Atlas 300T 训练卡
1.0.7 至 1.0.10

NPU 驱动和固件安装升级指南（型号 9000）

文档版本 07

发布日期 2022-03-07

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

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驱动与固件包的安装和升级流程以及具体的操作指导，同时提供了常见的问题解答及故障处理方法。

读者对象

本文档主要适用于升级的操作人员。操作人员必须具备以下经验和技能：

- 熟悉当前网络的组网和相关网元的版本信息。
- 有该设备维护经验，熟悉设备的操作维护方式。

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。

版权所有 © 华为技术有限公司

修改记录

文档版本	发布日期	修改说明
07	2022-03-07	第七次正式发布。 优化默认数据。
06	2021-07-31	第六次正式发布。
05	2021-06-03	修改了**1.4 卸载驱动和固件**。
04	2021-05-08	修改了**1.2 安装前准备**、**2.2 升级前准备**和**2.3.3 升级MCU**。 增加了**4.6 npu-smi查询MCU健康状态出现1C1F mcu flash app error告警**、**4.7 npu-smi查询MCU健康状态出现1DAF vrd PSU firmware needs to be upgraded告警**和**1.2.6（可选）重构驱动包**。
03	2021-02-02	修改了**1.2 安装前准备**和**2.2 升级前准备**。 增加了**4.5 驱动20.2.0及以上版本回退低版本失败问题**。
02	2020-11-28	修改了**1.2.5 创建运行用户**。 增加了**A.4 添加内核补丁**。
01	2020-09-30	第一次正式发布。

 版权所有 © 华为技术有限公司

目 录

前言 ii
1 安装与维护 1
1.1 安装前必读 1
1.2 安装前准备 1
1.2.1 检查环境 1
1.2.2 获取软件包 2
1.2.3 检验软件包完整性 4
1.2.4 系统要求和环境检查 4
1.2.5 创建运行用户 6
1.2.6 （可选）重构驱动包 8
1.3 安装驱动和固件 10
1.4 卸载驱动和固件 12
2 升级 14
2.1 升级前必读 14
2.2 升级前准备 15
2.3 升级组件 18
2.3.1 升级 Ascend 910 芯片固件 18
2.3.2 升级 Ascend 910 芯片驱动 19
2.3.3 升级 MCU 20
2.3.3.1 通过 npu-smi 升级 21
2.3.3.2 通过 iBMC 升级 21
2.3.3.2.1 通过 Web 界面升级 22
2.3.3.2.2 通过命令行升级 23
3 版本回退 24
4 FAQ 25
4.1 驱动源码编译 25
4.2 软件包不可用 26
4.3 如何检查 device 是否正常运行 27
4.4 BC_Linux 镜像源配置 28
4.5 驱动 20.2.0 及以上版本回退低版本失败问题 29
4.6 npu-smi 查询 MCU 健康状态出现 1C1F mcu flash app error 告警 29
4.7 npu-smi 查询 MCU 健康状态出现 1DAF vrd PSU firmware needs to be upgraded 告警 30

 版权所有 © 华为技术有限公司

A 附录 31
A.1 参数说明 31
A.2 相关脚本 33
A.3 相关工具 34
A.4 添加内核补丁 38

 版权所有 © 华为技术有限公司

1 安装与维护

1.1 安装前必读
1.2 安装前准备
1.3 安装驱动和固件
1.4 卸载驱动和固件

1.1 安装前必读

- NPU芯片是Ascend 910 Pro B的训练卡，首次安装NPU驱动需使用20.2.0.spc300及以上版本，NPU固件使用配套驱动同步发布的版本，且均不支持回退至20.2.0.spc200及更低版本。
- 登录HOST查询NPU芯片名称命令：***npu-smi info***

  通过iBMC查询NPU芯片名称方法：登录iBMC Web管理网站，点击“系统管理 > 系统信息 > 其他 > PCIe卡”，第一列描述显示有芯片名称信息。
- MCU固件不支持安装和卸载，预置固件升级新的MCU固件版本请参见**2.3.3 升级MCU**。

1.2 安装前准备

1.2.1 检查环境

安装前，需要检查的环境信息如下。

- 按硬件形态检查系统版本信息：版本信息如**表1-3**所示，检查命令如**表1-1**所示。

表 1-1 检查命令

Host操作系统版本/Host操作系统架构	Host操作系统内核版本
uname -m && cat /etc/*release	uname -r

版权所有 © 华为技术有限公司

- 检查Atlas 300T 训练卡 (型号 9000)是否正常在位。

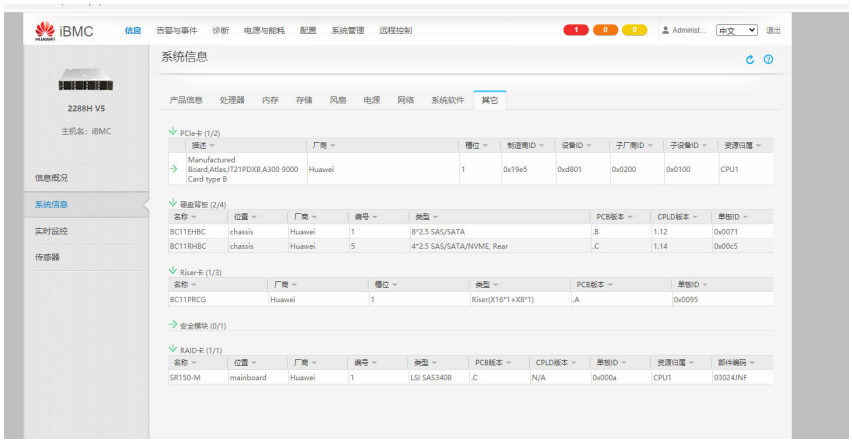
  通过以下两种方式可以检查Atlas 300T 训练卡 (型号 9000)是否正常在位：

  - 执行**lspci | grep d801**命令，显示如下：

    ```
    3b:00.0 Processing accelerators: Device 19e5:d801 (rev 20)
    86:00.0 Processing accelerators: Device 19e5:d801 (rev 20)
    ```

  - 如果Atlas 300T 训练卡 (型号 9000)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 > 系统信息 > 其它”，如图1-1所示，Atlas 300T 训练卡 (型号 9000)的PCIe卡信息在“PCIe卡”列表中，表示Atlas 300T 训练卡 (型号 9000)正常在位。

    图 1-1 系统信息

1.2.2 获取软件包

安装前，请根据运行环境的操作系统获取对应的软件包。具体如表1-2所示：

表 1-2 软件包名称

组件	操作系统	软件包名	执行用户
固件包	**Ubuntu 18.04/ CentOS 7.6/ Debian 9.9/ CentOS 8.2/ kylinosv10SP1/ EulerOS 2.8** **说明** **kylinosv10SP1**在21.0.rc1及以上版本才支持。	A300t-9000-npu-firmware_<*version*>.run	root
驱动包	**Debian 9.9+x86_64**	A300t-9000-npu-driver_*x.x.x*_debian9.9-x86_64.run	root

 版权所有 © 华为技术有限公司

组件	操作系统	软件包名	执行用户
驱动包	**CentOS 7.6+x86_64** **说明** 标卡环境，CentOS x86低版本内核（低于4.5）无法使用时，需要将CentOS升级至8.0及以上或添加内核补丁。添加内核补丁的方法请参见**A.4 添加内核补丁**。	A300t-9000-npu-driver_*x.x.x*_centos7.6-x86_64.run	root
驱动包	**CentOS 7.6+aarch64**	A300t-9000-npu-driver_*x.x.x*_centos7.6-aarch64.run	root
驱动包	**Ubuntu 18.04+x86_64**	A300t-9000-npu-driver_*x.x.x*_ubuntu18.04-x86_64.run	root
驱动包	**Ubuntu 18.04+aarch64**	A300t-9000-npu-driver_*x.x.x*_ubuntu18.04-aarch64.run	root
驱动包	**CentOS 8.2+x86_64**	A300t-9000-npu-driver_*x.x.x*_linux-x86_64.run	root
驱动包	**CentOS 8.2+aarch64**	A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run	root
驱动包	**EulerOS 2.8+aarch64**	A300t-9000-npu-driver_*x.x.x*_euleros2.8-aarch64.run	root
驱动包	**kylinosv10SP1+aarch64** **说明** **kylinosv10SP1**在21.0.rc1及以上版本才支持。	A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run	root

