Atlas 300I 推理卡
5.1.RC2

NPU 驱动和固件安装指南（型号 3000）

文档版本	02
发布日期	2022-08-18

华为技术有限公司

版权所有 © 华为技术有限公司 2022。 保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明

HUAWEI和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： https://e.huawei.com

前言

概述

本文档详细的描述了软件包的安装、卸载等流程的具体操作指导，同时提供了常见的问题解答及故障处理方法。

读者对象

本文档主要适用于以下人员：

- 企业管理员
- 企业终端用户

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
危险	表示如不避免则将会导致死亡或严重伤害的具有高等级风险的危害。
警告	表示如不避免则可能导致死亡或严重伤害的具有中等级风险的危害。
注意	表示如不避免则可能导致轻微或中度伤害的具有低等级风险的危害。
须知	用于传递设备或环境安全警示信息。如不避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

命令行格式约定

格式	意义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | ... }	表示从两个或多个选项中选取一个。
[x | y | ...]	表示从两个或多个选项中选取一个或者不选。
{ x | y | ... }*	表示从两个或多个选项中选取多个，最少选取一个，最多选取所有选项。
[x | y | ...]*	表示从两个或多个选项中选取多个或者不选。
&<1-n>	表示符号&前面的参数可以重复1～n次。
#	由“#”开始的行表示为注释行。

修改记录

文档版本	发布日期	修改说明
02	2022-08-18	第二次正式发布。 **3.3 获取软件包**章节增加软件包说明。
01	2021-07-28	第一次正式发布。

目 录

前言 ii
1 用户必读 1
2 版本配套关系 3
3 物理机安装 4
3.1 安装流程 5
3.2 确认操作系统 6
3.3 获取软件包 7
3.4 创建运行用户 12
3.5 安装驱动 13
3.5.1 确认安装方式 13
3.5.2 二进制文件直接安装 16
3.5.2.1 安装驱动（适用于.run 格式） 16
3.5.2.2 安装驱动（适用于.rpm 格式） 19
3.5.2.3 安装驱动（适用于.deb 格式） 20
3.5.3 源码编译安装 24
3.5.3.1 安装依赖 24
3.5.3.2 安装驱动（适用于.run 格式） 27
3.5.3.3 安装驱动（适用于.rpm 格式） 30
3.5.3.4 安装驱动（适用于.deb 格式） 32
3.5.4 重构驱动包安装 35
3.5.4.1 安装依赖 36
3.5.4.2 重构驱动包 36
3.5.4.3 安装驱动（适用于.run 格式） 38
3.6 安装固件 41
3.6.1 安装固件（适用于.run 格式） 41
3.6.2 安装固件（适用于.rpm 格式） 42
3.6.3 安装固件（适用于.deb 格式） 44
3.7 卸载驱动 45
3.7.1 确认卸载方式 45
3.7.2 卸载驱动（适用于.run 格式） 46
3.7.3 卸载驱动（适用于.rpm 格式） 47
3.7.4 卸载驱动（适用于.deb 格式） 48

3.8 卸载固件 49
3.8.1 卸载固件（适用于.run 格式） 49
3.8.2 卸载固件（适用于.rpm 格式） 49
3.8.3 卸载固件（适用于.deb 格式） 50
4 容器内安装 52
4.1 安装前必读 52
4.2 安装场景 53
4.2.1 宿主机目录挂载容器 53
4.2.2 宿主机目录不挂载容器 55
4.3 容器内软件包卸载 57
5 常用操作 58
5.1 通过 PuTTY 登录（网口方式） 58
5.2 使用 WinSCP 传输文件 60
5.3 设置用户有效期 62
5.4 配置 Host 侧 IP 地址通过 SSH 方式登录 Device 63
6 FAQ 66
6.1 升级系统内核版本时未卸载 run 包导致 run 包不可用 66
6.2 deb 包安装、升级或卸载失败后的处理方式 67
6.3 root 用户切换到普通用户需要输入密码权限 68
6.4 如何检查 device 是否正常运行 68
6.5 复位或下电导致 Device 无法正常启动 69
6.6 软件包格式混用安装或升级恢复方法 69
6.7 安装软件包时提示解压失败 70
6.8 驱动加载超时，CPU 打印“soft lockup”报错 70
6.9 驱动卸载过程中出现进程占用报错 72
6.10 安装 deb 包时出现“binary opetator expected”报错 73
6.11 算力切分场景下 npu-smi 命令无法使用 73
7 参考 75
7.1 参数说明/常用命令 75
7.2 驱动安装过程中所需的 Linux 工具 79
7.3 相关工具 81

1 用户必读

软件包简介

组件	软件包名称		
驱动&npu-smi工具包	*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run	*{product name}*-npu-driver-*x.x.x*-*<release>*.*{arch}*.rpm	*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.deb
固件包	*{product name}*-npu-firmware_*x.x.x*.run	*{product name}*-npu-firmware-*x.x.x*-*<release>*.noarch.rpm	*{product name}*-npu-firmware_*x.x.x*.deb
说明： *{product name}*表示产品名称，*x.x.x*表示NPU软件包版本号，*<release>表示RPM包的发布版本号*，{arch}表示系统架构，具体请根据实际情况进行替换。 *{product name}* 具体为：A300-3000			

注意事项

不支持run包、rpm包和deb包混用安装或升级。若用户已使用其中一种软件包安装，想要切换为其他类型的软件包，则需要先使用对应类型软件包的卸载命令进行卸载，再重新安装同版本新软件包。

若用户误操作run包、rpm包和deb包的混用安装或升级，可以参考**6.6 软件包格式混用安装或升级恢复方法**。

安装场景说明

表 1-1 安装场景说明

安装说明	参考章节
如果用户推理服务部署在物理机，则在物理机安装驱动固件即可。	**3 物理机安装**

安装说明	参考章节
如果用户推理服务部署在容器，则需先在物理机安装驱动固件，再在容器内安装驱动。	3 物理机安装 4 容器内安装

OS EOS 声明

EOS后的操作系统将由二进制转为源码编译交付；对于Ubuntu 18.04.1操作系统，官方EOS时间为2023年4月，二进制驱动交付时间将支持至2023年4月，后续版本驱动将使用源码编译方式安装。

2 版本配套关系

硬件产品版本	软件项目版本	NPU软件包版本(驱动和固件包）	NPU驱动版本&npu-smi工具版本	NPU固件版本
1.0.16	Ascend HDK 22.0.RC2	5.1.RC2	22.0.2	1.82.22.2.X

硬件产品版本是指华为发布昇腾硬件的项目版本号，迭代增加，如果是服务器类型硬件，会随版本发布服务器RAID卡和网卡配套的驱动固件包，如果硬件配置了昇腾AI处理器，那么硬件版本会同步发布昇腾AI处理器对应的驱动固件。

软件项目版本是指华为发布昇腾硬件配置的昇腾AI处理器对应的驱动固件，如推理卡、训练卡或者服务器配置的昇腾AI处理器，本文以NPU驱动固件称呼。

昇腾AI处理器对应的驱动固件软件包名称中包含的版本号为5.1.RC2。

备注：NPU驱动和固件版本可通过**npu-smi info -t board -i** *NPU ID*命令查询。回显信息中的“Software Version”字段值表示NPU驱动版本，“Firmware Version”字段值表示NPU固件版本，**NPU驱动和固件包名称中包含的版本为5.1.RC2，但是部署驱动和固件后使用npu-smi命令查询获取的驱动版本为22.0.2，固件版本为1.82.22.2.X。**

本文中的NPU版本号表示npu-smi工具版本，可通过**npu-smi -v**命令查询。

说明

*NPU ID*表示设备编号，可通过**npu-smi info -l**命令查询。

3 物理机安装

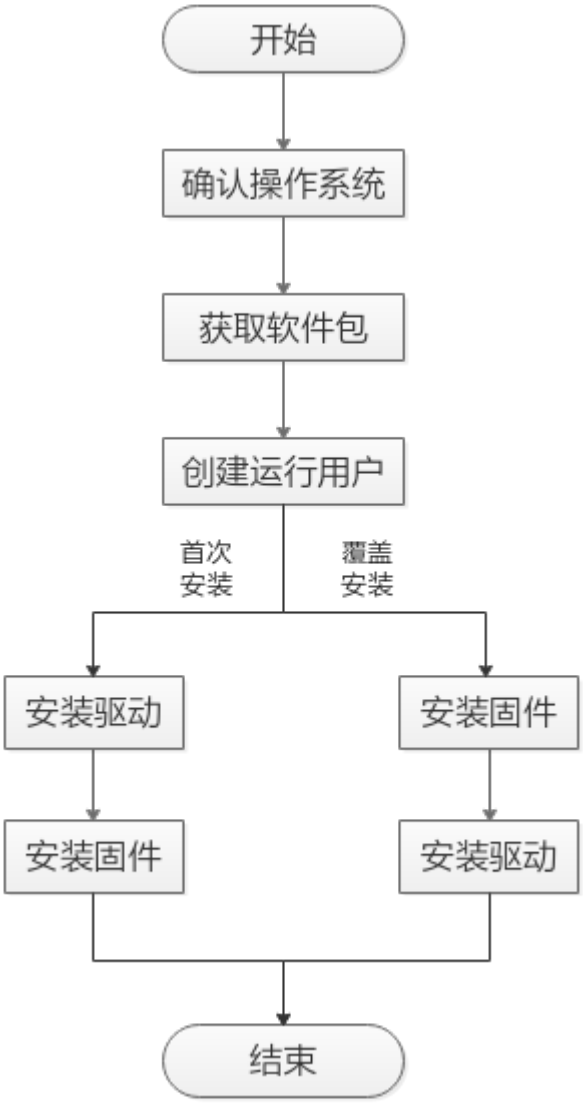
本章节主要介绍Linux系统下驱动和固件的安装与卸载。

3.1 安装流程
3.2 确认操作系统
3.3 获取软件包
3.4 创建运行用户
3.5 安装驱动
3.6 安装固件
3.7 卸载驱动
3.8 卸载固件

3.1 安装流程

安装流程

图 3-1 软件包安装流程



说明

首次安装场景：硬件设备刚出厂时未安装驱动，或者硬件设备前期安装过驱动固件但是当前已卸载，上述场景属于首次安装场景，需按照“**驱动->固件**”的顺序安装驱动固件。

覆盖安装场景：硬件设备前期安装过驱动固件且未卸载，当前要再次安装驱动固件，此场景属于覆盖安装场景，需按照“**固件->驱动**”的顺序安装固件驱动。

表 3-1 安装流程说明

关键步骤	说明	参考章节
确认操作系统	确认目标环境支持的操作系统。	3.2 确认操作系统
获取软件包	根据目标环境所使用的操作系统下载所需软件包。	3.3 获取软件包
创建运行用户	使用root用户安装软件包，但安装完之后要求使用root用户（该场景下权限控制可能存在安全风险）或非root用户（不能为root用户属组）运行相关进程。	3.4 创建运行用户
安装驱动	支持以下三种安装场景。 ● 二进制文件直接安装 ● 源码编译安装 ● 重构驱动包安装 **说明** 首次安装请按照“**驱动->固件**”的顺序；覆盖安装或升级请按照“**固件->驱动**”的顺序，分别安装软件包。	3.5 安装驱动
安装固件	根据不同的软件包格式指导用户安装固件。 **说明** 首次安装请按照“**驱动->固件**”的顺序；覆盖安装或升级请按照“**固件->驱动**”的顺序，分别安装软件包。	3.6 安装固件

3.2 确认操作系统

操作步骤

步骤1 查询服务器及配套PCIe卡支持的操作系统，详细请参考计算产品兼容性查询助手。

- 若使用华为服务器，则“产品型号”选择对应服务器型号，“PCIe卡”信息选择对应卡型号，单击“查询”。
- 若使用第三方服务器，则“产品型号”选择对应卡型号，单击“查询”。

步骤2 执行如下命令，查询服务器当前运行环境的操作系统架构及版本。

```
uname -m && cat /etc/*release
```

说明

如果查询的操作系统版本不在兼容性列表中，请替换为兼容性列表中支持的操作系统。

----结束

3.3 获取软件包

操作步骤

- 企业网用户：
  a. 登录**Ascend HDK**的软件页签。
  b. 选择目标版本。
  c. 根据下表选择所需的软件包。

表 3-2 软件包说明

软件包	配套操作系统版本	架构	说明
A300-3000-npu-driver-*x.x.x*-1.aarch64.rpm	银河麒麟V10SP1、中标麒麟 7.6、Tlinux 2.4、CentOS 8.2、EulerOS 2.9、Suse12Sp5、CentOS 7.6、EulerOS 2.8、EulerOS-virtual 2.9、OpenEuler 20.03、Kylinosv10、EulerOS Virtual V200R009C10、Euler 2.10、EulerOS-Virtual-V200R010C00SPC500B020-aarch64	ARM	A300 3000昇腾NPU驱动包(rpm格式)
A300-3000-npu-driver_*x.x.x*_linux-aarch64.deb	UOS 20、Linx 6.0、Ubuntu 18.04.1、Ubuntu 18.04.5、Ubuntu 20.04	ARM	A300 3000昇腾NPU驱动包(deb格式)

软件包	配套操作系统版本	架构	说明
A300-3000-npu-driver_*x.x.x*_linux-aarch64.run	UOS 20、银河麒麟 V10SP1、中标麒麟 7.6、Tlinux 2.4、CentOS 8.2、EulerOS 2.9、Linx 6.0、Suse12Sp5、Ubuntu 18.04.1、CentOS 7.6、EulerOS 2.8、EulerOS-virtual 2.9、Ubuntu 18.04.5、OpenEuler 20.03、Kylinosv10、EulerOS Virtual V200R009C10、Ubuntu 20.04、Euler 2.10、EulerOS-Virtual-V200R010C00SPC500B020-aarch64	ARM	A300 3000昇腾NPU驱动包(run格式)

软件包	配套操作系统版本	架构	说明
A300-3000-npu-firmware-*x.x.x*-1.noarch.rpm	银河麒麟V10SP1、中标麒麟 7.6、Tlinux 2.4、CentOS 8.2、EulerOS 2.9、Suse12Sp5、CentOS 7.6、EulerOS 2.8、EulerOS-virtual 2.9、OpenEuler 20.03、Kylinosv10、EulerOS Virtual V200R009C10、Euler2.10、EulerOS-Virtual-V200R010C00SPC500B020-aarch64	ARM	A300 3000昇腾固件安装包(rpm格式)
A300-3000-npu-firmware_*x.x.x*.deb	UOS 20、Linx 6.0、Ubuntu 18.04.1、Ubuntu 18.04.5、Ubuntu 20.04	ARM	A300 3000昇腾固件安装包(deb格式)

软件包	配套操作系统版本	架构	说明
A300-3000-npu-firmware_*x.x.x*.run	UOS 20、银河麒麟 V10SP1、中标麒麟 7.6、Tlinux 2.4、CentOS 8.2、EulerOS 2.9、Linx 6.0、Suse12Sp5、Ubuntu 18.04.1、CentOS 7.6、EulerOS 2.8、EulerOS-virtual 2.9、Ubuntu 18.04.5、OpenEuler 20.03、Kylinosv10、EulerOS Virtual V200R009C10、Ubuntu 20.04、Euler 2.10、EulerOS-Virtual-V200R010C00SPC500B020-aarch64	ARM	A300 3000昇腾固件安装包(run格式)

软件包	配套操作系统版本	架构	说明
A300-3000-npu_*x.x.x*_linux-aarch64.zip	UOS 20、银河麒麟 V10SP1、中标麒麟 7.6、Tlinux 2.4、CentOS 8.2、EulerOS 2.9、Linx 6.0、Suse12Sp5、Ubuntu 18.04.1、CentOS 7.6、EulerOS 2.8、EulerOS-virtual2.9、Ubuntu 18.04.5、OpenEuler 20.03、Kylinosv10、EulerOS Virtual V200R009C10、Ubuntu 20.04、Euler 2.10、EulerOS-Virtual-V200R010C00SPC500B020-aarch64	ARM	A300 3000一键式升级包（含NPU固件和驱动，仅支持全默认完整安装升级模式）
A300-3000-driver-repack-tools_*x.x.x*.tar.gz	-	-	Atlas300 3000 驱动包重新打包工具
A300-3000-mcu_*X*.zip	-	-	FusionDirector升级MCU固件zip包（内含带内升级MCU固件bin文件和带外升级MCU固件hpm包）

说明

*x.x.x*表示NPU软件包版本号，X表示MCU版本号，具体请根据实际情况进行替换。

驱动和固件包格式支持run、rpm、deb，每种格式软件包安装效果无区别，但是支持的OS范围有区别，请用户根据现场OS系统需求选择软件包。

一种格式的软件包可能支持多种OS，如Ubuntu OS支持run和deb格式，用户可以任意选择一种格式的软件包，如果用户对软件包格式无特殊需求，那么在OS支持的前提下优先推荐下载run格式的软件包。

d. 单击软件包后面的和，可获取软件包和数字签名文件。

说明

请获取同一版本下的驱动包、芯片固件包、MCU固件包和驱动包重新打包工具。

- 运营商用户：联系华为技术有限公司驻当地办事处的技术支持人员。

软件数字签名验证

为了防止软件包在传递过程或存储期间被恶意篡改，下载软件包时需下载对应的数字签名文件用于完整性验证。

在软件包下载之后，请参考《OpenPGP签名验证指南》，对从Support网站下载的软件包进行PGP数字签名校验。如果校验失败，请不要使用该软件包，先联系华为技术支持工程师解决。

使用软件包安装/升级之前，也需要按上述过程先验证软件包的数字签名，确保软件包未被篡改。

运营商客户请访问：**http://support.huawei.com/carrier/digitalSignatureAction**

企业客户请访问：**https://support.huawei.com/enterprise/zh/tool/pgp-verify-TL1000000054**

3.4 创建运行用户

首先介绍安装用户和运行用户概念，安装用户为安装驱动和固件所使用的用户，运行用户为驱动固件安装完成后，后续运行推理或训练业务时启动运行驱动和固件的用户。

驱动和固件的安装用户必须为root用户，驱动和固件的运行用户可以为root用户或者非root用户，如果后续用户对运行业务有用户权限控制或者多用户隔离的需求，可创建非root用户作为后续运行用户。本章节介绍创建运行用户的步骤。

若使用rpm包安装时，系统会自动创建运行用户HwHiAiUser，用户无需手动创建。请跳过此章节。

若使用FusionDirector工具纳管，首次安装驱动包（.zip格式）系统会自动创建用户组和用户HwHiAiUser，无需手动创建。请跳过此章节。

表 3-3 创建运行用户

运行用户	操作步骤

• root用户
• 非root用户（推荐）
  – HwHiAiUser
  – 非HwHiAiUser
说明
如果使用root用户作为运行用户，则需要与--install-for-all参数配合使用，详情请参见7.1 参数说明/常用命令，该场景下权限控制可能存在安全风险；如果使用非root用户作为运行用户，则创建的该用户不能为root用户属组。
	• 如果创建的用户是HwHiAiUser，安装软件包时无需指定运行用户，默认即为HwHiAiUser。
	• 如果创建的用户是非HwHiAiUser（含root），安装软件包时必须指定运行用户（通过--install-username=*username* --install-usergroup=*usergroup*参数指定）。因此如果对运行用户名称没有特殊要求，建议使用HwHiAiUser。
	请参见如下方法创建运行用户。
	1. 以root用户登录服务器。
	2. 执行如下命令，创建运行用户。
	# groupadd *usergroup*
	# useradd -g *usergroup* **-d /home/**_username_ **-m** *username* **-s /bin/bash**

