Install .NET Core SDK or .NET Core Runtime on Ubuntu

06/04/2020 • 21 minutes to read •   

In this article

.NET Core is supported on Ubuntu. This article describes how to install .NET Core on Ubuntu. When an Ubuntu version falls out of support, .NET Core is no longer supported with that version. However, these instructions may help you to get .NET Core running on those versions, even though it isn't supported.

Install the SDK (which includes the runtime) if you want to develop .NET apps. Or, if you only need to run apps, install the runtime. If you're installing the runtime, we suggest you install the **ASP.NET Core runtime** as it includes both .NET Core and ASP.NET Core runtimes.

If you've already installed the SDK or runtime, use the `dotnet --list-sdks` and `dotnet --list-runtimes` commands to see which versions are installed. For more information, see How to check that .NET Core is already installed.

*Package manager installs are only supported on the **x64** architecture*. Other architectures, such as **ARM**, must manually install the .NET Core SDK or .NET Core Runtime. For more information, see the manually install section below.

Supported distributions

The following table is a list of currently supported .NET Core releases and the versions of Ubuntu they're supported on. These versions remain supported until either the version of .NET Core reaches end-of-support or the version of Ubuntu reaches end-of-life.

- A ✔ indicates that the version of Ubuntu or .NET Core is still supported.
- A ✘ indicates that the version of Ubuntu or .NET Core isn't supported on that Ubuntu release.
- When both a version of Ubuntu and a version of .NET Core have ✔, that OS and .NET combination are supported.

Ubuntu	.NET Core 2.1	.NET Core 3.1	.NET 5 Preview (manual install only)
✔ 20.04 (LTS)	✔ 2.1	✔ 3.1	✔ 5.0 Preview
✘ 19.10	✔ 2.1	✔ 3.1	✔ 5.0 Preview
✘ 19.04	✔ 2.1	✔ 3.1	✘ 5.0 Preview
✘ 18.10	✔ 2.1	✘ 3.1	✘ 5.0 Preview
✔ 18.04 (LTS)	✔ 2.1	✔ 3.1	✔ 5.0 Preview
✘ 17.10	✔ 2.1	✘ 3.1	✘ 5.0 Preview
✘ 17.04	✔ 2.1	✘ 3.1	✘ 5.0 Preview
✘ 16.10	✘ 2.1	✘ 3.1	✘ 5.0 Preview
✔ 16.04 (LTS)	✔ 2.1	✔ 3.1	✔ 5.0 Preview

The following versions of .NET Core are no longer supported. The downloads for these still remain published:

- 3.0
- 2.2
- 2.0

How to install other versions

The packages added to package manager feeds are named in a hackable format: `{product}-{type}-{version}`.

- **product**
  The type of .NET product to install. Valid options are:
  - dotnet
  - aspnetcore

- **type**
  Chooses the SDK or the runtime. Valid options are:
  - sdk
  - runtime

- **version**
  The version of the SDK or runtime to install. This article will always give the instructions for the latest supported version. Valid options are any released version, such as:
  - 3.1
  - 3.0
  - 2.1

  It's possible the SDK/runtime you're trying to download is not available for your Linux distribution. For a list of supported distributions, see .NET Core dependencies and requirements.

Examples

- Install the ASP.NET Core 3.1 runtime: `aspnetcore-runtime-3.1`
- Install the .NET Core 2.1 runtime: `dotnet-runtime-2.1`
- Install the .NET Core 3.1 SDK: `dotnet-sdk-3.1`

Package missing

If the package-version combination doesn't work, it's not available. For example, there isn't an ASP.NET Core SDK, the SDK components are included with the .NET Core SDK. The value `aspnetcore-sdk-2.2` is incorrect and should be `dotnet-sdk-2.2`. For a list of

Linux distributions supported by .NET Core, see .NET Core dependencies and requirements.

20.04 ✔

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package dotnet-sdk-3.1**, see the APT troubleshooting section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package aspnetcore-runtime-3.1**, see the APT troubleshooting section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-3.1` in the command above with `dotnet-runtime-3.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-3.1
```

19.10 ✕

✕ Please note that this version of Ubuntu is no longer supported.

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/19.10/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package dotnet-sdk-3.1**, see the APT troubleshooting section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package aspnetcore-runtime-3.1**, see the APT troubleshooting section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-3.1` in the command above with `dotnet-runtime-3.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-3.1
```

19.04 ✗

✕ Please note that this version of Ubuntu is no longer supported.

