

IoT 边缘

常见问题

文档版本
发布日期

1.1
2024-10-21



版权所有 © 华为技术有限公司 2024。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。

目录

1 概念相关	1
1.1 什么是 IoT 边缘?	1
1.2 开通边缘服务失败怎么处理?	1
1.3 IoT 边缘如何进行设备管理?	1
1.4 IoTEdge 与 IEF 的关系和区别是什么?	1
2 节点管理	3
2.1 安装专业版节点异常如何处理?	3
2.2 安装专业版节点, 执行安装命令时失败, 一直提示 “edgedaemon not found”?	6
2.3 安装专业版节点失败, 提示 “edge_core has existed”, 该如何处理?	6
2.4 专业版节点所在机器断网/断电一段时间后恢复, 节点模块一直处于 “部署中” 怎么处理?	7
2.5 专业版节点欠费充值后, 部署、升级模块仍然报错?	7
2.6 MQTT 设备无法接入边缘节点如何处理? 专业版节点所在机器断网/断电一段时间后恢复, 节点模块一直处于 “部署中” 怎么处理?	7
2.7 误删除 IoTDA 上的 edge_node/modbus_server 产品后, 在边缘服务创建节点/添加 Modbus 设备失败, 该怎么处理?	7
2.8 边缘节点离线排查指南	7
2.9 修改专业版 (Docker) 或基础版节点的云端日志级别, 但没有生效?	9
2.10 ubuntu 系统下节点 shell 命令兼容性导致的安装失败	9
3 子设备管理	11
3.1 边缘节点和子设备均在线, 为什么子设备上报的数据在 IoTDA 平台查看不到?	11
3.2 Modbus 设备与 MQTT 设备在数据上报流程有什么区别?	11
3.3 边缘节点断网离线后, 为何其子设备仍然在线?	12
3.4 如何让边端的设备数据不上云?	12
3.5 集成 ModuleSDK 后, 上报数据成功后, 设备状态显示为未激活, 如何上报子设备状态?	12
3.6 添加边缘设备的什么时候填写模块 ID?	13
4 应用管理	14
4.1 在边缘节点下部署自定义应用, 实例异常如何处理?	14
4.2 部署自定义应用时, 数据存储卷挂载没有按照配置进行挂载的原因?	14
4.3 自定义应用重启后报错“The module bind failed from cloud”?	15
4.4 在节点部署应用后, 节点由 “在线” 状态变为 “离线” 状态, 或者导致其他应用 “实例异常” ?	15
4.5 自定义应用报错 could not execute statement?	15
4.6 创建应用版本时, 如何添加自定义命令?	15
4.7 部署/升级应用出现“依赖的应用软件包没有部署”或“依赖的应用软件包版本不匹配”	16

5 账号与权限	17
5.1 在 IoT 边缘添加边缘设备或子设备，出现提示：“该用户没有权限 IOTDA.000022”	17
6 Module-SDK	18
6.1 Module-SDK 与 EdgeHub 如何选择对应版本?.....	18

1 概念相关

- 1.1 什么是IoT边缘?
- 1.2 开通边缘服务失败怎么处理?
- 1.3 IoT边缘如何进行设备管理?
- 1.4 IoTEdge与IEF的关系和区别是什么?

1.1 什么是 IoT 边缘?

IoT边缘是一组软件产品，包含云服务、边缘运行时软件、边缘模块应用，将云端能力快速拓展至边缘，提供数据采集、低时延自治、云边协同、边缘计算等能力，在园区、城市、工业等场景，作为数据源切入点，解决客户对设备上云、本地计算、数据预处理等诉求。

1.2 开通边缘服务失败怎么处理?