3.5 安装驱动

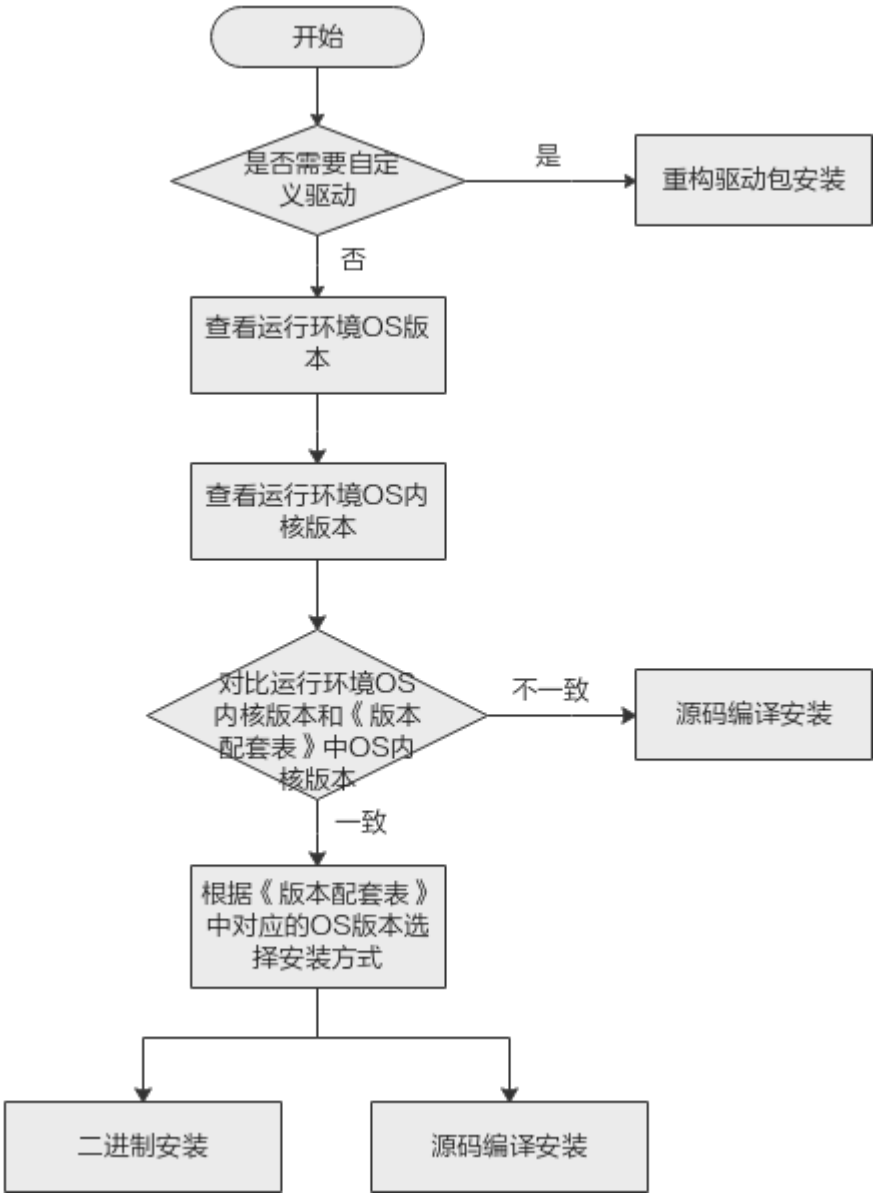
须知

如果设备已经纳管到FusionDirector，安装驱动后，FusionDirector界面上显示的驱动版本会延迟刷新，最长为1个小时。重启设备后版本会立即刷新。

3.5.1 确认安装方式

根据图3-2选择驱动安装方式。

图 3-2 确认安装方式流程



说明

- 如果运行环境的操作系统内核版本是《版本配套表》中内核版本的衍生版本，驱动也可以通过源码编译方式安装，但可能会存在无法正常编译的问题。
- 对于EulerOS-Virtual-V200R009C10操作系统，当前host操作系统的内核版本必须与《版本配套表》中要求一致，不支持源码编译。
- EulerOS 2.8场景下，操作系统内核版本必须与《版本配套表》中默认系统版本一致；如果不一致，只能通过升级内核版本的方式解决。否则会导致安装过程报错：部分芯片无法启动。

系统兼容性要求

- 支持PCIe MSI-X中断，每个Atlas 300I 推理卡至少可分配464个MSI-X中断。
- BAR空间大小要求：

  Atlas 300I 推理卡需要三个BAR，空间大小分别为512KB，64MB，256MB。

环境检查

表 3-4 环境检查

序号	检查项	检查命令	检查说明
1	操作系统内核版本	**uname -r**	如果Host系统中安装过run包，内核升级后会出现run包启动异常的问题。出现该问题时请参见**6.1 升级系统内核版本时未卸载run包导致run包不可用**解决。
2	GCC版本	**gcc -v**	• 若使用二进制文件直接安装驱动包，安装无须检查GCC版本。 • 若使用源码编译安装驱动包，GCC版本须不低于《版本配套表》中要求的版本。
3	查询系统是否安装过软件包	**lsmod| grep drv_pcie_host**	• 如无内容表示未安装过软件包。可以直接安装软件包。 • 如有内容，表示安装过软件包。需要先卸载驱动包后，再安装新版本软件包。卸载驱动包具体请参见**3.7 卸载驱动**。
4	检测卡是否正常在位	**lspci | grep d100**	• 如果服务器上有*N*（*N*>0）张Atlas 300I 推理卡（型号 3000），回显中含“d100”字段的行数为4*N*，则表示Atlas 300I 推理卡（型号 3000）正常在位。 88:00.0 Processing accelerators: Huawei Technologies Co., Ltd. Device d100 (rev 20) 89:00.0 Processing accelerators: Huawei Technologies Co., Ltd. Device d100 (rev 20) 8a:00.0 Processing accelerators: Huawei Technologies Co., Ltd. Device d100 (rev 20) 8b:00.0 Processing accelerators: Huawei Technologies Co., Ltd. Device d100 (rev 20) ..... • 如果Atlas 300I 推理卡（型号 3000）所在的服务器是华为服务器，可通过登录iBMC WebUI界面，进入“信息管理 > 系统信息 ”，单击“其他”，Atlas 300I 推理卡（型号 3000）的PCIe卡信息在“PCIe卡”列表中，表示Atlas 300I 推理卡（型号 3000）正常在位。
5（可选）	驱动安装过程中所需的Linux工具	NA	请根据**表7-2**检查所需的Linux工具并进行安装。如系统中缺失，安装过程中也会有回显信息提示，可以根据提示信息进行安装。

序号	检查项	检查命令	检查说明
6（可选）	相关配置文件	NA	• 对于**Linx 6.0.90**操作系统，需要检查“/etc/pam.d/su”配置文件的“auth sufficient pam_rootok.so”字段是否被注释掉，若被注释，请参见**6.3 root用户切换到普通用户需要输入密码权限**解决。 • 对于**SUSE12SP5**操作系统，需要检查配置文件“10-unsupported-modules.conf”中“allow_unsupported_modules ”的值是否为1。执行**cat /etc/modprobe.d/10-unsupported-modules.conf**命令，查看modprobe限制配置文件中“allow_unsupported_modules ”的值是否“1”。如果不是，使用**vi /etc/modprobe.d/10-unsupported-modules.conf**命令修改“allow_unsupported_modules ”的值为“1”。 **说明** 在**SUSE12SP5**操作系统下，可通过“10-unsupported-modules.conf”配置文件控制非系统自带驱动在系统启动过程中的加载。默认情况下，“10-unsupported-modules.conf”配置文件中“allow_unsupported_modules ”的值为“0”，表示不允许系统启动过程中加载非系统自带驱动；“allow_unsupported_modules ”的值设置为“1”时，将允许系统启动过程中加载非系统自带驱动。

3.5.2 二进制文件直接安装

3.5.2.1 安装驱动（适用于.run 格式）

本章节介绍宿主机上的安装方法，本章节以*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run包为例说明，具体操作以客户实际host系统对应驱动包为准。

注意事项

- 如果有安装NPU 1.X.X系列版本的驱动包，在安装NPU 20.X.X及以上系列版本的驱动包前需要先卸载低版本驱动。
- 在软件包安装或升级过程中，不允许对Host或Device进行复位或下电操作，否则会出现Device无法正常启动或者升级失败。

操作步骤

须知

首次安装请按照“**驱动->固件**”的顺序；覆盖安装或升级请按照“**固件->驱动**”的顺序，分别安装软件包。固件安装请参见**3.6.1 安装固件（适用于.run格式）**。

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

对于Atlas 300I 推理卡（型号 3000），为保证驱动版本、固件版本和MCU版本三者保持配套关系，请参考《**Atlas 300I 推理卡 5.1.RC2 NPU驱动和固件升级指南（型号3000,3010）**》中“升级MCU”章节升级对应的MCU组件。

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd /*opt*

步骤5 执行如下命令，增加软件包的可执行权限。

chmod +x *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run

步骤6 执行如下命令，校验run安装包的一致性和完整性。

./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--check**

若出现如下回显信息，表示软件包校验成功。

```
Verifying archive integrity...  100%   SHA256 checksums are OK. All good.
```

说明

软件包校验使用SHA256加密方式；校验过程中若提示：./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run does not contain an embedded MD5 checksum.和./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run does not contain a CRC checksum，表示没有使用MD5和CRC加密方式，可忽略。

步骤7 执行**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--full**命令，完成驱动安装，软件包默认安装路径为“/usr/local/Ascend”。

说明

- 安装详细日志路径：/var/log/ascend_seclog/ascend_install.log。
- 安装后软件包的安装路径、安装命令以及运行用户信息记录路径为“/etc/ascend_install.info”。
- 安装run包驱动时，会将动态库libdcmi.so和头文件dcmi_interface_api.h拷贝到“/usr/local/dcmi/”目录下。
- 若创建运行用户是非HwHiAiUser，安装驱动软件包时需要指定运行用户（通过**--install-username=***username* **--install-usergroup=***usergroup*参数指定）。
- 如果指定root用户为运行用户，则需要与--install-for-all参数配合使用，如下所示，该场景下权限控制可能存在安全风险。
  ```
  --install-username=root  --install-usergroup=root  --install-for-all
  ```
- 如果服务器无BMC时，只能采用默认路径安装驱动。如果采用指定路径安装，ipmi_si驱动加载时会出现阻塞，导致磁盘挂载慢，影响npu正常运行。
- Device侧系统类日志通过msnpureport工具传输到Host侧进行查看。导出操作以及导出日志的存储路径具体请参见对应产品和版本的《黑匣子日志参考》的“msnpureport工具使用”章节。容器内不支持查看Device侧系统类日志，也不支持通过msnpureport工具导出Device侧系统类日志。
- 安装驱动过程中出现：“driver ko vermagic(*) is different from the os(*), need rebuild driver”，请参见**3.5.3 源码编译安装**解决。

若用户需要指定安装路径，如以“/test/HiAI/”为例。

可执行**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--full --install-path=**/test/HiAI/命令，完成安装。

- 若指定路径不存在，则安装时会自动创建目录，若有多层目录，则只有最后一层目录不存在时会自动创建。
- 若指定路径已存在：
  - 若该路径下所有层级目录属主为root用户，则请确保所有层级目录权限至少为755。若不满足要求，请修改路径权限：**chmod 755 路径**
  - 若该路径下有一层目录属主为非root用户，则请用户自行修改为root属主，并确保所有层级目录权限为755。若不满足要求，请修改路径属主为root：**chown root:群组名 路径**

若系统出现如下关键回显信息，则表示驱动安装成功。
```
Driver package installed successfully!
```

步骤8 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤9 执行**npu-smi info**查看驱动加载是否成功。

若出现如下回显信息，说明加载成功。否则，说明加载失败。请联系华为技术支持处理。

```
+------------------------------------------------------------------------------------------------+
| npu-smi 22.0.2                       Version: 22.0.2                                           |
+-------------------+-----------------+-------------------------------------------------------+
| NPU     Name      | Health          | Power(W)          Temp(C)            Hugepages-Usage(page) |
| Chip    Device    | Bus-Id          | AICore(%)         Memory-Usage(MB)                         |
+===================+=================
+==============================================================+
| 6       310       | OK              | 12.8              73            0       / 970              |
| 0       3         | 0000:89:00.0    | 0                 2703 / 8192                              |
+-------------------+-----------------+--------------------------------------------------------------+
| 6       310       | OK              | 12.8              74            0       / 970              |
```

```
| 1      4        | 0000:8A:00.0    | 0                2867 / 8192                          |
+-------------------+-----------------+-----------------------------------------------------------------+
| 6      310      | OK              | 12.8             70          0        / 970              |
| 2      5        | 0000:8B:00.0    | 0                2867 / 8192                          |
+-------------------+-----------------+-----------------------------------------------------------------+
| 6      310      | OK              | 12.8             65          0        / 970              |
| 3      6        | 0000:8C:00.0    | 0                2867 / 8192                          |
+===================+=================
+=================================================================+
```

说明

- 加载失败，可以执行**dmesg**命令查看linux启动日志。如果出现 /*安装路径*/driver/device/davinci_mini.fd copy err信息，请参考**3.7.2 卸载驱动（适用于.run格式）**章节卸载驱动，并使用默认路径重新进行驱动安装。
- 回显中“npu-smi”后面的字段为npu-smi工具版本号；“Version:”后面的字段为驱动版本号。

----结束

3.5.2.2 安装驱动（适用于.rpm 格式）

本章以*{product name}*-npu-driver-*x.x.x-<release>.{arch}*.rpm为例介绍.rpm格式驱动包安装相关指导，具体请根据实际情况进行替换。

注意事项

- 当前rpm包安装不支持修改安装路径功能，请按照正常rpm流程安装，不支持rpm包的--force命令安装升级。
- Ascend 310内部部分功能，只支持HwHiAiUser用户访问和执行，用户首次安装时系统会自动创建HwHiAiUser用户。
- 如果已安装NPU 1.X.X系列版本的rpm包，在安装NPU 20.X.X及以上系列版本的rpm包前需要先卸载原版本再安装。
- 在软件包安装或升级过程中，不允许对Host或Device进行复位或下电操作，否则会出现Device无法正常启动或者升级失败。

操作步骤

须知

请按照“**驱动->固件**”的顺序，分别安装软件包。固件安装请参见**3.6.2 安装固件（适用于.rpm格式）**。

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 在软件包所在路径执行如下命令完成Host侧驱动包安装。

rpm -ivh *{product name}-npu-driver-x.x.x-<release>.{arch}.rpm*

说明

软件包安装过程中，若无HwHiAiUser用户系统会自动创建该用户。

若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤5 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤6 执行**npu-smi info**查看驱动加载是否成功。

若出现如下回显信息，说明加载成功。否则，说明加载失败。请联系华为技术支持处理。

```
+------------------------------------------------------------------------------------------------+
| npu-smi 22.0.2                       Version: 22.0.2                                           |
+-------------------+-----------------+-------------------------------------------------------+
| NPU     Name      | Health          | Power(W)        Temp(C)           Hugepages-Usage(page) |
| Chip    Device    | Bus-Id          | AICore(%)       Memory-Usage(MB)                        |
+===================+=================
+==========================================================================+
| 6       310       | OK              | 12.8            73          0       / 970             |
| 0       3         | 0000:89:00.0    | 0               2703 / 8192                           |
+-------------------+-----------------+-------------------------------------------------------+
| 6       310       | OK              | 12.8            74          0       / 970             |
| 1       4         | 0000:8A:00.0    | 0               2867 / 8192                           |
+-------------------+-----------------+-------------------------------------------------------+
| 6       310       | OK              | 12.8            70          0       / 970             |
| 2       5         | 0000:8B:00.0    | 0               2867 / 8192                           |
+-------------------+-----------------+-------------------------------------------------------+
| 6       310       | OK              | 12.8            65          0       / 970             |
| 3       6         | 0000:8C:00.0    | 0               2867 / 8192                           |
+===================+=================
+==========================================================================+
```

说明

- 加载失败，可以执行**dmesg**命令查看linux启动日志。如果出现 */安装路径*/driver/device/davinci_mini.fd copy err信息，请参考**3.7.2 卸载驱动（适用于.run格式）**章节卸载驱动，并使用默认路径重新进行驱动安装。
- 回显中“npu-smi”后面的字段为npu-smi工具版本号；“Version:”后面的字段为驱动版本号。

----结束

3.5.2.3 安装驱动（适用于.deb 格式）

通过 dpkg 安装

使用默认运行用户HwHiAiUser安装

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd /*opt*

步骤5 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤6 执行如下命令，安装deb驱动包（*驱动包名称请根据实际情况替换*）。

dpkg -i *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.deb

若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤7 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤8 执行如下命令，查询驱动安装信息。

dpkg -l | grep Ascend

----结束

使用指定运行用户安装

步骤1 将3.3 获取软件包准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd /*opt*

步骤5 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤6 设置环境变量。

说明

如果指定环境变量为root用户，则需要与export ASCEND_INSTALL_FOR_ALL=true配合使用，该场景下权限控制可能存在安全风险。

1. 通过export方式设置环境变量，该种方式设置的环境变量只在当前窗口有效，设置完后立即生效。

   export ASCEND_USER_GROUP=*usergroup*

   export ASCEND_USER_NAME=*username*

2. 通过~/.bashrc文件方式设置永久环境变量，操作如下：
   a. 以安装用户在任意目录下执行**vi ~/.bashrc**命令，打开.bashrc文件，在文件最后一行后面添加上述内容。
   b. 执行**:wq!**命令保存文件并退出。
   c. 执行**source ~/.bashrc**命令使其立即生效。

步骤7 将创建的用户名加入创建的用户组。

groupadd ${ASCEND_USER_GROUP}

useradd -g ${ASCEND_USER_GROUP} -d /home/${ASCEND_USER_NAME} -m ${ASCEND_USER_NAME} -s /bin/bash

步骤8 执行如下命令，安装deb驱动包（*驱动包名称请根据实际情况替换*）。

dpkg -i *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.deb

若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤9 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤10 执行如下命令，查询驱动安装信息。

dpkg -l | grep Ascend

步骤11 （**可选**）执行如下命令，清除相关环境变量。

unset ASCEND_USER_NAME

unset ASCEND_USER_GROUP

unset ASCEND_INSTALL_FOR_ALL

----结束

通过 apt-get 安装

默认用户安装

步骤1 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤2 执行以下命令，安装本地源中的deb驱动包。