说明

- 表中的{ *x.x.x* }表示具体版本号。
- A300t-9000-npu-driver_*x.x.x*_linux-x86_64.run适配x86所有操作系统；A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run适配arm所有操作系统。

版权所有 © 华为技术有限公司

操作步骤

步骤1 登录**A300T-9000**。

步骤2 打开目标版本“A300T-9000 *X.X.X*”。

固件、驱动与CANN的配套关系，请查看CANN软件版本的《**CANN 版本配套表**》。

步骤3 单击软件包（如A300t-9000-npu-firmware_<*version*>.run）后面的和，可获取软件包和数字签名文件。

----结束

1.2.3 检验软件包完整性

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**

1.2.4 系统要求和环境检查

软件包默认使用的操作系统和内核，如**表1-3**和**表1-4**所示。

表 1-3 二进制驱动包默认操作系统版本信息

硬件形态	Host操作系统版本	软件包默认的Host操作系统内核版本	GCC编译器版本
x86_64+Atlas 300T 9000	Debian 9.9	4.9.0-9-amd64	6.3.0
x86_64+Atlas 300T 9000	CentOS 7.6	3.10.0-957.el7	-
x86_64+Atlas 300T 9000	Ubuntu 18.04	4.15.0-45-generic **说明** 若内核版本和操作系统版本匹配失败，需先安装dkms，安装方法请参见**4.1 驱动源码编译**。	7.5.0
aarch_64+Atlas 300T 9000	CentOS 7.6	4.14.0	-

版权所有 © 华为技术有限公司

硬件形态	Host操作系统版本	软件包默认的Host操作系统内核版本	GCC编译器版本
aarch_64+Atlas 300T 9000	EulerOS 2.8	4.19	7.3.0
aarch_64+Atlas 300T 9000	Ubuntu 18.04	4.15.0-45-generic **说明** 若内核版本和操作系统版本匹配失败，需先安装dkms，安装方法请参见**4.1 驱动源码编译**。	7.4.0

表 1-4 通用驱动包兼容操作系统版本信息

硬件形态	Host操作系统版本	软件包默认的Host操作系统内核版本	GCC编译器版本
x86_64+Atlas 300T 9000	CentOS 8.2	4.18.X **说明** 支持升级到5.6.14。	8.3.1
aarch_64+Atlas 300T 9000	CentOS 8.2	4.18.X **说明** 支持升级到5.6.14。	8.3.1
aarch_64+Atlas 300T 9000	kylinosv10SP1 **说明** **kylinosv10SP1**在21.0.rc1及以上版本才支持。	4.19.90-17.ky10.aarch64	8.3.1-4.5

验证 Linux 操作系统版本信息

执行**uname -m && cat /etc/*release**命令，查询正在运行的操作系统版本和操作系统架构。

系统正在运行的操作系统版本和操作系统架构必须与**表1-3**或**表1-4**中要求一致。

通用驱动包特有检查项

- 确认make工具是否已安装；执行**make -v**命令，若能查询到make工具的版本，则make已安装。
- 以下2个检查项满足其中任意一个，驱动就可以安装成功。
  - 确认系统已安装dkms等安装依赖；具体请参见**4.1 驱动源码编译**。
  - 确认默认内核源码路径“/lib/modules/`uname -r`/build”是否存在；执行**ls /lib/modules/`uname -r`/build**命令，查看路径是否存在。
    - 若存在，则会自动使用内核进行驱动编译。

版权所有 © 华为技术有限公司

    - 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5**。
- 对于**Centos8.2+x86**系统，安装软件包前需要检查安装依赖“elfutils-libelf-devel”软件包是否已安装；执行**rpm -qa | grep elfutils-libelf-devel**命令，若能查询到“elfutils-libelf-devel”软件信息，则说明已安装。

验证 Linux 操作系统内核版本

执行**uname -r**命令，查询当前Host操作系统的内核版本。

- 若使用二进制驱动包安装，当前Host操作系统的内核版本必须与**表1-3**中要求一致。若不一致，有两种解决方法：
  - 用户需要重新进行源码编译，详细信息请参见**4.1 驱动源码编译**。
  - 参见下方**检查运行环境是否安装过软件包**检查系统是否安装过软件包，若未安装则直接升级系统内核版本；若安装过，则先卸载软件包，再升级内核。
- 若使用通用驱动包安装，当前host操作系统的内核版本与**表1-4**中要求一致，若不一致，可能导致驱动包安装不成功或者影响功能使用。

检查运行环境是否安装过软件包

如需要升级系统内核版本，在升级之前，请确保当前系统上未安装软件包。否则会出现内核升级后软件包启动异常的问题。出现该问题时请参见**4.2 软件包不可用**解决。请使用**lsmod|grep drv**命令查询是否安装过软件包。

- 如无内容表示未安装过软件包，可以直接升级系统内核版本。
- 如有内容，表示安装过软件包，需要先卸载软件包后，升级系统内核版本。卸载请参见**1.4 卸载驱动和固件**。

1.2.5 创建运行用户

- 若使用root用户安装

  可以使用root用户进行安装，但安装完之后要求使用非root用户运行，所以安装前需要先创建运行用户。
  - 如果创建的用户是**HwHiAiUser**，安装软件包时可以直接使用该运行用户，默认即为**HwHiAiUser**。
  - 如果创建的用户是非**HwHiAiUser**，安装软件包时需要指定运行用户（通过***--install-username=username --install-usergroup=usergroup***参数指定）。
- 若使用非root用户安装

  该场景下安装及运行用户必须相同。
  - 若已有非root用户，则无需再次创建。
  - 若想使用新的非root用户，则需要先创建该用户，请参见如下方法创建：

    创建非**root**用户操作方法如下，以下命令请以**root**用户执行。

    i. 创建非**root**用户。
    ```
    groupadd usergroup
    useradd -g usergroup -d /home/username -m username
    ```
    ii. 设置非**root**用户密码。
    ```
    passwd username
    ```

 版权所有 © 华为技术有限公司

说明

- 指定运行用户是指定Driver的运行用户，Firmware不支持指定，共用Driver的运行用户。
- 上述命令中的*username*为用户名，请根据实际情况替换。
- 由于权限控制可能存在安全风险，创建的运行用户不建议为root用户属组。
- 创建完HwHiAiUser用户后，请勿关闭该用户的登录认证功能。
- 密码有效期为90天，您可以在/etc/login.defs文件中修改有效期的天数，或者通过**chage**命令来设置用户的有效期，详情请参见**设置用户有效期**。

设置用户的有效期

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1-5**。

表 1-5 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，则用户密码不会超期，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。
-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1-5**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

 版权所有 © 华为技术有限公司

1.2.6 （可选）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。

重构驱动包目前支持以下3种方式：

1. 直接重构驱动包
2. 解压run包后重构驱动包
3. 使用重构工具重构驱动包

⚠ 注意

建议用户使用前两种方式。

- 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

- 确认目标OS及其内核版本号对应的GCC工具、内核源码已存在，具体请参见**1.2.4 系统要求和环境检查**。
- 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**。
- 检查pigz工具是否已安装，执行**which pigz**命令，若能查出pigz工具存在的位置则表示已安装。

直接重构驱动包

步骤1 获取通用驱动包，详细信息请参见**1.2.2 获取软件包**。

步骤2 上传通用驱动包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，增加通用驱动包的执行权限。

命令：**chmod +x** **.run*

示例：**A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run**

步骤5 在root用户下执行以下命令，构建新驱动包。

命令：**./****.run* **--repack** *[package_name]*

示例：**./A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run --repack A300t-9000-npu-driver-repack.run**

版权所有 © 华为技术有限公司

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1.3 安装驱动和固件，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1.2.2 获取软件包。

步骤2 上传通用驱动包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，增加通用驱动包的执行权限。

命令：**chmod +x** **.run*

示例：**A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run**

步骤5 执行以下命令，解压通用驱动包到指定目录。

命令：**./****.run* **--noexec --extract=***目标路径*

示例：**./A300t-9000-npu-driver_*x.x.x*_linux-aarch64.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：**./***.run **--repack-path=***<path> [package_name]*

