## Atlas 300 AI Accelerator Card
## 1.0.0

# DDK Installation Guide (CentOS, ARM) (Model 3000)

| | |
|---|---|
| **Issue** | 02 |
| **Date** | 2020-03-30 |

HUAWEI TECHNOLOGIES CO., LTD.

## Trademarks and Permissions

HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

## Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

# Contents

# 1 Introduction

This document describes the method of separately installing the device development kit (DDK) and troubleshooting of common problems. It is applicable to development in command line interface (CLI) mode.

The DDK is an NPU-based digital developer kit that can be used to build a compilation environment for projects. Different release packages integrate the DDKs of different NPU forms. The DDK of the current version contains the dependency libraries and header files of components such as the TE, DVPP, and process orchestration. Users can compile the project files by using **makefile**.

# 2 Obtaining Software Packages

Before installing the tool, prepare the following software package:

DDK installation package: Visit **https://support.huawei.com/enterprise**, input the product name in the search box, and press **Enter** or click the search icon. In the displayed window, click the **Software Download** tab to obtain the desired installation package.

**Table 2-1** describes the required software package.

**Table 2-1** Overview of the software package

| Installation Package | Content | Integrity Verification File for the Software Package |
|---|---|---|
| MSpore_DDK-{version}-aarch64.centOS7.6-aarch64.centOS7.6-aarch64.miniOS.tar.gz | DDK installation package | MSpore_DDK-{version}-aarch64.centOS7.6-aarch64.centOS7.6-aarch64.miniOS.tar.gz.asc |

# 3 Verifying the Software Package Integrity

To prevent a software package from being maliciously tampered with during transmission or storage, download the corresponding digital signature file for integrity verification when downloading the software package.

After the software package is downloaded, verify its PGP digital signature according to the OpenPGP Signature Verification Guide. If the software package fails the verification, do not use the software package, and contact Huawei technical support engineers.

Before a software package is used in installation or upgrade, its digital signature also needs to be verified according to the OpenPGP Signature Verification Guide to ensure that the software package is not tampered with.

Visit either of the following websites to obtain the OpenPGP Signature Verification Guide:

- Carrier customers: **https://support.huawei.com/carrier/digitalSignatureAction**
- Enterprise customers: **https://support.huawei.com/enterprise/en/tool/pgp-verify-TL1000000054**

# 4 Environment Preparation

## Environment Requirements

The DDK installation must meet the following environment requirements on the hardware and operating system (OS).

**Table 4-1** CentOS version information

| Category | Version | Obtaining Method | Precautions |
|---|---|---|---|
| OS | 7.6 | Download the following recommended version from **http://vault.centos.org/altarch/7.6.1810/isos/aarch64/**:<br>**CentOS-7-aarch64-Everything-1810.iso** | The DDK installation and offline installation packages are adapted and tested based on the Infrastructure Server of CentOS. Therefore, you must use the Infrastructure Server software for installation. Other software, for example, Minimal Install or Development and Creative Workstation, cannot be used. |
| Python | Python 2: 2.7+<br>Python3: 3.5+ | For details, see **Installing Dependencies**. | Install Python 2.7 or later for Python2 and Python 3.5 or later for Python3. Both Python versions need to be installed. |

## (Optional) Preparing the DDK Installation User

You can install the DDK as any user (**root** or non-**root**).

- If the **root** user is used for installation, skip this section.
- If you use an existing non-**root** user to install the DDK, run the following command as the **root** user to grant the existing user with permission 750 on the **$HOME** directory:
  ```
  chmod 750 /home/username
  ```
- If you want to install the DDK as a new non-**root** user, perform the following steps as the **root** user: This document describes how to install the DDK as a new non-**root** user.
  a. Run the following command to create a user for DDK installation and configure the **$HOME** directory of the user:
  ```
  useradd -d /home/username -m username
  ```
  b. Run the following command to set the password:
  ```
  passwd username
  ```
  c. Run the following command to set the permission:
  ```
  chmod 750 /home/username
  ```

  NOTE

  *username* indicates the user name for installing the DDK. The **umask** value of the user cannot be greater than **0027**.
  - To view the **umask** value, run the **umask** command.
  - To change the **umask** value, run the **umask** *New value* command.

