

Dell PowerStore

CLI User Guide

Version 4.1

Notes, cautions, and warnings

 **NOTE:** A NOTE indicates important information that helps you make better use of your product.

 **CAUTION:** A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

 **WARNING:** A WARNING indicates a potential for property damage, personal injury, or death.

Additional Resources	4
Chapter 1: Introduction	5
Overview.....	5
Use PowerStore CLI in scripts.....	5
Using CLI versions.....	5
CLI command reference.....	5
Set up the PowerStore CLI client.....	6
Install the PowerStore CLI client.....	6
Launch the PowerStore CLI client.....	6
Certificate verification.....	6
Passwords and special characters.....	7
PowerStore CLI command syntax.....	7
Executable.....	7
Switches.....	7
Object types.....	7
Object qualifiers.....	8
Actions.....	8
Action qualifiers.....	8
Size qualifiers.....	8
Action commands.....	9
The create action command.....	9
The set action command.....	9
The show action command.....	10
The delete action command.....	12
Get help.....	12
Help on the PowerStore CLI client.....	12
Help on object types.....	12
Help on actions.....	13
Manage SSL/TLS certificates.....	14
View the switches.....	14
Access the system.....	15
Display the header.....	16
Save PowerStore CLI settings.....	16

As part of an improvement effort, revisions of the software and hardware are periodically released. Some functions that are described in this document are not supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information about product features. Contact your service provider if a product does not function properly or does not function as described in this document.

 **NOTE:** PowerStore X model customers: For the latest how-to technical manuals and guides for your model, download the *PowerStore 3.2.x Documentation Set* from the PowerStore Documentation page at dell.com/powerstoredocs.

Where to get help

Support, product, and licensing information can be obtained as follows:

- **Product information**—For product and feature documentation or release notes, go to the PowerStore Documentation page at dell.com/powerstoredocs.
- **Troubleshooting**—For information about products, software updates, licensing, and service go to [Dell Support](#) and locate the appropriate product support page.
- **Technical support**—For technical support and service requests, go to [Dell Support](#) and locate the **Service Requests** page. To open a service request, you must have a valid support agreement. Contact your Sales Representative for details about obtaining a valid support agreement or to answer any questions about your account.

Introduction

This document provides a reference for using the PowerStore command line interface (CLI) to work with your system.

Topics:

- [Overview](#)
- [CLI command reference](#)
- [Set up the PowerStore CLI client](#)
- [PowerStore CLI command syntax](#)
- [Action commands](#)
- [Get help](#)
- [Manage SSL/TLS certificates](#)
- [View the switches](#)
- [Access the system](#)
- [Display the header](#)
- [Save PowerStore CLI settings](#)

Overview

The PowerStore Manager CLI enables you to run commands on a system through a prompt from a Microsoft Windows or Linux host. The PowerStore Manager CLI is intended for users who want to interactively manage a PowerStore system, or to use commands in scripts for automating routine tasks.

Use the PowerStore Manager CLI to manage a system. Tasks include:

- Configuring and monitoring the system
- Managing users
- Provisioning storage
- Protecting data
- Controlling host access to storage

Use PowerStore CLI in scripts

Use scripts with PowerStore CLI to automate routine tasks, such as provisioning storage or scheduling snapshots to protect stored data. For example, create a script to create a snapshot of an iSCSI LUN and delete the older snapshots created before it. Customer Support does not provide sample scripts or support for custom scripting.

Using CLI versions

When using the PowerStore CLI, it is recommended that you use the PowerStore CLI version that accompanies each major release of PowerStoreOS. For example, use PowerStore CLI version 3.0 with PowerStoreOS versions 3.0 and 3.2 and use PowerStore CLI version 3.5 with PowerStoreOS version 3.5.

CLI command reference

See the PowerStore CLI Reference Guide at dell.com/powerstoredocs for more information about the commands and associated qualifiers available with PowerStore.

Set up the PowerStore CLI client

You can install and launch the PowerStore CLI client on a Microsoft Windows or Linux system. PowerStore CLI sends commands to the system through the secure HTTPS protocol using the PowerStore REST API.