示例：**./A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run --repack-path=tmp/A300t-9000-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1.3 安装驱动和固件，安装驱动。

----结束

使用重构工具重构驱动包

步骤1 获取通用驱动包和重构工具包，详细信息请参见1.2.2 获取软件包。

步骤2 上传通用驱动包、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，增加通用驱动包的执行权限。

命令：**chmod +x** **.run*

 版权所有 © 华为技术有限公司

示例：**chmod +x A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run**

步骤5 执行如下命令，解压通用驱动包到指定目录。

命令：**./**.run* --noexec --extract=*目标路径***

示例：**./A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run --noexec --extract=./tmp**

步骤6 执行如下命令，解压重构工具包到当前目录。

命令：**tar -xzvf *重构工具包名称* --strip-components 1**

示例：**tar -xzvf A300t-9000-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 通用驱动包解压的路径 输出包名***

示例：**bash build.sh ./makeself-release-2.4.0/makeself.sh tmp/ A300t-9000-npu-driver-repack.run**

步骤9 请参见1.3 安装驱动和固件，安装驱动。

----结束

1.3 安装驱动和固件

前提条件

请参见1.2 安装前准备完成安装前准备。

操作步骤

请按照“驱动 > 固件”的顺序，分别安装组件软件包。每个组件软件包的安装步骤相同，详细操作如下。软件包中的*请根据实际情况进行替换。

步骤1 使用**root**用户登录到运行环境，将*.run软件包上传至运行环境任意路径下，如/opt下。

步骤2 增加安装用户对软件包的可执行权限。

在软件包所在路径执行**ls -l**命令检查安装用户是否有该文件的执行权限，若没有，请执行如下命令。

chmod +x *.run

步骤3 执行如下命令，校验软件包安装文件的一致性和完整性。

./*.run --check

步骤4 执行以下命令进行安装。

- 如果软件包安装路径是指定的，以“/test/HiAI/”为例。执行**./*.run --full --install-path=***/test/HiAI/*

 版权所有 © 华为技术有限公司

- 如果软件包安装路径是默认的，执行./***.run --full**

  说明

  - 指定软件包安装路径是指定Driver安装路径，Firmware不支持指定路径，共用Driver的安装路径。
  - 若指定路径安装驱动时：
    - 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**
  - 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
    ```
  - 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Driver/Firmware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

步骤5 对于通用驱动包，若出现以下回显，说明未安装dkms，默认内核源码路径/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/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6 若出现如下信息，则说明安装成功：

- 驱动：
  ```
  Driver package install success! Reboot needed for installation/upgrade to take effect!
  ```
- 固件：
  ```
  Firmware package install success! Reboot needed for installation/upgrade to take effect!
  ```

步骤7 重启运行环境。

步骤8 查看安装的驱动版本号。

 版权所有 © 华为技术有限公司

在软件包的安装路径下，例如root用户默认路径“/usr/local/Ascend/*{package_name}*”，执行如下命令查看所升级软件包版本是否正确。

cat version.info

```
Version=1.73.T105.0.B050
```

步骤9 查看安装的NPU固件版本号。

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

```
Get component version(1.73.5.0.b050) succeed for deviceId(0), componentType(0).
        {"device_id":0, "component":nve, "version":1.73.5.0.b050}
Get component version(1.73.5.0.b050) succeed for deviceId(0), componentType(3).
        {"device_id":0, "component":uefi, "version":1.73.5.0.b050}
Get component version(1.73.5.0.b050) succeed for deviceId(0), componentType(8).
        {"device_id":0, "component":imu, "version":1.73.5.0.b050}
Get component version(1.73.105.0.b050) succeed for deviceId(0), componentType(9).
        {"device_id":0, "component":imp, "version":1.73.105.0.b050}
```

步骤10 执行**npu-smi info**查看NPU工具安装是否成功。

回显如下类似信息，说明安装成功。否则，说明安装失败，请联系华为技术支持处理。

```
[root@localhost ~]# npu-smi info
+------------------------------------------------------------------------------------------------+
| npu-smi 1.5.8                     Version: 1.73.T105.0.B050                                    |
+-------------------+-----------------+----------------------------------------------------------+
| NPU     Name      | Health          | Power(W)         Temp(C)                                 |
| Chip              | Bus-Id          | AICore(%)        Memory-Usage(MB)      HBM-Usage(MB)     |
+===================+=================+==========================================================+
| 2       910       | OK              | 64.5             54                                      |
| 0                 | 0000:01:00.0    | 0                1092 / 15600          0    / 32255     |
+===================+=================+==========================================================+
```

说明

- 步骤回显中“npu-smi”后面的字段为NPU工具版本号，“Version:”后面的字段为NPU驱动版本号。
- 其他命令使用请参考**《Atlas 300T 训练卡 npu-smi 命令参考 (型号 9000)》**。

----结束

版本安装注意事项

- 使用run安装包后，不要手动设置环境变量export LD_LIBRARY_PATH指向之前rar包的旧SO文件，否则可能会出现run安装包内工具连接到之前版本的动态库。指向第三方库文件路径、非run安装包发布库文件路径的配置不受影响。
- 查看日志时需注意：日志时间采用的是系统时间，device侧时间与host侧时间保持同步，修改host侧时间可以使用**date**命令。

  例如：执行**date -s 17:55:55**命令，将系统时间设定为17点55分55秒。

1.4 卸载驱动和固件

操作步骤

驱动和固件的卸载没有先后顺序。软件包中的*请根据实际情况进行替换。

 版权所有 © 华为技术有限公司

步骤1 使用root账户登录运行环境。

步骤2 支持两种卸载方式，用户根据实际情况选择一种方式卸载即可：

- 使用软件包进行卸载，请在run包存放目录下执行以下命令，如/opt：

  ./*.run --uninstall

- 在任意路径执行如下命令卸载。

  bash *{install_path}*/*{package_name}*/script/uninstall.sh

  说明

  install_path请替换为安装时的实际路径；***package_name***请根据实际场景替换，固件包替换为firmware，驱动包替换为driver。

步骤3 如果卸载过程中无错误信息提示，则表示卸载成功，根据系统提示信息决定是否重启服务器，完成对软件包的卸载。

----结束

 版权所有 © 华为技术有限公司

2 升级

2.1 升级前必读
2.2 升级前准备
2.3 升级组件

2.1 升级前必读

升级影响

升级过程禁止进行其他维护操作动作。

Atlas 300T 训练卡软件版本升级过程中需要复位系统，会导致业务中断。

注意事项

Atlas 300T 训练卡版本升级时的注意事项如**表2-1**所示。

表 2-1 升级时注意事项

序号	描述
1	在进行升级操作之前，请仔细阅读本文档，确定已经理解全部内容。如果您对文档有任何意见或建议，请联系华为技术支持解决。
2	为了减少对业务的影响，请提前切走业务或在业务量低时进行升级操作。
3	升级后，请确保所有组件的版本保持一致。
4	需要确保系统已经正常运行且固件升级工具及依赖的驱动被正确加载、确保运行环境正常运行。如果不能确保，请复位后再升级。相关检查方法请参见**4.3 如何检查device是否正常运行**。
5	不建议用户修改“/etc/ascend_install.info”，如果修改，会导致系统功能不可用。

 版权所有 © 华为技术有限公司

序号	描述
6	NPU芯片是Ascend 910 Pro B的训练标卡，NPU驱动需使用20.2.0.spc100及以上版本，且不支持回退至20.1.0及其补丁版本或更低软件版本；NPU固件使用配套驱动同步发布的版本。

版本要求

建议使用同一个Atlas 300T 训练卡软件版本列表下驱动版本、固件版本、MCU版本，使三者保持配套关系。

升级流程

请按照“固件->驱动->MCU”的升级顺序，分别升级软件包。

2.2 升级前准备

升级前检查

升级操作前，请按照表2-2中序号顺序进行检查，并记录检查结果。

表 2-2 升级前检查表

序号	项目	检查标准
1	检查软件版本	1. 查询并记录当前系统中Atlas 300T 训练卡版本。 2. 确认需要升级的版本。
2	检查系统状态	检查Atlas 300T 训练卡告警： ● 未出现告警，可直接升级。 ● 出现告警时，请与华为技术支持确认告警情况，然后再决定下一步动作。

