



Salesforce DX Setup Guide

Version 41.0, Winter '18



CONTENTS

| | |
|------------------------------------------------------------------------------------------|----|
| Chapter 1: Before You Begin | 1 |
| Chapter 2: System Requirements | 2 |
| Chapter 3: Enable the Dev Hub in Your Org | 3 |
| Chapter 4: Enable Second-Generation Packaging (Beta) | 4 |
| Chapter 5: Add Salesforce DX Users | 5 |
| Chapter 6: Install the Salesforce CLI | 7 |
| Install the CLI on macOS | 8 |
| Install the CLI on Windows | 8 |
| Install the CLI on the Windows 10 Subsystem for Linux | 8 |
| Install the CLI on Linux | 9 |
| Install the CLI with npm | 10 |
| Verify Your Installation | 10 |
| Chapter 7: Use the Salesforce CLI from Behind a Company Firewall or Web Proxy ... | 12 |
| Chapter 8: Update the CLI | 14 |
| Chapter 9: Uninstall the CLI Binary or Plug-In | 15 |
| Chapter 10: Sample Source in GitHub | 16 |
| Chapter 11: Next Steps | 17 |

CHAPTER 1 Before You Begin

Before you begin to explore Salesforce DX, you must set up the Salesforce DX environment.

If you participated in the Salesforce DX Beta, follow the instructions in [Verify Your Installation](#). That's it. You're ready to go.

1. Enable the Developer Hub (Dev Hub) in your production org (if you're a customer) or your business org (if you're an AppExchange partner).

If you want to try out Salesforce DX first, you can sign up for a [trial org](#) that has Dev Hub enabled.

2. Enable Second-Generation Packaging (2GP) in your org so you can develop 2GP packages.
3. If you want to include your team, you can add users to your Dev Hub org.
4. Ensure that your computers meet all system requirements.
5. If you want to evaluate the Force.com IDE 2 (Beta) or Salesforce Development Tools for Visual Studio Code, download and install them.
6. Download the Salesforce DX samples from GitHub.

SEE ALSO:

[Salesforce DX Trial Org](#)

[Visual Studio Code Extension Pack for Salesforce DX \(Visual Studio | Marketplace\)](#)

CHAPTER 2 System Requirements

Use this list of system requirements to get the most out of Salesforce DX.

Operating Systems

The Salesforce CLI supports the following operating systems.

- Windows—Windows 7 (64 and 32-bit) or later
- Mac—macOS 10.11 or later
- Linux—Ubuntu 14.04

macOS Requirement: Developer Toolset

Install the base developer toolset, which includes Git. You can install these command-line tools without installing the entire Xcode developer package or having a developer account.

```
xcode-select --install
```

If you see these error messages, you can ignore them. It means that you already have Xcode installed.

```
Can't install the software because it is not currently available from
the Software Update server
xcode-select: error: command line tools are already installed, use
"Software Update" to
install updates
```

 **Note:** If you're updating to a new version of macOS, run this command again.

Code Editor or IDE

You can use any code editor Salesforce DX, including the redesigned Force.com IDE built for Salesforce DX.

Version Control System

You can use any version control system (VCS) with Salesforce DX. We recommend that you use GitHub to take advantage of the samples in our GitHub repository.


SEE ALSO:

[GitHub](#)

[PowerShell Documentation](#)

CHAPTER 3 Enable the Dev Hub in Your Org

Enable the Dev Hub in your org so you can create and manage scratch orgs from the command line and Lightning Experience. Scratch orgs are disposable Salesforce orgs to support development and testing.

 **Note:** Enabling Dev Hub in a production or business org is completely safe and won't cause any performance or customer issues. The Dev Hub is comprised of objects with permissions that allow admins to control the level of access available to a user and an org.

1. Log in to your production org (if you're a customer), your business org (if you're an ISV), or your trial org as the System Administrator.
2. From Setup, enter *Dev Hub* in the Quick Find box and select **Dev Hub**.

If you don't see Dev Hub in the Setup menu, make sure you're in your production or business org, and your org is one of the supported editions.