Install the PowerStore CLI client

About this task

To install the PowerStore CLI client:

Steps

1. Go to your support website.
2. Download the PowerStore CLI client for your operating system.
3. Perform the following actions based on your operating system:
 - On Windows, double-click the installer and follow the prompts. The default installation location is:
 - 64-bit systems: `C:\Program Files\Dell EMC\PowerStore CLI`
 - 32-bit systems: `C:\Program Files (x86)\Dell EMC\PowerStore Manager CLI`

 **NOTE:** Check the box to have the installation directory added to the PATH system variable.

- On Linux, type: `rpm -ihv <filename>`, where `filename` is the name of the installer. The default installation location is: `/opt/dellemc/pstcli-<version>/bin/`, where `<version>` is the version of the client installed.

Launch the PowerStore CLI client

About this task

After installing the PowerStore CLI client, you can launch the client on a Microsoft Windows or Linux system.

To launch the PowerStore CLI client, perform the following in a command prompt based on your operating system:

Steps

1. If you have a Windows operating system, type:
`pstcli.exe`
2. If you have a Linux operating system, type:
`/usr/bin/pstcli`

Certificate verification

In order to establish a secure connection between PowerStore CLI and its backend server, a Public Key infrastructure (PKI) is used. An important component of PKI is certificate verification. Certificate verification provides a way for a user to verify the backend server being contacted.

When PowerStore CLI connects to a server requesting a secure connection, the server sends its identification in the form of a digital certificate. The certificate usually contains the following:

- Server name
- Trusted certificate authority (CA)
- Server's public encryption key.

The PowerStore CLI client can contact the server that issued the certificate (the trusted CA) and confirm the validity of the certificate before proceeding. When the certificate is verified, PowerStore CLI and its backend server will establish the connection and begin to exchange data.

Passwords and special characters

When you use a password for PowerStore CLI commands, it is recommended that you add single quotation marks around the password. These single quotation marks ensure that the system can parse all special characters within the password.

The following example shows the password **IsAnyone\$Home56** used on the command line with quotation marks:

```
pstcli -d 10.10.220.100 -u admin -p 'IsAnyone$Home56' security_config show
```

PowerStore CLI command syntax

The syntax of an example command line is as follows:

```
pstcli <switches>] <object type> [<object qualifier>] <action> [<action qualifiers>
```

Executable

All command lines begin with the executable `pstcli`. If you do not start each command line with `pstcli`, the command fails and you must rerun the command. If you run only `pstcli`, without any switches or commands, the list of switches and their descriptions is displayed.

If you are in session mode, you do not have to include `pstcli` in the command line. Also, you can press TAB after typing the first letter of the switch to autocomplete the switch.

Switches

Use switches to configure PowerStore CLI and connect to a system. Type switches immediately after `pstcli`. When typing more than one switch on the same line, separate each switch with a space. All switches start with a hyphen (-).

[View the switches](#) provides details on all available switches.

Object types

Object types identify the type of entity on which to perform an action, such as a `local_user`, `host`, `volume`, or `cluster`. All actions require an object type. The one exception is the `-help` switch, which can be used without an object type. [Get help](#) explains how to use the `-help` switch.

The object instance on which you perform an action is designated by an identifier that is called an object qualifier, as explained in [Object qualifiers](#).

Example 1

In the following example for creating a user, the object type is `local_user`:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! local_user create -name user1 -password Password789! -role_id operator
```

Example 2

The following example for viewing all user accounts on the system shows that the object type is also `local_user`. An object identifier is not specified, so the `show` action displays a list of all user accounts:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! local_user show
```

Object qualifiers

Object qualifiers are unique identifiers for objects on the system. Object qualifiers are displayed in the following format:

- **-id** <value>
- **-name** <value>

All object types have identifiers that can be used as qualifiers to specify a unique object instance. Some object types have names. For object types that have names, the name qualifier can be used as an alternate to the ID qualifier.

When you create an object, such as a user or network interface, it receives an ID, which is the primary object qualifier for that object. The uniqueness of the ID is only guaranteed in the scope of the specified object type. When performing actions such as viewing, modifying, or deleting an object, you specify an object qualifier. All object qualifiers start with a hyphen (-).