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/19.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-3.1
```

ⓘ Important

If you receive an error message similar to **Unable to locate package dotnet-sdk-3.1**, see the **APT troubleshooting** section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
```

```
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package aspnetcore-runtime-3.1**, see the APT troubleshooting section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-3.1` in the command above with `dotnet-runtime-3.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-3.1
```

18.10 ✕

✕ Please note that this version of Ubuntu is no longer supported.

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/18.10/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
```

```
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-2.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package dotnet-sdk-2.1**, see the **APT troubleshooting** section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-2.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package aspnetcore-runtime-2.1**, see the **APT troubleshooting** section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-2.1` in the command above with `dotnet-runtime-2.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-2.1
```

18.04 ✔

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/18.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package dotnet-sdk-3.1**, see the **APT troubleshooting** section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-3.1
```

> **Important**

If you receive an error message similar to **Unable to locate package aspnetcore-runtime-3.1**, see the APT troubleshooting section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-3.1` in the command above with `dotnet-runtime-3.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-3.1
```

17.10 ✕

✕ Please note that this version of Ubuntu is no longer supported.

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/17.10/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-2.1
```

ⓘ Important

If you receive an error message similar to **Unable to locate package dotnet-sdk-2.1**, see the APT troubleshooting section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-2.1
```

Important

If you receive an error message similar to **Unable to locate package aspnetcore-runtime-2.1**, see the APT troubleshooting section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-2.1` in the command above with `dotnet-runtime-2.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-2.1
```

17.04 ✕

✕ Please note that this version of Ubuntu is no longer supported.

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/17.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-2.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package dotnet-sdk-2.1**, see the APT troubleshooting section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-2.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package aspnetcore-runtime-2.1**, see the APT troubleshooting section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-2.1` in the command above with `dotnet-runtime-2.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-2.1
```

16.10 ✕

✕ Please note that this version of Ubuntu is no longer supported.

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/16.10/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-2.1
```

> ⓘ **Important**
>
> If you receive an error message similar to **Unable to locate package dotnet-sdk-2.1**, see the **APT troubleshooting** section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-2.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package aspnetcore-runtime-2.1**, see the **APT troubleshooting** section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-2.1` in the command above with `dotnet-runtime-2.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-2.1
```

16.04 ✔

Installing with APT can be done with a few commands. Before you install .NET, run the following commands to add the Microsoft package signing key to your list of trusted keys and add the package repository.

Open a terminal and run the following commands:

Bash Copy

```
wget https://packages.microsoft.com/config/ubuntu/16.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
sudo dpkg -i packages-microsoft-prod.deb
```

Install the SDK

.NET Core SDK allows you to develop apps with .NET Core. If you install .NET Core SDK, you don't need to install the corresponding runtime. To install .NET Core SDK, run the following commands:

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y dotnet-sdk-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package dotnet-sdk-3.1**, see the APT troubleshooting section.

Install the runtime

The .NET Core Runtime allows you to run apps that were made with .NET Core that didn't include the runtime. The commands below install the ASP.NET Core Runtime, which is the most compatible runtime for .NET Core. In your terminal, run the following commands.

Bash Copy

```
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y aspnetcore-runtime-3.1
```

> **Important**
>
> If you receive an error message similar to **Unable to locate package aspnetcore-runtime-3.1**, see the APT troubleshooting section.

As an alternative to the ASP.NET Core Runtime, you can install the .NET Core Runtime that doesn't include ASP.NET Core support: replace `aspnetcore-runtime-3.1` in the command above with `dotnet-runtime-3.1`.

Bash Copy

```
sudo apt-get install -y dotnet-runtime-3.1
```

APT update SDK or runtime

When a new patch release is available for .NET Core, you can simply upgrade it through APT with the following commands:

Bash Copy

```
sudo apt-get update
sudo apt-get upgrade
```

APT troubleshooting

This section provides information on common errors you may get while using APT to install .NET Core.

Unable to locate

If you receive an error message similar to **Unable to locate package {netcore-package}**, run the following commands.

There are two placeholders in the following set of commands.

- `{dotnet-package}`
  This represents the .NET Core package you're installing, such as `aspnetcore-runtime-3.1`. This is used in the `sudo apt-get install` command below.
- `{os-version}`
  This represents the Linux version you are on. This is used in the `wget` command below.

Try purging the package list:

Bash Copy

```
sudo dpkg --purge packages-microsoft-prod && sudo dpkg -i packages-microsoft-prod.deb
sudo apt-get update
sudo apt-get install {dotnet-package}
```

If that doesn't work, you can run a manual install with the following commands:

Bash Copy