请检查开通边缘服务的账号是否为IAM用户。当前边缘服务不支持使用IAM用户（子账户），请使用主账号进行开通。

1.3 IoT 边缘如何进行设备管理?

IoT边缘服务和IoTDA设备接入服务协同提供统一的设备管理。当边缘节点与网络异常时，边缘本地控制台提供设备状态和数据查询等能力。

1.4 IoTEdge 与 IEF 的关系和区别是什么?

IoTEdge是一组软件产品，包含云服务、边缘运行时软件、边缘模块应用，将云端能力快速拓展至边缘，提供数据采集、低时延自治、云边协同、边缘计算等能力，在园区、城市、工业等场景，作为数据源切入点，解决客户对设备上云、本地计算、数据预处理等诉求。IoTEdge支持进程级应用部署、容器化应用部署、Helm应用模板部署，提供基础版、专业版、企业版，三种运行引擎。其中基础版可以部署在Linux系统或OpenHarmony系统的轻量级网关上，专业版可以部署在Docker环境的网关或服务上，企业版可以部署在边缘服务器集群上，自带Kubernetes。IEF基于开源框架

KubeEdge开发，提供容器化应用部署能力，具备Kubernetes的集群管理能力。
IoTEdge专业版兼容IEF部署底座。

2 节点管理

- 2.1 安装专业版节点异常如何处理？
- 2.2 安装专业版节点，执行安装命令时失败，一直提示“edgedaemon not found”？
- 2.3 安装专业版节点失败，提示“edge_core has existed”，该如何处理？
- 2.4 专业版节点所在机器断网/断电一段时间后恢复，节点模块一直处于“部署中”怎么处理？
- 2.5 专业版节点欠费充值后，部署、升级模块仍然报错？
- 2.6 MQTT设备无法接入边缘节点如何处理？专业版节点所在机器断网/断电一段时间后恢复，节点模块一直处于“部署中”怎么处理？
- 2.7 误删除IoTDA上的edge_node/modbus_server产品后，在边缘服务创建节点/添加Modbus设备失败，该怎么处理？
- 2.8 边缘节点离线排查指南
- 2.9 修改专业版（Docker）或基础版节点的云端日志级别，但没有生效？
- 2.10 ubuntu系统下节点shell命令兼容性导致的安装失败

2.1 安装专业版节点异常如何处理？

2.2.1 执行安装命令下载obs包失败

1. 域名解析失败，无法建立连接。请确认自己的主机的网络是否能够访问一些特定域名，检查域名服务器地址是否正确；若自身没有配置域名解析，或者不知道域名服务器地址，需要将下列用到的域名配置对应的hosts解析。

北京四需要验证的域名：

ioe-installer.obs.cn-north-4.myhwclouds.com;

swr.cn-north-4.myhuaweicloud.com;

iot-mqtts.cn-north-4.myhuaweicloud.com;

iotedge-south.cn-north-4.myhuaweicloud.com

2. 出现connected 说明网络没有问题，请求资源出现403错误，请确认当前执行的命令是否过期，在节点列表->选择安装，重新获取安装命令。

- 3. 出现404 说明资源不存在，需确认下载的安装包是否在对应的OBS中存在，请联系华为工程师。

```

root@iotlite-edge-node:~# cd /tmp && wget https://iot-installer.obs.cn-north-7.ulanqab.huawei.com/443/edgehuntime-0.3.072-20210111180508_x86_64/edgehuntime-0.3.072-20210111180508_x86_64.tar.gz
AccessKeyId=VEDM0M0WFTVBI8BL0M6eXpRes=161069445655signature=M0K1Wf2ckqWdXKXt8G0LTA30 -O edgehuntime_signed.tar.gz && tar -zxvf edgehuntime_signed.tar.gz && echo "----BEGIN PUBLIC KEY-----MIBIjAM8p8
p9A L0W9B4QEFAADQAGAM1BICgcARZAss1wV0p8ePKFmnd4S/0u1Lo1y0NcYwYvZFFMh3M3Wz0Q8B0C8K0q27FN7M4F0uSp00D0ueYrr/cv941Rt5r0u/NQ24W2L3h206jy0d0e0w0r4136G1Bh4N2JtL1xGAcQ0M6SpC0KE0E/P11ImNDKXrS30
0p18S0w0m0Lp000180W0s441g074W014113013K0m0m0u0u011d0s35k0y0t0e0y0p0a0s179W0u7172v0x01500p0w0y0s10K4W02030p0w0k0s0r0m0e0s0701h042060d0e0z0x06070u0m0w0t0a0b0----END PUBLIC KEY-----" > edge_in
staller_pub.pem && openssl dgst -sha256 -verify edge_installer_pub.pem -signature edgehuntime.tar.gz.sig edgehuntime.tar.gz && tar -zxvf edgehuntime.tar.gz && sh install.sh 100.94.60.201 8943 5474589800674220
0 sig edge_daemon host80000001G0P40070P0t0e0d070p0w0m0j0n0d0r0e070w0e0d0f0000427470d0322020827 cloud 66 cd
-2021-01-15 14:16:56 - https://iot-installer.obs.cn-north-7.ulanqab.huawei.com/edgehuntime-0.3.072-20210111180508_x86_64/edgehuntime-0.3.072-20210111180508_x86_64.tar.gz?AccessKeyId=VEDM0M0WFTV
BI8BL0M6eXpRes=161069445655signature=M0K1Wf2ckqWdXKXt8G0LTA30
Connecting to iot-installer.obs.cn-north-7.ulanqab.huawei.com [100.95.181.129]... 100.95.181.129
HTTP request sent, awaiting response... 200 OK
length: 27279194 [zstd] (application/gzip)
Saving to: 'edgehuntime_signed.tar.gz'

100%[=====] 27,279,194 100MB/s in 0.3s

2021-01-15 14:16:57 (100 MB/s) - "edgehuntime_signed.tar.gz" saved [27279194/27279194]
unable to load key file