Example

In the following example for changing the password of a user account, the object type is `local_user`, and the qualifier that is used is **-name**, with a value of `operator`:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! local_user -name operator set -password NewPassword123! -current_password MyPassword456!
```

Actions

Actions are the operations that are performed on an object instance or object type, including creating, modifying, viewing, and deleting. Actions are always required. [Action commands](#) provides details on each of the action commands.

Example

In the following example for changing the password of a user account, the action is `set`:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! local_user -name operator set -password NewPassword456! -current_password OldPassword456!
```

Action qualifiers

Action qualifiers are parameters specific to actions, such as attributes or settings to modify when changing an object. All action qualifiers start with a hyphen (-).

Example

In the following example for changing a role and password for a user account, the action qualifiers are `-password`, `-current_password`, and `-role`:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! local_user -name lab3_operator set -password NewPassword123! -current_password MyPassword456! -role administrator
```

Size qualifiers

Use size qualifiers to indicate a specific capacity-size value. In this case, 1 KB equals 1024 bytes.

To specify a fraction, use a period. For example, type 2.4T for 2.4 terabytes. The output for a size value displays the exact number of bytes and the specified size value:

```
Size = 1209462790557 (1.1TB)
```

The following table lists the size qualifiers. The qualifiers are case-sensitive.

Table 1. Size qualifiers

Qualifier	Measurement
K	Kilobyte
M	Megabyte
G	Gigabyte
T	Terabyte
P	Petabyte

Action commands

When using PowerStore CLI, there are four primary action commands that you can perform on object types or objects: creating, modifying or configuring, viewing, and deleting. This section explains each of these four action commands. [PowerStore CLI command syntax](#) explains the relationship between action commands, object types, and objects.

The create action command

The `create` action command creates an object on the system that is based on the specified object type. If the command is successful, the new object receives an object qualifier, or ID, that identifies the object on the system.

Format

```
<object type> create [<object qualifiers>]
```

Example

The following example uses the `create` action command to create a volume. The new volume receives the ID 12345:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! -header volume create -name test -size 819200000
```

 **NOTE:** The following output shows the header because the `-header` switch is included in the command. By default, the header is not displayed:

```
PowerStore system: 10.0.0.1
Timezone:         Eastern Daylight Time (UTC-04:00)
User:             admin

Created
1:      id = 12345
```

The set action command

The `set` action command modifies an object instance that is specified by object type and object qualifier. If the object identified by the object qualifier does not exist, an error results.

Format

```
<object type> [<object qualifiers>] set [<action qualifiers>]
```

Example

The following example uses the `set` action command to change the name of a volume:

```
psstcli -d 10.0.0.1 -u admin -p MyPassword456! volume -name old_name set -name new_name
```

The following output should be displayed:

```
Success.
```

The show action command

The `show` action command displays a list of objects that exist on the system and the attributes of those objects. You can specify an object qualifier to view the attributes for a single object. The `show` action command provides qualifiers for changing the display of the output, including the format and the attributes to include. The available output formats are name-value pair (NVP), table, JSON, and comma-separated values (CSV).

Format

```
<object> [<object qualifiers>] show [-select <field>,...] [-sort <sort field>(+|-)] [-offset <value>] [{ -limit <value> | -all }] [-output {nvp | table | json [-raw]| csv [-table] [-noformat]}]
```

Action qualifiers

Qualifier	Description
-o, -output	Specify the output format. Value is one of the following: <ul style="list-style-type: none">• <code>nvp</code>—The name-value pair (NVP) format displays the output as <code>name=value</code>.• <code>table</code>—The table format displays output as a table, with column headers and rows. By default, values that are too long to fit in a column are cut off.• <code>json</code>—The JSON format is the JSON representation of the data, similar to the corresponding REST API response body.<ul style="list-style-type: none">◦ <code>-raw</code>—Removes human-readable formatting such as new lines and indentation for JSON output.• <code>csv</code>—The comma-separated values (CSV) format is similar to the table format, but commas separate the names and values.<ul style="list-style-type: none">◦ <code>-noformat</code>—Removes human-readable formatting for size and speed values to ease data import.◦ <code>-table</code>—Exports <code>.csv</code> with table formatting; instances with embedded arrays are split into several lines after importing <code>.csv</code> as a table.
-select	Display the list of fields specified instead of the default fields.
-sort	Sort the output by the specified fields. Append a field with <code>+</code> to sort ascending (default), or <code>-</code> to sort descending.
-offset	Specifies where to start in a result set. Offset 0 is the normal start of the list. Offset 100 skips the first 100 instances and start the output with the 100th instance.
-limit -all	Specifies how many instances to show. The default is 100. <code>-all</code> displays all instances, up to a limit of 2000. For longer lists, use <code>-offset</code> and <code>-limit</code> together to show chunks of data iteratively.