  If the created non-**root** user is **HwHiAiUser**, the DDK is deployed on the host server, and **HwHiAiUser** is used for offline model conversion, logs are stored in the **var/dlog** directory on the host by default. To enable the output of logs to the screen by default, you can set the following environment variable:
  ```
  export  SLOG_PRINT_TO_STDOUT=1
  ```
- Change the size of the file descriptor.
  a. Open the system configuration file **/etc/security/limits.conf** as the **root** user by running the following command:
  ```
  vi /etc/security/limits.conf
  ```
  b. Add the following content to the end of the **limits.conf** file:
  ```
  * soft nproc 65535
  * hard nproc 65535
  * soft nofile 65535
  * hard nofile 65535
  ```

  NOTE

  ***** indicates all users. You are advised to replace it with the name of the user who installs the Mind Studio project.

  c. Run the **:wq!** command to save the file and exit.

## (Optional) Configuring Permissions of the DDK Installation User

When a non-**root** user is used for installation, perform the following operations. Skip the following operations if the **root** user is used for installation.

Before installing the DDK, you need to download the dependent software with the **sudo yum** permission. Perform the following operations as the **root** user:

1. Run the following commands to open the **/etc/sudoers** file:

```
chmod u+w /etc/sudoers
vi /etc/sudoers
```

2. Under **root ALL=(ALL:ALL) ALL** in the file, add the following content:

```
username ALL=(ALL:ALL) NOPASSWD:SETENV: /usr/bin/yum,/bin/make install,/bin/ln -
s /usr/local/python3/bin/python3.5 /usr/bin/python3,/bin/ln -s /usr/local/python3/bin/
python3.5 /usr/bin/python3.5,/bin/ln -s /usr/local/python3/bin/pip3 /usr/bin/pip3,/bin/ln -
s /usr/local/python3/bin/pip3 /usr/bin/pip3.5
```

   Replace **username** with the name of the common user who executes the installation script.

   NOTE

   When you run other **sudo** commands, such as **sudo rm -rf python3** for deleting the soft link to Python 3, the following error may be reported:

   Sorry, user sam is not allowed to execute '/bin/rm -rf python3' as root on localhost.localdomain.

   Open the **/etc/sudoers** file as the **root** user, add the **/bin/rm -rf python3** command to the last line of 2 to obtain the operation permissions, and switch to the common user.

3. Run the **:wq!** command to save the file.
4. Run the following command to remove the write permission on the **/etc/sudoers** file:

```
chmod u-w /etc/sudoers
```

NOTE

Ensure that the last line of the **/etc/sudoers** file is **#includedir /etc/sudoers.d**. Otherwise, add it manually.

## Configuring the Network

If CentOS cannot be connected to the network through a remote terminal after CentOS is started, perform the following steps to configure the network adapter:

Run the following command as the **root** user to open the NIC configuration file:

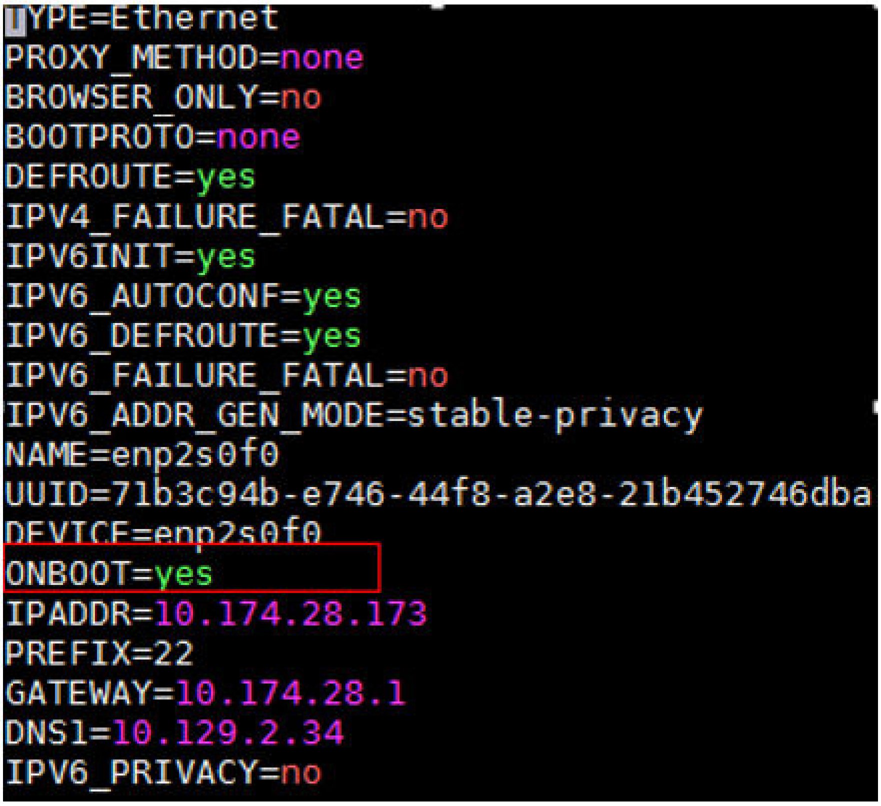
```
vi /etc/sysconfig/network-scripts/ifcfg-enp2s0f0
```

NOTE

Replace **ifcfg-enp2s0f0** with the actual name of the network adapter in use.

