


NVIDIA HPC SDK Installation Guide

Release 26.11

NVIDIA Corporation

Nov 17, 2026

Contents

1 Prepare to Install on Linux 3

2 Installation Steps for Linux 5

3 End-user Environment Settings 7

NVIDIA HPC SDK Installation Guide

This section describes how to install the HPC SDK from the tar file installer on Linux x86_64 or Arm Server systems with NVIDIA GPUs. It covers both local and network installations.

For package manager installations (e.g., apt, dnf/yum, zypper), please refer to the instructions on the HPC SDK download page.

For a complete description of supported processors, Linux distributions, and CUDA versions please see the HPC SDK Release Notes.

Chapter 1. Prepare to Install on Linux

Linux installations require some version of the GNU Compiler Collection (including gcc, g++, and gfortran compilers) to be installed and in your $PATH prior to installing HPC SDK software. To determine if such a compiler is installed on your system, do the following:

1. Create a `hello.c` program.

```
#include <stdio.h>
int main() {
  printf("hello, world!\n");
  return 0;
}
```

2. Compile to create an executable.

```
$ gcc -o hello_c hello.c
```

Run the `file` command on the produced executable. The output should look similar to the following:

```
$ file ./hello_c
hello_64_c: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), for
GNU/Linux 2.6.9, dynamically linked (uses shared libs), for GNU/Linux
2.6.9, not stripped
```

3. For support with C++ compilation, g++ version 4.4 is required at a minimum. A more recent version will suffice. Create a `hello.cpp` program and invoke g++. Make sure you are able to compile, link, and run the simple hello.cpp program first before proceeding.

```
#include <iostream>
int main() {
  std::cout << "hello, world!\n";
  return 0;
}
```

```
$ g++ -o hello_cpp hello.cpp
```

The `file` command on the `hello_cpp` binary should produce similar results as the C example.

Note: Any changes to your gcc compilers requires you to reinstall the HPC SDK.

In a typical local installation, the default installation base directory is `/opt/nvidia/hpc_sdk`.

If you choose to perform a network installation, the installation location must be on a shared file system accessible to all nodes.

To Prepare for the Installation:

- After downloading the HPC SDK installation package, bring up a shell command window on your system.

  The installation instructions assume you are using csh, sh, ksh, bash, or some compatible shell. If you are using a shell that is not compatible with one of these shells, appropriate modifications are necessary when setting environment variables.

- Verify you have enough free disk space for the HPC SDK installation.
  - The uncompressed installation packages requires 9.5 GB of total free disk space for the HPC SDK slim packages, and 20 GB for the HPC SDK multi pacakges.

Chapter 2. Installation Steps for Linux

Follow these instructions to install the software:

1. Unpack the HPC SDK software. In the instructions that follow, replace `<tarfile>` with the name of the file that you downloaded. Use the following command sequence to unpack the tar file before installation.

   ```
   % tar xpfz <tarfile>.tar.gz
   ```

   The tar file will extract an `install` script and an `install_components` folder to a directory with the same name as the tar file.

2. Run the installation script(s). Install the compilers by running `[sudo] ./install` from the `<tarfile>` directory.

   Important: The installation script must run to completion to properly install the software.

   To successfully run this script to completion, be prepared to do the following:

   - Determine the type of installation to be performed.
   - Define where to place the installation directory. The default is `/opt/nvidia/hpc_sdk`.

   Note: Linux users have the option of automating the installation of the HPC compiler suite without interacting with the usual prompts. This may be useful in a large institutional setting, for example, where automated installation of HPC compilers over many systems can be efficiently done with a script.

   To enable the silent installation feature, set the appropriate environment variables prior to running the installation script. These variables are as follows:

NVHPC_SILENT	(required) Set this variable to “true” to enable silent installation.
NVHPC_INSTALL_DIR	(required) Set this variable to a string containing the desired installation location, e.g. /opt/nvidia/hpc_sdk.
NVHPC_INSTALL_TYPE	(required) Set this variable to select the type of install. The accepted values are “single” for a single system install, “network” for a network install, or “auto” for a installation suitable for both.
NVHPC_DEFAULT_CUDA	(optional) Set this variable to the desired CUDA version in the form of XX.Y, e.g. ${OLDEST_CUDA_SHIPPED} or ${NEWEST_CUDA_SHIPPED}
NVHPC_STDPAR_CUDACC	(optional) Set this variable to force C++ stdpar GPU-compilation to target a specific compute capability by default, e.g. 60, 70, 75, etc.

The HPC SDK installation scripts install all of the binaries, tools, and libraries for the HPC SDK in the appropriate subdirectories within the specified installation directory. If the “auto” installation type is selected, some configuration files will also be created in $HOME/.config/NVIDIA the first time the HPC SDK is used on a system. If your home directory is not accessible or not writable on some nodes, do not select the “auto” installation type.

3. Review documentation.

   NVIDIA HPC Compiler documentation is available online in both HTML and PDF formats.

4. Complete network installation tasks.

Note: Skip this step if you are not installing a network installation.

For a network installation, you must run the local installation script on each system on the network where the compilers and tools will be available for use.

If your installation base directory is /opt/nvidia/hpc_sdk then run the following command on each system on the network.

```
/opt/nvidia/hpc_sdk/$NVARCH/26.11/compilers/bin/add_network_host
```

This command creates a system-dependent file localrc.machinename in the /opt/nvidia/hpc_sdk/$NVARCH/26.11/compilers/bin directory.

Installation of the HPC SDK for Linux is now complete. For assistance with difficulties related to the installation, please reach out on the NVIDIA Developer Forums.