Name-value pair format

```
1:      ID           = 1a0_SPA
        SP           = SPA
        Ports        = eth0_SPA,eth1_SPA
        Health state = OK (5)

2:      ID           = 1a0_SPB
        SP           = SPB
        Ports        = eth0_SPB,eth1_SPB
        Health state = OK (5)
```

Table format

ID	SP	Ports	Health state
la0_SPA	SPA	eth0_SPA,eth1_SPA	OK (5)
la0_SPB	SPB	eth0_SPB,eth1_SPB	OK (5)

JSON format

```
[
  {
    "id": "1a477f78-6880-44bc-9dbe-81eb6a570e25",
    "name": "tal3",
    "description": "",
    "type": "Primary",
    "wwn": "naa.68ccf09800fdeb5b95b22ab6f61c352d",
    "state": "Ready",
    "size": 81920000
  },
  {
    "id": "59a5f876-fe37-45a1-8ed3-7dae3d493386",
    "name": "tal5",
    "description": "",
    "type": "Primary",
    "wwn": "naa.68ccf098007d921192aec35c4449c1b4",
    "state": "Ready",
    "size": 81920000
  },
  {
    "id": "616b80db-f8ad-46cb-b958-f9c8a7940770",
    "name": "tal2",
    "description": "",
    "type": "Primary",
    "wwn": "naa.68ccf09800c6c60a72678f4c3c71ddff",
    "state": "Ready",
    "size": 819200000
  },
  {
    "id": "cab12dab-a421-49e2-a9f0-8f0812b88bd9",
    "name": "tal4",
    "description": "",
    "type": "Primary",
    "wwn": "naa.68ccf09800c6f0e48a9361937ffbb10f",
    "state": "Ready",
    "size": 81920000
  },
  {
    "id": "e97a7760-a4d9-462b-a6f1-67bcc5f8f367",
    "name": "tal1_new",
    "description": "",
    "type": "Primary",
    "wwn": "naa.68ccf09800efe4b0ae44dadec26a0a59",
    "state": "Ready",
    "size": 819200000
  }
]
```

Comma-separated values format

```
ID,SP,Ports,Health state
la0_SPA,SPA,"eth0_SPA,eth1_SPA",OK (5)
la0_SPB,SPB,"eth0_SPB,eth1_SPB",OK (5)
```

The delete action command

The `delete` action command removes an object from the system based on the specified object and object qualifier.

Format

```
<object type> [<object qualifiers>] delete
```

Example

The following command deletes `user1`:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! local_user -name user1 delete
```

Get help

For help with using the CLI, use the `-help`, `--help`, `-h`, `--h` or `-?`.

Depending on the context where this command is used, it can provide the following information:

- List of CLI object types
- Details about a specific object such as attributes and action syntax
- Details about a specific action such as syntax and description of each parameter

Help on the PowerStore CLI client

For help on the client, including the use cases, switches, and system requests, type only `pstcli` or include the `-help|-?` switch. [View the switches](#) provides details on all available switches.

Example

The following command displays information about the syntax and switches:

```
pstcli -?
```

```
Enter interactive session.
pstcli [-d <address>] [-port <number>] [-u <user_name>] [-p <password>] [-ssl {
interactive | reject | accept | store }] [-header] -session

Perform an action on an object on the destination system.
pstcli [-d <address>] [-port <number>] [-u <user_name>] [-p <password>] [-ssl {
interactive | reject | accept | store }] [-header] <object> <action> [<qualifiers>]