Change the value of **ONBOOT** to **yes**, and set network parameters as shown in the following figure.

**Figure 4-1** Setting network parameters



```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=enp2s0f0
UUID=71b3c94b-e746-44f8-a2e8-21b452746dba
DEVICE=enp2s0f0
ONBOOT=yes
IPADDR=10.174.28.173
PREFIX=22
GATEWAY=10.174.28.1
DNS1=10.129.2.34
IPV6_PRIVACY=no
```

NOTE

**BOOTPROTO** indicates the IP address type of the device. For a static IP address, set this parameter to **static** or **none**. For a dynamic IP address, set this parameter to **dhcp** so that the IP address can be obtained automatically.

Save the file and exit. Run the following command to restart the network service:

```
service network restart
```

## Configuring the Sources

DDK installation requires related dependencies, which must be downloaded in advance. Ensure that the server where the DDK is installed can successfully connect to the network. Perform the following steps as the **root** user:

**Step 1** Run the following command to check source validity:

```
yum makecache
```

If an error is reported during the command execution, check whether the network connection is normal or replace the following content with the source in the **/etc/yum.repos.d/CentOS-Base.repo** file:

```
[base]
 name=CentOS-$releasever - Base
 mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os&infra=$infra
 gpgcheck=1
 gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
        file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7-aarch64
 #released updates
 [updates]
 name=CentOS-$releasever - Updates
 mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=
$basearch&repo=updates&infra=$infra
 gpgcheck=1
 gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
        file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7-aarch64
 #additional packages that may be useful
 [extras]
 name=CentOS-$releasever - Extras
 mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=extras&infra=
$infra
 gpgcheck=1
 gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
       file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7-aarch64
 enabled=1
 #additional packages that extend functionality of existing packages
 [centosplus]
 name=CentOS-$releasever - Plus
 mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=
$basearch&repo=centosplus&infra=$infra
 gpgcheck=1
 enabled=0
 gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
       file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7-aarch64
```

**Step 2** Configure a special source to support Python download and installation.

```
yum install -y epel-release
```

**NOTE**

If **epel-release** fails to be installed by using the **yum** source, rectify the fault by referring to **FAQs > What Do I Do If epel-release Installation Fails?**.

**Step 3** Run the following commands to update the software information:

```
yum clean all
yum makecache
```

**----End**

## Installing Dependencies

Run the **su -** ***username*** command to switch to the DDK installation user and perform the following operations to install components such as the GCC and Python on which the DDK depends:

**Step 1** Install Python 3.5.

CentOS does not provide a proper source for installing Python 3.5. Therefore, install Python 3.5 by using an installation package.

1. Install the dependencies that may be used in Python 3.5 installation: (If a command line is wrapped, copy the command to Word or Notepad, replace the line wrap point with a space to merge the lines into one and copy it back to the server for execution.)
```
sudo -E yum install -y openssl-devel zlib-devel bzip2-devel expat-devel gdbm-devel gcc readline-devel
sqlite-devel ncurses-devel
```
2. Run the **wget** command to download the Python 3.5 installation package to any directory on the CentOS server:

```
wget https://www.python.org/ftp/python/3.5.1/Python-3.5.1.tgz
```

3. Run the following command to go to the download directory and decompress the installation package:

```
tar -zxvf Python-3.5.1.tgz
```

4. Go to the decompressed folder and run the following configuration, compilation, and installation commands:

```
cd Python-3.5.1
./configure --prefix=/usr/local/python3
make
sudo make install
```

5. Run the following commands to set the soft link:

```
sudo ln -s /usr/local/python3/bin/python3.5 /usr/bin/python3
sudo ln -s /usr/local/python3/bin/python3.5 /usr/bin/python3.5
sudo ln -s /usr/local/python3/bin/pip3 /usr/bin/pip3
sudo ln -s /usr/local/python3/bin/pip3 /usr/bin/pip3.5
```