3. To enable the Dev Hub, click **Enable**.

After you enable the Dev Hub, you can't disable it. If you're using a trial org, Dev Hub is already enabled.

SEE ALSO:

[Salesforce DX Trial Org](#)

[Add Salesforce DX Users](#)

EDITIONS

Available in: Salesforce Classic and Lightning Experience

Dev Hub available in: **Enterprise, Performance, and Unlimited** Editions

Scratch orgs available in: **Developer, Enterprise, Group, and Professional** Editions

CHAPTER 4 Enable Second-Generation Packaging (Beta)

Enable Second-Generation Packaging (2GP) in your org so you can develop 2GP packages. You can work with the packages in scratch orgs, sandbox orgs, and target subscriber orgs.

Enable the Dev Hub in your org.

1. Log in to the org where you've enabled the Dev Hub.
2. From Setup, enter *Dev Hub* in the Quick Find box and select **Dev Hub**.
3. In the section for Second-Generation Packaging, click **Non-GA Service Agreement** to read the service agreement.
4. To enable second-generation packaging, click **Enable**.

After you enable Second-Generation Packaging, you can't disable it.

CHAPTER 5 Add Salesforce DX Users

System administrators can access the Dev Hub by default. You can enable more users to access the Dev Hub so they can also create scratch orgs.

You can use Salesforce DX with these Standard user licenses: Salesforce, Salesforce Platform, and Salesforce Limited Access - Free (partners only).

You can add a user with the System Administrator profile. You can also add a user with a Standard User profile as long as you apply the set of permissions required for Salesforce DX. Because you're adding users to a Dev Hub org, avoid adding them as system administrators unless their work requires that level of authority.

1. Add the user to your Dev Hub org, if necessary.
 - a. In Setup, enter *Users* in the Quick Find box, then select **Users**.
 - b. Click **New User**.
 - c. Fill out the form, and assign the System Administrator or Standard User profile.
 - d. Click **Save**.If you're adding a System Administrator user, you can stop here.
2. If you're adding a Standard User, create a permission set for Salesforce DX users if you don't have one.
 - a. From Setup, enter *Permission Sets* in the Quick Find box, then select **Permission Sets**.
 - b. Click **New**.
 - c. Enter a label, API name, and description. The API name is a unique name used by the Force.com API and managed packages.
 - d. Select a user license option. If you plan to assign this permission set to multiple users with different licenses, select **None**.
 - e. Click **Save**. The permission set overview page appears. From here, you can navigate to the permissions you want to add or change for Salesforce DX. For the required permissions, see [Permission Set for Salesforce DX Users](#) on page 5.
3. Apply the Salesforce DX permission set to the Standard User.
 - a. From Setup, enter *Permission Sets* in the Quick Find box, then select **Permission Sets**.
 - b. Select the Salesforce DX permission set.
 - c. In the permission set toolbar, click **Manage Assignments**.
 - d. Click **Add Assignments**.
 - e. Select the user to assign the permission set to.
 - f. Click **Assign**.
 - g. Click **Done**.

Permission Set for Salesforce DX Users

To give full access to the Dev Hub, the permission set must contain these permissions.

Add Salesforce DX Users

- Object Settings -> Scratch Org Info -> Read, Create, and Delete
- Object Settings -> Active Scratch Org -> Read and Delete
- Object Settings -> Namespace Registry -> Read, Create, and Delete

You can limit a user's access by modifying the permissions.

SEE ALSO:

[Permission Sets \(Salesforce Help\)](#)

[Salesforce DX Developer Guide](#)

CHAPTER 6 Install the Salesforce CLI

In this chapter ...

- [Install the CLI on macOS](#)
- [Install the CLI on Windows](#)
- [Install the CLI on Linux](#)
- [Install the CLI with npm](#)
- [Verify Your Installation](#)

Use the CLI commands to create environments for development and testing, synchronize source code between your scratch orgs and version control system, and execute test suites.

Installation Options

By default, the latest version of the Salesforce CLI and salesforcedx plug-in is installed. When you run a command, the Salesforce CLI checks to see if you have the latest version. If not, the CLI automatically updates itself and the salesforcedx plug-in. Sometimes, you'll want to use a specific version of the salesforcedx plug-in, or remain on the current version.