```

2.2.2 执行安装脚本阶段，执行edgeInstaller失败。

- 1. 现象一：机器之前安装过边缘节点。

```

2022-10-14T15:28:33,485713 | INFO |
2022-10-14T15:28:33,492844 | INFO | (5/14) start verify duplicate install iot edge lite
2022-10-14T15:28:33,495273 | ERROR | IoT Edge Lite have already installed in the host, please uninstall first.
2022-10-14T15:28:33,502834 | ERROR | (5/14) install failed
/root
[root@ecs-tjj-01 ~]#

```

解决办法：执行卸载命令（sh /opt/IoTEdge-Installer/uninstall.sh），卸载之前安装过的节点后，再次执行安装命令。

- 2. 现象二：cannot execute binary file, 安装进程和机器架构不匹配，导致二进制进程包不能运行。

```

2022-10-14T15:32:39,856760 | INFO | (13/14) systemd enable and start edgedaemon success
2022-10-14T15:32:39,856604 | INFO |
2022-10-14T15:32:39,858486 | INFO | (14/14) backing up service file /lib/systemd/system/edgedaemon.service and environment file /lib/systemd/system/edgedaemon.env
2022-10-14T15:32:39,860372 | INFO | (14/14) not config backing up dir
2022-10-14T15:32:39,862873 | INFO | (14/14) backing up service file /lib/systemd/system/edgedaemon.service and environment file /lib/systemd/system/edgedaemon.env success
2022-10-14T15:32:39,864672 | INFO |
2022-10-14T15:32:39,866548 | INFO | run edgeInstaller...
edge_install.sh: line 723: /opt/IoTEdge/IoTEdge/edgeInstaller/edgeInstaller: cannot execute binary file
/root

```

导致此问题原因可能是：在创建节点后获取安装命令选择的架构与机器本身不匹配。

解决办法：

- 1. 执行：sh /opt/IoTEdge-Installer/uninstall.sh
- 2. 在节点列表->安装->选择和机器匹配的架构，重新获取安装命令，在机器上执行安装命令。