不同芯片的软件版本支持情况如表2-3所示。

表 2-3 软件版本配套表

芯片名称	类型	适配版本
Ascend 910 A	Atlas 300T 训练卡 (型号 9000)	20.0.0、20.1.0、20.2.0
Ascend 910 B	Atlas 300T 训练卡 (型号 9000)	20.0.0、20.1.0、20.2.0
Ascend 910 Pro B	Atlas 300T 训练卡 (型号 9000)	20.2.0.SPC300

 版权所有 © 华为技术有限公司

获取升级包

请通过下述方法获取升级包。

1. 登录**A300T-9000**。
2. 打开目标版本“A300T-9000 *X.X.X*”。
   固件、驱动与CANN的配套关系，请查看CANN软件版本的《**CANN 版本配套表**》。
3. 在指定版本页面下载对应升级包到客户端（本地PC）。获取的升级包如**表2-4**所示。

说明

- 下文中所有x.x.x代表软件版本号。
- A300t-9000-npu-driver_*x.x.x*_linux-x86_64.run适配x86所有操作系统；A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run适配arm所有操作系统。

表 2-4 升级包信息

硬件形态	Host操作系统版本	Host操作系统内核版本	驱动包名称	固件包名称	MCU包名称
x86_64+Atlas 300T 训练卡 (型号 9000) aarch_64+Atlas 300T 训练卡 (型号 9000)	Debian 9.9	4.9.0-9-amd64	A300t-9000-npu-driver_*x.x.x*_debian9.9-x86_64.run	A300t-9000-npu-firmware_<*version*>.run **说明** Atlas 300T 训练卡 (型号 9000) 中芯片固件升级包。	• A300T-9000-MCU_Firmware-*x.x.x*.bin **说明** Atlas 300T 训练卡 (型号 9000)的加速模块MCU固件升级包，用于npu-smi命令方式升级。 • A300T-9000-MCU_Firmware-*x.x.x*.hpm **说明** Atlas 300T 训练卡 (型号 9000)的加速模块MCU固件升级包，用于iBMC工具方式升级MCU。
	CentOS 7.6 4.14.0	3.10.0-957.el7	A300t-9000-npu-driver_*x.x.x*_centos7.6-x86_64.run A300t-9000-npu-driver_*x.x.x*_centos7.6-aarch64.run		

 版权所有 © 华为技术有限公司

硬件形态	Host操作系统版本	Host操作系统内核版本	驱动包名称	固件包名称	MCU包名称
	Ubuntu 18.04	4.15.0-45-generic **说明** 若内核版本和操作系统版本匹配失败，需先安装dkms，安装方法请参见**4.1 驱动源码编译**。	A300t-9000-npu-driver_*x.x.x*_ubuntu18.04-x86_64.run A300t-9000-npu-driver_*x.x.x*_ubuntu18.04-aarch64.run		
	EulerOS 2.8	4.19	A300t-9000-npu-driver_*x.x.x*_euleros2.8-aarch64.run		
	CentOS 8.2	4.18.X **说明** 支持升级到5.6.14。	A300t-9000-npu-driver_*x.x.x*_linux-x86_64.run A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run		
	kylinos v10SP1 **说明** **kylinosv10SP1**在21.0.rc1及以上版本才支持。	4.19.90-17.ky10.aarch64	A300t-9000-npu-driver_*x.x.x*_linux-aarch64.run		

 版权所有 © 华为技术有限公司

检查软件包完整性

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**

2.3 升级组件

本章介绍各组件的升级操作，请按照“固件->驱动->MCU”的升级顺序，分别升级软件包。

2.3.1 升级 Ascend 910 芯片固件

Atlas 300T 训练卡支持Ascend 910芯片固件升级。本章节以Atlas 300T 训练卡的A300t-9000-npu-firmware_*<version>*.run包为例介绍升级固件的相关操作。

操作步骤

步骤1 参照**2.2 升级前准备**准备软件包“A300t-9000-npu-firmware_*<version>*.run”。

步骤2 使用root帐户登录Atlas 300T 训练卡所在服务器。

步骤3 将“A300t-9000-npu-firmware_*<version>*.run”上传至Linux系统任意目录下，如/opt下。

步骤4 进入软件包“A300t-9000-npu-firmware_*<version>*.run”所在路径，如/opt下。

cd */opt*

步骤5 执行如下命令，更改“A300t-9000-npu-firmware_*<version>*.run”软件包权限。

chmod u+x *A300t-9000-npu-firmware_<version>.run*

步骤6 执行**./***A300t-9000-npu-firmware_<version>.run* **--check**命令，校验run安装包的一致性和完整性。

步骤7 执行升级命令完成升级操作。

升级过程支持一键式升级操作，可执行**./***A300t-9000-npu-firmware_<version>.run* **--upgrade**命令，完成升级。

出现以下类似回显，表示升级成功。

```
Firmware package install success! Reboot needed for installation/upgrade to take effect!
```

 版权所有 © 华为技术有限公司

说明

- 软件包名称中的*<version>*表示固件版本号。
- 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8 重启系统。

reboot

步骤9 检查升级后的版本号。

在软件包安装路径下，执行以下命令查看所升级软件包版本是否正确。

cat version.info

说明

软件包安装默认路径：“/usr/local/Ascend/firmware”。

----结束

2.3.2 升级 Ascend 910 芯片驱动

操作场景

该任务指导用户升级Atlas 300T 训练卡驱动。

run升级包支持一键式升级，本章节以Atlas 300T 训练卡的A300t-9000-npu-driver_*x.x.x*_debian9.9-x86_64.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。

说明

驱动升级不会改变系统的用户名和密码。

对系统的影响

Atlas 300T 训练卡驱动升级过程中需要复位系统，会导致业务中断。为了减少对业务的影响，执行升级操作前请提前切走业务。

操作步骤

步骤1 参照**2.2 升级前准备**准备驱动包“A300t-9000-npu-driver_*x.x.x*_debian9.9-x86_64.run”。

步骤2 使用root帐户登录Atlas 300T 训练卡所在服务器。

步骤3 将“A300t-9000-npu-driver_*x.x.x*_debian9.9-x86_64.run”上传至Linux系统任意目录下，如/opt下。

步骤4 进入软件包“A300t-9000-npu-driver_*x.x.x*_debian9.9-x86_64.run”所在路径，如/opt下。

cd /*opt*

步骤5 执行如下命令，更改“A300t-9000-npu-driver_*x.x.x*_debian9.9-x86_64.run”软件包权限。

 版权所有 © 华为技术有限公司

chmod u+x *A300t-9000-npu-driver_x.x.x_debian9.9-x86_64.run*

步骤6 执行**./***A300t-9000-npu-driver_x.x.x_debian9.9-x86_64.run* **--check**命令，校验run安装包的一致性和完整性。

步骤7 执行升级命令完成升级操作。

升级过程支持一键式升级操作，可执行**./***A300t-9000-npu-driver_x.x.x_debian9.9-x86_64.run* **--upgrade**命令，完成升级。

出现以下类似回显，表示升级成功。

```
Driver package install success! Reboot needed for installation/upgrade to take effect!
```

说明

- 升级驱动时，会将动态库libdcmi.so和头文件dcmi_interface_api.h拷贝到“/usr/local/dcmi/”目录下。
- 驱动升级过程中出现的异常场景及其解决方法请参见**4 FAQ**。
- 驱动升级过程中Ascend910驱动的日志信息输出在“/var/log/ascend_seclog/ascend_install.log”文件中。

步骤8 重启系统。

reboot

步骤9 检查升级后的版本号。

在软件包安装路径下，执行以下命令查看所升级软件包版本是否正确。

cat version.info

说明

软件包安装默认路径：“/usr/local/Ascend/driver”。

- 如果升级后无法登录Host系统，请联系华为技术支持解决。
- 如果升级后不是目标版本或者升级失败，请重新进行升级。如果依然升级失败，请记录故障现象和操作步骤，并联系华为技术支持解决。