- **首次安装**

  apt-get install ascend310-driver

- **重新安装**

  apt-get --reinstall install ascend310-driver

  若系统出现如下关键回显信息，则表示驱动安装成功。

  ```
  Driver package installed successfully!
  ```

步骤3 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤4 执行如下命令，查询驱动安装信息。

apt list ascend310-driver

----结束

指定用户安装

步骤1 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤2 设置环境变量。

说明

如果指定环境变量为root用户，则需要与export ASCEND_INSTALL_FOR_ALL=true配合使用，该场景下权限控制可能存在安全风险。

1. 通过export方式设置环境变量，该种方式设置的环境变量只在当前窗口有效，设置完后立即生效。

   export ASCEND_USER_GROUP=*usergroup*

   export ASCEND_USER_NAME=*username*

2. 通过~/.bashrc文件方式设置永久环境变量，操作如下：

   a. 以安装用户在任意目录下执行**vi ~/.bashrc**命令，打开.bashrc文件，在文件最后一行后面添加上述内容。

   b. 执行**:wq!**命令保存文件并退出。

   c. 执行**source ~/.bashrc**命令使其立即生效。

步骤3 将创建的用户名加入创建的用户组。

groupadd ${ASCEND_USER_GROUP}

useradd -g ${ASCEND_USER_GROUP} -d /home/${ASCEND_USER_NAME} -m ${ASCEND_USER_NAME} -s /bin/bash

步骤4 执行以下命令，安装本地源中的deb驱动包。

- **首次安装**

apt-get install ascend310-driver

- **重新安装**

  apt-get --reinstall install ascend310-driver

  若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤5 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤6 执行如下命令，查询驱动安装信息。

apt list ascend310-driver

步骤7 （**可选**）执行如下命令，清除相关环境变量。

unset ASCEND_USER_NAME

unset ASCEND_USER_GROUP

unset ASCEND_INSTALL_FOR_ALL

----结束

3.5.3 源码编译安装

当用户当前操作系统内核版本更新或与默认不一致，可以基于安装包驱动源码重新构建并安装使用，该场景下不会触发内核版本升级，会基于用户当前内核版本编译出相应的内核驱动。软件包在安装时会自动触发驱动源码编译，编译完成后自动进行软件包安装。

如果使用该功能，则需要提前检查并安装相关依赖工具。具体操作请切换到root用户执行。

3.5.3.1 安装依赖

make 工具

执行**make -v**命令，若能查询到make工具的版本，则make已安装。

默认内核源码路径或 dkms、gcc、linux-header 等安装依赖

- 对于以下各操作系统，请执行如下命令检查dkms、gcc、linux-header等安装依赖是否存在。
- 对于其他操作系统，检查默认内核源码路径是否存在。例如：/lib/modules/`uname -r`/build

  执行**ls /lib/modules/`uname -r`/build**命令，查看路径是否存在。

  – 若存在，安装驱动包时则会自动使用内核进行驱动编译。
  – 若不存在，您可以在安装时提供对应的源码路径，具体请参见**步骤8**。

说明

以下操作系统中的DKMS软件包建议使用dkms-2.6.1-1.el7.noarch.rpm。

Ubuntu 18.04 操作系统

需要安装dkms、gcc、linux-header软件包。执行如下命令检查：

dpkg-query -s dkms

dpkg-query -s gcc

dpkg-query -s linux-headers-$(uname -r)

若未安装请自行在相应操作系统厂商网站获取并安装。

Ubuntu 20.04 操作系统

需要安装dkms、gcc、linux-header软件包。执行如下命令检查：

dpkg-query -s dkms

dpkg-query -s gcc

dpkg-query -s linux-headers-$(uname -r)

若未安装请自行在相应操作系统厂商网站获取并安装

CentOS 7.6 操作系统

需要安装dkms、gcc、kernel-headers、kernel-devel软件包。执行如下命令检查：

rpm -qa | grep dkms

rpm -qa | grep gcc

rpm -qa | grep kernel-headers-$(uname -r)

rpm -qa | grep kernel-devel-$(uname -r)

若未安装请自行在相应操作系统厂商网站获取并安装。

CentOS 8.2 操作系统

需要安装dkms、gcc、kernel-headers、kernel-devel软件包。执行如下命令检查：

rpm -qa | grep dkms

rpm -qa | grep gcc

rpm -qa | grep kernel-headers-$(uname -r)

rpm -qa | grep kernel-devel-$(uname -r)

若未安装请自行在相应操作系统厂商网站获取并安装。

EulerOS 2.8 操作系统

需要安装gcc、kernel-headers、kernel-devel软件包。执行如下命令检查：

rpm -qa | grep gcc

rpm -qa | grep kernel-headers-$(uname -r)

rpm -qa | grep kernel-devel-$(uname -r)

若未安装请自行在相应操作系统厂商网站获取并安装。

EulerOS 2.10 操作系统

需要安装gcc、kernel-headers、kernel-devel软件包。执行如下命令检查：

rpm -qa | grep gcc

rpm -qa | grep kernel-headers-$(uname -r)

rpm -qa | grep kernel-devel-$(uname -r)

若未安装请自行在相应操作系统厂商网站获取并安装。

SUSE12SP5 操作系统

需要安装gcc、kernel-default、kernel-default-devel软件包。执行如下命令检查：

rpm -qa | grep gcc

rpm -qa | grep kernel-default

rpm -qa | grep kernel-default-devel

若未安装请自行在相应操作系统厂商网站获取并安装。

EulerOS 2.9 操作系统

需要安装kernel-headers、gcc 7.3.0、elfutils-devel、make等软件包。操作如下：

1. 请获取对应版本OS的“devel_tools.tar.gz”工具组件。
2. 上传“devel_tools.tar.gz”到运行环境的任一目录。如：/opt下。
3. 进入“devel_tools.tar.gz”所在路径并进行解压。

   cd /opt

   tar -xzvf devel_tools.tar.gz
4. 在3的解压目录下，执行以下命令安装kernel-headers、gcc 7.3.0、elfutils-devel、make等软件包。
   - 对于**EulerOS 2.9+aarch64**系统，请执行如下命令：

     rpm -ivh kernel-headers-4.19.90-vhulk2009.2.0.h269.eulerosv2r9.aarch64.rpm kernel-devel-4.19.90-vhulk2009.2.0.h269.eulerosv2r9.aarch64.rpm

     rpm -ivh binutils-2.34-1.h11.eulerosv2r9.aarch64.rpm binutils-extra-2.34-1.h11.eulerosv2r9.aarch64.rpm binutils-devel-2.34-1.h11.eulerosv2r9.aarch64.rpm zlib-devel-1.2.11-17.eulerosv2r9.aarch64.rpm cpp-7.3.0-20190804.h39.eulerosv2r9.aarch64.rpm libmpc-1.1.0-4.eulerosv2r9.aarch64.rpm glibc-devel-2.28-59.eulerosv2r9.aarch64.rpm libxcrypt-devel-4.4.8-4.h2.eulerosv2r9.aarch64.rpm

     rpm -ivh gcc-7.3.0-20190804.h39.eulerosv2r9.aarch64.rpm

     rpm -ivh elfutils-devel-0.179-1.h3.eulerosv2r9.aarch64.rpm

     rpm -ivh make-4.2.1-15.eulerosv2r9.aarch64.rpm

Tlinux 2.4 操作系统

由于内核进行过针对性适配，需要使用版本为8.2.1的GCC编译器，因此编译驱动前需保证8.2.1的GCC编译器已经部署。操作步骤如下：

1. 获取GCC 8.2.1版本编译器的安装包。软件包获取网址：**https://mirrors.tencent.com/tlinux/2.4/arm64/tlinux-sclo/aarch64/tl/devtoolset-8/devtoolset-8-gcc-8.2.1-3.tl2.aarch64.rpm**。
2. 登录运行环境。
3. 上传gcc编译器的安装包到指定路径下，如“/tmp”。
4. 执行以下命令安装GCC编译器。

   rpm -Uvh devtoolset-8-gcc-8.2.1-3.tl2.aarch64.rpm --force --nodeps
5. 安装完成后，执行以下命令查找GCC 8.2.1的可以执行文件所在路径。

   find / -name gcc

   执行文件路径如下红框所示：

   ```
   /opt/rh/devtoolset-8/root/usr/libexec/gcc
   /opt/rh/devtoolset-8/root/usr/bin/gcc
   /opt/rh/devtoolset-8/root/usr/lib/gcc
   ```
6. 配置环境变量。

   export PATH=/opt/rh/devtoolset-8/root/usr/bin/:$PATH
7. 执行以下命令，验证GCC编译器是否安装成功。

   gcc --version

   如出现以下回显，安装后的gcc版本和目标版本一致，则表示安装成功。

   ```
   [root@TENCENT64 bin]# gcc --version
   gcc (GCC) 8.2.1 20180905 (Red Hat 8.2.1-3)
   Copyright (C) 2018 Free Software Foundation, Inc.
   This is free software; see the source for copying conditions.  There is NO
   warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
   ```

3.5.3.2 安装驱动（适用于.run 格式）

本章节介绍宿主机上的安装方法，本章节以*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run包为例说明，具体操作以客户实际host系统对应驱动包为准。

前提条件

驱动包安装前需要检查并安装相关依赖，具体请参见**3.5.3.1 安装依赖**。

注意事项

- 如果有安装NPU 1.X.X系列版本的驱动包，在安装NPU 20.X.X及以上系列版本的驱动包前需要先卸载低版本驱动。
- 在软件包安装或升级过程中，不允许对Host或Device进行复位或下电操作，否则会出现Device无法正常启动或者升级失败。

操作步骤

须知

首次安装请按照“**驱动->固件**”的顺序；覆盖安装或升级请按照“**固件->驱动**”的顺序，分别安装软件包。固件安装请参见**3.6.1 安装固件（适用于.run格式）**。

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

对于Atlas 300I 推理卡（型号 3000），为保证驱动版本、固件版本和MCU版本三者保持配套关系，请参考《**Atlas 300I 推理卡 5.1.RC2 NPU驱动和固件升级指南（型号3000,3010）**》中“升级MCU”章节升级对应的MCU组件。

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd /*opt*

步骤5 执行如下命令，增加软件包的可执行权限。

chmod +x *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run

步骤6 执行如下命令，校验run安装包的一致性和完整性。

./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--check**

若出现如下回显信息，表示软件包校验成功。

```
Verifying archive integrity...  100%   SHA256 checksums are OK. All good.
```

说明

软件包校验使用SHA256加密方式；校验过程中若提示：./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run does not contain an embedded MD5 checksum.和./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run does not contain a CRC checksum，表示没有使用MD5和CRC加密方式，可忽略。

步骤7 执行**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--full**命令，完成驱动安装，软件包默认安装路径为“/usr/local/Ascend”。

说明

- 安装详细日志路径：/var/log/ascend_seclog/ascend_install.log。
- 安装后软件包的安装路径、安装命令以及运行用户信息记录路径为“/etc/ascend_install.info”。
- 安装run包驱动时，会将动态库libdcmi.so和头文件dcmi_interface_api.h拷贝到“/usr/local/dcmi/”目录下。
- 若创建运行用户是非HwHiAiUser，安装驱动软件包时需要指定运行用户（通过**--install-username=***username* **--install-usergroup=***usergroup*参数指定）。
- 如果指定root用户为运行用户，则需要与--install-for-all参数配合使用，如下所示，该场景下权限控制可能存在安全风险。
  ```
  --install-username=root  --install-usergroup=root  --install-for-all
  ```
- 如果服务器无BMC时，只能采用默认路径安装驱动。如果采用指定路径安装，ipmi_si驱动加载时会出现阻塞，导致磁盘挂载慢，影响npu正常运行。
- Device侧系统类日志通过msnpureport工具传输到Host侧进行查看。导出操作以及导出日志的存储路径具体请参见对应产品和版本的《黑匣子日志参考》的“msnpureport工具使用”章节。容器内不支持查看Device侧系统类日志，也不支持通过msnpureport工具导出Device侧系统类日志。

若用户需要指定安装路径，如以“/test/HiAI/”为例。

可执行**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--full --install-path=**/test/HiAI/命令，完成安装。

- 若指定路径不存在，则安装时会自动创建目录，若有多层目录，则只有最后一层目录不存在时会自动创建。
- 若指定路径已存在：
  - 若该路径下所有层级目录属主为root用户，则请确保所有层级目录权限至少为755。若不满足要求，请修改路径权限：**chmod 755 路径**
  - 若该路径下有一层目录属主为非root用户，则请用户自行修改为root属主，并确保所有层级目录权限为755。若不满足要求，请修改路径属主为root：**chown root:群组名 路径**

步骤8 （**可选**）安装时若出现以下回显，说明未安装dkms，默认内核源码路径如“/lib/modules/`uname -r`/build”不存在。请根据系统提示输入相应内容：

```
[WARNING]rebuild ko has something wrong, detail in /var/log/ascend_seclog/ascend_rebuild.log
Do you want to try build driver after input kernel absolute path? [y/n]:
```

若需要继续安装，请输入**y**。

出现如下回显后，请根据提示输入您提供的内核源码的实际路径，如：**/lib/modules/`uname -r`/build-bak**。

```
Please input your kernel absolute path:
```

按“Enter”键继续安装。

说明

- 若已安装dkms及相应kernel-header、kernel-devel等组件，则会自动执行dkms驱动编译安装。
- 若未安装dkms，但默认内核源码路径如“/lib/modules/`uname -r`/build”已存在，则会自动使用该内核进行驱动编译。

步骤9 若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤10 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤11 执行**npu-smi info**查看驱动加载是否成功。

若出现如下回显信息，说明加载成功。否则，说明加载失败。请联系华为技术支持处理。

```
+------------------------------------------------------------------------------------------------+
| npu-smi 22.0.2                       Version: 22.0.2                                           |
+-------------------+-----------------+-----------------------------------------------------------+
| NPU     Name      | Health          | Power(W)         Temp(C)            Hugepages-Usage(page) |
| Chip    Device    | Bus-Id          | AICore(%)        Memory-Usage(MB)                         |
+===================+=================
+===========================================================================+
| 6       310       | OK              | 12.8             73          0       / 970              |
| 0       3         | 0000:89:00.0    | 0                2703 / 8192                          |
+-------------------+-----------------+-----------------------------------------------------------+
| 6       310       | OK              | 12.8             74          0       / 970              |
| 1       4         | 0000:8A:00.0    | 0                2867 / 8192                          |
+-------------------+-----------------+-----------------------------------------------------------+
| 6       310       | OK              | 12.8             70          0       / 970              |
| 2       5         | 0000:8B:00.0    | 0                2867 / 8192                          |
+-------------------+-----------------+-----------------------------------------------------------+
| 6       310       | OK              | 12.8             65          0       / 970              |
| 3       6         | 0000:8C:00.0    | 0                2867 / 8192                          |
+===================+=================
+===========================================================================+
```

说明

- 加载失败，可以执行**dmesg**命令查看linux启动日志。如果出现 /*安装路径*/driver/device/davinci_mini.fd copy err信息，请参考**3.7.2 卸载驱动（适用于.run格式）**章节卸载驱动，并使用默认路径重新进行驱动安装。
- 回显中“npu-smi”后面的字段为npu-smi工具版本号；“Version:”后面的字段为驱动版本号。

----结束

3.5.3.3 安装驱动（适用于.rpm 格式）

本章以*{product name}*-npu-driver-*x.x.x*-*<release>*.*{arch}*.rpm为例介绍.rpm格式驱动包安装相关指导，具体请根据实际情况进行替换。

前提条件

驱动包安装前需要检查并安装相关依赖，具体请参见**3.5.3.1 安装依赖**。

注意事项

- 当前rpm包安装不支持修改安装路径功能，请按照正常rpm流程安装，不支持rpm包的--force命令安装升级。
- Ascend 310内部部分功能，只支持HwHiAiUser用户访问和执行，用户首次安装时系统会自动创建HwHiAiUser用户。
- 如果已安装NPU 1.X.X系列版本的rpm包，在安装NPU 20.X.X及以上系列版本的rpm包前需要先卸载原版本再安装。
- 在软件包安装或升级过程中，不允许对Host或Device进行复位或下电操作，否则会出现Device无法正常启动或者升级失败。

操作步骤

须知

请按照“**驱动->固件**”的顺序，分别安装软件包。固件安装请参见**3.6.2 安装固件（适用于.rpm格式）**。

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 在软件包所在路径执行如下命令完成Host侧驱动包安装。

rpm -ivh *{product name}-npu-driver-x.x.x-<release>.{arch}.rpm*

说明

软件包安装过程中，若无HwHiAiUser用户系统会自动创建该用户。

若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤5 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤6 执行**npu-smi info**查看驱动加载是否成功。

若出现如下回显信息，说明加载成功。否则，说明加载失败。请联系华为技术支持处理。

```
+------------------------------------------------------------------------------------------------+
| npu-smi 22.0.2                       Version: 22.0.2                                           |
+-------------------+-----------------+-------------------------------------------------------+
| NPU     Name      | Health          | Power(W)          Temp(C)            Hugepages-Usage(page) |
| Chip    Device    | Bus-Id          | AICore(%)         Memory-Usage(MB)                         |
+===================+=================
+==========================================================+
| 6       310       | OK              | 12.8              73            0       / 970              |
| 0       3         | 0000:89:00.0    | 0                 2703 / 8192                            |
+-------------------+-----------------+-------------------------------------------------------+
| 6       310       | OK              | 12.8              74            0       / 970              |
| 1       4         | 0000:8A:00.0    | 0                 2867 / 8192                            |
+-------------------+-----------------+-------------------------------------------------------+
| 6       310       | OK              | 12.8              70            0       / 970              |
| 2       5         | 0000:8B:00.0    | 0                 2867 / 8192                            |
+-------------------+-----------------+-------------------------------------------------------+
| 6       310       | OK              | 12.8              65            0       / 970              |
| 3       6         | 0000:8C:00.0    | 0                 2867 / 8192                            |
+===================+=================
+==========================================================+
```

说明

- 加载失败，可以执行**dmesg**命令查看linux启动日志。如果出现 /*安装路径*/driver/device/davinci_mini.fd copy err信息，请参考**3.7.2 卸载驱动（适用于.run格式）**章节卸载驱动，并使用默认路径重新进行驱动安装。
- 回显中“npu-smi”后面的字段为npu-smi工具版本号；“Version:”后面的字段为驱动版本号。

