

搭建环境

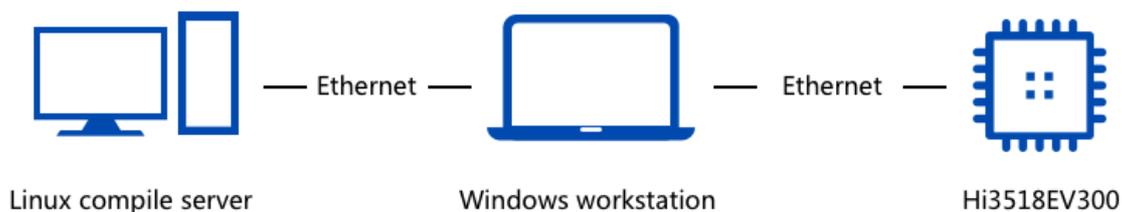
- [环境要求](#)
- [硬件要求](#)
- [软件系统要求](#)
- [Linux构建工具要求](#)
- [Windows开发工具要求](#)
- [安装编译基础环境](#)
- [连接Linux服务器](#)
- [将Linux shell改为bash](#)
- [安装Python环境](#)
- [安装文件打包工具](#)
- [安装编译工具环境](#)
- [安装gn](#)
- [安装ninja](#)
- [安装LLVM编译工具链](#)
- [安装hc-gen](#)
- [安装IDE开发工具](#)

环境要求

硬件要求

- Linux服务器
 - Windows工作台（主机电脑）
 - Hi3518EV300 IoT Camera开发板
 - USB转串口线、网线（Windows工作台通过USB转串口线、网线与开发板连接）
- 各硬件连接关系如下图所示。

图 1 硬件连线图



软件系统要求

表 1 开发板开发平台要求

硬件	软件	描述	备注
Linux编译服务器	操作系统	<p>Ubuntu16.04及以上64位系统版本，Shell使用bash。</p> <p>说明： 通常系统默认安装samba、vim等常用软件，需要做适当适配以支持linux文件共享。</p>	开发人员可以在Windows工作台中进行程序开发，或者远程登录到Linux服务器进行程序开发。
Windows工作台	操作系统	Windows XP/Windows7/Windows10	
	USB转串口驱动	<p>http://www.hihope.org/download/AllDocuments</p> <p>驱动软件名：USB-to-Serial Comm Port.exe</p>	

Linux构建工具要求

Linux服务器通用环境配置需要的工具及其获取途径如下表所示。

表 2 Linux服务器开发工具及获取途径

开发工具	用途	获取途径
Python3.7+	执行编译脚本工具	https://www.python.org/ftp/python/3.8.5/Python-3.8.5.tgz
bash	执行命令行工具	通过互联网获取
gn	产生ninja编译脚本	https://repo.huaweicloud.com/harmonyos/compiler/gn/1523/linux/gn.1523.tar
ninja	执行ninja编译脚本	https://repo.huaweicloud.com/harmonyos/compiler/ninja/1.9.0/linux/ninja.1.9.0.tar
LLVM	编译工具链	https://repo.huaweicloud.com/harmonyos/compiler/clang/9.0.0-34042/linux/llvm-linux-9.0.0-34042.tar

开发工具	用途	获取途径
hc-gen	驱动配置编译工具	https://repo.huaweicloud.com/harmonyos/compiler/hc-gen/0.65/linux/hc-gen-0.65-linux.tar
IP OP、PuTTY 或其他超级终端	连接Linux服务器工具（任选其一）	通过互联网获取（如： https://www.putty.org/ ）
Java 虚拟机环境	编译、调试和运行Java程序	通过apt-get install安装

开发工具	用途	获取途径
hmos_app_packing_tool	hap打包工具	https://repo.huaweicloud.com/harmonyos/develop_tools/hmos_app_packing_tool.jar
hapsegn_tool	hap签名工具	https://repo.huaweicloud.com/harmonyos/develop_tools/hapsegn_toolv2.jar

Windows开发工具要求

表 3 Windows工作台开发工具及获取途径

开发工具	用途	获取途径
Visual Studio Code	代码编辑工具	https://code.visualstudio.com/
HUAWEI DevEco Device Tool	<p>IDE开发工具，Visual Studio Code插件，支持代码编译、烧录和调试等功能，支持C/C++语言</p> <p>说明： HUAWEI DevEco Device Tool是OpenHarmony面向智能设备开发者提供的一站式集成开发环境，支持OpenHarmony的组件按需定制，支持代码编辑、编译、烧录、调试等功能，支持C/C++语言，以插件的形式部署在Visual Studio Code上。</p>	https://device.harmonyos.com/cn/ide