The following sections contain information detailing the directory structure of the HPC SDK installation, and instructions for end-users to initialize environment and path settings to use the compilers and tools.

Chapter 3. End-user Environment Settings

After the software installation is complete, each user's shell environment must be initialized to use the HPC SDK.

Note: Each user must issue the following sequence of commands to initialize the shell environment before using the HPC SDK.

The HPC SDK keeps version numbers under an architecture type directory, e.g. `Linux_x86_64/26.11`. The name of the architecture is in the form of `` `uname -s`_`uname -m` ``. For Arm Server platforms the expected architecture name is "Linux_aarch64." The guide below sets the value of the necessary uname commands to "NVARCH", but you can explicitly specify the name of the architecture if desired.

To make the HPC SDK available:

In csh, use these commands:

```
% setenv NVARCH `uname -s`_`uname -m`
% setenv NVCOMPILERS /opt/nvidia/hpc_sdk
% setenv MANPATH "$MANPATH":$NVCOMPILERS/$NVARCH/26.11/compilers/man
% set path = ($NVCOMPILERS/$NVARCH/26.11/compilers/bin $path)
```

In bash, sh, or ksh, use these commands:

```
$ NVARCH=`uname -s`_`uname -m`; export NVARCH
$ NVCOMPILERS=/opt/nvidia/hpc_sdk; export NVCOMPILERS
$ MANPATH=$MANPATH:$NVCOMPILERS/$NVARCH/26.11/compilers/man; export MANPATH
$ PATH=$NVCOMPILERS/$NVARCH/26.11/compilers/bin:$PATH; export PATH
```

Once the compilers are available, you can make the OpenMPI commands and man pages accessible using these commands.

```
% set path = ($NVCOMPILERS/$NVARCH/26.11/comm_libs/mpi/bin $path)
% setenv MANPATH "$MANPATH":$NVCOMPILERS/$NVARCH/26.11/comm_libs/mpi/man
```

And the equivalent in bash, sh, and ksh:

```
$ export PATH=$NVCOMPILERS/$NVARCH/26.11/comm_libs/mpi/bin:$PATH
$ export MANPATH=$MANPATH:$NVCOMPILERS/$NVARCH/26.11/comm_libs/mpi/man
```

Alternatively, the HPC SDK also provides environment modules to configure the shell environment. In csh, use these commands:

```
% setenv MODULEPATH $NVCOMPILERS/modulefiles:"$MODULEPATH"
% module load nvhpc
```

And the equivalent in bash, sh, and ksh:

```
$ export MODULEPATH=$NVCOMPILERS/modulefiles:$MODULEPATH
$ module load nvhpc
```

Notices

Notice and Disclaimers

All information provided in this document is provided as-is, for your informational purposes only and is subject to change at any time without notice. Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing. To obtain the latest information, please contact your NVIDIA representative. Product or service performance varies by use, configuration and other factors. Your costs and results may vary. No product or component is absolutely secure. TO THE FULLEST EXTENT PERMITTED BY APPLICABLE LAW, NVIDIA DISCLAIMS ALL WARRANTIES AND REPRESENTATIONS OF ANY KIND, WHETHER EXPRESS, IMPLIED OR STATUTORY, RELATING TO OR ARISING UNDER THIS DOCUMENT, INCLUDING, WITHOUT LIMITATION, THE WARRANTIES OF TITLE, NONINFRINGEMENT, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, USAGE OF TRADE AND COURSE OF DEALING. NVIDIA products are not intended or authorized for use as critical components in a system or application where the use of or failure of such system or application developed with products, technology, software or services provided by NVIDIA could result in injury, death or catastrophic damage.

Except for your permitted use of the information contained in this document, no license or right is granted by implication, estoppel or otherwise. If this document directly includes or links to third-party websites, products, services or information, please consult other sources to evaluate if and how to use that information since NVIDIA does not support, endorse or assume any responsibility for any third party offerings or its accuracy or usefulness.

TO THE FULLEST EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY (I) INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, OR (II) DAMAGES FOR THE (A) COST OF PROCURING SUBSTITUTE GOODS OR (B) LOSS OF PROFITS, REVENUES, USE, DATA OR GOODWILL ARISING OUT OF OR RELATED TO THIS DOCUMENT, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), STRICT LIABILITY, OR OTHERWISE, AND EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND EVEN IF A PARTY'S REMEDIES FAIL THEIR ESSENTIAL PURPOSE. ADDITIONALLY, TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, NVIDIA'S TOTAL CUMULATIVE AGGREGATE LIABILITY FOR ANY AND ALL LIABILITIES, OBLIGATIONS OR CLAIMS ARISING OUT OF OR RELATED TO THIS DOCUMENT WILL NOT EXCEED FIVE U.S. DOLLARS (US$5).

Statements in this document that refer to future plans or expectations are forward-looking statements. These statements are based on currently available information, beliefs, assumptions and involve many risks and uncertainties that could cause actual results to differ materially from those expressed or implied in these statements. For more information on the factors that could cause actual results to differ materially, see our most recent earnings release and SEC filings at NVIDIA Corporation SEC Filings.

© NVIDIA Corporation. All rights reserved. NVIDIA, the NVIDIA logo, and other NVIDIA marks are trademarks of NVIDIA Corporation or its affiliates. Other names and brands may be claimed as the property of others.

Trademarks

NVIDIA, the NVIDIA logo, CUDA, CUDA-X, GPUDirect, HPC SDK, NGC, NVIDIA Volta, NVIDIA DGX, NVIDIA Nsight, NVLink, NVSwitch, and Tesla are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

©2022-2026, NVIDIA Corporation & affiliates. All rights reserved