----结束

2.3.3 升级 MCU

MCU是Atlas 300T 训练卡的带外管理模块，具备单板监控、故障上报等功能。Atlas 300T 训练卡已集成了初始版本，为了保障所有功能正常使用，请将MCU升级到最新版本。

MCU升级有两种方式：

- 通过npu-smi工具升级
- 通过iBMC升级

Ascend 910 Pro B乘加计算能力训练标卡的MCU，只能使用2.2.1及以后版本。

MCU版本升级有如下限制：

- MCU的2.0.4及以上版本，均不支持回退到2.0.4之前版本。
- MCU版本升级至2.2.1及以上版本，需先升级至2.1.6中间版本进行跳转，且不支持回退至2.1.3及以下版本。

 版权所有 © 华为技术有限公司

- MCU 1.4.9、1.4.10、2.0.5、2.1.1和2.2.1版本存在VRD变更，如果升级前后VRD版本有变化则需要在升级完成30秒后，对Host侧进行上下电以更新VRD固件。

MCU版本号查询命令：**npu-smi upgrade -b mcu -i** *NPU ID*

2.3.3.1 通过 npu-smi 升级

npu-smi工具可以将单个Atlas 300T 训练卡的MCU升级到相应版本，如果配备了多个Atlas 300T 训练卡，需要逐个升级。本章以Atlas 300T 训练卡的A300t-9000-mcu_*x.x.x*.bin包为例介绍MCU升级的相关操作。

操作步骤

步骤1 参照**2.2 升级前准备**准备安装包“A300t-9000-mcu_*x.x.x*.bin”。

步骤2 使用root帐户登录Atlas 300T 训练卡所在服务器。

步骤3 将“A300t-9000-mcu_*x.x.x*.bin”上传至Linux系统任意目录下，如/opt下。

步骤4 进入“A300t-9000-mcu_*x.x.x*.bin”所在路径，如/opt下。

cd /*opt*

步骤5 执行如下命令启动升级。

npu-smi upgrade -t mcu -i *NPU ID* **-f** *A300t-9000-mcu_x.x.x.bin*

步骤6 执行如下命令使新版本生效。

npu-smi upgrade -a mcu -i *NPU ID*

步骤7 在生效新版本之后，等待30s，查询MCU版本号，确保升级成功。

npu-smi upgrade -b mcu -i *NPU ID*

说明

- *NPU ID*表示Atlas 300T 训练卡的设备编号，可通过**npu-smi info -l**命令查询。详细信息请参见**《Atlas 300T 训练卡 npu-smi 命令参考 (型号 9000)》**。
- *x.x.x*表示MCU的版本号。

步骤8 如果升级后不是目标版本或者升级失败，请重新进行升级。如果依然升级失败，请记录故障现象和操作步骤，并联系华为技术支持解决。

----结束

2.3.3.2 通过 iBMC 升级

iBMC是服务器的管理软件，具备升级MCU的功能。该功能能够将服务器上配备的所有Atlas 300T 训练卡的MCU升级到相应版本，根据执行方式不同有如下两种形式：

- 通过Web界面升级
- 通过命令行升级

下面以Atlas 300T 训练卡的A300t-9000-mcu_*x.x.x*.hpm包为例介绍MCU升级的相关操作。

 版权所有 © 华为技术有限公司

2.3.3.2.1 通过 Web 界面升级

操作步骤

步骤1 参照**2.2 升级前准备**准备安装包“A300t-9000-mcu_*x.x.x*.hpm”。

步骤2 在浏览器的地址栏中输入“*https://iBMC管理网口IP地址*”，可打开用户登录界面。

输入“用户名”和“密码”登录iBMC Web界面，默认用户名和密码具体请参见**《Atlas硬件产品 用户清单》**。

步骤3 单击“固件升级”。

步骤4 选择升级文件，单击“开始升级”。

步骤5 升级完成。

 版权所有 © 华为技术有限公司

步骤6 检查升级后的版本号。

1. 在iBMC Web主界面，选择“系统管理 > 系统信息 > 其他”。
2. 在打开界面中的“PCIe卡”区域查看“MCU固件版本”信息。

如果升级后不是目标版本或者升级失败，请重新进行升级。如果依然升级失败，请记录故障现象和操作步骤，并联系华为技术支持解决。

----结束

2.3.3.2.2 通过命令行升级

操作步骤

步骤1 参照**2.2 升级前准备**准备安装包“A300t-9000-mcu_*x.x.x*.hpm”。

步骤2 使用PuTTY工具，输入iBMC的主机IP地址、用户名和密码登录iBMC命令行。

默认用户名和密码具体请参见**《Atlas硬件产品 用户清单》**。

步骤3 将“A300t-9000-mcu_*x.x.x*.hpm”上传至服务器的/tmp路径下。

步骤4 执行如下iBMC命令启动升级。

ipmcset -d upgrade -v */tmp/A300t-9000-mcu_x.x.x.hpm*

例如，

```
ipmcset -d upgrade -v /tmp/A300t-9000-mcu_x.x.x.hpm
```

回显如下信息：

```
Please make sure the iBMC is working while upgrading.
Updating...
100%
Upgrade successfully.
```

说明

- */tmp/A300t-9000-mcu_x.x.x.hpm*：带有绝对路径的升级包。
- *x.x.x*表示MCU的版本号。

如果升级后不是目标版本或者升级失败，请重新进行升级。如果依然升级失败，请记录故障现象和操作步骤，并联系华为技术支持解决。

----结束

 版权所有 © 华为技术有限公司

3 版本回退

回退原则

版本回退执行步骤：先回退Ascend910固件，再回退Atlas 300T 训练卡 (型号 9000)的驱动。

版本升级失败后，固件版本依然为升级前版本，不需要进行回退操作。当出现以下问题，请及时决策是否进行回退操作。

- 升级过程中出现重大异常，无法定位问题原因，在规定时间内无法排障。
- 完成升级后，出现新版本功能异常等问题。
- 留给完成后续升级操作的时间太短，无法在预定时间内完成升级，为避免因升级操作继续中断业务的情况持续到早晨业务高峰期来临时段，执行回退。

NPU芯片是Ascend 910 Pro B的训练标卡回退驱动版本时，请以实际是否能查询到NPU固件版本号来判断设备是否可用，查询方式参考**步骤9**。

回退操作

1. 确定需要回退的版本号。

   建议回退到升级前版本。

2. 回退软件版本。

   回退到指定版本时，回退操作方式和升级操作相同。具体操作过程请参见**2.3 升级组件**回退所有部件版本，保持版本一致。

 版权所有 © 华为技术有限公司

4 FAQ

4.1 驱动源码编译

4.2 软件包不可用

4.3 如何检查device是否正常运行

4.4 BC_Linux镜像源配置

4.5 驱动20.2.0及以上版本回退低版本失败问题

4.6 npu-smi查询MCU健康状态出现1C1F mcu flash app error告警

4.7 npu-smi查询MCU健康状态出现1DAF vrd PSU firmware needs to be upgraded告警

4.1 驱动源码编译

驱动源码编译支持自动编译和手动编译两种方式，其中自动编译是通过dkms工具自动完成，需要先安装dkms工具。

如下操作请切换到root用户执行。

前提条件

通过如下命令查询dkms、gcc、linux-header等软件包是否已安装：

- Ubuntu/Debian操作系统：
  dpkg-query -s dkms（*仅自动编译方式需要查询dkms*）
  dpkg-query -s gcc
  dpkg-query -s linux-headers-$(uname -r)
- CentOS操作系统：
  rpm -qa | grep dkms（*仅自动编译方式需要查询dkms*）
  rpm -qa | grep gcc
  rpm -qa | grep kernel-*-headers-$(uname -r)
  rpm -qa | grep kernel-*-devel-$(uname -r)
- EulerOS操作系统：