----结束

3.5.3.4 安装驱动（适用于.deb 格式）

驱动包安装前需要检查并安装相关依赖，具体请参见**3.5.3.1 安装依赖**。

通过 dpkg 安装

使用默认运行用户HwHiAiUser安装

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd /*opt*

步骤5 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤6 执行如下命令，安装deb驱动包（*驱动包名称请根据实际情况替换*）。

dpkg -i *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.deb

若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤7 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤8 执行如下命令，查询驱动安装信息。

dpkg -l | grep Ascend

----结束

使用指定运行用户安装

步骤1 将3.3 获取软件包准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd /*opt*

步骤5 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤6 设置环境变量。

说明

如果指定环境变量为root用户，则需要与export ASCEND_INSTALL_FOR_ALL=true配合使用，该场景下权限控制可能存在安全风险。

1. 通过export方式设置环境变量，该种方式设置的环境变量只在当前窗口有效，设置完后立即生效。

   export ASCEND_USER_GROUP=*usergroup*

   export ASCEND_USER_NAME=*username*

2. 通过~/.bashrc文件方式设置永久环境变量，操作如下：
   a. 以安装用户在任意目录下执行**vi ~/.bashrc**命令，打开.bashrc文件，在文件最后一行后面添加上述内容。
   b. 执行**:wq!**命令保存文件并退出。
   c. 执行**source ~/.bashrc**命令使其立即生效。

步骤7 将创建的用户名加入创建的用户组。

groupadd ${ASCEND_USER_GROUP}

useradd -g ${ASCEND_USER_GROUP} -d /home/${ASCEND_USER_NAME} -m ${ASCEND_USER_NAME} -s /bin/bash

步骤8 执行如下命令，安装deb驱动包（*驱动包名称请根据实际情况替换*）。

dpkg -i *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.deb

若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤9 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤10 执行如下命令，查询驱动安装信息。

dpkg -l | grep Ascend

步骤11 （**可选**）执行如下命令，清除相关环境变量。

unset ASCEND_USER_NAME

unset ASCEND_USER_GROUP

unset ASCEND_INSTALL_FOR_ALL

----结束

通过 apt-get 安装

默认用户安装

步骤1 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤2 执行以下命令，安装本地源中的deb驱动包。

- **首次安装**

  apt-get install ascend310-driver

- **重新安装**

  apt-get --reinstall install ascend310-driver

  若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤3 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤4 执行如下命令，查询驱动安装信息。

apt list ascend310-driver

----结束

指定用户安装

步骤1 （**可选**）执行如下命令，调用--install-for-all参数，允许其他用户具有安装群组权限。

export ASCEND_INSTALL_FOR_ALL=true

说明

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

当安装或者升级携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。

步骤2 设置环境变量。

说明

如果指定环境变量为root用户，则需要与export ASCEND_INSTALL_FOR_ALL=true配合使用，该场景下权限控制可能存在安全风险。

1. 通过export方式设置环境变量，该种方式设置的环境变量只在当前窗口有效，设置完后立即生效。

   export ASCEND_USER_GROUP=*usergroup*

   export ASCEND_USER_NAME=*username*

2. 通过~/.bashrc文件方式设置永久环境变量，操作如下：

   a. 以安装用户在任意目录下执行**vi ~/.bashrc**命令，打开.bashrc文件，在文件最后一行后面添加上述内容。

   b. 执行**:wq!**命令保存文件并退出。

   c. 执行**source ~/.bashrc**命令使其立即生效。

步骤3 将创建的用户名加入创建的用户组。

groupadd ${ASCEND_USER_GROUP}

useradd -g ${ASCEND_USER_GROUP} -d /home/${ASCEND_USER_NAME} -m ${ASCEND_USER_NAME} -s /bin/bash

步骤4 执行以下命令，安装本地源中的deb驱动包。

- **首次安装**

  apt-get install ascend310-driver

- **重新安装**

  apt-get --reinstall install ascend310-driver

  若系统出现如下关键回显信息，则表示驱动安装成功。

  ```
  Driver package installed successfully!
  ```

步骤5 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤6 执行如下命令，查询驱动安装信息。

apt list ascend310-driver

步骤7 （**可选**）执行如下命令，清除相关环境变量。

unset ASCEND_USER_NAME

unset ASCEND_USER_GROUP

unset ASCEND_INSTALL_FOR_ALL

----结束

3.5.4 重构驱动包安装

3.5.4.1 安装依赖

GCC 和内核源码

确认目标OS及其内核版本号对应的GCC及内核源码是否已存在，具体请参见**3.5.3.1 安装依赖**。

makeself 2.4.0 工具

下载开源工具：makeself 2.4.0。下载地址请参见：**https://github.com/megastep/makeself**或**https://github.com/megastep/makeself/releases/tag/release-2.4.0**。

pigz 工具

执行如下命令，检查pigz工具是否已安装。例如：

- Ubuntu操作系统：

  dpkg-query -s pigz

- CentOS操作系统：

  rpm -qa |grep pigz

若未安装请使用对应的软件管理器进行安装。例如：

- Ubuntu操作系统：

  apt install pigz

- CentOS操作系统：

  yum install pigz

3.5.4.2 重构驱动包

用户可以根据本章节内容，构建一个满足目标OS版本，内核版本，gcc版本要求的驱动安装包。本章节以*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run包为例说明，具体操作以客户实际host系统对应驱动包为准。

重构驱动包目前支持如下几种方式：

- 直接重构驱动包
- 解压run包后重构驱动包
- 使用重构工具重构驱动包

注意

建议用户使用如下两种方式。

- 若不需要更改run包中源码的场景；则直接重构驱动包。具体请参见**直接重构驱动包**。
- 若需要修改驱动源码，则解压run包后重构驱动包。具体请参见**解压后重构驱动包**。

前提条件

编译前需要安装相关依赖，具体请参见**3.5.4.1 安装依赖**。

直接重构驱动包

步骤1 获取通用驱动包，详细信息请参见**3.3 获取软件包**。

步骤2 上传通用驱动包到运行环境的任意目录。如：/opt下。

步骤3 执行以下命令，进入**步骤2**软件包所在路径，如/opt下。

cd /*opt*

步骤4 执行以下命令，增加通用驱动包的执行权限。

chmod +x *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run

步骤5 在root用户下执行以下命令，构建新驱动包。

命令：**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--repack** *[package_name]*

示例：**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--repack** *{product name}*-npu-driver-repack.run

说明

若package_name参数为空，则会在当前路径下生成一个命名为“原驱动包名称-custom.run”的文件。

步骤6 请参见**3.5.4.3 安装驱动（适用于.run格式）**，安装驱动。

----结束

解压后重构驱动包

步骤1 获取通用驱动包，详细信息请参见**3.3 获取软件包**。

步骤2 上传通用驱动包到运行环境的任意目录。如：/opt下。

步骤3 执行以下命令，进入**步骤2**软件包所在路径，如/opt下。

cd /*opt*

步骤4 执行以下命令，增加通用驱动包的执行权限。

chmod +x *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run

步骤5 执行以下命令，解压通用驱动包到指定目录。

命令：**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--noexec --extract=***目标路径*

示例：**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--noexec --extract=./tmp**

步骤6 在root用户下执行以下命令，构建新驱动包。

命令：**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--repack-path=***<path>* *[package_name]*

示例：**./***{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--repack-path=tmp/***{product name}*-npu-driver-repack.run

说明

若package_name参数为空，则会在当前路径下生成一个命名为“原驱动包名称-custom.run”的文件。

步骤7 请参见**3.5.4.3 安装驱动（适用于.run格式）**，安装驱动。

----结束

使用重构工具重构驱动包

步骤1 根据**3.3 获取软件包**获取源码驱动包和重构工具包。

步骤2 上传源码驱动包、重构工具包、开源软件“makeself-release-2.4.0.zip”到运行环境的同一目录。如：/opt下。

步骤3 执行以下命令，进入**步骤2**软件包所在路径，如/opt下。

cd /*opt*

步骤4 执行以下命令，增加源码驱动包的执行权限。

chmod +x *{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run

步骤5 执行以下命令，解压源码驱动包到指定目录。

./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--noexec --extract=./tmp**

步骤6 执行以下命令，解压重构工具包到当前目录。

tar -xzvf *{product name}-npu-driver-repack-tools_x.x.x.tar.gz* **--strip-components 1**

步骤7 执行以下命令，解压软件包“makeself-release-2.4.0.zip”。

unzip makeself-release-2.4.0.zip

步骤8 执行以下命令，构建新驱动包。

命令：**bash build.sh** *makeself.sh路径 包解压的路径 输出包名*

示例：**bash build.sh ./makeself-release-2.4.0/makeself.sh tmp/** *{product name}*-npu-driver-repack.run

步骤9 请参见**3.5.4.3 安装驱动（适用于.run格式）**，安装驱动。

----结束

3.5.4.3 安装驱动（适用于.run 格式）

本章以**3.5.4.2 重构驱动包**的*{product name}*-npu-driver-repack.run为例介绍.run格式驱动包安装相关指导，具体请根据实际情况进行替换。

注意事项

- 如果有安装NPU 1.X.X系列版本的驱动包，在安装NPU 20.X.X及以上系列版本的驱动包前需要先卸载低版本驱动。

- 在软件包安装或升级过程中，不允许对Host或Device进行复位或下电操作，否则会出现Device无法正常启动或者升级失败。

操作步骤

须知

首次安装请按照“**驱动->固件**”的顺序；覆盖安装或升级请按照“**固件->驱动**”的顺序，分别安装软件包。固件安装请参见**3.6.1 安装固件（适用于.run格式）**。

首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。

对于Atlas 300I 推理卡（型号 3000），为保证驱动版本、固件版本和MCU版本三者保持配套关系，请参考《**Atlas 300I 推理卡 5.1.RC2 NPU驱动和固件升级指南（型号3000,3010）**》中“升级MCU”章节升级对应的MCU组件。

步骤1 将**3.5.4.2 重构驱动包**构建的驱动包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd */opt*

步骤5 执行如下命令，校验run安装包的一致性和完整性。

./*{product name}-npu-driver-repack.run* **--check**

出现如下回显信息，表示软件包校验成功。

```
Verifying archive integrity...  100%   SHA256 checksums are OK. All good.
```

说明

软件包校验使用SHA256加密方式；校验过程中若提示：*./{product name}-npu-driver-repack.run* does not contain an embedded MD5 checksum.和*./{product name}-npu-driver-repack.run* does not contain a CRC checksum，表示没有使用MD5和CRC加密方式，可忽略。

步骤6 执行安装命令完成安装。

- 若不指定安装路径，可执行**./***{product name}*-npu-driver-repack.run **--full**命令，完成安装。
- 若用户指定安装路径，以“/test/HiAI/”为例。

  可执行**./***{product name}-npu-driver-repack.run* **--full --install-path=** /test/HiAI/命令，完成安装。

说明

- 安装详细日志路径：/var/log/ascend_seclog/ascend_install.log。
- 安装后软件包的安装路径、安装命令以及运行用户信息记录路径为“/etc/ascend_install.info”。
- 安装run包驱动时，会将动态库libdcmi.so和头文件dcmi_interface_api.h拷贝到“/usr/local/dcmi/”目录下。
- 若创建运行用户是非HwHiAiUser，安装驱动软件包时需要指定运行用户（通过**--install-username=***username* **--install-usergroup=***usergroup*参数指定）。
- 如果指定root用户为运行用户，则需要与--install-for-all参数配合使用，如下所示，该场景下权限控制可能存在安全风险。
  ```
  --install-username=root  --install-usergroup=root  --install-for-all
  ```
- 如果服务器无BMC时，只能采用默认路径安装驱动。如果采用指定路径安装，ipmi_si驱动加载时会出现阻塞，导致磁盘挂载慢，影响npu正常运行。
- Device侧系统类日志通过msnpureport工具传输到Host侧进行查看。导出操作以及导出日志的存储路径具体请参见对应产品和版本的《黑匣子日志参考》的“msnpureport工具使用”章节。容器内不支持查看Device侧系统类日志，也不支持通过msnpureport工具导出Device侧系统类日志。
- 安装驱动过程中出现：“driver ko vermagic(*) is different from the os(*), need rebuild driver”，请参见**3.5.3 源码编译安装**解决。

若系统出现如下关键回显信息，则表示驱动安装成功。

```
Driver package installed successfully!
```

步骤7 根据系统提示信息决定是否重启系统，若需要重启，请执行以下命令；否则，请跳过此步骤。

reboot

步骤8 执行**npu-smi info**查看驱动加载是否成功。

若出现如下回显信息，说明加载成功。否则，说明加载失败。请联系华为技术支持处理。

```
+------------------------------------------------------------------------------------------------+
| npu-smi 22.0.2                       Version: 22.0.2                                           |
+-------------------+-----------------+-----------------------------------------------------+
| NPU     Name      | Health          | Power(W)          Temp(C)            Hugepages-Usage(page) |
| Chip    Device    | Bus-Id          | AICore(%)         Memory-Usage(MB)                         |
+===================+=================
+=====================================================================+
| 6       310       | OK              | 12.8              73            0       / 970              |
| 0       3         | 0000:89:00.0    | 0                 2703 / 8192                              |
+-------------------+-----------------+-----------------------------------------------------+
| 6       310       | OK              | 12.8              74            0       / 970              |
| 1       4         | 0000:8A:00.0    | 0                 2867 / 8192                              |
+-------------------+-----------------+-----------------------------------------------------+
| 6       310       | OK              | 12.8              70            0       / 970              |
| 2       5         | 0000:8B:00.0    | 0                 2867 / 8192                              |
+-------------------+-----------------+-----------------------------------------------------+
| 6       310       | OK              | 12.8              65            0       / 970              |
| 3       6         | 0000:8C:00.0    | 0                 2867 / 8192                              |
+===================+=================
+=====================================================================+
```

说明

- 加载失败，可以执行**dmesg**命令查看linux启动日志。如果出现 /*安装路径*/driver/device/davinci_mini.fd copy err信息，请参考**3.7.2 卸载驱动（适用于.run格式）**章节卸载驱动，并使用默认路径重新进行驱动安装。
- 回显中“npu-smi”后面的字段为npu-smi工具版本号；“Version:”后面的字段为驱动版本号。

----结束

3.6 安装固件

本章节主要介绍固件在宿主机上的安装方法，不支持在docker容器内安装。

3.6.1 安装固件（适用于.run 格式）

首次安装请按照“**驱动->固件**”的顺序；覆盖安装或升级请按照“**固件->驱动**”的顺序，分别安装软件包，驱动安装请参见**3.5 安装驱动**，本章节以*{product name}*-npu-firmware_*x.x.x*.run为例介绍宿主机上.run格式固件包的安装方法。

注意事项

- 在软件包安装或升级过程中，不允许对Host或Device进行复位或下电操作，否则会出现Device无法正常启动或者升级失败。
- 扩容或备件替换场景下，请执行**npu-smi info -t board -i** *id* 命令查询相应的版本号，建议服务器上所有Atlas 300I 推理卡的固件版本保持一致，若不一致，请先进行固件升级，否则可能导致服务器无法识别Atlas 300I 推理卡或Atlas 300I 推理卡在业务运行过程中出现丢卡等问题。

操作步骤

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd /*opt*

步骤5 执行如下命令，增加软件包的可执行权限。

chmod +x *{product name}-npu-firmware_x.x.x.run*

步骤6 执行如下命令，校验run安装包的一致性和完整性。

./*{product name}-npu-firmware_x.x.x.run* **--check**

出现如下回显信息，表示软件包校验成功。

```
Verifying archive integrity...  100%   SHA256 checksums are OK. All good.
```

说明

软件包校验使用SHA256加密方式；校验过程中若提示：*./{product name}-npu-firmware_x.x.x.run* does not contain an embedded MD5 checksum.和./*{product name}-npu-firmware_x.x.x.run* does not contain a CRC checksum，表示没有使用MD5和CRC加密方式，可忽略。

步骤7 执行如下安装命令完成安装。

./*{product name}-npu-firmware_x.x.x.run* **--full**

若系统出现如下关键回显信息，表示固件安装成功，并根据提示信息决定是否立即重启系统。

```
Firmware package installed successfully!
```

说明

- 若重新安装固件后安装配套版本的驱动，则无需重启系统，驱动安装后固件版本会自动生效。
- 若单独安装固件，则需执行**reboot**命令重启系统或执行**npu-smi set -t reset**命令复位标卡或芯片生效。npu-smi命令详细请参见**《Atlas 300I 推理卡 5.1.RC2 npu-smi 命令参考（型号 3000,3010）》**。需保证固件与驱动版本配套，否则可能导致无法启动的兼容性问题。
- 安装固件软件包不支持指定运行用户名和用户组，共用驱动的运行用户名和用户组。
- 安装固件软件包不支持指定路径，共用驱动的安装路径。
- 软件包默认安装路径：/usr/local/Ascend
- 安装详细日志路径：/var/log/ascend_seclog/ascend_install.log。
- 安装后软件包的安装路径、安装命令以及运行用户信息记录路径：/etc/ascend_install.info
- Device侧系统类日志通过msnpureport工具传输到Host侧进行查看。导出操作以及导出日志的存储路径具体请参见对应产品和版本的《黑匣子日志参考》的“msnpureport工具使用”章节。容器内不支持查看Device侧系统类日志，也不支持通过msnpureport工具导出Device侧系统类日志。

步骤8 执行如下命令查看芯片固件版本号。若与目标版本一致，则说明安装成功。

*/usr/local/Ascend/***driver/tools/upgrade-tool --device_index -1 --component -1 --version**

说明

如果安装驱动时指定安装路径，则命令中的“*/usr/local/Ascend*”请根据实际情况替换。

----结束

3.6.2 安装固件（适用于.rpm 格式）

本章以*{product name}*-npu-firmware-*x.x.x-<release>*.noarch.rpm为例介绍宿主机上.rpm格式固件包的安装方法，具体请根据实际情况进行替换。

注意事项

- 当前rpm包安装不支持修改安装路径功能，请按照正常rpm流程安装，不支持rpm包的--force命令安装升级。
- Ascend 310内部部分功能，只支持HwHiAiUser用户访问和执行，用户首次安装时系统会自动创建HwHiAiUser用户。
- 安装完成后需要重启生效。卸载完成后，如需再次安装，需要先重启再安装。

- 如果已安装NPU 1.X.X系列版本的rpm包，在安装NPU 20.X.X及以上系列版本的rpm包前需要先卸载原版本再安装。
- 在软件包安装或升级过程中，不允许对Host或Device进行复位或下电操作，否则会出现Device无法正常启动或者升级失败。
- 扩容或备件替换场景下，请执行**npu-smi info -t board -i** *id* 命令查询相应的版本号，建议服务器上所有Atlas 300I 推理卡的固件版本保持一致，若不一致，请先进行固件升级，否则可能导致服务器无法识别Atlas 300I 推理卡或Atlas 300I 推理卡在业务运行过程中出现丢卡等问题。

操作步骤

须知

请按照“**驱动->固件**”的顺序，分别安装组件软件包。驱动安装请参见**3.5 安装驱动**。

步骤1 将**3.3 获取软件包**准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 在软件包所在路径执行如下命令完成Host侧固件包安装。

rpm -ivh *{product name}-npu-firmware-x.x.x-<release>.noarch.rpm*

若系统出现如下关键回显信息，表示固件安装成功，并根据提示信息决定是否立即重启系统。

```
Firmware package installed successfully!
```