6. After the installation is complete, run the following commands to check the installation version. If the required version information is displayed, the installation is successful.

```
python3 -V
pip3 -V
```

**Step 2** Run the following command to install the dependencies:

```
sudo -E yum install -y gcc-c++ cmake  make python python-devel python2-pip gcc-gfortran
```

**Step 3** Install the Python dependencies.

- Before installing the software on which Python 2.7 depends, ensure that the preceding operations are successful. Run the **pip2 -V** command to check the pip version. If the following information is displayed, it indicates that Python 2.7 has been installed correctly:

  ```
  pip 8.1.2 from /usr/lib/python2.7/site-packages (python 2.7)
  ```

  The preceding information indicates that pip has been installed correctly. Run the following commands to install the software dependencies required by Python 2.7 (NumPy versions 1.11.0–1.16.0 are supported and the following uses **1.16.0** as an example):

  ```
  pip2 install numpy==1.16.0 --user
  pip2 install decorator --user
  pip2 install setuptools --upgrade --user
  ```

- Before installing the software on which Python 3.5 depends, run the **pip3.5 -V** command to check the version information. If the following information is displayed, it indicates that Python 3.5 has been installed correctly.

  ```
  pip 7.1.2 from /usr/lib/python3.5/site-packages (python 3.5)
  ```

  The preceding information indicates that pip3.5 has been installed correctly. Run the following commands to install the software dependencies required by Python 3.5 (NumPy versions 1.11.0–1.16.0 are supported and the following uses **1.16.0** as an example):

  ```
  pip3.5 install numpy==1.16.0 --user
  pip3.5 install decorator --user
  pip3.5 install setuptools --upgrade --user
  ```

NOTE

If the message "Import Error:cannot import name main" is displayed when you run the **pip3.5 -V** command, rectify the fault by referring to **6.1 What Do I Do If the Message "Import Error:cannot import name main" Is Displayed When the pip3/pip3.5 -V Command Is Run after pip3/pip3.5 Upgrade?**.

**----End**

# 5 Installing the DDK

This section describes the installation method of the DDK and common operations.

This section describes how to install and use the DDK.

## 5.1 Installation

### Prerequisites

The operations required in **4 Environment Preparation** and **3 Verifying the Software Package Integrity** have been completed.

### Procedure

Switch to the DDK installation user and perform the following operations:

**Step 1** Run the following command to decompress the installation package:

```
tar -zxvf MSpore_DDK****tar.gz
```

In actual operations, replace *MSpore_DDK****.tar.gz.asc* with the actual installation package. For details about the decompressed files, see **Table A-1**.

**Step 2** Run the following command to install the DDK:

```
bash install.sh DDK installation directory
```

The DDK installation directory is automatically created during the installation. For example, if the installation directory is set to **$HOME/tools/che/ddk**, the **tools/che/ddk** directory is automatically created during the installation. Alternatively, you can specify an installation path.

If the message "Successfully installed the DDK!" is displayed, the DDK has been installed successfully.

NOTE

- If the message "Warning: DDK functions except TE have been successfully installed." is displayed during the installation, rectify the fault by referring to **6.3 What Do I Do If the Message "Warning: DDK functions except TE have been successfully installed." Is Displayed During DDK Installation?**.
- After the DDK is installed, the profiling function under the installation path **$HOME/tools/che/ddk/ddk/toolchains/profiler** is unavailable. If you need to use the profiling function, install Mind Studio and the DDK by referring to *Ascend 310 Mind Studio Installation Guide* (select an installation guide based on the Mind Studio server).

**----End**

## 5.2 Common Operations

This section describes how to install and use the DDK.

### 5.2.1 Uninstalling the DDK

To update the DDK version, uninstall the current version and then install a new version according to the installation procedure described in 5.1. This section describes how to uninstall the DDK.

Go to the DDK installation path, for example, *$HOME/tools/che/ddk*/**ddk/scripts**. Run the following command as the DDK installation user to uninstall the DDK:

```
./uninstall.sh
```

If the following information is displayed, the DDK is uninstalled successfully:

```
Info: ide_daemon pem uninstall succ
Starting to remove ddk dir.
```

### 5.2.2 Querying the DDK Version

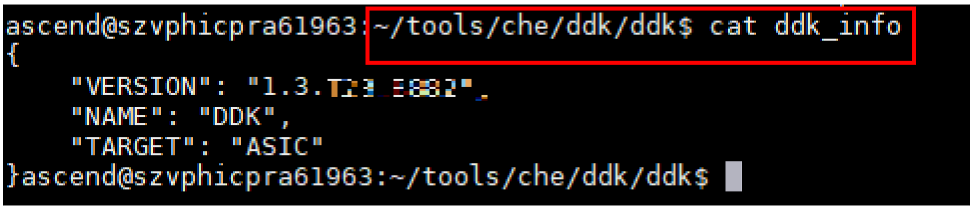
After the DDK is installed, you can view the DDK version number in the DDK installation directory.