To install a specific version of the Salesforce CLI, in this example, version 41.3.0:

```
sfdx plugins:install salesforcedx@41.3.0
```

To remain on the current version of the CLI, set this environment variable:

```
export SFDX_AUTOUPDATE_DISABLE=true
```

Install the CLI on macOS

You install the Salesforce CLI on macOS with a `.pkg` file.

1. [Download](#) the `.pkg` file.
2. Double-click the `.pkg` file.

SEE ALSO:

[Verify Your Installation](#)

Install the CLI on Windows

You install the Salesforce CLI on Windows with an `.exe` file.

1. [Download](#) and run the Windows installer. If the installer does not find Git, it is also installed.
2. Check that your PATH environment variable includes the location of the Git binaries. For example:

```
set PATH=%PATH%;C:\Program Files\Git\bin
```



Warning: The Salesforce CLI works best within the native Windows command prompt (`cmd.exe`) and the Microsoft Powershell. We do not recommend using the Salesforce CLI with a Linux terminal emulator, such as cygwin or MinGW, because support for bugs is limited.

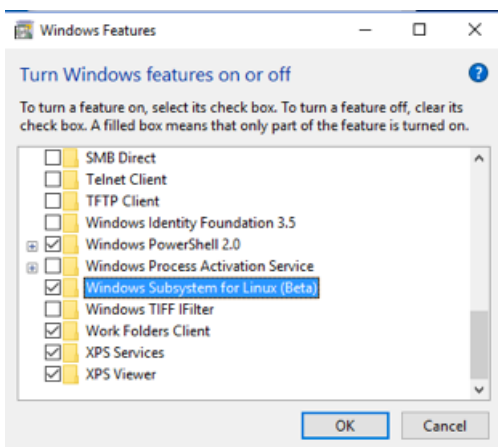
SEE ALSO:

[Verify Your Installation](#)

Install the CLI on the Windows 10 Subsystem for Linux

The Windows Subsystem for Linux (WSL) allows you to run native Linux command-line tools on Windows 10 alongside your traditional Windows desktop.

1. From the Run dialog box (Windows key + R), run `optionalfeatures.exe`.
2. In Windows Features, select **Windows Subsystem for Linux**.



- From the command prompt, enter `bash`, which triggers the installation of Ubuntu. At the prompt, enter `y` to agree to install the components.

```
> bash
```

- When the Ubuntu install is finished, create the directory `/mnt/c/ProgramFilesLinux`.

```
$ mkdir /mnt/c/ProgramFilesLinux
```

- Change to the new directory.

```
$ cd /mnt/c/ProgramFilesLinux
```

- Run `wget` with the download URL of the latest Linux (amd64) tarball. Find the appropriate URL in [this manifest file](#).

```
$ wget  
https://developer.salesforce.com/media/salesforce-cli/sfdx-v5.7.6-d42cf65-linux-amd64.tar.xz
```

- Unpack the contents.

```
$ tar xvf sfdx-v5.7.6-d42cf65-linux-amd64.tar.xz
```

- Change to the `sfdx` directory.


```
$ cd sfdx
```

- Run the install script.

```
$ ./install
```

- Install Git.

```
$ sudo apt-get install git
```

 **Note:** You must install Git for the Salesforce CLI to work correctly.

Install the CLI on Linux

The Linux version of the Salesforce CLI is distributed as a tarball.

The tarball is called `sfdx-version-OS-ARCH.tar.xz`, where:

- `version` is a version string, such as `v5.9.9-d42cf65`
- `OS` is either `darwin`, `linux`, `freebsd`, `openbsd`, or `windows`
- `ARCH` is either `amd64` or `386`

- Download or `wget` the tarball. Find the download URL for your tarball from [this manifest file](#). Select the URL based on your target platform. For example:

```
$ wget  
https://developer.salesforce.com/media/salesforce-cli/sfdx-v5.99.1-d7efd75-linux-amd64.tar.xz
```

2. Unpack the contents. For example, if you downloaded the tarball in your current directory:

```
$ tar -xvJf sfdx-v5.9.9-d42cf65-linux-amd64.tar.xz
```

3. Change to the `sfdx` directory.

```
$ cd sfdx
```


4. Run the install script.

```
$ ./install
```

The Salesforce CLI is installed in `/usr/local/bin/sfdx`.