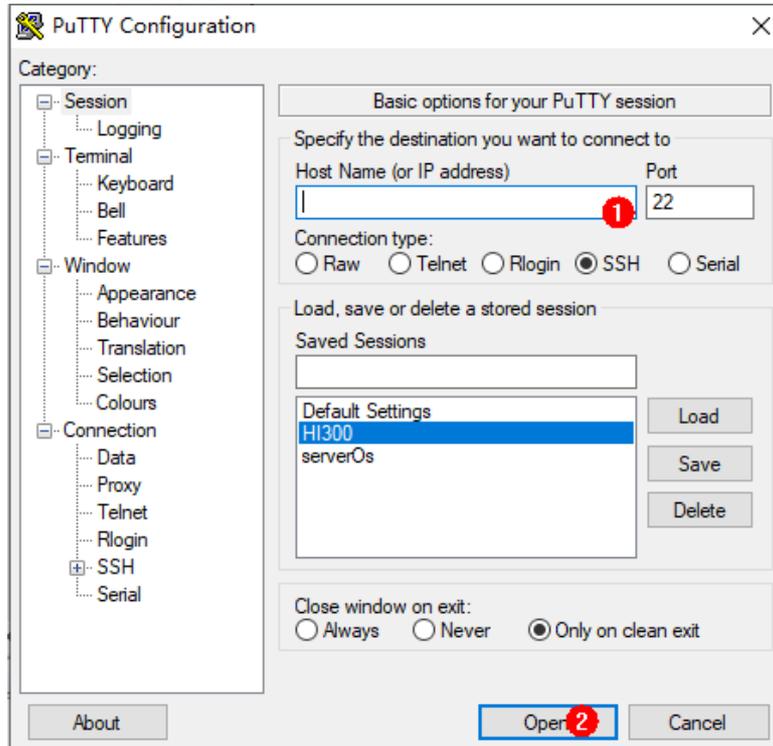
安装编译基础环境

连接Linux服务器

使用Windows远程登录Linux，以PUTTY工具为例：

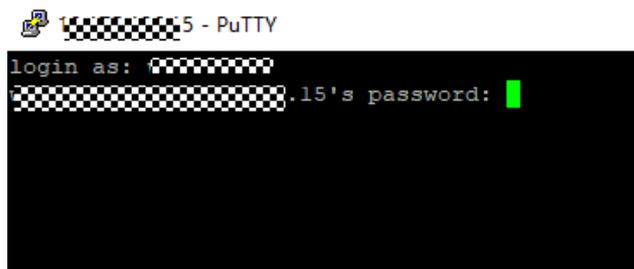
1. 打开[PuTTY工具](#)，输入Linux服务器IP地址，后点击“open”，打开连接。

图 2 PuTTY界面示例图



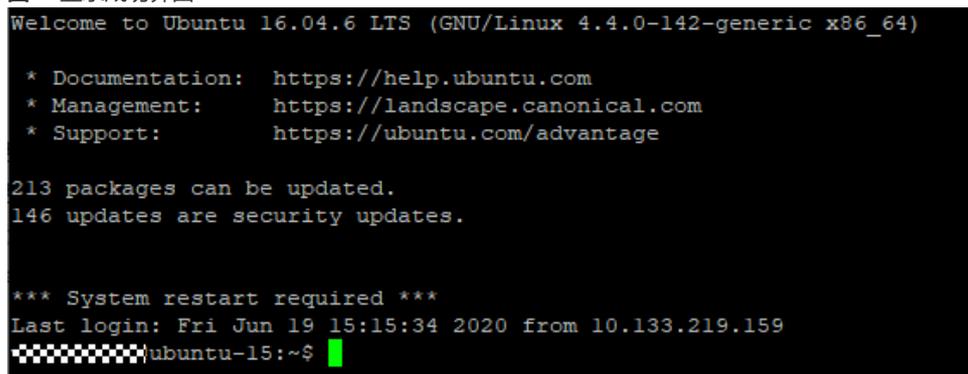
2. 弹框告警选择“是(Y)”。
3. 弹出界面中输入帐号并按回车键，继续输入密码并按回车键。

图 3 登录界面图



4. 登录成功。

图 4 登录成功界面



将Linux shell改为bash

查看shell是否为bash，在终端运行如下命令

```
ls -l /bin/sh
```

如果显示为“/bin/sh -> bash”则为正常，否则请按以下方式修改：

方法一：在终端运行如下命令，然后选择 no。

```
sudo dpkg-reconfigure dash
```

方法二：先删除sh，再创建软链接。

```
sudo rm -rf /bin/sh
sudo ln -s /bin/bash /bin/sh
```

安装Python环境

1. 打开Linux编译服务器终端。
2. 输入如下命令，查看python版本号，需使用python3.7以上版本。

```
python3 --version
```

以python3.8为例，按照以下步骤安装python。

1. 运行如下命令，查看Ubuntu版本：

```
cat /etc/issue
```

1. 根据Ubuntu不同版本，安装python。

- 如果Ubuntu 版本为18+，运行如下命令。

```
sudo apt-get install python3.8
```

- 如果Ubuntu版本为16，请下载包的方式安装python。

1. python安装环境依赖(gcc, g++, make, zlib, libffi)请运行如下命令：

```
sudo apt-get install gcc && sudo apt-get install g++ && sudo apt-get install make &&
sudo apt-get install zlib* && sudo apt-get install libffi-dev
```