For example, if the installation directory of the DDK is *$HOME/tools/che/ddk*, run the following command in the *$HOME/tools/che/ddk* directory to view the DDK version number:

```
cat ddk_info
```

**Figure 5-1** shows the return result.

**Figure 5-1** Querying the DDK version



```
ascend@szvphicpra61963:~/tools/che/ddk/ddk$ cat ddk_info
{
    "VERSION": "1.3.[redacted]",
    "NAME": "DDK",
    "TARGET": "ASIC"
}ascend@szvphicpra61963:~/tools/che/ddk/ddk$
```

In **Figure 5-1**:

- **VERSION**: Indicates the version number of the DDK. The query result varies according to the actual situations.
- **TARGET**: Indicates the running environment of the DDK, which is **ASIC** or **Atlas DK**.

## 5.2.3 Configuring the OpenPGP Public Key

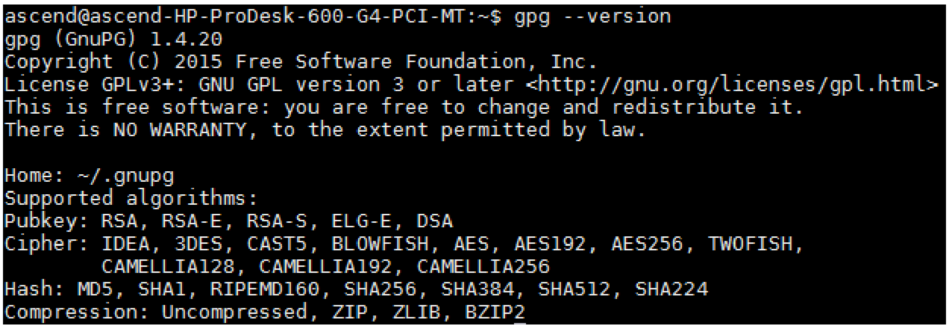
### Prerequisites

- The public key is configured by the DDK installation user.
- The GnuPG tool is installed on Linux.

  **Verification:**

  – If the GnuPG tool has been installed, run the **gpg --version** command in the shell. Information in **Figure 5-2** is displayed:

  **Figure 5-2** Command output



```
ascend@ascend-HP-ProDesk-600-G4-PCI-MT:~$ gpg --version
gpg (GnuPG) 1.4.20
Copyright (C) 2015 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Home: ~/.gnupg
Supported algorithms:
Pubkey: RSA, RSA-E, RSA-S, ELG-E, DSA
Cipher: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
        CAMELLIA128, CAMELLIA192, CAMELLIA256
Hash: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compression: Uncompressed, ZIP, ZLIB, BZIP2
```

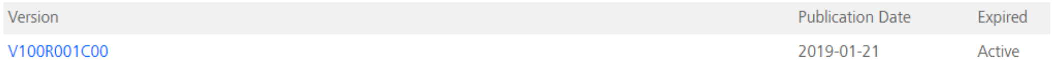
  – If the GnuPG tool is not installed, install the tool by following the instructions provided in its official website **https://www.gnupg.org/**.

### Configuring a Public Key

**Step 1** Obtain the public key file.

Go to the **OpenPGP download page** and click the download link, as shown in **Figure 5-3**. The file download page is displayed.

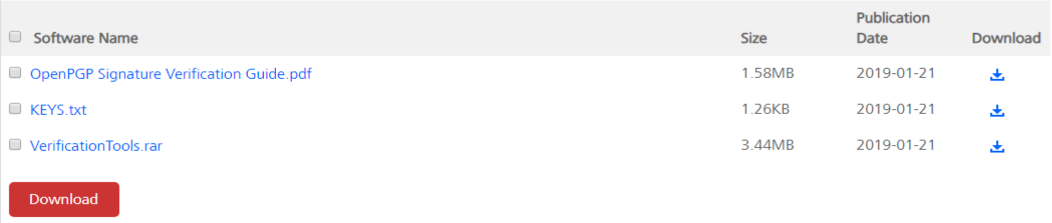
**Figure 5-3** Click to download the file.



| Version | Publication Date | Expired |
|---|---|---|
| V100R001C00 | 2019-01-21 | Active |

The **KEYS** file is the public key file, as shown in **Figure 5-4**.