 版权所有 © 华为技术有限公司

rpm -qa | grep gcc

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

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

说明

- 对于Ubuntu/Debian操作系统，需要安装dkms、gcc、linux-header软件包，若未安装请自行在相应操作系统厂商网站获取并安装。
- 对于CentOS操作系统，需要安装dkms、gcc、kernel-*-headers、kernel-*-devel软件包，若未安装请自行在相应操作系统厂商网站获取并安装。
- 对于EulerOS操作系统，需要安装gcc、kernel-headers、kernel-devel软件包，若未安装请自行在相应操作系统厂商网站获取并安装。
- DKMS软件包建议使用dkms-2.6.1-1.el7.noarch.rpm。

自动编译

安装run安装包时，如果内核版本与软件包中驱动镜像文件版本不一致，则run安装包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。

安装run软件包请参见**1.3 安装驱动和固件**。

手动编译

安装run安装包时，如果内核版本与run安装包中驱动镜像文件一致，此时不会触发源码编译。但在后续使用中，用户升级了内核，由于run安装包已经安装，无法触发驱动源码自动编译，此时用户需要去run安装包安装路径下手动执行驱动源码编码脚本。

- 脚本路径：“run安装路径/driver/script”。
- 脚本内容及作用如下。
  - **run_driver_dkms_install.sh**

    说明

    驱动源码编译脚本，并添加到dkms框架下。脚本执行完毕后，会自动更新run安装包链接的驱动镜像。
  - **run_driver_dkms_uninstall.sh**

    说明

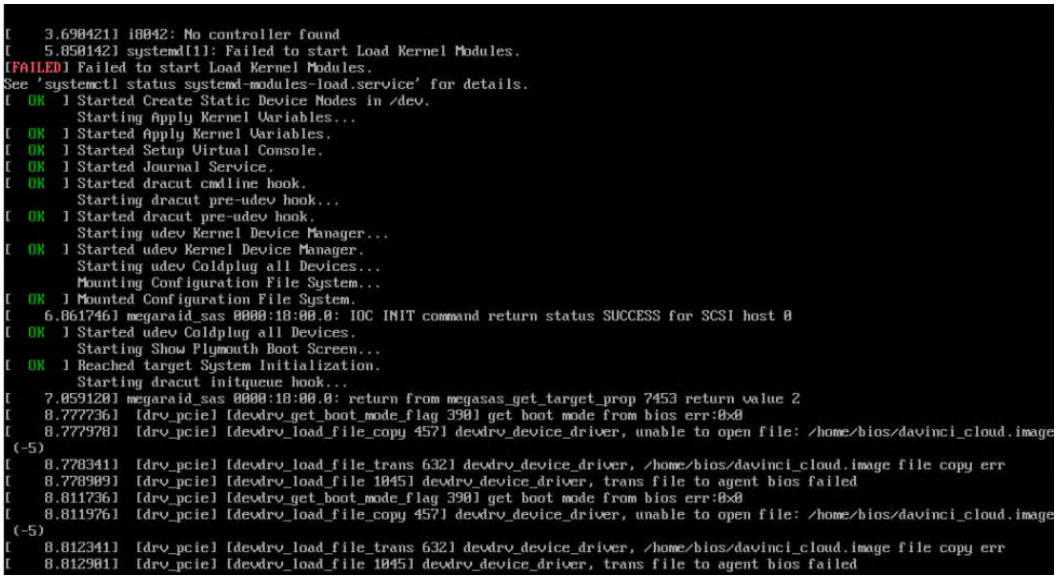
    驱动源码dkms卸载脚本，从dkms框架移除。移除不影响当前已安装run安装包，只是后续无自动编译功能。

4.2 软件包不可用

如果在安装了软件包的环境直接升级内核，再次重启时环境会出现启动异常。如果后续在该环境上再安装软件包也无法使用。这是由于安装软件包时驱动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。

 版权所有 © 华为技术有限公司

图 4-1 软件包不可用



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

4.3 如何检查 device 是否正常运行

步骤1 以**root**用户登录运行环境，查询软件包安装路径。

cat /etc/ascend_install.info

系统显示示例如下：

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

步骤2 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。

 版权所有 © 华为技术有限公司

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

----结束

4.4 BC_Linux 镜像源配置

安装driver时，如果安装失败，显示缺少dkms和kernel-headers的问题，则通过配置镜像源的方式来解决。下面以BC_Linux为例进行说明。

操作步骤

步骤1 使用**root**用户登录到运行环境。

步骤2 将原有的repo文件保存为备用文件。

mv /etc/yum.repo.d/BCLinux-Base.repo /etc/yum.repo.d/BCLinux-Base.repo.bak

步骤3 在KVM界面挂载系统安装镜像ISO文件。

1. 打开远程虚拟控制台。
2. 在工具栏中单击 。
   打开虚拟光驱的操作窗口，如图4-2所示。

   图 4-2 虚拟光驱操作窗口

3. 选择“本地文件夹”并单击“浏览”。
4. 选中客户端上已下载的ISO文件所在的文件夹。
5. 单击虚拟光驱操作窗口中的“Insert”。

步骤4 打开命令行操作台。

步骤5 将ISO镜像挂载至指定目录，例如/mnt目录下。

mount /dev/cdrom /mnt

步骤6 编辑文件/etc/yum.repo.d/iso.repo，

vi /etc/yum.repo.d/iso.repo

按“i”编辑文件，在文件中增加如下内容：

```
[base-local]
name=iso
baseurl=file:///mnt
enable=1
gpgcheck=0
```

 版权所有 © 华为技术有限公司

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤7 更新yum源。

yum clean all

yum makecache

步骤8 查看repo列表。

yum repolist

回显如下信息，说明配置本地源成功。

----结束

4.5 驱动 20.2.0 及以上版本回退低版本失败问题

回退至低版本若报**Device_images_crl_check failed**失败，解决方法如下：

步骤1 以root用户登录，执行如下驱动卸载脚本（以默认安装路径/usr/local/Ascend/driver/为例）。

```
[root@localhost ~]# /usr/local/Ascend/driver/script/uninstall.sh
```

步骤2 卸载成功后，若提示立即生效则跳过该步骤；若提示重启使卸载生效，请执行如下热复位命令。

```
[root@localhost ~]# reboot
```

步骤3 删除root路径下的ascend_check目录。

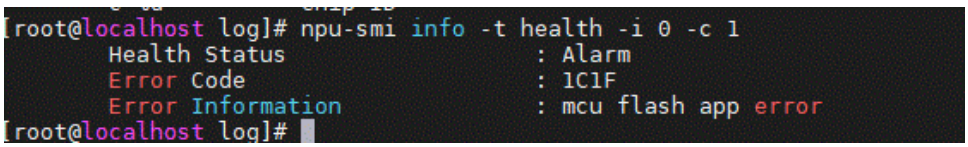
```
[root@localhost ~]# rm -rf /root/ascend_check
```

步骤4 重新尝试版本回退操作。

----结束

4.6 npu-smi 查询 MCU 健康状态出现 1C1F mcu flash app error 告警

步骤1 执行**npu-smi info -t health -i *NPU ID* -c 1**命令查询MCU健康状态，系统显示信息如下：



步骤2 执行**npu-smi upgrade -b mcu -i *NPU ID***命令查询MCU版本。

 版权所有 © 华为技术有限公司

若MCU版本号为2.1.6，升级至2.2.1及以上版本。

步骤3 升级完成后，等待半小时以上，再次查询MCU健康状态：

- 若告警消失，表示问题已解决。
- 若告警未消失，联系华为工程师处理。
- 若MCU为其他版本，联系华为工程师处理。

----结束

4.7 npu-smi 查询 MCU 健康状态出现 1DAF vrd PSU firmware needs to be upgraded 告警

步骤1 执行**npu-smi info -t health -i *NPU ID* -c 1**命令查询MCU健康状态，系统显示信息如下：

```
[root@localhost g00522086]# npu-smi info -t health -i 5 -c 1
        Health Status                   : Warning
        Error Code                      : 1DAF
        Error Information               : vrd PSU firmware needs to be upgraded
```

步骤2 系统下电，等待10秒后，给系统上电。

步骤3 上电完成后，再次查询MCU健康状态：

- 若告警消失，表示问题已解决。
- 若告警未消失，联系华为工程师处理。

----结束

 版权所有 © 华为技术有限公司

A 附录

A.1 参数说明

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： **./*.run** [options]