1. 获取[python3.8.5安装包](#)，将其放至linux服务器中，运行如下命令：

```
tar -xvzf Python-3.8.5.tgz && cd Python-3.8.5 && sudo ./configure && sudo make &&
sudo make install
```

3. 确定python安装好后，将python路径链接到"/usr/bin/python"。

输入如下命令，查看python是否正确软链接到python3

```
python --version
```

如果回显不是python 3.8.5，则运行如下命令，查看python3.8所在目录：

```
which python3.8
```

将以下命令中的 "python3.8-path" 替换为 "which python3.8" 命令执行后的回显路径

```
cd /usr/bin && sudo rm python && sudo ln -s python3.8-path python && python --version
```

4. 安装并升级Python包管理工具 (pip3)，任选如下一种方式。

- **命令行方式：**

```
sudo apt-get install python3-setuptools python3-pip -y
sudo pip3 install --upgrade pip
```

- **安装包方式：**

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
python get-pip.py
```

5. 运行如下命令，安装python模块setuptools。

```
pip3 install setuptools
```

6. 安装GUI menuconfig工具 (Kconfiglib)，建议安装Kconfiglib 13.2.0+版本，任选如下一种方式。

o 命令行方式:

```
sudo pip3 install kconfiglib
```

- **安装包方式:**

1. 下载.whl文件（例如：kconfiglib-13.2.0-py2.py3-none-any.whl）。

下载路径：“[<https://pypi.org/project/kconfiglib/#files>]
(<https://pypi.org/project/kconfiglib/#files>)”

1. 运行如下命令，安装.whl文件。

...

```
sudo pip3 install kconfiglib-13.2.0-py2.py3-none-any.whl
```

...

安装文件打包工具

1. 打开Linux编译服务器终端。

2. 运行如下命令，安装dosfstools。

```
sudo apt-get install dosfstools
```

3. 运行如下命令，安装mtools。

```
sudo apt-get install mtools
```

4. 运行如下命令，安装mtd-utils。

```
sudo apt-get install mtd-utils
```

安装编译工具环境



须知:

- 如果后续通过“HPM组件方式”或“HPM包管理器命令行工具方式”获取源码，不需要安装gn、ninja、LLVM、hc-gen编译工具。
- （推荐）如果后续通过“镜像站点方式”或“代码仓库方式”获取源码，需要安装gn、ninja、LLVM、hc-gen编译工具。
安装gn、ninja、LLVM、hc-gen编译工具时，请确保编译工具的环境变量路径唯一。

安装gn

1. 打开Linux编译服务器终端。

2. [下载gn工具](#)。

3. 解压gn安装包至~/gn路径下。

```
tar -xvf gn.1523.tar -C ~/
```

4. 设置环境变量。

```
vim ~/.bashrc
```

将以下命令拷贝到.bashrc文件的最后一行，保存并退出。

```
export PATH=~/.gn:$PATH
```

5. 生效环境变量。

```
source ~/.bashrc
```

安装ninja

1. 打开Linux编译服务器终端。
2. [下载ninja工具](#)。
3. 解压ninja安装包至~/ninja路径下。

```
tar -xvf ninja.1.9.0.tar -C ~/
```

4. 设置环境变量。

```
vim ~/.bashrc
```

将以下命令拷贝到.bashrc文件的最后一行，保存并退出。

```
export PATH=~/.ninja:$PATH
```

5. 生效环境变量。

```
source ~/.bashrc
```

安装LLVM编译工具链

1. 打开Linux编译服务器终端。
2. [下载LLVM工具](#)。
3. 解压LLVM安装包至~/llvm路径下。

```
tar -xvf llvm-linux-9.0.0-34042.tar -C ~/
```

4. 设置环境变量。

```
vim ~/.bashrc
```

将以下命令拷贝到.bashrc文件的最后一行，保存并退出。

```
export PATH=~/.llvm/bin:$PATH
```

5. 生效环境变量。

```
source ~/.bashrc
```

安装hc-gen

1. 打开Linux编译服务器终端。
2. [下载hc-gen工具](#)。

3. 解压hc-gen安装包到Linux服务器~/hc-gen路径下。

```
tar -xvf hc-gen-0.65-linux.tar -C ~/
```

4. 设置环境变量。

```
vim ~/.bashrc
```

将以下命令拷贝到.bashrc文件的最后一行，保存并退出。

```
export PATH=~/hc-gen:$PATH
```

5. 生效环境变量。

```
source ~/.bashrc
```

安装IDE开发工具

OpenHarmony IDE工具DevEco Device Tool集成了终端工具的能力，工具的安装步骤请参考[DevEco Device Tool环境搭建、导入和配置工程](#)完成。