```
sudo apt-get install -y gpg
wget -O - https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor
-o microsoft.asc.gpg
sudo mv microsoft.asc.gpg /etc/apt/trusted.gpg.d/
wget https://packages.microsoft.com/config/ubuntu/{os-version}/prod.list
sudo mv prod.list /etc/apt/sources.list.d/microsoft-prod.list
sudo chown root:root /etc/apt/trusted.gpg.d/microsoft.asc.gpg
sudo chown root:root /etc/apt/sources.list.d/microsoft-prod.list
sudo apt-get update; \
  sudo apt-get install -y apt-transport-https && \
  sudo apt-get update && \
  sudo apt-get install -y {dotnet-package}
```

Failed to fetch

While installing the .NET Core package, you may see an error similar to `Failed to fetch ... File has unexpected size ... Mirror sync in progress?`. This error could mean that the package feed for .NET Core is being upgraded with newer package versions, and that you should try again later. During an upgrade, the package feed shouldn't be unavailable for more than 30 minutes. If you continually receive this error for more than 30 minutes, please file an issue at https://github.com/dotnet/core/issues.

Snap

.NET Core is available from the Snap Store.

A snap is a bundle of an app and its dependencies that works without modification across many different Linux distributions. Snaps are discoverable and installable from the Snap Store. For more information about Snap, see Getting started with Snap.

Only supported versions of .NET Core are available through Snap.

Install the SDK

Snap packages for .NET Core SDK are all published under the same identifier: `dotnet-sdk`. A specific version of the SDK can be installed by specifying the channel. The SDK includes the coresponding runtime. The following table list the channels:

.NET Core version	Snap package
3.1 (LTS)	`3.1` or `latest/stable`
2.1 (LTS)	`2.1`

.NET Core version	Snap package
.NET 5.0 preview	`5.0/beta`

Use the `snap install` command to install a .NET Core SDK snap package. Use the `--channel` parameter to indicate which version to install. If this parameter is omitted, `latest/stable` is used. In this example, `3.1` is specified:

Bash Copy

```
sudo snap install dotnet-sdk --classic --channel=3.1
```

Next, register the `dotnet` command for the system with the `snap alias` command:

Bash Copy

```
sudo snap alias dotnet-sdk.dotnet dotnet
```

This command is formatted as: `sudo snap alias {package}.{command} {alias}`. You can choose any `{alias}` name you would like. For example, you could name the command after the specific version installed by snap: `sudo snap alias dotnet-sdk.dotnet dotnet31`. When you use the command `dotnet31`, you'll invoke this specific version of .NET. But this is incompatible with most tutorials and examples as they expect a `dotnet` command to be available.

Install the runtime

Snap packages for .NET Core Runtime are each published under their own package identifier. The following table lists the package identifiers:

.NET Core version	Snap package
3.1 (LTS)	`dotnet-runtime-31`
3.0	`dotnet-runtime-30`
2.2	`dotnet-runtime-22`
2.1 (LTS)	`dotnet-runtime-21`

Use the `snap install` command to install a .NET Core Runtime snap package. In this example, .NET Core 3.1 is installed:

Bash Copy

```
sudo snap install dotnet-runtime-31 --classic
```

Next, register the `dotnet` command for the system with the `snap alias` command:

Bash Copy

```
sudo snap alias dotnet-runtime-31.dotnet dotnet
```

This command is formatted as: `sudo snap alias {package}.{command} {alias}`. You can choose any `{alias}` name you would like. For example, you could name the command after the specific version installed by snap: `sudo snap alias dotnet-runtime-31.dotnet dotnet31`. When you use the command `dotnet31`, you'll invoke this specific version of .NET. But this is incompatible with most tutorials and examples as they expect a `dotnet` command to be available.

SSL Certificate errors

When .NET is installed through Snap, it's possible that on some distros the .NET SSL certificates may not be found and you may receive an error similar to the following during `restore`:

Bash Copy

```
Processing post-creation actions...
Running 'dotnet restore' on /home/myhome/test/test.csproj...
  Restoring packages for /home/myhome/test/test.csproj...
/snap/dotnet-sdk/27/sdk/2.2.103/NuGet.targets(114,5): error : Unable to load
the service index for source https://api.nuget.org/v3/index.json.
[/home/myhome/test/test.csproj]
/snap/dotnet-sdk/27/sdk/2.2.103/NuGet.targets(114,5): error :   The SSL
connection could not be established, see inner exception.
[/home/myhome/test/test.csproj]
/snap/dotnet-sdk/27/sdk/2.2.103/NuGet.targets(114,5): error :   The remote
certificate is invalid according to the validation procedure.
[/home/myhome/test/test.csproj]
```

To resolve this issue, set a few enviornment variables:

Bash Copy