详细参数请参见**表A-1**。

须知

如果通过**./*.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 A-1 安装包支持的参数说明

参数	说明
--run	运行态模式：只安装运行场景需安装的文件。
--devel	开发态模式：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	完整安装模式：安装所有文件。
--docker	容器场景下安装，仅用于driver子包，其他子包默认按full模式。

版权所有 © 华为技术有限公司

参数	说明
--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** • 不建议指定root用户，如果指定root用户，权限控制可能存在安全风险。 • 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。 • 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（**1.2.5 创建运行用户**）保持一致。
--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（**1.2.5 创建运行用户**）保持一致。
--install-path=<path>	指定安装目录，如不指定，则默认安装路径为：/usr/local/Ascend。 若通过该参数指定了安装目录，运行用户需要对指定的安装路径有可读写权限。 **说明** Firmware安装不支持指定路径，共用Driver的安装路径。
--uninstall	卸载。
--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。
--upgrade	升级，立即生效。仅支持从软件包所在路径升级。
--help或者-h	查看帮助。
--check	校验安装包的完整性。
--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run安装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。

版权所有 © 华为技术有限公司

参数	说明
--install-for-all	当安装或者升级携带该参数时，子包中创建的目录及文件，权限设置为：其它用户权限修改为group的权限。 该参数需要与--run、--devel、--full、--upgrade等其中一个参数配合使用，例如**./*.run** --full --install-for-all。 **说明** Firmware不支持该参数。
--repack [package_name]	构建新驱动包。 package_name为新构建的驱动包名称，格式为”*.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”*.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。

举例说明：

- full模式下安装
  - 不指定安装目录：**./*.run** --full
  - 指定安装目录：**./*.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：**./*.run** --docker
  - 指定安装目录**./*.run** --docker --install-path=*安装路径*
- run模式下安装
  - 不指定安装目录：**./*.run** --run
  - 指定安装目录：**./*.run** --run --install-path=*安装路径*

A.2 相关脚本

在执行run安装包安装、卸载、更新等命令的过程中会调用表A-2中的各脚本，因此不需要单独手动执行各脚本。脚本存放路径以实际安装过程中指定的路径为准。如下以默认路径“/usr/local/Ascend”为例。

表 A-2 相关脚本

名称	路径	说明
run_firmware_install.sh	/usr/local/Ascend/firmware/	实现固件安装。

版权所有 © 华为技术有限公司

名称	路径	说明
run_firmware_uninstall.sh	/usr/local/Ascend/firmware/	实现固件的卸载。
install_common_parser.sh	/usr/local/Ascend/firmware/	filelist.csv的解析脚本。
install.sh	/usr/local/Ascend/firmware/	Firmware安装包的安装入口脚本。
host_sys_stop.sh	/usr/local/Ascend/	关闭后台进程。
host_sys_init.sh	/usr/local/Ascend/	调起后台进程。
host_server_setup.sh	/usr/local/Ascend/	获取系统配置，调用host_sys_init.sh脚本。
host_servers_remove.sh	/usr/local/Ascend/	移除host_sys_init.sh脚本开机启动和相关配置文件。
install.sh	/usr/local/Ascend/driver/script/	Driver安装包的安装入口脚本。
run_driver_dkms_install.sh	/usr/local/Ascend/driver/script/	DKMS加载某些模块。
run_driver_dkms_uninstall.sh	/usr/local/Ascend/driver/script/	DKMS卸载某些模块。
run_driver_install.sh	/usr/local/Ascend/driver/script/	实现驱动安装、卸载、更新或延迟更新的功能。
run_driver_uninstall.sh	/usr/local/Ascend/driver/script/	实现驱动的卸载。
run_driver_tool_install.sh	/usr/local/Ascend/driver/script/	用于安装工具。
run_driver_tool_uninstall.sh	/usr/local/Ascend/driver/script/	用于卸载工具。
install_common_parser.sh	/usr/local/Ascend/driver/script/	filelist.csv的解析脚本。
install_npudrv.sh	/usr/local/Ascend/driver/tools/	特权容器脚本，用于在容器内给宿主机升级驱动。

A.3 相关工具

表A-3中工具的执行方法如下：

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。

版权所有 © 华为技术有限公司

步骤2 执行命令切换到工具所在的路径下。例如切换到“/usr/local/Ascend/driver/tools”路径。命令为：**cd /usr/local/Ascend/driver/tools**

步骤3 执行表A-3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。

 版权所有 © 华为技术有限公司

表 A-3 相关工具

名称	路径	作用	执行命令
upgrade-tool	/usr/local/Ascend/driver/tools	查看固件版本、升级单个或多个固件等。	**说明** upgrade-tool必须在**/usr/local/Ascend/firmware/tools**路径下执行。 **cd /usr/local/Ascend/firmware/tools** dev_id=-1时，表示所有设备。 ● 列举所有Device：**/usr/local/Ascend/driver/tools/upgrade-tool --mini_devices** ● 获取指定设备的版本：**/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --system_version** ● 获取指定设备的组件信息：**/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --components** ● 查询某个设备中某一个组件对应的版本：**/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --component <type> --version** ● 查询设备状态：**/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --status** ● 热复位device：**/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --hot_reset** dev_id仅支持-1，表示热复位所有device，不支持热复位单个device。 **说明** – 执行热复位前，需要先关闭业务。 – 当前仅ARM架构支持热复位。 ● 查询设备是否是物理机：

版权所有 © 华为技术有限公司

名称	路径	作用	执行命令
			/usr/local/Ascend/driver/tools/upgrade-tool --device_index <dev_id> --phymachflag **说明** 当前版本仅支持物理机升级固件包。 • 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>** 如上命令中涉及的各参数含义如下。 – --mini_devices：所有设备列表。 – --device_index：设备编号。取值可以是【0~7】和-1，【0~7】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（当前aicpu仅支持单个升级）。升级所有组件时，需要输入-1，升级所有组件并重置密码时，需要输入9。 – --version：组件的版本。 – --status：设备状态。 – --path：固件包相对路径：若升级全部组件，则为**--path ./conf/upgrade.cfg**；若升级单个组件，例如nve.bin，则为**--path ../image/nve.bin**。 – --hot_reset：热复位device。 – --phymachflag：查询设备是否是物理机。如果不

版权所有 © 华为技术有限公司

名称	路径	作用	执行命令
			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。
hccn_tool	/usr/local/Ascend/driver/tools	集群网络工具，只允许root用户执行。	具体命令请参见《Ascend 910 HCCN Tool 接口参考》。

----结束

A.4 添加内核补丁

下载网址

内核源码rpm包（例如：kernel-3.10.0-957.el7.src.rpm）地址如下：

须知

内核源码rpm包的版本需与操作系统内核版本一致。

redhat官网：**https://access.redhat.com/downloads/**

centos网站：**http://vault.centos.org/**

补丁链接地址如下：

https://lore.kernel.org/lkml/1448575344-19107-1-git-send-email-jordan_hargrave@dell.com/#Z30drivers:firmware:dmi_scan.c

操作步骤

步骤1 安装并解压源码。

1. 执行如下命令，安装源码包。
   rpm -ivh kernel-3.10.0-957.el7.src.rpm
2. 执行如下命令，进入/root/rpmbuild/SOURCES目录。
   cd /root/rpmbuild/SOURCES/
3. 执行如下命令，解压源码包。
   xz -d linux-3.10.0-957.el7.tar.xz

版权所有 © 华为技术有限公司

tar xvf linux-3.10.0-957.el7.tar

步骤2 修改内核代码。

1. 执行如下命令，进入linux-3.10.0-957.el7目录。

   cd linux-3.10.0-957.el7

2. 修改相关函数。具体修改内容请参见**https://lore.kernel.org/lkml/1448575344-19107-1-git-send-email-jordan_hargrave@dell.com/#Z30drivers:firmware:dmi_scan.c**。