说明

- 若重新安装固件后安装配套版本的驱动，则无需重启系统，驱动安装后固件版本会自动生效。
- 若单独安装固件，则需执行**reboot**命令重启系统或执行**npu-smi set -t reset**命令复位标卡或芯片生效。npu-smi命令详细请参见**《Atlas 300I 推理卡 5.1.RC2 npu-smi 命令参考（型号 3000,3010）》**。需保证固件与驱动版本配套，否则可能导致无法启动的兼容性问题。
- 软件包安装过程中，若无HwHiAiUser用户系统会自动创建该用户。
- 安装过程中的日志信息输出在“/var/log/ascend_seclog/ascend_install.log”文件中，用户可以执行**vim /var/log/ascend_seclog/ascend_install.log**命令打开日志。

步骤5 执行如下命令查看芯片固件版本号。若与目标版本一致，则说明安装成功。

/usr/local/Ascend/driver/tools/upgrade-tool --device_index -1 --component -1 --version

----结束

3.6.3 安装固件（适用于.deb 格式）

dpkg 安装

步骤1 将3.3 获取软件包准备的安装包上传至Linux系统任意目录下（如/opt）。

步骤2 使用PuTTY登录服务器的OS命令行。

步骤3 执行如下命令，切换至root用户。

su - root

步骤4 执行如下命令，进入软件包所在路径（如/opt）。

cd */opt*

步骤5 执行如下命令，安装deb固件包（*固件包名称请根据实际情况替换*）。

dpkg -i *{product name}-npu-firmware_x.x.x.deb*

若系统出现如下关键回显信息，表示固件安装成功，并根据提示信息决定是否立即重启系统。

```
Firmware package installed successfully!
```

说明

- 若重新安装固件后安装配套版本的驱动，则无需重启系统，驱动安装后固件版本会自动生效。
- 若单独安装固件，则需执行**reboot**命令重启系统或执行**npu-smi set -t reset**命令复位标卡或芯片生效。npu-smi命令详细请参见《Atlas 300I 推理卡 5.1.RC2 npu-smi 命令参考（型号 3000,3010）》。需保证固件与驱动版本配套，否则可能导致无法启动的兼容性问题。

步骤6 执行如下命令，查询固件安装信息。

dpkg -l | grep Ascend

----结束

apt-get 安装

步骤1 执行如下命令，安装本地源中的deb固件包。

- 首次安装

  apt-get install ascend310-firmware

- 重新安装

  apt-get --reinstall install ascend310-firmware

  若系统出现如下关键回显信息，表示固件安装成功，并根据提示信息决定是否立即重启系统。

  ```
  Firmware package installed successfully!
  ```

说明

- 固件对驱动有依赖关系，执行此命令会先安装好驱动包，再装固件。
- 若重新安装固件后安装配套版本的驱动，则无需重启系统，驱动安装后固件版本会自动生效。
- 若单独安装固件，则需执行**reboot**命令重启系统或执行**npu-smi set -t reset**命令复位标卡或芯片生效。npu-smi命令详细请参见《**Atlas 300I 推理卡 5.1.RC2 npu-smi 命令参考（型号 3000,3010）**》。需保证固件与驱动版本配套，否则可能导致无法启动的兼容性问题。

步骤2 执行如下命令，查询固件安装信息。

apt list ascend310-firmware

----结束

3.7 卸载驱动

须知

请在软件包卸载前确保无相关业务进程占用昇腾驱动节点（包含容器映射场景），否则热复位将无法生效。

如果设备已经纳管到FusionDirector，卸载驱动后，FusionDirector界面上显示的驱动版本会延迟刷新，最长为1个小时。重启设备后版本会立即刷新。

3.7.1 确认卸载方式

表 3-5 卸载方式说明

查询命令	说明	卸载方式
dpkg -l | grep Ascend 或**apt list ascend310***	若可以查询到驱动或固件，则说明使用deb包安装。	请参见**3.7.4 卸载驱动（适用于.deb格式）**或**3.8.3 卸载固件（适用于.deb格式）**的“dpkg卸载”或“apt-get卸载”。
rpm -qa | grep npu	若可以查询到驱动或固件，则说明使用rpm包安装。	请参见**3.7.3 卸载驱动（适用于.rpm格式）**或**3.8.2 卸载固件（适用于.rpm格式）**。

查询命令	说明	卸载方式
• **dpkg -l | grep Ascend** • **apt list ascend310*** • **rpm -qa | grep npu**	若使用这几种命令都查询不到驱动或固件，则说明使用run包安装。	请参见**3.7.2 卸载驱动（适用于.run格式）**或**3.8.1 卸载固件（适用于.run格式）**。

3.7.2 卸载驱动（适用于.run 格式）

目前发布的run包在宿主机上的卸载方式相同。本章以*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run为例介绍.run格式驱动包卸载相关操作，具体请根据实际情况进行替换。

前提条件

未安装run安装包时不能使用卸载功能。

操作步骤

步骤1 使用PuTTY登录服务器的OS命令行。

步骤2 执行如下命令，切换至root用户。

su - root

步骤3 支持以下两种卸载方式，用户根据实际情况选择其中一个方式卸载即可：

- 在任意路径执行如下命令卸载软件包：

  <install-path>/**driver/script/uninstall.sh**

  说明

  *<install-path>*表示软件包安装路径，可以执行**cat /etc/ascend_install.info**命令查询安装路径；请根据实际情况替换。

- 在软件包所在路径执行以下命令进行卸载：

  ./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--uninstall**

  注意

  若业务或容器占用了davinci节点，卸载过程中系统会提示“The davinci nodes are occupied by some processes, please stop processes and install or uninstall again, details in : /var/log/ascend_seclog/ascend_install.log”报错，可以选择使用**--force**参数进行强制卸载，详细请参见**7.1 参数说明/常用命令**。但可能导致当前正在运行的业务受到影响。

  若出现如下关键回显信息，则表示驱动卸载成功。

  ```
  Driver package uninstalled successfully!
  ```

步骤4 根据系统提示信息决定是否重启服务器，若需要重启系统，请执行以下命令；否则，请跳过此步骤。

reboot

须知

驱动卸载后，npu-smi工具不可用。

----结束

3.7.3 卸载驱动（适用于.rpm 格式）

本章主要介绍Host侧驱动卸载的相关指导。

前提条件

未安装rpm安装包时不能使用卸载功能。

操作步骤

步骤1 使用PuTTY登录服务器的OS命令行。

步骤2 执行如下命令，切换至root用户。

su - root

步骤3 执行如下命令查看rpm驱动列表。

rpm -qa | grep npu

回显中包含“npu-driver”字段表示存在驱动，如下所示：

```
npu-driver-22.0.2-1.aarch64
```

步骤4 执行如下命令卸载驱动。

rpm -e *npu-driver-22.0.2-1.aarch64*

说明

上述命令中软件名称请根据**步骤3**的查询情况进行替换。

若出现如下关键回显信息，则表示驱动卸载成功。

```
Driver package uninstalled successfully!
```

步骤5 根据系统提示信息决定是否重启服务器，若需要重启系统，请执行以下命令；否则，请跳过此步骤。

reboot

须知

驱动卸载后，npu-smi工具不可用。

----结束

3.7.4 卸载驱动（适用于.deb 格式）

dpkg 卸载

步骤1 执行以下命令，查询驱动安装信息。

dpkg -l | grep Ascend

步骤2 根据步骤1的查询结果，执行以下命令进行卸载。

dpkg -r ascend310-driver

若出现如下关键回显信息，则表示驱动卸载成功。

```
Driver package uninstalled successfully!
```

说明

- 固件对驱动有依赖关系，执行此命令需先卸载固件包，再卸载驱动。
- 如果在卸载软件包时出现重新安装其他包的情景，说明被安装的软件包曾经安装失败，需要对安装失败的软件包重新安装，确保安装成功。安装失败软件包先参考**6.2 deb包安装、升级或卸载失败后的处理方式**进行处理。

步骤3 根据系统提示信息决定是否重启服务器，若需要重启系统，请执行以下命令；否则，请跳过此步骤。

reboot

须知

驱动卸载后，npu-smi工具不可用。

----结束

apt-get 卸载

步骤1 执行以下命令，查询驱动安装信息。

apt list ascend310*

步骤2 根据步骤1的查询结果，执行以下命令进行卸载。

apt-get remove ascend310-driver

若出现如下关键回显信息，则表示驱动卸载成功。

```
Driver package uninstalled successfully!
```

说明

- 固件对驱动有依赖关系，执行此命令需先卸载固件包，再卸载驱动。
- 如果在卸载软件包时出现重新安装其他包的情景，说明被安装的软件包曾经安装失败，需要对安装失败的软件包重新安装，确保安装成功。安装失败软件包先参考**6.2 deb包安装、升级或卸载失败后的处理方式**进行处理。

步骤3 根据系统提示信息决定是否重启服务器，若需要重启系统，请执行以下命令；否则，请跳过此步骤。

reboot

须知

驱动卸载后，npu-smi工具不可用。

----结束

3.8 卸载固件

3.8.1 卸载固件（适用于.run 格式）

驱动和固件的卸载没有先后顺序要求。驱动卸载请参见**3.7.2 卸载驱动（适用于.run格式）**。本章以*{product name}*-npu-firmware_*x.x.x*.run为例介绍固件卸载相关指导，具体请根据实际情况进行替换。

前提条件

未安装run安装包时不能使用卸载功能。

操作步骤

步骤1 使用PuTTY登录服务器的OS命令行。

步骤2 执行如下命令，切换至root用户。

su - root

步骤3 支持以下两种方式，用户根据实际情况选择其中一个方式卸载即可：

- 在任意路径执行如下命令卸载软件包：

  <install-path>/**firmware/script/uninstall.sh**

  说明

  *<install-path>*表示软件包安装路径，可以执行**cat /etc/ascend_install.info**命令查询安装路径；请根据实际情况替换。

- 在软件包所在路径执行以下命令进行卸载：

  ./*{product name}*-npu-firmware_*x.x.x*.run **--uninstall**

  若出现如下关键回显信息，则表示固件卸载成功。

  ```
  Firmware package uninstalled successfully!
  ```

步骤4 根据系统提示信息决定是否重启服务器，若需要重启系统，请执行以下命令；否则，请跳过此步骤。

reboot

----结束

3.8.2 卸载固件（适用于.rpm 格式）

本章主要介绍Host侧固件卸载的相关指导。

前提条件

未安装rpm安装包时不能使用卸载功能。

操作步骤

步骤1 使用PuTTY登录服务器的OS命令行。

步骤2 执行如下命令，切换至root用户。

su - root

步骤3 执行如下命令查看rpm固件列表。

rpm -qa | grep npu

回显中包含“npu-firmware”字段表示存在固件，如下所示：

```
npu-firmware-1.82.22.2.220-1.noarch
```

步骤4 执行如下命令卸载固件。

rpm -e *npu-firmware-1.82.22.2.220-1.noarch*

说明

上述命令中软件名称请根据**步骤3**的查询情况进行替换。

若出现如下关键回显信息，则表示固件卸载成功。

```
Firmware package uninstalled successfully!
```

步骤5 根据系统提示信息决定是否重启服务器，若需要重启系统，请执行以下命令；否则，请跳过此步骤。

reboot

----结束

3.8.3 卸载固件（适用于.deb 格式）

dpkg 卸载

步骤1 执行以下命令，查询固件安装信息。

dpkg -l | grep Ascend

步骤2 根据**步骤1**的查询结果，执行以下命令进行卸载。

dpkg -r ascend310-firmware

若出现如下关键回显信息，则表示固件卸载成功。

```
Firmware package uninstalled successfully!
```

说明

如果在卸载软件包时出现重新安装其他包的情景，说明被安装的软件包曾经安装失败，需要对安装失败的软件包重新安装，确保安装成功。安装失败软件包先参考**6.2 deb包安装、升级或卸载失败后的处理方式**进行处理。

----结束

apt-get 卸载

步骤1 执行以下命令，查询固件安装信息。

apt list ascend310*

步骤2 根据步骤2的查询结果，执行以下命令进行卸载。

apt-get remove ascend310-firmware

若出现如下关键回显信息，则表示固件卸载成功。

```
Firmware package uninstalled successfully!
```

说明

如果在卸载软件包时出现重新安装其他包的情景，说明被安装的软件包曾经安装失败，需要对安装失败的软件包重新安装，确保安装成功。安装失败软件包先参考**6.2 deb包安装、升级或卸载失败后的处理方式**进行处理。

----结束

4 容器内安装

4.1 安装前必读
4.2 安装场景
4.3 容器内软件包卸载

4.1 安装前必读

本章节介绍软件包在容器内的安装以及卸载的详细流程。使用容器安装软件包之前，请务必查看如下限制：

- Firmware软件包不支持docker内安装。
- 容器内安装的Driver软件包版本需要与宿主机安装的Driver软件包版本保持一致，不支持独立升级。
- 运行环境中的一个device只能被一个容器使用，只有当使用该device的容器退出后，该device才可以被其他容器使用。
- Driver支持docker内安装，只安装驱动的.so，即安装驱动包的用户态部分。根据宿主机目录（默认路径为：/usr/local/Ascend/driver/lib64/ ）是否挂载到容器，分为如下两种安装方式，两种方式二选一，用户根据实际情况进行选择。
  - 将宿主机的目录挂载到容器中，请参见**4.2.1 宿主机目录挂载容器**。
  - 宿主机的目录不挂载容器，请参见**4.2.2 宿主机目录不挂载容器**。

宿主机与容器操作系统兼容性关系

宿主机操作系统与容器操作系统的兼容性关系如**表4-1**。

表 4-1 兼容性关系

操作系统架构	宿主机操作系统版本	宿主机操作系统内核版本	容器操作系统版本	说明
ARM	Ubuntu 18.04.1	4.15.0-29-generic	Ubuntu 18.04	容器OS目前支持的Ubuntu操作系统版本为18.04.1、18.04.4和18.04.5。

操作系统架构	宿主机操作系统版本	宿主机操作系统内核版本	容器操作系统版本	说明
			CentOS 7.6	NA
			CentOS 7.8	不支持使用宿主机目录挂载容器方式安装Driver软件包。
	CentOS 7.6	4.14.0-115.el7a.0.1.aarch64	CentOS 7.6	NA
			CentOS 7.8	NA
			Ubuntu 18.04	容器OS目前支持的Ubuntu操作系统版本为18.04.1、18.04.4和18.04.5。
	统信服务器操作系统V20（1020e）	4.19.90-2106.3.0.0095.up2.uel20.aarch64	Ubuntu 18.04.5	NA
			CentOS 7.6	NA
	Ubuntu 20.04	5.4.0-26-generic	Ubuntu 18.04.5	NA
			CentOS 7.6	NA
	EulerOS 2.10	4.19.90-vhulk2111.1.0.h963.eulerosv2r10.aarch64	EulerOS 2.10	NA

说明

以上表格中的容器操作系统为已验证过的版本，具体请以实际获取的操作系统镜像版本为准。

4.2 安装场景

4.2.1 宿主机目录挂载容器

前提条件

1. 宿主机已经安装过Driver和Firmware，详情请参见**3.5 安装驱动**和**3.6 安装固件**。
2. 需用户在宿主机自行安装docker。
3. 重启服务与容器，命令为：

```
systemctl daemon-reload
systemctl restart docker
```

4. 获取OS镜像，例如：容器镜像为Ubuntu 18.04，命令为：

```
docker pull ubuntu:18.04
```

如果获取CentOS 7.6容器镜像，建议从**Ascend Hub**上获取。

5. 查看镜像，命令为：

```
docker images
```

安装步骤

1. 在宿主机使用**id HwHiAiUser**命令查看宿主机HwHiAiUser的gid，并记录该gid的取值，如图4-1所示，则记录gid为1001。

   如果用户安装Driver包时使用**--install-username=*username* --install-usergroup=*usergroup***参数指定了其他非root用户，则需要使用同样的方法记录此非root用户的gid，容器内创建非root用户时，需要使用该gid，确保宿主机和容器内相关进程的非root用户同属组。

   图 4-1 查看宿主机 HwHiAiUser 的 gid

```
root@ascend-HP-ProDesk-600-G4-PCI-MT:~# id HwHiAiUser
uid=1027(HwHiAiUser) gid=1001(HwHiAiUser) groups=1001(HwHiAiUser)
```

2. 在宿主机创建并启动docker容器。

   说明

   多容器场景下，为了防止某个容器的恶意代码大量申请甚至耗尽内核态内存，从而对其它容器造成影响。需要用户根据实际情况通过”--kernel-memory”参数指定容器能够占用的最大内核态内存。此功能依赖客户主机侧内核开启kmem accounting功能。

   需要将表4-2所示参数挂载到容器：

   表 4-2 参数说明

参数	参数说明
--device	表示映射的设备，可以挂载一个或者多个设备。
-v /usr/local/bin/npu-smi:/usr/local/bin/npu-smi	将宿主机npu-smi工具“/usr/local/bin/npu-smi”挂载到容器中，请根据实际情况修改。
-v /usr/local/Ascend/driver/lib64:/usr/local/Ascend/driver/lib64	将宿主机目录“/usr/local/Ascend/driver/lib64/”挂载到容器，请根据driver的驱动.so所在路径修改。
-v /etc/ascend_install.info:/etc/ascend_install.info	将宿主机安装信息文件“/etc/ascend_install.info”挂载到容器中。
-v /usr/local/Ascend/driver/version.info:/usr/local/Ascend/driver/version.info	将宿主机版本信息文件“/usr/local/Ascend/driver/version.info”挂载到容器中，请根据实际情况修改。

   容器启动后，执行以下命令查看当前docker容器中可以使用的davinci设备：

```
ls /dev/ | grep davinci*
```

命令执行情况参见图4-2。

图 4-2 查询 docker 容器中使用的 davinci 设备

```
root@ascend-HP-ProDesk-600-G4-PCI-MT:/# ls /dev/ | grep davinci*
davinci0
davinci_manager
```

其中：

- davinci_manager为管理模块的字符设备节点。
- davinci0为该容器使用的davinci设备。

3. 在容器内创建HwHiAiUser用户（用户也可以创建其他非root用户），用于启动相关进程。

```
groupadd -g gid HwHiAiUser && useradd -g HwHiAiUser -d /home/HwHiAiUser -m HwHiAiUser &&
echo ok
```

请将上述命令中的gid替换为1中查出的gid值。若返回“ok”，则说明创建成功。

如果用户创建了其他非root用户，请确保该用户所属的属组必须和Driver运行用户所属属组相同；如果不同，请用户自行添加到Driver运行用户属组。

4. 设置如下环境变量，用于在容器中加载驱动so：

```
export LD_LIBRARY_PATH=/usr/local/Ascend/driver/lib64:${LD_LIBRARY_PATH}
```

