line option, 16
 - devices-bulk-deactivate command line option, 18
 - devices-bulk-reactivate command line option, 19
 - devices-bulk-rename command line option, 20
 - devices-list command line option, 22
 - devices-list-backup-sets command line option, 22
 - legal-hold-list command line option, 26
 - legal-hold-search-events command line option, 28
 - users-bulk-add-roles command line option, 33
 - users-bulk-deactivate command line option, 34
 - users-bulk-move command line option, 35
 - users-bulk-reactivate command line option, 35
 - users-bulk-remove-roles command line option, 36
 - users-bulk-update command line option, 36
 - users-list command line option, 38
 - users-orgs-list command line option, 39
 - users-orgs-show command line option, 39
 - users-show command line option, 41
- m
 - legal-hold-add-user command line option, 25
 - legal-hold-remove-user command line option, 27
 - legal-hold-search-events command line option, 27
- n
 - devices-rename command line option, 23
 - profile-create command line option, 29
 - profile-create-api-client command line option, 29
 - profile-update command line option, 31
- p
 - audit-logs-send-to command line option, 17
 - devices-bulk-generate-template command line option, 19
 - legal-hold-bulk-generate-template command line option, 26
 - users-bulk-generate-template command line option, 34
- s
 - profile-create command line option, 29
 - profile-create-api-client command line option, 29
 - profile-update command line option, 31
- u
 - legal-hold-add-user command line option, 25
 - legal-hold-remove-user command line option, 27
 - profile-create command line option, 29

profile-update command line option, 32

-y

profile-delete command line option, 30

profile-delete-all command line option, 30

A

audit-logs-clear-checkpoint command line option

- debug, 15
- profile, 15
- totp, 15
- d, 15
- CHECKPOINT_NAME, 15

audit-logs-search command line option

- actor-ip, 16
- actor-user-id, 16
- actor-username, 16
- affected-user-id, 16
- affected-username, 16
- begin, 16
- debug, 16
- end, 16
- event-type, 16
- format, 16
- profile, 16
- totp, 16
- use-checkpoint, 16
- b, 16
- c, 16
- d, 16
- e, 16
- f, 16

audit-logs-send-to command line option

- actor-ip, 17
- actor-user-id, 17
- actor-username, 17
- affected-user-id, 17
- affected-username, 17
- begin, 17
- certs, 17
- debug, 17
- end, 17
- event-type, 17
- ignore-cert-validation, 17
- profile, 18
- protocol, 17
- totp, 18
- use-checkpoint, 17
- b, 17
- c, 17
- d, 17
- e, 17
- p, 17

HOSTNAME, 18

C

CHECKPOINT_NAME

- audit-logs-clear-checkpoint command line option, 15

CMD

- devices-bulk-generate-template command line option, 19
- legal-hold-bulk-generate-template command line option, 26
- users-bulk-generate-template command line option, 35

CSV_FILE

- devices-bulk-deactivate command line option, 19
- devices-bulk-reactivate command line option, 20
- devices-bulk-rename command line option, 20
- legal-hold-bulk-add command line option, 25
- legal-hold-bulk-remove command line option, 26
- users-bulk-add-roles command line option, 34
- users-bulk-deactivate command line option, 34
- users-bulk-move command line option, 35
- users-bulk-reactivate command line option, 36
- users-bulk-remove-roles command line option, 36
- users-bulk-update command line option, 37

D

DEVICE_GUID

- devices-deactivate command line option, 21
- devices-reactivate command line option, 23
- devices-rename command line option, 24
- devices-show command line option, 24

devices-bulk-deactivate command line option

- change-device-name, 18
- debug, 18
- format, 18
- profile, 19
- purge-date, 18
- totp, 19
- d, 18
- f, 18
- CSV_FILE, 19

devices-bulk-generate-template command line option
 --path, 19
 -p, 19
 CMD, 19
 devices-bulk-reactivate command line option
 --debug, 19
 --format, 19
 --profile, 19
 --totp, 19
 -d, 19
 -f, 19
 CSV_FILE, 20
 devices-bulk-rename command line option
 --debug, 20
 --format, 20
 --profile, 20
 --totp, 20
 -d, 20
 -f, 20
 CSV_FILE, 20
 devices-deactivate command line option
 --change-device-name, 20
 --debug, 20
 --profile, 21
 --purge-date, 20
 --totp, 20
 -d, 20
 DEVICE_GUID, 21
 devices-list command line option
 --active, 21
 --created-after, 22
 --created-before, 22
 --debug, 22
 --exclude-most-recently-connected, 21
 --format, 22
 --inactive, 21
 --include-backup-usage, 21
 --include-legal-hold-membership, 21
 --include-settings, 21
 --include-total-storage, 21
 --include-usernames, 21
 --last-connected-after, 21
 --last-connected-before, 21
 --org-uid, 21
 --page-size, 22
 --profile, 22
 --totp, 22
 -d, 22
 -f, 22
 devices-list-backup-sets command line option
 --active, 22
 --debug, 23
 --format, 22
 --inactive, 22
 --include-usernames, 22
 --org-uid, 22
 --page-size, 22
 --profile, 23
 --totp, 23
 -d, 23
 -f, 22
 devices-reactivate command line option
 --debug, 23
 --profile, 23
 --totp, 23
 -d, 23
 DEVICE_GUID, 23
 devices-rename command line option
 --debug, 23
 --new-device-name, 23
 --profile, 24
 --totp, 23
 -d, 23
 -n, 23
 DEVICE_GUID, 24
 devices-show command line option
 --debug, 24
 --profile, 24
 --totp, 24
 -d, 24
 DEVICE_GUID, 24