Install the CLI with `npm`

If you've installed `Node.js` on your computer, you can use `npm` to install the Salesforce CLI. This method lets you install the Salesforce CLI from the command line and can be especially useful for continuous integration (CI) use cases.

 **Note:** Currently, you can install the Salesforce CLI using `npm` on only macOS or Linux. Check back later for Windows support.

1. Ensure that Node v8.6.0 or later is installed on your computer. For example, on macOS or Linux:

```
$ node --version
v8.6.0
```

2. Run this command.

```
$ npm install sfdx-cli --global
```

If you receive a permission error when installing the CLI using `npm`, we recommend not using `sudo`. See [Fixing npm permissions](#).

SEE ALSO:

[Verify Your Installation](#)

[npm Documentation](#)

Verify Your Installation

Verify your Salesforce CLI installation and plug-in versions.

Run this command to verify the Salesforce CLI version:

```
sfdx --version
sfdx-cli/6.0.10-3713d7b alpha (darwin-x64) node-v8.6.0
```

Run this command to verify the Salesforce CLI plug-in version:

```
$ sfdx plugins --core
salesforcedx 41.2.0 (core)
```

Run this command to return a list of the command families in the `force` topic:

```
sfdx force --help
```

This command returns all the `force` commands:

```
sfdx force:doc:commands:list
```

CHAPTER 7 Use the Salesforce CLI from Behind a Company Firewall or Web Proxy

If you install or update the Salesforce CLI on a computer that's behind a company firewall or web proxy, you might receive error messages. In this case, you must further configure your system.

You get an error similar to the following when you run a command after installing the CLI binary behind a firewall or web proxy. This error is from a Linux computer, but Windows and macOS users sometimes see similar errors.

```
panic: Get
https://developer.salesforce.com/media/salesforce-cli/sfdx-cli/channels/stable/darwin-x64:
EOF

goroutine 1 [running]:
panic(0x265c00, 0xc4200f0060)
/usr/local/go/src/runtime/panic.go:500 +0x1a1
main.must(0x3c0280, 0xc4200f0060)
/home/ubuntu/.go_workspace/src/github.com/heroku/cli/io.go:115 +0x5c
main.GetUpdateManifest(0x2a734b, 0x6, 0x2a6b5c, 0x3, 0x3c0280)
/home/ubuntu/.go_workspace/src/github.com/heroku/cli/install.go:104
+0x26b
main.GetUpdateManifest(0x2a734b, 0x6, 0x2a6b5c, 0x3, 0x0)
/home/ubuntu/.go_workspace/src/github.com/heroku/cli/install.go:102
+0x2ed
main.Install()
/home/ubuntu/.go_workspace/src/github.com/heroku/cli/install.go:33
+0xf9
main.main()
/home/ubuntu/.go_workspace/src/github.com/heroku/cli/main.go:18 +0x3
```

Updating the CLI behind a firewall or web proxy might return these errors.

```
sfdx-cli: Updating CLI... !
❑ 'ECONNRESET': tunneling socket could not be established,
cause=connect EHOSTUNREACH 0.0.23.221:8080 - Local
(10.126.148.39:53107)
```

To address this issue, run these commands from your terminal or Windows command prompt, replacing `username:pwd` with your web proxy username and password. If your proxy doesn't require these values, omit them. Also replace `proxy.company.com:8080` with the URL and port of your company proxy.

```
npm config set https-proxy https://username:pwd@proxy.company.com:8080
npm config set proxy https://username:pwd@proxy.company.com:8080
npm config set sslVerify false
npm config set strict-ssl false
```


Use the Salesforce CLI from Behind a Company Firewall or Web Proxy

Then set the HTTP_PROXY or HTTPS_PROXY environment variable to the full URL of the proxy. For example, on UNIX:

```
export HTTP_PROXY=http://username:pwd@proxy.company.com:8080
```

```
export HTTPS_PROXY=https://username:pwd@proxy.company.com:8080
```

On a Windows machine:

```
set HTTP_PROXY=http://username:pwd@proxy.company.com:8080
```

```
set HTTPS_PROXY=https://username:pwd@proxy.company.com:8080
```

CHAPTER 8 Update the CLI

If you want to ensure that you are running the latest version of Salesforce CLI, you can manually update it.