其中“/usr/local/Ascend”为默认安装路径，请根据实际情况修改。通过export方式设置环境变量，只在当前窗口有效，设置完成后立即生效。

4.2.2 宿主机目录不挂载容器

前提条件

1. 宿主机已经安装过Driver和Firmware，详情请参见3.5 安装驱动和3.6 安装固件。
2. 需用户在宿主机自行安装docker。
3. 重启服务与容器，命令为：

```
systemctl daemon-reload
systemctl restart docker
```

4. 获取OS镜像，例如：容器镜像为Ubuntu 18.04，命令为：

```
docker pull ubuntu:18.04
```

如果获取CentOS 7.6容器镜像，建议从**Ascend Hub**上获取。

5. 查看镜像，命令为：

```
docker images
```

操作步骤

1. 在宿主机使用**id HwHiAiUser**命令查看宿主机HwHiAiUser的gid，并记录该gid的取值，如图4-3所示，则记录gid为1001。

如果用户安装Driver包时使用**--install-username=*username* --install-usergroup=*usergroup***参数指定了其他非root用户，则需要使用同样的方法记录此非root用户的gid，容器内创建非root用户时，需要使用该gid，确保宿主机和容器内相关进程的非root用户同属组。

图 4-3 查看宿主机 HwHiAiUser 的 gid

```
root@ascend-HP-ProDesk-600-G4-PCI-MT:~# id HwHiAiUser
uid=1027(HwHiAiUser) gid=1001(HwHiAiUser) groups=1001(HwHiAiUser)
```

2. 在宿主机创建并启动docker容器。

说明

多容器场景下，为了防止某个容器的恶意代码大量申请甚至耗尽内核态内存，从而对其它容器造成影响。需要用户根据实际情况通过”--kernel-memory”参数指定容器能够占用的最大内核态内存。此功能依赖客户主机侧内核开启kmem accounting功能。

需要将表4-3参数挂载到容器：

表 4-3 参数说明

参数	参数说明
--device	表示映射的设备，可以挂载一个或者多个设备。
-v /usr/local/bin/npu-smi:/usr/local/bin/npu-smi	将宿主机npu-smi工具“/usr/local/bin/npu-smi”挂载到容器中，请根据实际情况修改。

容器启动后，执行以下命令查看当前docker容器中可以使用的davinci设备：

```
ls /dev/ | grep davinci*
```

命令执行情况参见图4-4。

图 4-4 查询 docker 容器中使用的 davinci 设备

```
root@ascend-HP-ProDesk-600-G4-PCI-MT:/# ls /dev/ | grep davinci*
davinci0
davinci_manager
```

其中：

- davinci_manager为管理模块的字符设备节点。
- davinci0为该容器使用的davinci设备。

3. 在容器内创建HwHiAiUser用户（用户也可以创建其他非root用户），用于启动相关进程。

```
groupadd -g gid HwHiAiUser && useradd -g HwHiAiUser -d /home/HwHiAiUser -m HwHiAiUser &&
echo ok
```

请将上述命令中的gid替换为1中查出的gid值。若返回“ok”，则说明创建成功。

如果用户创建了其他非root用户，请确保该用户所属的属组必须和Driver运行用户所属属组相同；如果不同，请用户自行添加到Driver运行用户属组。

4. 使用**exit**命令退出容器，在宿主机Driver软件包所在路径，执行如下命令将软件包复制到容器内部。

```
docker cp /xxx/{product name}-npu-driver_x.x.x_linux-{arch}.run container_id:/home/HwHiAiUser/
software
```

其中xxx为宿主机上软件包的存放路径，*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run请替换为具体软件包名，*container_id*为容器ID，可以使用**docker ps -a**命令查看所使用容器的ID，*/home/HwHiAiUser/software*为容器内软件包的存放路径，如果没有该路径，请先手动创建；所有路径请用户根据实际情况进行修改。

5. 使用如下命令重新进入容器，并进入软件包所在的*software*目录，根据表4-4所示安装命令安装相应软件包。

```
docker start xxx
docker attach xxx
```

其中xxx请替换为具体容器ID或容器名。

说明

如下安装路径为默认路径举例说明如何安装。更多安装参数请参考**7.1 参数说明/常用命令**。

表 4-4 安装命令

组件	安装命令
Driver	./{product name}-npu-driver_x.x.x_linux-{arch}.run --docker 默认路径：/usr/local/Ascend 如果用户需要指定安装路径，需要通过**--install-path=<path>**参数指定。

若安装过程中无错误信息提示，则表示安装成功。

- 软件包默认安装路径：/usr/local/Ascend
- 安装详细日志路径：/var/log/ascend_seclog/ascend_install.log
- 安装后软件包的安装路径、安装命令以及运行用户信息记录路径：/etc/ascend_install.info

6. 设置如下环境变量，用于在容器中加载驱动so：

```
export LD_LIBRARY_PATH=/usr/local/Ascend/driver/lib64:${LD_LIBRARY_PATH}
```

其中“/usr/local/Ascend”为默认安装路径，请根据实际情况修改。通过export方式设置环境变量，只在当前窗口有效，设置完成后立即生效。

4.3 容器内软件包卸载

软件包卸载

容器内驱动包的卸载与宿主机一样，用户请参见**3.7 卸载驱动**卸载软件包即可。

5 常用操作

5.1 通过PuTTY登录（网口方式）
5.2 使用WinSCP传输文件
5.3 设置用户有效期
5.4 配置Host侧IP地址通过SSH方式登录Device

5.1 通过 PuTTY 登录（网口方式）

操作场景

该任务指导用户使用PuTTY通过局域网远程访问服务器，对服务器实施配置、维护操作。

说明

- 您可以访问chiark网站主页下载PuTTY软件。
- 低版本的PuTTY软件可能导致登录服务器系统失败，建议使用最新版本的PuTTY软件。

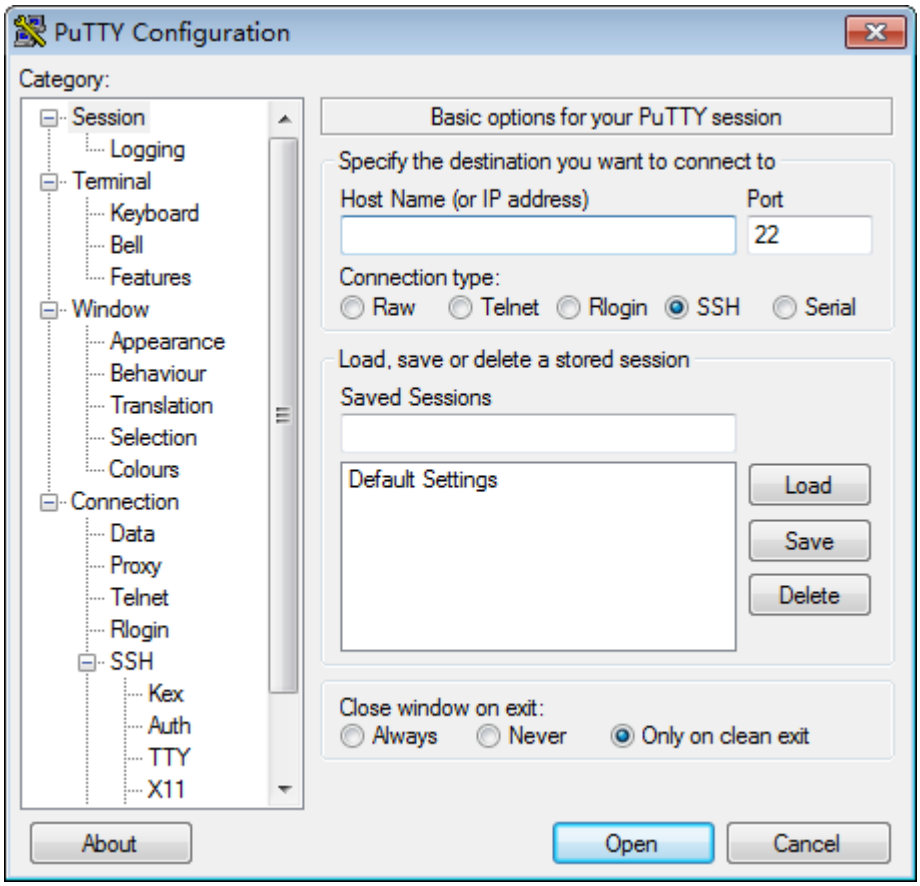
操作步骤

步骤1 设置本地PC的IP地址、子网掩码或者路由，使本地PC能和服务器网络互通。

步骤2 在本地PC上，双击“PuTTY.exe”。

弹出“PuTTY Configuration”窗口。

图 5-1 PuTTY Configuration



步骤3 在左侧导航树中，选择“Session”。

步骤4 设置登录参数。

参数说明如下：

- Host Name (or IP address)：输入要登录服务器的IP地址，如“192.168.2.10”。
- Port：默认设置为“22”。
- Connection type：默认选择“SSH”。
- Close window on exit：默认选择为“Only on clean exit”。

说明

配置“Host Name”后，再配置“Saved Sessions”并单击“Save”保存，则后续使用时直接双击“Saved Sessions”下保存的记录即可登录服务器。

步骤5 单击“Open”。

进入“PuTTY”运行界面，提示“login as:”，等待用户输入用户名。

说明

- 如果首次登录该目标服务器，则会弹出“PuTTY Security Alert”窗口。单击“是”表示信任此站点，进入“PuTTY”运行界面。
- 如果帐号输入错误，必须重新连接PuTTY。

步骤6 按提示分别输入用户名和密码。

登录完成后，命令提示符左侧显示出当前登录服务器的主机名。

----结束

5.2 使用 WinSCP 传输文件

操作场景

在PC机上使用WinSCP工具进行文件传输。

必备事项

前提条件

目的设备已开启SFTP服务。

数据

需准备如下数据：

- 待连接服务器的IP地址
- 登录待连接服务器的用户名和密码

软件

WinSCP.exe：此工具为第三方免费软件。

操作步骤

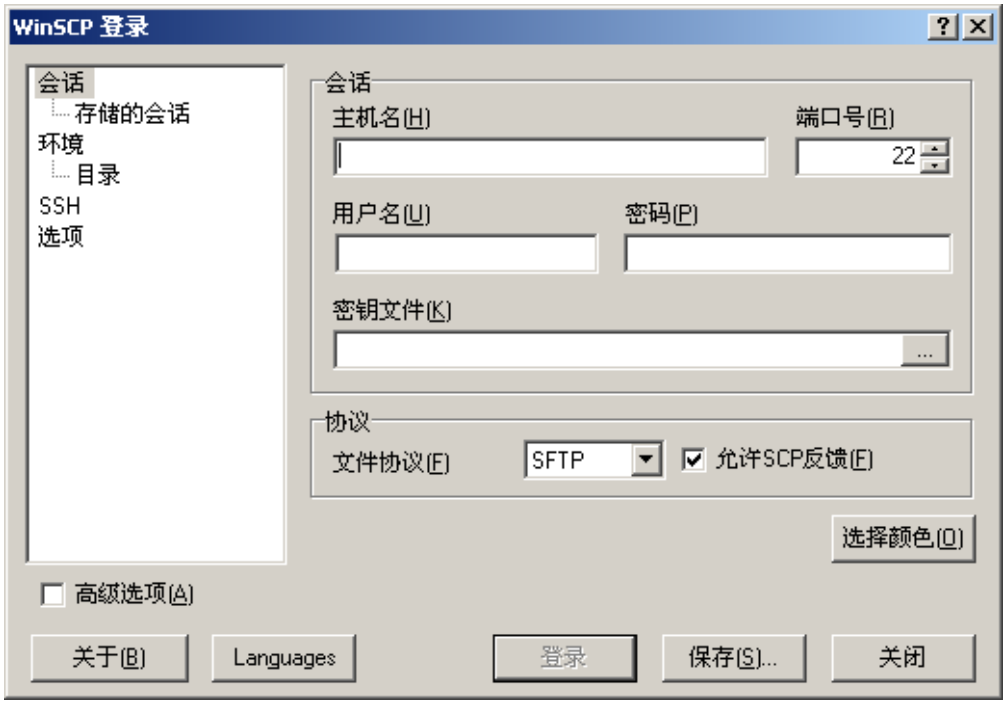
步骤1 打开“WinSCP”文件夹，双击“WinSCP.exe”。

弹出“WinSCP 登录”对话框，如图5-2所示。

说明

若系统非中文操作系统，可以单击“Languages”进行界面语言的选择。

图 5-2 WinSCP 登录



步骤2 设置登录参数。

参数说明如下。

- 主机名（H）：输入待连接设备的IP地址。例如：“192.168.2.10”。
- 端口号（R）：默认为“22”。
- 用户名（U）：输入用户名。例如：admin123。
- 密码（P）：输入密码。例如：admin123。
- 密钥文件（K）：默认为空，保留默认值。
- 协议：选择默认文件协议“SFTP”，并勾选“允许SCP反馈（F）”。

步骤3 单击“登录”。

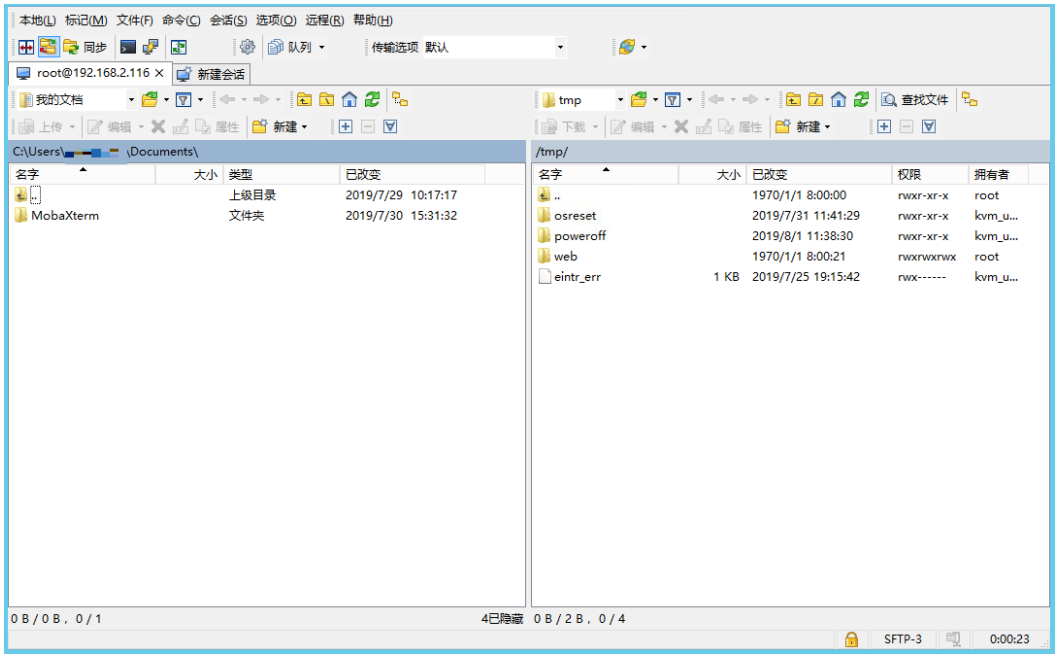
进入“WinSCP”文件传输界面。

说明

- 如果首次登录时没有选择密钥文件，此时会弹出一个警告提示框，询问“是否连接并添加密钥到缓存？”，单击“是（Y）”，进入“WinSCP”文件传输界面。
- 以Windows 7为例，界面左侧区域默认打开本地计算机的“C:\Users\Administrator\Documents”目录，右侧区域默认打开设备的“/root”目录。

步骤4 根据实际需求，在界面左右区的指定目录中进行文件夹的创建、删除和复制等操作。

图 5-3 WinSCP 界面



----结束

5.3 设置用户有效期

为保证用户的安全性，应设置用户的有效期，使用系统命令**chage**来设置用户的有效期。

命令为：

chage [**-m** *mindays*] [**-M** *maxdays*] [**-d** *lastday*] [**-I** *inactive*] [**-E** *expiredate*] [**-W** *warndays*] *user*

相关参数请参见**表5-1**。

表 5-1 设置用户有效期

参数	参数说明
-m	密码可更改的最小天数。设置为“0”表示任何时候都可以更改密码。
-M	密码保持有效的最大天数。设置为“-1”表示可删除这项密码的检测，用户密码则不会超期，存在安全风险，请谨慎使用。
-d	上一次更改密码的日期。
-I	停滞时期。过期指定天数后，设定密码为失效状态。
-E	用户到期的日期。超过该日期，此用户将不可用。
-W	密码到期前，提前收到警告信息的天数。

参数	参数说明
-l	列出用户以及密码的有效期。由非特权用户来确定用户或密码何时过期。

说明

- 表5-1只列举出常用的参数，用户可通过**chage --help**命令查询详细的参数说明。
- 日期格式为YYYY-MM-DD，如**chage -E** 2019-12-01 *test*表示用户*test*的用户在2019年12月1日过期。
- User如果不填写，默认为root用户。

例如：修改用户*test*的有效期为2019年12月31日。可执行如下命令：

chage -E *2019-12-31 test*

5.4 配置 Host 侧 IP 地址通过 SSH 方式登录 Device

操作步骤

步骤1 使用**npu-smi set -t ssh-enable**命令打开SSH开关，命令的具体使用方法请参见《Atlas 300I 推理卡 5.1.RC2 npu-smi 命令参考（型号 3000,3010）》。

步骤2 在Host侧配置IP。

IP必须与Device侧的IP为同一网段，如：192.168.1.100。Device默认IP地址从192.168.1.199依次递减，例如用户有多个Device，则IP地址分别为192.168.1.199，192.168.1.198，192.168.1.197，...，最多支持配置64个Device。

本文仅给出Host操作系统为Ubuntu 18.04和CentOS 7.6配置IP的方法，其他OS配置IP的方法可参考对应OS官网社区的FAQ去解决。

- 在Host操作系统为Ubuntu 18.04配置动态IP（动态添加IP，每次重启后都要重新操作）
  a. 执行以下命令，配置动态IP。

     ifconfig endvnic *192.168.1.100*

  b. 执行**ifconfig**命令，查看网络配置是否成功。

     若出现如下回显，表示网络配置成功。否则，请联系华为技术支持处理。

```
endvnic: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.100  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::xxxx:xxxx:xxxx:74d3  prefixlen 64  scopeid 0x20<link>
        ...
```

- 在Host操作系统为Ubuntu 18.04配置静态IP
  a. 修改“/etc/netplan”目录下的yaml配置文件。
     i. 执行以下命令，进入“/etc/netplan”目录下查看配置文件名称。

     cd /etc/netplan

     ls

     查询结果如下：

```
01-netcfg.yaml
```

ii. 执行以下命令，打开配置文件。

vi /etc/netplan/*01-netcfg.yaml*

在该文件中增加以下内容：

```
endvnic:
        addresses:  [192.168.1.100/24]
        nameservers:
            addresses:  [255.255.254.0]
```