```
export SSL_CERT_FILE=[path-to-certificate-file]
export SSL_CERT_DIR=/dev/null
```

The certificate location will vary by distro. Here are the locations for the distros where we have experienced the issue.

- Fedora - `/etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem`
- OpenSUSE - `/etc/ssl/ca-bundle.pem`
- Solus - `/etc/ssl/certs/ca-certificates.crt`

Dependencies

When you install with a package manager, these libraries are installed for you. But, if you manually install .NET Core or you publish a self-contained app, you'll need to make sure these libraries are installed:

- libc6
- libgcc1
- libgssapi-krb5-2
- libicu52 (for 14.x)
- libicu55 (for 16.x)
- libicu60 (for 18.x)
- libicu66 (for 20.x)
- libssl1.0.0 (for 14.x, 16.x)
- libssl1.1 (for 18.x, 20.x)
- libstdc++6
- zlib1g

For .NET Core apps that use the *System.Drawing.Common* assembly, you also need the following dependency:

- libgdiplus (version 6.0.1 or later)

> ⚠ **Warning**
>
> You can install a recent version of *libgdiplus* by adding the Mono repository to your system. For more information, see **https://www.mono-project.com/download/stable/**.

Scripted install

The dotnet-install scripts are used for automation and non-admin installs of the **SDK** and **Runtime**. You can download the script from https://dot.net/v1/dotnet-install.sh.

The script defaults to installing the latest SDK long term support (LTS) version, which is .NET Core 3.1. To install the current release, which may not be an (LTS) version, use the `-c Current` parameter.

```
Bash
./dotnet-install.sh -c Current
```

To install .NET Core Runtime instead of the SDK, use the `--runtime` parameter.

```
Bash
./dotnet-install.sh -c Current --runtime aspnetcore
```

You can install a specific version by altering the `-c` parameter to indicate the specific version. The following command installs .NET Core SDK 3.1.

```
Bash
./dotnet-install.sh -c 3.1
```

For more information, see dotnet-install scripts reference.

Manual install

As an alternative to the package managers, you can download and manually install the SDK and runtime. Manual install is usually performed as part of continuous integration testing or on an unsupported Linux distribution. For a developer or user, it's generally better to use a package manager.

If you install .NET Core SDK, you don't need to install the corresponding runtime. First, download a **binary** release for either the SDK or the runtime from one of the following sites:

- ✔ .NET 5.0 preview downloads
- ✔ .NET Core 3.1 downloads
- ✔ .NET Core 2.1 downloads
- All .NET Core downloads

Next, extract the downloaded file and use the `export` command to set variables used by .NET Core and then ensure .NET Core is in PATH.

To extract the runtime and make the .NET Core CLI commands available at the terminal, first download a .NET Core binary release. Then, open a terminal and run the following commands from the directory where the file was saved. The archive file name may be different depending on what you downloaded.

Use the following command to extract the runtime:

Bash Copy

```
mkdir -p "$HOME/dotnet" && tar zxf aspnetcore-runtime-3.1.0-linux-x64.tar.gz
-C "$HOME/dotnet"
export DOTNET_ROOT=$HOME/dotnet
export PATH=$PATH:$HOME/dotnet
```

Use the following command to extract the SDK:

Bash Copy

```
mkdir -p "$HOME/dotnet" && tar zxf dotnet-sdk-3.1.301-linux-x64.tar.gz -C
"$HOME/dotnet"
export DOTNET_ROOT=$HOME/dotnet
export PATH=$PATH:$HOME/dotnet
```

Tip

The preceding `export` commands only make the .NET Core CLI commands available for the terminal session in which it was run.

You can edit your shell profile to permanently add the commands. There are a number of different shells available for Linux and each has a different profile. For example:

- **Bash Shell**: *~/.bash_profile*, *~/.bashrc*
- **Korn Shell**: *~/.kshrc* or *.profile*
- **Z Shell**: *~/.zshrc* or *.zprofile*

Edit the appropriate source file for your shell and add `:$HOME/dotnet` to the end of the existing `PATH` statement. If no `PATH` statement is included, add a new line with `export PATH=$PATH:$HOME/dotnet`.

Also, add `export DOTNET_ROOT=$HOME/dotnet` to the end of the file.

This approach lets you install different versions into separate locations and choose explicitly which one to use by which application.

Next steps

- Tutorial: Create a console application with .NET Core SDK using Visual Studio Code

Is this page helpful?

Yes No