**Figure 5-4** Selecting the KEYS file



| □ Software Name | Size | Publication Date | Download |
|---|---|---|---|
| □ OpenPGP Signature Verification Guide.pdf | 1.58MB | 2019-01-21 | ⤓ |
| □ KEYS.txt | 1.26KB | 2019-01-21 | ⤓ |
| □ VerificationTools.rar | 3.44MB | 2019-01-21 | ⤓ |

NOTE

If you want to switch to the English interface, click Worldwide in the upper right corner.

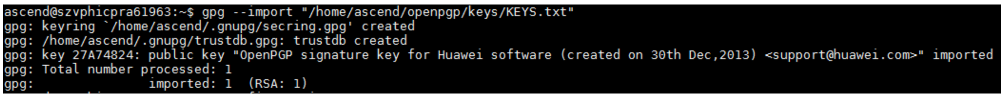
**Step 2** Upload the downloaded **KEYS.txt** file to the Linux OS where the DDK is located, for example, **/home/*username*/openpgp/keys**.

**Step 3** Import the public key file.

Run the following command to go to the directory that contains the public key file **KEYS**.

```
gpg --import "/home/username/openpgp/keys/KEYS.txt"
```

**Figure 5-5** Import the public key file.



```
ascend@szvphicpra61963:~$ gpg --import "/home/ascend/openpgp/keys/KEYS.txt"
gpg: keyring `/home/ascend/.gnupg/secring.gpg' created
gpg: /home/ascend/.gnupg/trustdb.gpg: trustdb created
gpg: key 27A74824: public key "OpenPGP signature key for Huawei software (created on 30th Dec,2013) <support@huawei.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

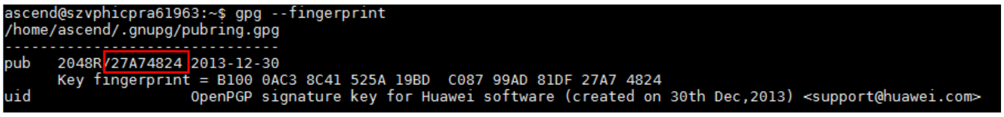
NOTE

**/home/*username*/openpgp/keys** indicates the absolute path of the public key file **KEYS**. ***username*** must be replaced with the name of the DDK installation user.

**Step 4** Run the following command to view the import result:

```
gpg --fingerprint
```

**Figure 5-6** Checking the import result



```
ascend@szvphicpra61963:~$ gpg --fingerprint
/home/ascend/.gnupg/pubring.gpg
-------------------------------
pub   2048R/27A74824 2013-12-30
      Key fingerprint = B100 0AC3 8C41 525A 19BD  C087 99AD 81DF 27A7 4824
uid                  OpenPGP signature key for Huawei software (created on 30th Dec,2013) <support@huawei.com>
```

**Step 5** Verify the public key.

- The validity of the OpenPGP public key must be verified based on the public key ID, fingerprint, UID, and the publisher of the public key. The published information of the OpenPGP public key is as follows:
  - Public key ID: 27A74824
  - Key fingerprint: B100 0AC3 8C41 525A 19BD C087 99AD 81DF 27A7 4824
  - UID: OpenPGP signature key for Huawei software (created on 30th Dec, 2013) <support@huawei.com>

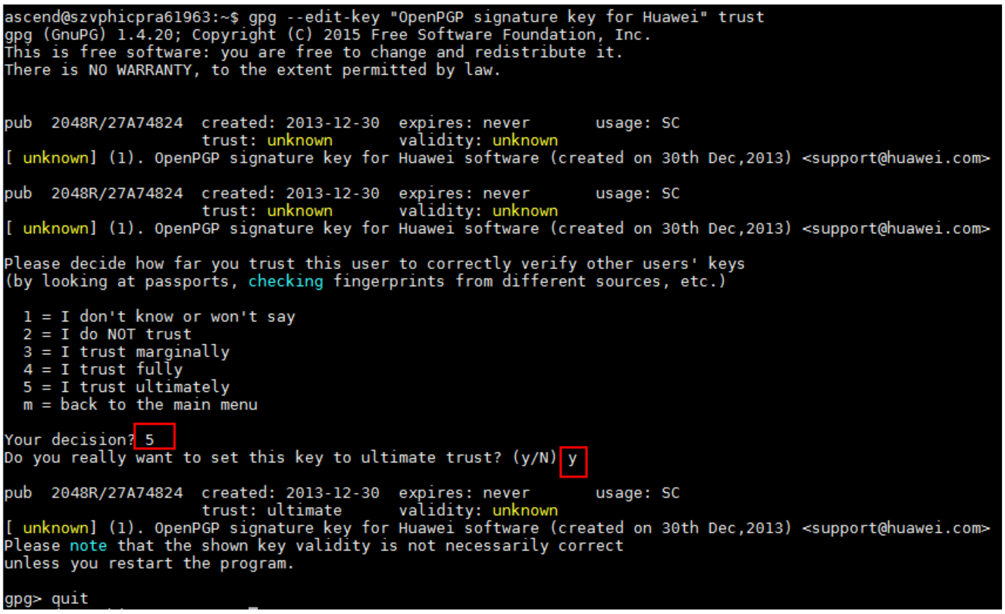
  After the information is verified, you can set the trust level for the public key

- by running the following command:

  ```
  gpg --edit-key "OpenPGP signature key for Huawei software" trust
  ```

  Information similar to the following is displayed. Enter **5** after **Your decision?**, which indicates **I trust ultimately**. Enter **y** after **Do you really want to set this key to ultimate trust? (y/N)**.

**Figure 5-7** Setting the trust level for the public key



```
ascend@szvphicpra61963:~$ gpg --edit-key "OpenPGP signature key for Huawei" trust
gpg (GnuPG) 1.4.20; Copyright (C) 2015 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.


pub  2048R/27A74824  created: 2013-12-30  expires: never       usage: SC
                     trust: unknown       validity: unknown
[ unknown] (1). OpenPGP signature key for Huawei software (created on 30th Dec,2013) <support@huawei.com>

pub  2048R/27A74824  created: 2013-12-30  expires: never       usage: SC
                     trust: unknown       validity: unknown
[ unknown] (1). OpenPGP signature key for Huawei software (created on 30th Dec,2013) <support@huawei.com>

Please decide how far you trust this user to correctly verify other users' keys
(by looking at passports, checking fingerprints from different sources, etc.)

  1 = I don't know or won't say
  2 = I do NOT trust
  3 = I trust marginally
  4 = I trust fully
  5 = I trust ultimately
  m = back to the main menu

Your decision? 5
Do you really want to set this key to ultimate trust? (y/N) y

pub  2048R/27A74824  created: 2013-12-30  expires: never       usage: SC
                     trust: ultimate      validity: unknown
[ unknown] (1). OpenPGP signature key for Huawei software (created on 30th Dec,2013) <support@huawei.com>
Please note that the shown key validity is not necessarily correct
unless you restart the program.

gpg> quit
```

**Step 6** Run the **quit** command to exit.

**----End**

# 6 FAQs

## 6.1 What Do I Do If the Message "Import Error:cannot import name main" Is Displayed When the pip3/pip3.5 -V Command Is Run after pip3/pip3.5 Upgrade?

### Symptom

After pip3/pip3.5 upgrade, the Python library functions are changed. Therefore, the "Import Error:cannot import name main" error occurs upon a pip3/pip3.5 command.

### Solution 1

Run the following upgrade command:

- For Python 3:
```
python3 -m pip install --upgrade pip --user
```
- For Python 3.5:
```
python3.5 -m pip install --upgrade pip --user
```

If the fault persists, follow **Solution 2**.

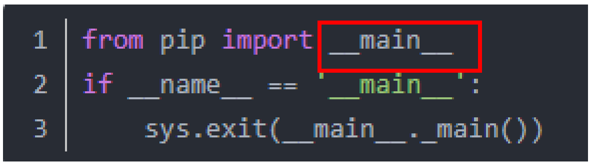
### Solution 2

For Python 3:

Open **vim /usr/bin/pip3**.

Modify the code as shown in **Figure 6-1**.

**Figure 6-1** Modifying pip3 code