步骤3 重新压缩内核代码。

1. 执行如下命令，进入linux-3.10.0-957.el7文件夹所在目录。

   cd /root/rpmbuild/SOURCES

2. 执行如下命令，压缩源码文件。

   tar -cf linux-3.10.0-957.el7.tar linux-3.10.0-957.el7

   xz -z linux-3.10.0-957.el7.tar

3. 删除源码文件夹。

   rm -rf linux-3.10.0-957.el7

步骤4 编译源码包。

1. 执行如下命令，进入/root/rpmbuild/SPECS/目录。

   cd ../SPECS/

2. 执行如下命令，重构源码包。

   rpmbuild -ba kernel.spec

   说明

   生成的rpm包在/root/rpmbuild/RPMS目录。

步骤5 卸载原OS上的rpm包，安装新编译好的rpm包。

1. 执行如下命令，查询OS上的原有的内核rpm包。

   rpm -qa |grep kernel

   如下所示：

   ```
   [root@localhost wp]# rpm -qa |grep kernel
   kernel-devel-3.10.0-1160.21.1.el7.x86_64
   kernel-tools-3.10.0-1160.21.1.el7.x86_64
   kernel-headers-3.10.0-1160.21.1.el7.x86_64
   abrt-addon-kerneloops-2.1.11-60.el7.centos.x86_64
   kernel-3.10.0-1160.21.1.el7.x86_64
   kernel-devel-3.10.0-957.el7.x86_64
   kernel-tools-libs-3.10.0-1160.21.1.el7.x86_64
   kernel-3.10.0-957.el7.x86_64
   [root@localhost wp]#
   ```

2. 执行如下命令，卸载OS上原有的内核rpm包。

   rpm -e *文件名*

   如下所示：

```
[root@localhost wp]#
[root@localhost wp]# rpm -e kernel-devel-3.10.0-1160.21.1.el7.x86_64
[root@localhost wp]# rpm -e kernel-tools-3.10.0-1160.21.1.el7.x86_64
[root@localhost wp]# rpm -e kernel-headers-3.10.0-1160.21.1.el7.x86_64
error: Failed dependencies:
        kernel-headers >= 2.2.1 is needed by (installed) glibc-headers-2.17-323.el7_9.x86_64
[root@localhost wp]# rpm -e kernel-headers-3.10.0-1160.21.1.el7.x86_64 --nodeps
[root@localhost wp]# rpm -e kernel-3.10.0-1160.21.1.el7.x86_64
error: Failed dependencies:
        kernel >= 3.10.0-1133.el7 is needed by (installed) kmod-kvdo-6.1.3.23-5.el7.x86_64
        kernel(dm_get_device) = 0xa425b4a9 is needed by (installed) kmod-kvdo-6.1.3.23-5.el7.x86_64
        kernel(dm_put_device) = 0x412d503d is needed by (installed) kmod-kvdo-6.1.3.23-5.el7.x86_64
        kernel(dm_register_target) = 0x979ddcc5 is needed by (installed) kmod-kvdo-6.1.3.23-5.el7.x86_64
        kernel(dm_unregister_target) = 0xeeeae71b is needed by (installed) kmod-kvdo-6.1.3.23-5.el7.x86_64
[root@localhost wp]# rpm -e kernel-3.10.0-1160.21.1.el7.x86_64 --nodeps
warning: file /lib/modules/3.10.0-1160.21.1.el7.x86_64/updates: remove failed: No such file or directory
[root@localhost wp]# rpm -e kernel-devel-3.10.0-957.el7.x86_64 --nodeps
[root@localhost wp]# rpm -e kernel-tools-libs-3.10.0-1160.21.1.el7.x86_64
[root@localhost wp]# rpm -e kernel-3.10.0-957.el7.x86_64 --nodeps
[root@localhost wp]# rpm -qa | grep kernel
abrt-addon-kerneloops-2.1.11-60.el7.centos.x86_64
[root@localhost wp]#
[root@localhost wp]# ll
total 96256
```

须知

保留brt-addon-kerneloops-2.1.11-60.el7.centos.x86_64文件。

3. 执行如下命令，安装4新编译的kernel、kernel-devel、kernel-headers、kernel-tools-libs、kernel-tools内核rpm包。

   rpm -Uvh kernel-3.10.0-957.el7.x86_64.rpm

   rpm -Uvh kernel-devel-3.10.0-957.el7.x86_64.rpm

   rpm -Uvh kernel-headers-3.10.0-957.el7.x86_64.rpm

   rpm -Uvh kernel-tools-libs-3.10.0-957.el7.x86_64.rpm

   rpm -Uvh kernel-tools-3.10.0-957.el7.x86_64.rpm

   如下所示：

```
[root@localhost x86_64]# rpm -Uvh kernel-3.10.0-957.el7.x86_64.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:kernel-3.10.0-957.el7            ################################# [100%]
[root@localhost x86_64]# rpm -Uvh kernel-headers-3.10.0-957.el7.x86_64.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:kernel-headers-3.10.0-957.el7    ################################# [100%]
[root@localhost x86_64]# rpm -Uvh kernel-tools-libs-3.10.0-957.el7.x86_64.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:kernel-tools-libs-3.10.0-957.el7 ################################# [100%]
[root@localhost x86_64]# rpm -Uvh kernel-devel-3.10.0-957.el7.x86_64.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:kernel-devel-3.10.0-957.el7      ################################# [100%]
[root@localhost x86_64]# rpm -Uvh kernel-tools-3.10.0-957.el7.x86_64.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:kernel-tools-3.10.0-957.el7      ################################# [100%]
[root@localhost x86_64]#
```

----结束

故障处理

执行rpmbuild命令报编译错误，请添加如下依赖。

报错内容：

```
error: Failed build dependencies:
        xmlto is needed by kernel-3.10.0-957.el7.x86_64
        asciidoc is needed by kernel-3.10.0-957.el7.x86_64
        hmaccalc is needed by kernel-3.10.0-957.el7.x86_64
```


版权所有 © 华为技术有限公司

```
pesign >= 0.109-4 is needed by kernel-3.10.0-957.el7.x86_64
elfutils-devel is needed by kernel-3.10.0-957.el7.x86_64
zlib-devel is needed by kernel-3.10.0-957.el7.x86_64
binutils-devel is needed by kernel-3.10.0-957.el7.x86_64
newt-devel is needed by kernel-3.10.0-957.el7.x86_64
python-devel is needed by kernel-3.10.0-957.el7.x86_64
perl(ExtUtils::Embed) is needed by kernel-3.10.0-957.el7.x86_64
audit-libs-devel is needed by kernel-3.10.0-957.el7.x86_64
numactl-devel is needed by kernel-3.10.0-957.el7.x86_64
pciutils-devel is needed by kernel-3.10.0-957.el7.x86_64
```

添加如下依赖：

```
rpm -ivh zlib-devel-1.2.7-13.el7.x86_64.rpm
rpm -ivh boost-regex-1.53.0-18.el7.x86_64.rpm
rpm -ivh source-highlight-3.1.6-6.el7.x86_64.rpm
rpm -ivh asciidoc-8.6.8-5.el7.noarch.rpm
rpm -ivh hmaccalc-0.9.13-4.el7.x86_64.rpm
rpm -ivh opensc-0.13.0-9.el7.x86_64.rpm
rpm -ivh pesign-0.109-6.el7.x86_64.rpm
rpm -ivh elfutils-libelf-devel-0.158-3.el7.x86_64.rpm
rpm -ivh elfutils-devel-0.158-3.el7.x86_64.rpm
rpm -ivh binutils-devel-2.23.52.0.1-16.el7.x86_64.rpm
rpm -ivh newt-0.52.15-4.el7.x86_64.rpm
rpm -ivh slang-devel-2.2.4-11.el7.x86_64.rpm
rpm -ivh newt-devel-0.52.15-4.el7.x86_64.rpm
rpm -ivh python-devel-2.7.5-16.el7.x86_64.rpm
rpm -ivh perl-ExtUtils-Embed-1.30-283.el7.noarch.rpm
rpm -ivh audit-libs-devel-2.3.3-4.el7.x86_64.rpm
rpm -ivh numactl-devel-2.0.9-2.el7.x86_64.rpm
rpm -ivh pciutils-3.2.1-4.el7.x86_64.rpm
rpm -ivh pciutils-devel-3.2.1-4.el7.x86_64.rpm
rpm -ivh xmlto-0.0.25-7.el7.x86_64.rpm –nodeps
```

 版权所有 © 华为技术有限公司