H

HOSTNAME
 audit-logs-send-to command line option, 18

L

legal-hold-add-user command line option
 --debug, 25
 --matter-id, 25
 --profile, 25
 --totp, 25
 --username, 25
 -d, 25
 -m, 25
 -u, 25
 legal-hold-bulk-add command line option
 --debug, 25
 --profile, 25
 --totp, 25
 -d, 25
 CSV_FILE, 25
 legal-hold-bulk-generate-template command line option
 --path, 26

- p, 26
- CMD, 26
- legal-hold-bulk-remove command line option
 - debug, 26
 - profile, 26
 - totp, 26
 - d, 26
 - CSV_FILE, 26
- legal-hold-list command line option
 - debug, 26
 - format, 26
 - profile, 27
 - totp, 27
 - d, 26
 - f, 26
- legal-hold-remove-user command line option
 - debug, 27
 - matter-id, 27
 - profile, 27
 - totp, 27
 - username, 27
 - d, 27
 - m, 27
 - u, 27
- legal-hold-search-events command line option
 - begin, 27
 - debug, 28
 - end, 28
 - event-type, 27
 - format, 28
 - matter-id, 27
 - profile, 28
 - totp, 28
 - d, 28
 - f, 28
 - m, 27
- legal-hold-show command line option
 - debug, 28
 - include-inactive, 28
 - include-policy, 28
 - profile, 28
 - totp, 28
 - d, 28
 - MATTER_ID, 28

M

MATTER_ID
 legal-hold-show command line option, 28

O

ORG_UID
 users-orgs-show command line option, 40

P

PROFILE_NAME
 profile-delete command line option, 30
 profile-reset-pw command line option, 31
 profile-show command line option, 31
 profile-use command line option, 32

profile-create command line option

- debug, 29
- disable-ssl-errors, 29
- name, 29
- password, 29
- server, 29
- totp, 29
- username, 29
- d, 29
- n, 29
- s, 29
- u, 29

profile-create-api-client command line option

- api-client-id, 30
- debug, 30
- disable-ssl-errors, 30
- name, 29
- secret, 30
- server, 29
- d, 30
- n, 29
- s, 29

profile-delete command line option

- assume-yes, 30
- y, 30
- PROFILE_NAME, 30

profile-delete-all command line option

- assume-yes, 30
- y, 30

profile-reset-pw command line option

- debug, 31
- d, 31
- PROFILE_NAME, 31

profile-show command line option
 PROFILE_NAME, 31

profile-update command line option

- api-client-id, 31
- debug, 32
- disable-ssl-errors, 32
- name, 31
- password, 32
- secret, 32
- server, 31
- totp, 32
- username, 32
- d, 32
- n, 31

-s, 31
 -u, 32
 profile-use command line option
 PROFILE_NAME, 32

U

USERNAME

users-deactivate command line option, 37
 users-reactivate command line option, 40
 users-show command line option, 41
 users-add-role command line option
 --debug, 33
 --profile, 33
 --role-name, 33
 --totp, 33
 --username, 33
 -d, 33
 users-bulk-add-roles command line option
 --debug, 33
 --format, 33
 --profile, 33
 --totp, 33
 -d, 33
 -f, 33
 CSV_FILE, 34
 users-bulk-deactivate command line option
 --debug, 34
 --format, 34
 --profile, 34
 --purge-date, 34
 --totp, 34
 -d, 34
 -f, 34
 CSV_FILE, 34
 users-bulk-generate-template command line
 option
 --path, 34
 -p, 34
 CMD, 35
 users-bulk-move command line option
 --debug, 35
 --format, 35
 --profile, 35
 --totp, 35
 -d, 35
 -f, 35
 CSV_FILE, 35
 users-bulk-reactivate command line option
 --debug, 35
 --format, 35
 --profile, 35
 --totp, 35
 -d, 35
 -f, 35

CSV_FILE, 36
 users-bulk-remove-roles command line option
 --debug, 36
 --format, 36
 --profile, 36
 --totp, 36
 -d, 36
 -f, 36
 CSV_FILE, 36
 users-bulk-update command line option
 --debug, 36
 --format, 36
 --profile, 37
 --totp, 37
 -d, 36
 -f, 36
 CSV_FILE, 37
 users-deactivate command line option
 --debug, 37
 --profile, 37
 --purge-date, 37
 --totp, 37
 -d, 37
 USERNAME, 37
 users-list command line option
 --active, 38
 --debug, 38
 --format, 38
 --inactive, 38
 --include-legal-hold-membership, 38
 --include-roles, 38
 --org-uid, 37
 --profile, 38
 --role-name, 37
 --totp, 38
 -d, 38
 -f, 38
 users-move command line option
 --debug, 38
 --org-id, 38
 --profile, 38
 --totp, 38
 --username, 38
 -d, 38
 users-orgs-list command line option
 --debug, 39
 --format, 39
 --profile, 39
 --totp, 39
 -d, 39
 -f, 39
 users-orgs-show command line option
 --debug, 39
 --format, 39

- profile, 39
- totp, 39
- d, 39
- f, 39
- ORG_UID, 40
- users-reactivate command line option
 - debug, 40
 - profile, 40
 - totp, 40
 - d, 40
 - USERNAME, 40
- users-remove-role command line option
 - debug, 40
 - profile, 40
 - role-name, 40
 - totp, 40
 - username, 40
 - d, 40
- users-show command line option
 - debug, 41
 - format, 41
 - include-legal-hold-membership, 41
 - profile, 41
 - totp, 41
 - d, 41
 - f, 41
 - USERNAME, 41
- users-update command line option
 - archive-size-quota, 42
 - debug, 42
 - email, 41
 - first-name, 41
 - last-name, 42
 - notes, 42
 - password, 41
 - profile, 42
 - totp, 42
 - user-uid, 41
 - username, 41
 - d, 42