```
from pip import __main__
if __name__ == '__main__':
    sys.exit(__main__._main())
```

The solution for Python 3.5 is the same as that for Python 3.

# 6.2 What Do I Do If epel-release Installation Fails?

If epel-release fails to be installed using the **sudo -E yum install -y epel-release** command, perform the following steps to manually install EPEL:

1. Run the following command to check if the system version matches the EPEL version:
   ```
   cat /etc/redhat-release
   ```
   If "CentOS Linux release 7.***" is returned, EPEL 7 must be installed.
2. Run the following command to download the EPEL installation package of the required version from the EPEL official website **https://dl.fedoraproject.org/pub/epel**:
   ```
   wget https://dl.fedoraproject.org/pub/epel/7Server/aarch64/Packages/e/epel-release-7-11.noarch.rpm --no-check-certificate
   ```
3. Run the following command to install EPEL (for example, EPEL 7):
   ```
   rpm -vih epel-release-7-11.noarch.rpm
   ```
4. Run the following command to update makecache:
   ```
   yum clean all && yum makecache
   ```

# 6.3 What Do I Do If the Message "Warning: DDK functions except TE have been successfully installed." Is Displayed During DDK Installation?

## Symptom

After the DDK installation script is executed, messages such as "error: install te-0.4.0.egg failed" and "error: install topi-0.4.0.egg failed" are displayed during the installation. However, the message "Warning: DDK functions except TE have been successfully installed." is displayed when the installation is complete. The DDK has been successfully installed, but TE functions are unavailable.

## Possible Cause

The **root** user adopts an earlier version (0.9.8) of setuptools. When the **root** user is switched to the DDK installation user to install the DDK, the system still reads the setuptools path of the **root** user. As a result, setuptools used by the DDK installation user does not take effect.

## Solution

Run the following command as the DDK installation user to set the environment variable for the setuptools run by the DDK installation user to take effect, and then reinstall the DDK:

```
export PATH=/usr/local/python3/bin:$HOME/bin:$HOME/.local/bin:$PATH
```

# A Overview of Software Packages

For details about the decompressed DDK installation package, see **Table A-1**.

**Table A-1** Overview of the software packages

| Installation Package | Content | Description |
|---|---|---|
| MSpore_DDK-{version}-<uihost arch.os>-<host arch.os>-<device arch.os>.tar.gz<br>For details about the parameter description, see **Table A-2**. | ddk.tar.gz | DDK installation package |
| | install.sh | Installation script |
| | check_sha.sh | Script used to verify the integrity of the preceding files. This script is automatically called to perform integrity check during the execution of **install.sh**. |

**Table A-2** Naming rules of the DDK installation package

| Parameter | Description |
|---|---|
| {version} | Version number |
| <uihost arch.os> | CPU architecture, OS, and version on the UI host side, for example, x86_64.ubuntu16.04 and x86_64.centOS7.4 |
| <host arch.os> | CPU architecture, OS, and version on the host side, for example, x86_64.ubuntu16.04 and x86_64.centOS7.4 |
| <device arch.os> | CPU architecture, OS, and version on the device side, for example, aarch64.ubuntu16.04 and aarch64.miniOS<br>● Device-side architecture of the DK form (Atlas 200 DK), for example, aarch64.ubuntu16.04<br>● Device-side architecture of the non-DK form (Atlas 300), for example, aarch64.miniOS |

# B Open Source Third-Party Libraries

## cereal

cereal is a header-only BSD-licensed C++11 serialization library. It is designed to be fast, light-weight, and easy to extend. cereal takes arbitrary data types and reversibly turns them into different representations, such as compact binary encodings, XML, or JSON. The version currently used is 1.2.2.

For details, visit the cereal official website at http://uscilab.github.io/cereal/.

## gflags

Google GFlags (GFlags), the Global Flags Editor, contains a Python-friendly C++ library that implements command-line flags processing, replacing systems like **getopt()**. The version currently used is 2.2.1.

For details, visit the GFlags official website at https://github.com/gflags/gflags.

## glog

Google glog is a library that implements application-level logging. This library provides logging APIs based on C++-style streams and various helper macros. Similar to assert defined in the standard C library, it provides more output information and flexibility.

For details, visit the glog official website at https://github.com/google/glog.

## opencv

OpenCV (short for Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. This cross-platform library sets its focus on real-time image processing, computer vision, and pattern recognition programs. The version currently used is 3.4.2.

For details, visit the OpenCV official website at https://opencv.org/.

## Protobuf

Protobuf (short for Protocol buffers) are Google's language-neutral, platform-neutral, extensible mechanism for serializing structured data. It is useful in developing programs to communicate with each other over a wire or for storing data. The version currently used is 3.5.1.

For details, visit the official website at https://developers.google.com/protocol-buffers/.

# C Change History

| Release Date | Description |
|---|---|
| 2020-05-30 | This issue is the first official release. |