If You Installed the Salesforce CLI Using the Installer

By default, the CLI periodically checks for and installs updates. To disable auto-update, set the `SFDX_AUTOUPDATE_DISABLE` environment variable to `true`.

```
export SFDX_AUTOUPDATE_DISABLE=true
```

To install the latest Salesforce CLI version, run:

```
$ sfdx update
sfdx-cli: Updating CLI from 6.0.0-0743bea5 to 6.0.0-aebbfd66
(alpha)... done
sfdx-cli: Updating CLI... already on latest version: 6.0.0-aebbfd66
sfdx-cli: Updating plugins... done
```

If You Installed the Salesforce CLI Using npm

The auto-update option is not available. To update the Salesforce CLI using npm, run:

```
npm install --global sfdx-cli
```

Windows: Improve CLI Update Speed

Anti-virus software, such as Windows Defender, slows a CLI update because it scans all the files as they are downloaded. To improve the update speed, add the `%LOCALAPPDATA%\sfdx` directory to the list of directories that are excluded from the scan. We recommend that you consult your security administrator before making this configuration change.

CHAPTER 9 Uninstall the CLI Binary or Plug-In

Uninstalling the CLI removes it entirely from your computer.

macOS or Linux

Enter all these commands in a terminal:

```
sudo rm -rf /usr/local/sfdx
sudo rm -rf /usr/local/lib/sfdx
sudo rm -rf /usr/local/bin/sfdx
sudo rm -rf ~/.local/share/sfdx ~/.config/sfdx ~/.cache/sfdx
sudo rm -rf ~/Library/Caches/sfdx
```

Windows

1. Select **Start > Control Panel > Programs > Programs and Features**.
2. Select **SFDX CLI**, and click **Uninstall**.
3. Inside your home directory, delete the `.config\sfdx` directory.

If the CLI is still installed, delete the `%LOCALAPPDATA%\sfdx` directory in Program Files.

Uninstall the `salesforcedx` Plug-In

Enter this command from a terminal or Windows command prompt:

```
sfdx plugins:uninstall salesforcedx
```

CHAPTER 10 Sample Source in GitHub

While you're exploring Salesforce DX, look at some sample repos we've provided on GitHub to help you understand the concepts.

- `sfdx-simple`—A sample with two Visualforce pages, a controller, and Apex tests.
- `dreamhouse-sfdx`—A more complex sample app with multiple Apex classes, configuration items, Lightning components, Visualforce components, and custom objects. The `dreamhouse-sfdx` sample is a standalone application.
- `sfdx-travisci`—A continuous integration (CI) sample that shows how to use Salesforce DX with Travis CI. Travis CI is a cloud-based continuous integration (CI) service for building and testing software projects hosted on GitHub.

SEE ALSO:

[sfdx-simple GitHub Sample](#)

[dreamhouse-sfdx GitHub Sample](#)

[sfdx-travisci GitHub Sample](#)

CHAPTER 11 Next Steps

Read on to learn what to do after you've installed the Salesforce CLI.

For a hands-on exploration of how the Salesforce DX components work together using the DreamHouse sample, see the [dreamhouse-sfdx Readme](#).

To learn more about how to use Salesforce DX, see the *Salesforce DX Developer Guide*.

For the complete list of CLI commands and how to use them, see the *Salesforce CLI Command Reference*.

SEE ALSO:

[dreamhouse-sfdx GitHub Sample Readme](#)

[Salesforce CLI Command Reference](#)

[Salesforce DX Developer Guide](#)