Get detailed help on client options:
pstcli help { -session | CMD | -version | -save_cred | -remove_cred | -remove_all_creds
| -default | -cert_list | -cert_del | -cert_clear | -clear_cache }

Additional help topics will be provided once you connect to the remote server. Please
provide destination address to obtain remote server command help.
```

Help on object types

For help on object types, specify the object type with the `-help` switch to view a list of supported action commands.

Example

The following command displays the action commands to set (change) and show a DNS server setting with the `dns` object type. In the output, the items in brackets are the actions, such as creating and modifying, that can be performed on the object type:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! dns -?
```

```
Configure system DNS client settings.
Actions:
[Set]
dns set -nameServer <value>

[Show]
dns show [-output {nvp|csv|table[-wrap]}] [{-brief|-detail}]
```

Help on actions

For help on an action command, type the object type and action command followed by `-h`, `-help`, or `-?`.

Example

The following command displays the list of volume attributes that you can change:

```
pstcli -destination 10.55.34.87 -u admin -p MyPassword456! /volume set -?
```

```
volume { -id <value> | -name <value> } set [ -name <value> ] [ -description
<value> ] [ -size <less than 281474976710656> ] [ -expiration_timestamp <date> ] [
-protection_policy_id <value> ] [ -performance_policy_id <value> ] [
-is_replication_destination { yes | true | no | false } ] [ -async ]
```

Modify the parameters of a volume.

Where:

```
[Choice] { -id <value> | -name <value> }
-id <value>
```

Unique identifier of the volume to modify.

```
-name <value>
```

Reference object (volume) by name.

```
[Optional] -name <value>
```

New name of the volume. This value must contain 128 or fewer printable Unicode characters.

```
[Optional] -description <value>
```

New description of the volume. This value must contain 128 or fewer printable Unicode characters.

```
[Optional] -size <less than 281474976710656>
```

New size of the volume in bytes, must be a multiple of 8192, must be bigger than the current volume size. Maximum volume size is 256TB.

```
[Optional] -expiration_timestamp <date>
```

New expiration time of the snapshot. Expired snapshots are deleted by the snapshot aging service that runs periodically in the background. If not specified, the snapshot never expires.

Use a maximum timestamp value to set an expiration to never expire.

```
[Optional] -protection_policy_id <value>
```

Unique identifier of the protection policy assigned to the volume.

```
[Optional] -performance_policy_id <value>
```

Unique identifier of the performance policy assigned to the volume.

```
[Optional] -is_replication_destination { yes | true | no | false }
```

New value for `is_replication_destination` property. The modification is only supported for primary and clone volume, only when the current value is true and there is no longer

a replication session using this volume as a destination, and only to false.

[Optional] `-async`
Perform asynchronous operation

Manage SSL/TLS certificates

When logging in to the system through PowerStore CLI, the system uses SSL/TLS certificates to secure communications between the CLI client and the system. You can manage these certificates and configure a policy for the PowerStore CLI to use when receiving unknown certificates. All downloaded certificates are stored in the secure, local lockbox on the client system. [Save PowerStore CLI settings](#) explains how settings are saved.

View the switches

The PowerStore CLI switches apply only to your installed PowerStore CLI client. Use the switches to access a system, upload files to the system, and manage security certificates.