按“Esc”切换到命令模式，输入**:wq**保存并退出。

b. 执行以下命令，使网络配置生效。

netplan apply

c. 执行**ifconfig**命令，查看网络配置是否成功。

若出现如下回显，表示网络配置成功。否则，请联系华为技术支持处理。

```
endvnic: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.1.100  netmask 255.255.255.0  broadcast 192.168.1.255
    inet6 fe80::xxxx:xxxx:xxxx:74d3  prefixlen 64  scopeid 0x20<link>
    ...
```

- 在Host操作系统为CentOS 7.6配置动态IP（动态添加IP，每次重启后都要重新操作）
  a. **systemctl status NetworkManager**
     如果不是active的，需要启动**systemctl start NetworkManager**
  b. **ifconfig -s endvnic 192.168.1.100**
  c. **systemctl disable NetworkManager**
  d. **systemctl stop NetworkManager**
- 在Host操作系统为CentOS 7.6配置静态IP（静态设置，每次重启无需重复操作）
  a. **systemctl status NetworkManager**
     如果不是active的，需要通过如下命令配置开机自启动。
     systemctl start NetworkManager
     systemctl enable NetworkManager
  b. **nmcli connect add autoconnect no type ethernet con-name endvnic ifname endvnic** 生成“/etc/sysconfig/network-scripts/ifcfg-endvnic”。
  c. 使用**ip a** 查看MAC地址，例如：

```
3: endvnic: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
link/ether 10:1b:54:xx:xx:d3 brd ff:ff:ff:ff:ff:ff
inet 192.168.1.100/24 brd 192.168.1.255 scope global endvnic
valid_lft forever preferred_lft forever
inet6 fe80::xxxx:xxxx:xxxx:f0d3/64 scope linkifconfig
valid_lft forever preferred_lft forever
其中 10:1b:54:xx:xx:d3 为MAC地址
```

须知

虚拟网口MAC地址说明

Driver包安装后生成的虚拟网口没有固定的MAC地址，Driver包升级或安装后生成随机数作为MAC地址，虚拟网口MAC地址固定格式：10:1b:54:xx:xx:d3，其中第四和第五位为随机数，10:1b:54前三位MAC地址属于华为公司。

Driver包升级或安装后，虚拟网口MAC地址会变化，需要重新进行MAC地址与IP地址的绑定。

d. 修改“/etc/sysconfig/network-scripts/ifcfg-endvnic”内容（请确保该路径下只有ifcfg-endvnic一个文件，备份文件或重命名文件例如ifcfg-endvnic1，都会影响添加网卡的结果）。

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
#BOOTPROTO=dhcp  #注释掉
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=endvnic
UUID=6c2db480-1993-40ba-bc21-3541ae728719
DEVICE=endvnic
IPADDR=192.168.1.100  #配置为Host侧虚拟网卡地址
ONBOOT=yes
HWADDR="10:1b:54:xx:xx:d3"  #配置虚拟网卡MAC地址
其中，HWADDR为MAC地址
```

e. 使用**ifup endvnic**命令添加网卡。

f. 执行如下命令，查看网络配置是否成功。

ifconfig

若出现如下回显，表示网络配置成功。否则，请联系华为技术支持处理。

```
endvnic: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.100  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::xxxx:xxxx:xxxx:78d3  prefixlen 64  scopeid 0x20<link>
        ether 10:1b:54:xx:xx:d3  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 42  bytes 2940 (2.9 KB)
        TX errors 8  dropped 8 overruns 0  carrier 0  collisions 0
```

步骤3 登录Device。通过**ssh HwHiAiUser@*xxx*** 命令登录device（xxx为device侧IP地址，如：192.168.1.199）。

说明

- Device重启后，Device侧的配置文件会恢复默认配置（默认账号的超期时间恢复为90天），若用户需要自定义过期时间，可以修改配置文件“/etc/login.defs”中有效期的天数；或者建议用户在自定义的启动脚本中添加对应的chage命令，重新配置过期时间，详情请参见**5.3 设置用户有效期**。
- 首次登录Device侧需要修改密码，默认用户名和密码具体请参见**《Atlas硬件产品 用户清单》**。新密码设置规则：密码必须包含字母、数字、特殊字符（特殊字符不能用**@**、**%**、**空格**），长度不小于8个字符。

----结束

6 FAQ

6.1 升级系统内核版本时未卸载run包导致run包不可用
6.2 deb包安装、升级或卸载失败后的处理方式
6.3 root用户切换到普通用户需要输入密码权限
6.4 如何检查device是否正常运行
6.5 复位或下电导致Device无法正常启动
6.6 软件包格式混用安装或升级恢复方法
6.7 安装软件包时提示解压失败
6.8 驱动加载超时，CPU打印“soft lockup”报错
6.9 驱动卸载过程中出现进程占用报错
6.10 安装deb包时出现“binary opetator expected”报错
6.11 算力切分场景下npu-smi命令无法使用

6.1 升级系统内核版本时未卸载 run 包导致 run 包不可用

问题描述

如果在安装了run包的环境直接升级内核，再次重启时环境会出现启动异常，如**图6-1**所示。如果后续在该环境上再安装run包也无法使用。这是由于安装run包时驱动ko加载到内核，而升级内核版本会触发根文件系统打包，会将原来环境中ko打包到根文件系统，导致启动异常。

图 6-1 run 包不可用

```
[    3.690421] i8042: No controller found
[    5.850142] systemd[1]: Failed to start Load Kernel Modules.
[FAILED] Failed to start Load Kernel Modules.
See 'systemctl status systemd-modules-load.service' for details.
[  OK  ] Started Create Static Device Nodes in /dev.
         Starting Apply Kernel Variables...
[  OK  ] Started Apply Kernel Variables.
[  OK  ] Started Setup Virtual Console.
[  OK  ] Started Journal Service.
[  OK  ] Started dracut cmdline hook.
         Starting dracut pre-udev hook...
[  OK  ] Started dracut pre-udev hook.
         Starting udev Kernel Device Manager...
[  OK  ] Started udev Kernel Device Manager.
         Starting udev Coldplug all Devices...
         Mounting Configuration File System...
[  OK  ] Mounted Configuration File System.
[    6.861746] megaraid_sas 0000:18:00.0: IOC INIT command return status SUCCESS for SCSI host 0
[  OK  ] Started udev Coldplug all Devices.
         Starting Show Plymouth Boot Screen...
[  OK  ] Reached target System Initialization.
         Starting dracut initqueue hook...
[    7.059120] megaraid_sas 0000:18:00.0: return from megasas_get_target_prop 7453 return value 2
[    8.777736]  [drv_pcie] [devdrv_get_boot_mode_flag 390] get boot mode from bios err:0x0
[    8.777978]  [drv_pcie] [devdrv_load_file_copy 457] devdrv_device_driver, unable to open file: /home/bios/davinci_cloud.image
 (-5)
[    8.778341]  [drv_pcie] [devdrv_load_file_trans 632] devdrv_device_driver, /home/bios/davinci_cloud.image file copy err
[    8.778909]  [drv_pcie] [devdrv_load_file 1045] devdrv_device_driver, trans file to agent bios failed
[    8.811736]  [drv_pcie] [devdrv_get_boot_mode_flag 390] get boot mode from bios err:0x0
[    8.811976]  [drv_pcie] [devdrv_load_file_copy 457] devdrv_device_driver, unable to open file: /home/bios/davinci_cloud.image
 (-5)
[    8.812341]  [drv_pcie] [devdrv_load_file_trans 632] devdrv_device_driver, /home/bios/davinci_cloud.image file copy err
[    8.812901]  [drv_pcie] [devdrv_load_file 1045] devdrv_device_driver, trans file to agent bios failed
```

解决方案

步骤1 手动卸载ko。

1. 查询当前系统存在ko列表。

   lsmod|grep drv

2. 将查询到的ko一次全部卸载，多个ko名字之间需要加空格。两个ko的示例如下（ko1，ko2请用实际ko名称替换）。

   rmmod *ko1 ko2*

步骤2 打包根文件系统。

dracut --force

步骤3 重启环境。

reboot

----结束

6.2 deb 包安装、升级或卸载失败后的处理方式

问题描述

deb包安装、升级或卸载失败后，残留的缓存信息会导致再次安装或升级异常。

解决方案

deb包安装、升级或卸载失败后，需要手动删除缓存信息等相关操作，步骤如下：

步骤1 删除失败包的配置文件列表，执行命令如下（以ascend310-driver包为例）。

```
rm /var/lib/dpkg/info/ascend310-driver*
```

步骤2 执行强制删除命令。

```
dpkg --remove --force-remove-reinstreq ascend310-driver
```

步骤3 删除原安装失败软件目录下的残留文件（示例目录为默认安装路径，以实际安装目录为准）。

```
rm -rf /usr/local/Ascend/driver
```

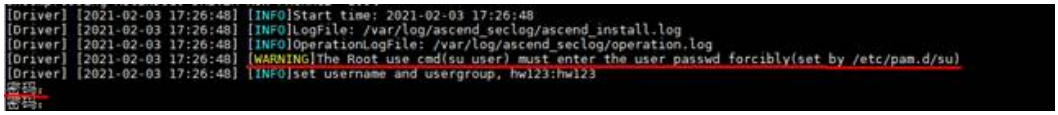
----结束

6.3 root 用户切换到普通用户需要输入密码权限

问题描述

在软件包安装过程中，可能会出现“The Root use cmd(su user) must enter the user passwd forcibly(set by /etc/pam.d/su)”报错。如图6-2所示：

图 6-2 报错信息



可能原因

root用户执行**su**命令进行用户切换时需要输入用户密码。

解决方案

- 直接输入运行用户密码继续完成后续安装。
- 修改系统配置文件“/etc/pam.d/su”，设置root用户执行**su**命令进行用户切换时不需要输入用户密码。操作如下：
  a. 执行如下命令，打开配置文件“/etc/pam.d/su”。
     vi /etc/pam.d/su
  b. 将“auth sufficient pam_rootok.so”前面"**#**"号删除，保存退出。

6.4 如何检查 device 是否正常运行

步骤1 登录运行环境，并在root用户下执行以下操作。

步骤2 查询run包安装路径。

cat /etc/ascend_install.info

系统显示示例如下：

```
Driver_Install_Path_Param=/usr/local/Ascend
```

步骤3 进入Driver安装路径，通过upgrade-tool工具查看下device侧运行文件系统版本。

cd */usr/local/Ascend/*driver/tools/

./upgrade-tool --device_index -1 --system_version

如果能正常查询，则说明device侧已经正常启动，正常查询返回信息类似如下。

```
[root@localhost tools]# /usr/local/Ascend/driver/tools/upgrade-tool --device_index -1 --system_version
{
Get system version[illegible] succeed, deviceId(0)
        {"device_id":0, "version"[illegible]
Get system version[illegible]) succeed, deviceId(1)
        {"device_id":1, "version":[illegible]
}
```

----结束

6.5 复位或下电导致 Device 无法正常启动

问题描述 1

在软件包安装或升级过程中，对Device进行了复位或下电操作，出现固件损坏，导致Device无法正常启动。

解决方案 1

1. 对Device连续进行3次复位（非下电，使用**reboot**命令）操作，3次复位之间间隔建议10秒以上。
2. 如Device仍无法启动，则对Device进行下电重启。

问题描述 2

在软件包安装或升级过程中，对Host进行了复位或下电操作，导致升级失败或Device无法正常启动。

解决方案 2

1. 若Device能正常启动，重新升级软件包。
2. 若Device无法正常启动，解决方法请参见**解决方案1**。

6.6 软件包格式混用安装或升级恢复方法

问题描述

当前环境已安装run包，覆盖安装或升级时使用rpm包或者deb包。或当前环境已安装rpm包或者deb包，覆盖安装或升级时使用run包。

解决方案

环境已安装软件包格式	覆盖安装或升级的软件包格式	处理方法
.rpm	.run	1. 卸载rpm包，具体请参见**3.7.3 卸载驱动（适用于.rpm格式）**和**3.8.2 卸载固件（适用于.rpm格式）**。 2. 重新安装软件包，具体请参见**3.5 安装驱动**和**3.6 安装固件**。
.run	.rpm	

环境已安装软件包格式	覆盖安装或升级的软件包格式	处理方法
.deb	.run	1. 卸载deb包，具体请参见3.7.4 卸载驱动（适用于.deb格式）和3.8.3 卸载固件（适用于.deb格式）。若卸载失败，请参见6.2 deb包安装、升级或卸载失败后的处理方式处理。 2. 重新安装软件包，具体请参见3.5 安装驱动和3.6 安装固件。
.run	.deb	

6.7 安装软件包时提示解压失败

问题描述

软件包安装时出现“Extraction failed”或“Decompression failed”报错。如图6-3所示：

图 6-3 报错打印

```
Verifying archive integrity...  100%   SHA256 checksums are OK. All good.
Uncompressing npu-driver-run-package ... Extraction failed.
 ... Decompression failed.
  100%  Signal caught, cleaning up
```

可能原因

系统缺少tar工具包。

执行**tar**命令，若出现如下回显，则说明缺少tar工具包。

```
-bash: tar: command not found
```

解决方案

请根据相应操作系统自行安装tar工具。

6.8 驱动加载超时，CPU 打印“soft lockup”报错

问题描述

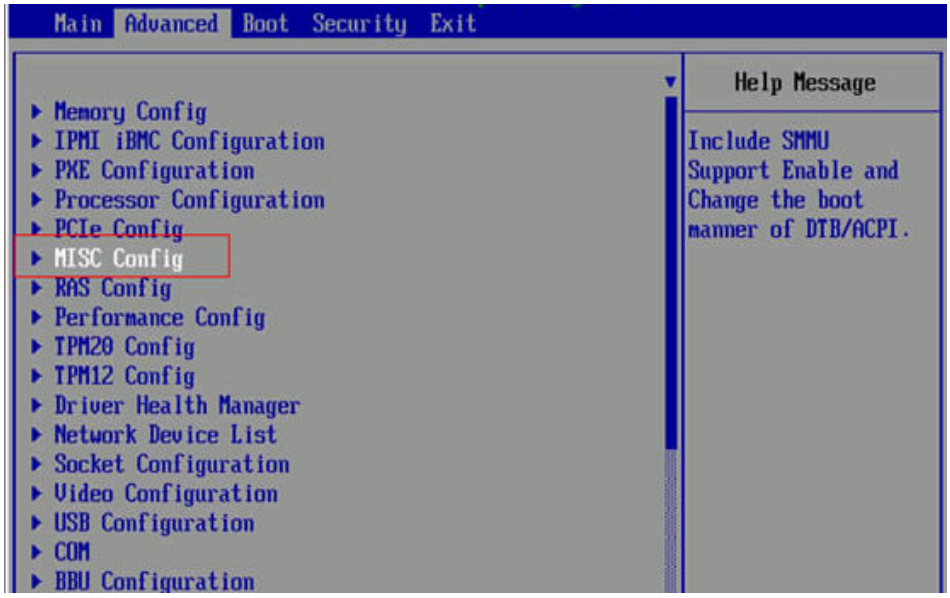
加载驱动出现概率性system-udevd服务加载超时和“soft lockup”报错。

可能原因

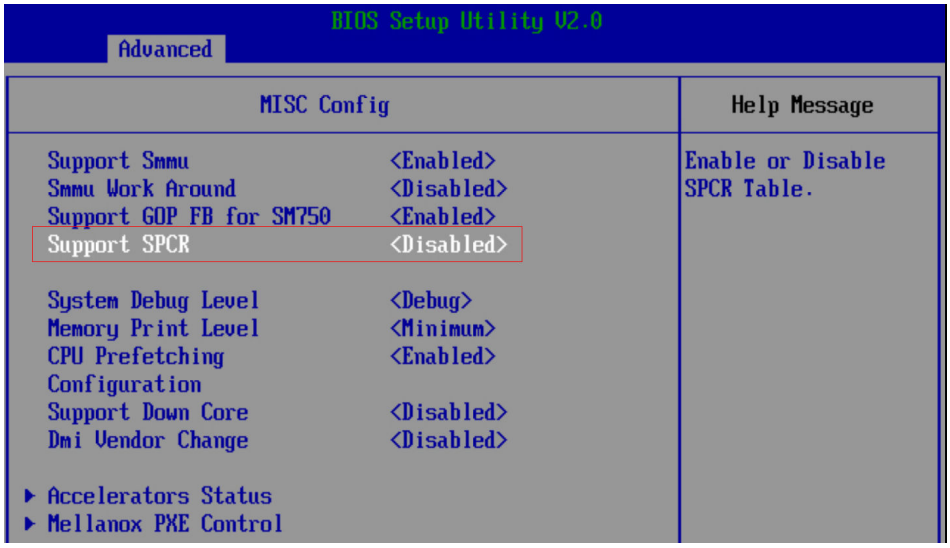
在开启BMC串口日志重定向后，当前系统默认将内核printk日志级别设置为DEBUG，因此驱动加载过程中的日志会打印到串口，导致CPU调度变慢，出现概率性system-udevd服务加载超时和“soft lockup”报错。

解决方案

- Atlas 800 推理服务器（型号：3000）环境关闭BMC串口日志重定向具体操作如下：
  a. 系统启动时按Delete键进入BIOS设置界面。
  b. 选择“Advanced > MISC Config”。



  c. 将“Support SPCR”设置为“Disabled”。



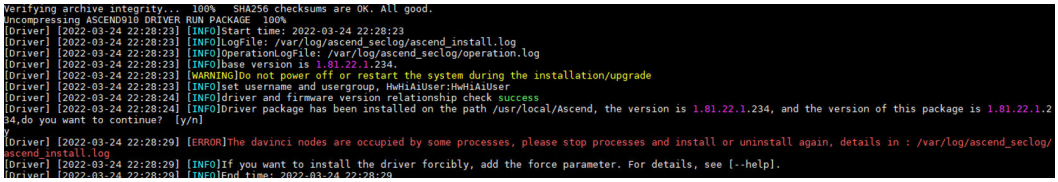
  d. 按“F10”。
  e. 弹出“Save configuration changes and exit? ”对话框。
  f. 选择“Yes”，按Enter键保存BIOS参数设置，此时服务器会自动重启。

6.9 驱动卸载过程中出现进程占用报错

问题描述

驱动卸载过程中出现“The davinci nodes are occupied by some processes, please stop processes and install or uninstall again, details in : /var/log/ascend_seclog/ascend_install.log”报错，如图6-4所示。

图 6-4 报错信息



可能原因

用户业务进程未关闭，卸载过程被拦截。

解决方案

说明

若卸载rpm、deb包时，提示“The davinci nodes are occupied by some processes”，请停止业务进程后重新卸载驱动。

步骤1 执行如下命令查看“/var/log/ascend_seclog/ascend_install.log”日志，确定未关闭的业务进程及其占用的节点信息。

cat /var/log/ascend_seclog/ascend_install.log

回显信息如下：