Format

```
pstcli [{-help|-h|-?}]
```

Switches

- | | |
|-------------------------------|--|
| <code>-destination -d</code> | IP (IPv4 or IPv6) address or network name of the destination system. If you do not include this switch, the client uses the addresses that are specified for <code>-default</code> . |
| <code>-port</code> | Port number on the system. |
| <code>-user -u</code> | Username for logging in to the system. |
| <code>-password -p</code> | Password for logging in to the system. |
| <code>-ssl_policy -ssl</code> | Policy for handling unknown SSL certificates downloaded from the system. Value is one of the following: <ul style="list-style-type: none">• <code>interactive</code> — Prompt the user to accept the certificates for the current session (default).• <code>reject</code> — Automatically reject the certificates.• <code>accept</code> — Automatically accept the certificates.• <code>store</code> — Automatically accept and store the certificates. |
| <code>-cert_list</code> | List of all certificates stored locally in the lockbox. |
| <code>-cert_clear</code> | Delete all certificates stored locally in the lockbox. |
| <code>-cert_del</code> | Delete one or more certificates from the lockbox. Type a comma-separated list of certificate IDs.
 NOTE: Use <code>-cert_list</code> to view a list of stored certificates with their IDs. |
| <code>-help -h -?</code> | Display information about the syntax and switches, a list of command categories available on the target system, and information about getting further help on specific topics. |
| <code>-save_cred</code> | Save the access credentials that are specified for the <code>-user</code> and <code>-password</code> switches to a local security file in the lockbox. With the access credentials saved, PowerStore CLI automatically applies them to the specified system destination and port pair each time you run a command. Save PowerStore CLI settings explains saving user account credentials to the local client system. |
| <code>-remove_cred</code> | Remove the specified user account from the lockbox. |
| <code>-remove_all_cred</code> | Remove all user accounts from the lockbox. |

s

-default	Save the destination, port pair, and SSL policy as defaults. You can use the <code>-default</code> switch with the <code>-destination</code> , <code>-port</code> , or <code>-ssl_policy</code> switches, or with any combination of those switches. The saved SSL policy applies to the client for all destinations. Also, you can update the destination and port separately. Save PowerStore CLI settings explains saving user account credentials to the local client system.
-header	Show the header message (system IP address, port number, and so on), which is hidden by default, above the command output.
-v -version	Display the version of your PowerStore CLI.
-clear_cache	Clear cached data.

Example 1

The following example accesses the destination system 10.0.0.1 as user `admin` with password **MyPassword456!**:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456!
```

Example 2

The following example saves the access credentials for the specified user:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! -save_cred
```

Example 3

The following example sets the destination system as the default:

```
pstcli -d 10.0.0.1 -u admin -p MyPassword456! -default
```

Example 4

The following example removes the saved access credentials from destination system 10.0.0.1:

```
pstcli -d 10.0.0.1 -remove_cred
```

Access the system

To access and run commands on a system through PowerStore CLI, specify the network name or management IP address of the system, your username, and your password.

Format

```
[{-d|-destination} <value>] [{-u|-user} <user_name>] [{-p|-password} <password>]
```

Switches

-destination -d	IP address or network name of the destination system. If you do not include this switch, the client uses the addresses that are specified for <code>-default</code> . If no default address exists, the client uses the localhost address 127.0.0.1.
-user -u	Username for logging in to the system. For example, <code>admin</code> .
-password -p	Password for logging in to the system.
-port	Specify the port number through which to access the system.

 **NOTE:** If you do not include the `-port` switch, PowerStore CLI accesses the system through default port 443.

-default

Save the destination, port pair, and SSL policy as defaults. You can use the `-default` switch with the `-destination`, `-port`, or `-ssl_policy` switches, or with any combination of those switches. The saved SSL policy applies to the client for all destinations. Also, you can update the destination and port separately. [Save PowerStore CLI settings](#) explains saving user account credentials to the local client system.

Display the header

You can display the header each time you run a switch or command. The header displays the destination system, system port number, the syntax, and communication protocol used (HTTPS).

The following example shows the header that is displayed after including the `-header` switch in a command:

```
pstcli -header volume show -o table
PowerStore system: 10.55.34.87
Timezone: Eastern Daylight Time (UTC-03:00)
User: admin

1: id = DNS1
addresses = 10.244.53.108, 10.244.53.109
```

Save PowerStore CLI settings

You can save the following settings on the host on which you run PowerStore CLI:

- User access credentials, including your username and password, for each system you access
- SSL certificates imported from the system
- Information about the default system to access through PowerStore CLI, including the system name or IP address and the system port number

PowerStore CLI saves the settings to a secure lockbox that resides locally on the host on which PowerStore CLI is installed. The stored data is only available on the host where it was saved and to the user who saved it. The lockbox resides in the following locations:

- On Windows XP: `C:\Documents and Settings\<<account_name>\Local Settings \Application Data\.pstcli`
- On Windows 7 and Windows 10: `C:\Users\${user_name}\AppData\Local\.pstcli`
- On Linux: `<home_directory>/ .pstcli`