```
[Driver] [2022-03-25 01:10:49] [WARNING]The setting hot reset flag does not take effect.
[Driver] [2022-03-25 01:10:49] [WARNING]/dev/davinci_manager has user process:  47355
 root      47355  23097  0 01:10 pts/1    00:00:00 ./hlt_hdc_test 65 0 6 100
[Driver] [2022-03-25 01:10:51] [ERROR]The davinci nodes are occupied by some processes, please stop
processes and install or uninstall again.
```

步骤2 执行以下操作继续卸载驱动。

- 停止业务进程后，重新卸载驱动。具体请参见3.7 卸载驱动。
- 不停止业务使用**--force**参数进行强制卸载驱动，具体操作如下。但正在运行的业务可能会受到影响，且执行后必须重启host驱动才可以生效。
  - 使用软件包卸载。

    ./*{product name}*-npu-driver_*x.x.x*_linux-*{arch}*.run **--uninstall --force**
  - 在任意路径执行如下命令卸载。

    *<install-path>***/driver/script/uninstall.sh --force**

    *<install-path>*表示软件包安装路径，可以执行**cat /etc/ascend_install.info**命令查询安装路径；请根据实际情况替换。

----结束

6.10 安装 deb 包时出现“binary opetator expected”报错

问题描述

安装deb包失败，出现“binary operator expected”报错，如图6-5所示。

图 6-5 安装失败报错

```
install firmware run fail, now to uninstall run pkg
Usage: ./[redacted]-npu-firmware_1.81.22.5.220.run [options]
Options:
  --help | -h                   Print this message
  --info                        Print embedded info : title, default target directory, embedded script ...
  --list                        Print the list of files in the archive
  --check                       Checks integrity and version dependency of the archive
  --quiet                       Quiet install mode, skip human-computer interactions
  --nox11                       Do not spawn an xterm
  --noexec                      Do not run embedded script
  --extract=<path>              Extract directly to a target directory (absolute or relative)
                                Usually used with --noexec to just extract files without running
  --tar arg1 [arg2 ...]         Access the contents of the archive through the tar command
  --full                        Install full mode
  --run                         Install run mode
  --upgrade                     Upgrade product immediately
  --reset                       Resetting user config
  --uninstall                   Uninstall product
  --version                     Query the package version
/var/lib/dpkg/info/[redacted]-firmware.postinst: line 3: [: ./[redacted]-npu-firmware_1.81.22.5.220.run: binary operator expected
dpkg: error processing package ascend310-firmware (--install):
 installed [redacted]-firmware package post-installation script subprocess returned error exit status 1
Errors were encountered while processing:
 [redacted]-firmware
```

可能原因

安装deb包时，会将deb包内的run包解压到当前运行环境的根目录“./”下，若用户根目录下已存在相同版本的run包，则会使安装脚本检索到2个安装包，出现“binary operator expected”报错。

解决方案

移除根目录“./”下的run包，重新安装软件包。

6.11 算力切分场景下 npu-smi 命令无法使用

问题描述

算力切分场景下，容器内使用切分后的设备运行推理或训练业务，执行**Ctrl+Z**挂起业务进程后，概率性出现**npu-smi**相关命令无法使用，新业务无法调度，返回失败。报错如下：

```
I20220718 06:39:54.367962   676 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder4) fps (21).
I20220718 06:39:54.371521   679 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder3) fps (21).
I20220718 06:39:54.375070   682 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder2) fps (21).
I20220718 06:39:54.378576   685 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder1) fps (21).
I20220718 06:39:54.382223   688 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder0) fps (22).
I20220718 06:39:55.346460   663 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder8) fps (31).
I20220718 06:39:55.356722   667 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder7) fps (30).
I20220718 06:39:55.361157   670 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder6) fps (30).
I20220718 06:39:55.364678   673 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder5) fps (30).
I20220718 06:39:55.368136   676 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder4) fps (30).
I20220718 06:39:55.371685   679 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder3) fps (30).
I20220718 06:39:55.375218   682 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder2) fps (30).
I20220718 06:39:55.378721   685 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder1) fps (30).
I20220718 06:39:55.382375   688 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder0) fps (30).
I20220718 06:39:56.346639   663 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder8) fps (29).
I20220718 06:39:56.356884   667 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder7) fps (30).
I20220718 06:39:56.361311   670 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder6) fps (30).
I20220718 06:39:56.364831   673 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder5) fps (30).
I20220718 06:39:56.368304   676 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder4) fps (30).
I20220718 06:39:56.371845   679 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder3) fps (30).
I20220718 06:39:56.375358   682 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder2) fps (30).
I20220718 06:39:56.378865   685 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder1) fps (30).
I20220718 06:39:56.382517   688 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder0) fps (29).
I20220718 06:39:57.346810   663 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder8) fps (31).
I20220718 06:39:57.357043   667 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder7) fps (30).
I20220718 06:39:57.361467   670 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder6) fps (30).
I20220718 06:39:57.364987   673 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder5) fps (30).
I20220718 06:39:57.368468   676 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder4) fps (30).
I20220718 06:39:57.372004   679 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder3) fps (30).
I20220718 06:39:57.375500   682 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder2) fps (30).
I20220718 06:39:57.379005   685 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder1) fps (30).
I20220718 06:39:57.382673   688 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder0) fps (30).
I20220718 06:39:58.346984   663 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder8) fps (29).
I20220718 06:39:58.357203   667 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder7) fps (30).
I20220718 06:39:58.361618   670 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder6) fps (30).
I20220718 06:39:58.365147   673 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder5) fps (30).
I20220718 06:39:58.368640   676 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder4) fps (30).
I20220718 06:39:58.372161   679 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder3) fps (30).
I20220718 06:39:58.375646   682 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder2) fps (30).
I20220718 06:39:58.379151   685 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder1) fps (30).
I20220718 06:39:58.382825   688 MxpiVideoEncoder.cpp:351] Plugin(mxpi_videoencoder0) fps (31).
^Z[4]   Killed                  ./mxVisionMediaCodec pipeline/test0.pipeline

[5]+  Stopped                 ./mxVisionMediaCodec pipeline/test0.pipeline
root@ef97c11c44af:/home/ld/mxVision/VideoCodec_1P_in_docker# npu-smi info
dcmi module initialize failed. ret is -8010
root@ef97c11c44af:/home/ld/mxVision/VideoCodec_1P_in_docker#
```

可能原因

业务进程调用内核态相关操作，挂起后CPU调度异常导致内核处理无法完成。

解决方案

- 使用**bg**命令重新调度挂起的进程。
- 使用**kill**命令杀掉挂起的进程即可恢复正常。

7 参考

7.1 参数说明/常用命令
7.2 驱动安装过程中所需的Linux工具
7.3 相关工具

7.1 参数说明/常用命令

参数说明

软件包支持根据命令行完成一键式安装，各个命令之间可以配合使用，用户根据安装需要选择对应参数完成安装，所有参数都是可选参数。

安装命令格式： **./***xx*.run [options]

详细参数请参见**表7-1**。

须知

软件包中的**xx**请根据实际情况进行替换。

如果通过**./***xx*.run **--help**命令查询出的参数未解释在如下表格，则说明该参数预留或适用于其他芯片版本，用户无需关注。

表 7-1 安装包支持的参数说明

参数	说明
--run	运行态类型：只安装运行场景需安装的文件。
--devel	开发态类型：包含用户开发时需要用到的头文件。 Firmware子包不支持该参数。
--full	完整安装类型：安装所有文件。

参数	说明
--docker	容器场景下安装，仅用于driver子包，其他子包默认按full模式。
--install-username=<username>	首次安装：可以指定运行用户名，若不指定，则默认是HwHiAiUser。 覆盖安装：沿用上次运行用户名。 **说明** ● 该参数需要配合“--install-usergroup=<usergroup>”一起使用，并且username需要与创建的用户（3.4 创建运行用户）保持一致。 ● 如果指定root用户为运行用户，则需要与--install-for-all参数配合使用，如下所示，该场景下权限控制可能存在安全风险。 `--install-username=root --install-usergroup=root --install-for-all` ● Firmware安装不支持指定运行用户名和用户组，共用Driver的运行用户名和用户组。
--install-usergroup=<usergroup>	首次安装：可以指定运行用户组，若不指定，则默认是HwHiAiUser。 覆盖安装：沿用上次运行用户组。 **说明** 该参数需要配合“--install-username=<username>”一起使用，并且usergroup需要与创建的用户组（3.4 创建运行用户）保持一致。
--install-path=<path>	指定安装目录，如不指定： ● 若使用root用户安装，默认安装路径为：/usr/local/Ascend。 ● 若使用非root用户安装，则默认安装路径为：$HOME/Ascend。 若通过该参数指定了安装路径，运行用户需要对指定的安装路径有可读写权限。安装路径支持大小写字母（a-z，A-Z）、数字（0-9）、下划线（_）、中划线（-）、句点（.（非相对路径））、单个/（文件名或目录不支持/）。 **说明** Firmware安装不支持指定路径，共用Driver的安装路径。

参数	说明

--install-for-all	安装时，允许其他用户具有安装群组的权限。
	当安装携带该参数时，软件包中创建的目录及文件，其他用户权限=安装群组权限。
	首次安装过程中如果使用了--install for all参数，覆盖安装中也需要保留--install for all参数。
	使用示例：**./***xx*.run **--full --install-for-all**
	说明
	● 使用该参数将会存在安全风险：其他所有用户都有权限访问安装目录，请谨慎使用。
	● Firmware不支持该参数。
	● 安装时如果使用该参数：
	  ● 如果安装路径不存在，则安装过程中会自动创建默认路径，例如$HOME/Ascend，该路径权限为755，其他用户可以正常使用。
	  ● 如果安装路径已存在，则请确保该目录权限至少为755，否则将会终止安装。

--reset	恢复初始化配置（该参数只适用于Firmware子包）： ● --full --reset参数通过安装方式恢复初始化配置，需立即重启生效；若安装失败，则恢复初始化配置失败。 ● --upgrade --reset参数通过升级方式恢复初始化配置，需立即重启生效；若升级失败，则恢复初始化配置失败。 恢复初始化配置后，原来用户对device侧的个人配置数据会失效，需要用户重新配置。 **说明** ● --reset参数只支持--full --reset与--upgrade --reset两种模式的使用方法，不支持与其它参数配合使用，也不支持单独使用。 ● 带有--reset参数命令执行完成后，重启生效前，禁止通过ssh、scp等方式访问Device侧的动作，否则会造成恢复初始化配置失败。
--uninstall	卸载。
--nox11	安装过程中不弹出图形终端窗口。 使用示例：**./***xx*.run **--full --nox11**
--noexec	不执行安装脚本。配套--extract=*path*使用。格式为：--noexec --extract=*path*
--extract=*<path>*	解压缩安装包中文件到指定目录。使用该参数时，用户无需手动建立指定的目录，解压过程中会自动创建，例如： --noexec --extract=./package，则命令执行后会自动将解压后的内容放在package目录。

参数	说明
--upgrade	升级。 ● 首次安装场景，系统默认安装到“/usr/local/Ascend”目录。 ● 非首次安装场景，仅支持从软件包所在路径升级。 **说明** 该参数不能与--install-username、--install-usergroup、--install-for-all其中一个参数配合使用。
--help或者-h	查看帮助。
--check	校验软件包的完整性、检查与已经安装的其他软件包之间的配套关系是否正确。
--version	查询版本号。
--tar arg1 [arg2 …]	对安装包执行tar命令，使用tar后面的参数作为命令的参数。例如执行**--tar xvf**命令，解压软件包的内容到当前目录。
--list	导出安装包文件列表。
--info	显示数据包详细信息。
--quiet	静默安装，跳过交互式信息。 使用该参数进行安装时，若安装目录权限大于755，则会提示安全风险，用户自行选择终止安装或选择其他权限为755的安装目录重新安装。 使用示例：**./***xx*.run **--full --quiet** **说明** Driver和Firmware软件包在使用该参数时，不能与--uninstall配合使用。
--repack [package_name]	构建新驱动包。 package_name为新构建的驱动包名称，格式为”*xx*.run”。 **说明** 若package_name参数为空，则会在当前路径下生成一个命名为“原驱动包名称-custom.run”的文件。
--repack-path=<path> [package_name]	在指定路径下构建新驱动包。 ● path为原run包的解压目录。 ● package_name为新构建的驱动包名称，格式为”*xx*.run”。 **说明** 若package_name参数为空，则会在当前路径下生成一个命名为“原驱动包名称-custom.run”的文件。

参数	说明
--force	控制仅安装不直接生效。使用该参数安装软件包时需要重启生效。（该参数仅适用于driver子包） 该参数需要与--run、--full、--upgrade、--uninstall等其中一个参数配合使用，例如./*xx*.run **--full --force** **说明** 若首次安装驱动使用该参数时，必须重启后才能卸载或覆盖安装，否则可能会导致卸载或覆盖安装失败。

举例说明：

- full模式下安装
  - 不指定安装目录：**./***xx*.run **--full**
  - 指定安装目录：**./***xx*.run **--full --install-path=***安装路径*
- docker模式下安装（仅支持driver子包）
  - 不指定安装目录：**./***xx*.run **--docker**
  - 指定安装目录./*xx*.run **--docker --install-path=***安装路径*
- run模式下安装
  - 不指定安装目录：**./***xx*.run **--run**
  - 指定安装目录：**./***xx*.run **--run --install-path=***安装路径*
- 开发态模式下安装
  - 不指定安装目录：**./***xx*.run **--devel**
  - 指定安装目录：**./***xx*.run **--devel --install-path=***安装路径*

7.2 驱动安装过程中所需的 Linux 工具

驱动安装过程中所需的Linux工具见表7-2。

表 7-2 驱动安装过程中所需的 Linux 工具

Linux工具	备注
which	shows the full path of (shell) commands.
update-rc.d,chkconfig	updates and queries runlevel information for system services
test	check file types and compare values
groups	print the groups a user is in
awk	pattern scanning and processing language
chattr	change file attributes on a Linux file system
lsattr	list file attributes on a Linux second extended file system

Linux工具	备注
ldconfig	configure dynamic linker run-time bindings
wc	print newline, word, and byte counts for each file
lspci	list all PCI devices
ifconfig	configure a network interface
chown	change file owner and group
chmod	change file mode bits
find	search for files in a directory hierarchy
sed	stream editor for filtering and transforming text
ln	make links between files
modprobe	Add and remove modules from the Linux Kernel
cksum	checksum and count the bytes in a file
tail	output the last part of files
head	output the first part of files
udevadm	udev management tool
modinfo	show information about a Linux Kernel module
useradd	create a new user or update default new user information
expr	evaluate expressions
basename	strip directory and suffix from filenames
tty	print the file name of the terminal connected to standard input
dirname	strip last component from file name
rev	reverse lines characterwise
id	print real and effective user and group IDs
touch	change file timestamps
grep	print lines matching a pattern
ps	report a snapshot of the current processes
useradd	create a new user or update default new user information

7.3 相关工具

须知

upgrade-tool工具不支持容器场景，部署容器时，禁止将upgrade-tool工具映射到容器内。

步骤1 以安装软件包时指定的运行用户（默认HwHiAiUser）登录Host侧的服务器。

步骤2 切换到软件包所在安装路径，根据以下表格中的命令调用工具，如下以Driver采用默认安装路径为例进行说明。

表 7-3 相关工具

名称	路径	作用	执行命令	参数
upgrade-tool	/usr/local/Ascend/driver/tools	查看固件版本、升级单个或多个固件等，只允许root用户执行。	请切换到/usr/local/Ascend/driver/tools路径执行如下命令： ● 列举所有Device：**./upgrade-tool --mini_devices** ● 获取指定设备的版本：**./upgrade-tool --device_index <dev_id> --system_version** ● 获取指定设备的组件信息：**./upgrade-tool --device_index <dev_id> --components** ● 查询某个设备中某一个组件对应的版本：**./upgrade-tool --device_index <dev_id>--component <type> --version** ● 查询设备状态：	● --mini_devices：所有设备列表。 ● --device_index：设备编号。取值可以是【0~63】和-1，【0~63】表示对应编号的设备。-1表示所有设备。 ● --system_version：系统版本。 ● --components：列举所有有效的组件。 ● --component：指定具体的组件。升级单个组件时，需要指定组件名称。升级所有组件时，需要输入-1。需要升级所有组件并清除用户配置数据时，请输入-9。 ● --version：组件的版本。 ● --status：设备状态。支持的设备

名称	路径	作用	执行命令	参数
			./upgrade-tool --device_index <dev_id> --status ● 查询设备是否是物理机：**./upgrade-tool --device_index <dev_id> --phymachflag** **说明** 仅支持通过物理机升级固件包。 ● 启动固件主备同步：**./upgrade-tool --sync** ● 查看所有分区的固件版本：**./upgrade-tool --device_index <dev_id> --component <type> --all --version** ● 配置算力功率等级：**./upgrade-tool --device_index <dev_id> --component nve --level**	状态具体包含： – idle：空闲。 – upgrading：正在升级。 – not support：不支持。 – failed：失败。 – waitting_restart：等待重启。 – waitting_sync：等待固件同步。 – synchronizing：正在同步。 – wrong status：错误状态。 ● --phymachflag：查询设备是否是物理机。如果不是物理机，不允许升级设备的固件包。 ● --async：支持异步升级，即device侧收到host侧发送的升级请求后，返回

名称	路径	作用	执行命令	参数
			<low/middle/high/full> ● 指定组件的主备区域进行升级：**./upgrade-tool --device_index <dev_id>--component <type> --media --mmc <main/backup> --path <firmware_path>** ● 查询主备分区的使能状态：**./upgrade-tool --fs_backup_status** ● 获取当前设备的启动状态：**./upgrade-tool --device_index <dev_id> --boot_status** ● 设置当前设备热复位标志：**./upgrade-tool --device_index <dev_id> --hot_reset_flag**	请求成功响应到host侧（具体固件是否升级成功，需要通过“--status”参数去查询）。 ● --help：查看帮助信息。 ● --sync：启动固件主备同步，仅支持RC场景。 ● --all --version：查看所有分区（Flash分区、主备分区）的固件版本，仅支持RC场景。 ● --level：配置算力功率等级，仅支持RC场景。 ● --media --mmc：指定组件的主备区域进行升级，仅支持RC场景。 ● --fs_backup_status：主备分区的使能状态，仅支持RC场景。 ● --boot_status：当前设

名称	路径	作用	执行命令	参数
				备的启动状态。 – 0：未初始化。 – 1：BIOS启动中。 – 2：OS启动中。 – 3：启动完成。 – 16：DSMI服务进程启动完成。 ● --hot_reset_flag：当前设备的热复位标志，不支持RC场景。

名称	路径	作用	执行命令	参数
			请切换到/usr/local/Ascend/firmware/tools路径执行如下命令： ● 升级指定设备的固件包：**/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --component <type> --path <firmware_path>** ● 异步升级指定设备的固件包：**/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --component <type> --async --path <firmware_path>** **--async** 必须紧跟在**component <type>**后面。	--path： ● 固件包相对路径：若升级全部组件，则为**--path ./conf/upgrade.cfg**；“upgrade.cfg”文件中包含各个组件的相对路径。若升级单个组件，例如nve.bin，则为**--path ../image/nve.bin**。

----结束