2.2.3 执行安装脚本阶段，注册节点失败，registerNode ERROR。

- 1. 现象一：注册节点时调用接口 connection time out.

```

2022-10-16T11:44:16,139327 | INFO | install | 366 | start install edge, target server:[iotedge-south.*****.com port:8943] nodeId:77*****56
2022-10-16T11:44:16,139319 | WARN | install | 371 | edgedaemon not found.
2022-10-16T11:44:21,139762 | INFO | SouthCloudService | 37 | SouthCloudService Create.
2022-10-16T11:44:21,139966 | INFO | DaemonService | 34 | DaemonService create.
2022-10-16T11:44:21,140210 | INFO | registerNode | 208 | registerNode nodeId:77*****56
2022-10-16T11:46:28,565366 | ERROR | doRequest | 178 | request url [s://south/02/7b2108fd8a0845248e626372d137294/edge-nodes/778980528939053056/bind
2022-10-16T11:46:28,565476 | ERROR | doRequest | 179 | Throw location unknown (consider using BOOST_THROW_EXCEPTION)
Dynamic exception type: boost::wrapexcept
std::exception: what: connect: Connection timed out
2022-10-16T11:46:28,569359 | INFO | install | 378 | install/uninstall edge BaseException thrown, /data/fuxi_ci_workspace/630ecba9db1f942f4b6663a4/code/common/httpclient/src/HttpClient.h(89): Thro
w in function HttpReponse HttpClient<Protocol>::doRequest(boost::string_ref, HttpHeadersMap&, boost::string_ref, boost::beast::http::verb) [with Protocol = boost::asio::ip::tcp; HttpReponse =
boost::beast::http::message&, boost::beast::http::basic_string_bodychar, boost::beast::http::basic_fieldsstd::allocatorchar >> ; boost::string_ref = boost::basic_string_refchar, std::
char_traitschar >; HttpHeadersMap = boost::unordered::unordered_mapstd::basic_stringchar, std::basic_stringchar >]
Dynamic exception type: ServiceException
std::exception: what: std::exception
[tag_errCode] = 100029
[tag_rspCode] = 500
[tag_exceptionMessage] = SYS_HTTP_CONNECT_FAILED

```

访问南向域名（iotedge-south.cn-north-4.myhuaweicloud.com）连接超时，可能原因：

- 安装边缘节点机器DNS服务器配置错误，域名解析失败，解决方案参照：2.2.1小节第一点。
- 边缘节点机器所在环境为内网环境无法访问公网ip，解决方法：在北京四购买ECS安装边缘节点。

- 2. 现象二：调用接口注册节点失败。

```

2022-10-16T11:31:22,282390 | INFO | registerNode | 208 | registerNode nodeId:77*****40
2022-10-16T11:31:22,340825 | ERROR | registerNode | 226 | code:404 body:{"error_code": "IoTEdge_02100305", "error_msg": "The edge node does not existed."}
2022-10-16T11:31:22,341401 | ERROR | start | 81 | install/uninstall edge BaseException thrown, /data/fuxi_ci_workspace/630ecba9db1f942f4b6663a4/code/app/edgeInstaller/SouthCloudService.cpp(228)
: Throw in function void SouthCloudService::registerNode(boost::string_ref, boost::string_ref, boost::string_ref)
Dynamic exception type: ServiceException
std::exception: what: std::exception
[tag_errCode] = 200015
[tag_rspCode] = 500
[tag_exceptionMessage] = service install south register fail.

```


根据error_code和error_msg确定失败原因:

- 节点不存在("error_code":"IoTEdge.02100305","error_msg":"The edge node does not existed.")

可能原因: 复制了错误的安装命令, 确认安装命令中的节点id是否在节点列表中存在。

- 验证码不正确("error_code":"IoTEdge.02100206","error_msg":"The verifyCode is not right.")

可能原因:

- 安装命令过期 (返回console节点列表, 节点状态为未安装)

解决方法:

- 1) 执行: sh /opt/IoTEdge-Installer/uninstall.sh
- 2) 在节点列表->安装->重新获取安装命令, 在机器上执行安装命令。

- 安装命令中验证码已经使用过了 (返回console节点列表, 节点状态为已安装/在线)

解决方法:

- 1) 机器执行: sh /opt/IoTEdge-Installer/uninstall.sh
- 2) console上删除原节点。
- 3) console上新建一个边缘节点->获取安装命令, 在机器上执行安装命令。

2.2.4 拉取镜像失败。

```
2021-01-18T02:13:34,903485 | INFO | pullImage | 252 | start pull_image, deploy type is edge_node
2021-01-18T02:13:34,903698 | DEBUG | handleModuleStatusChange | 63 | handleModuleStatusChange msg:{"modules":[{"module_id":"sys_ed
ner_state":"PENDING","update_time":1610698440842}]}
2021-01-18T02:13:35,111322 | ERROR | pullImage | 267 | code:500 body:{"errorCode":"200038","errorDesc":"SERVICE_INNER_ERROR"}
2021-01-18T02:13:35,111581 | ERROR | operator() | 32 | create module exception:/data/fuxi_ci_workspace/5feaa508c2ee9750da92c4ca/co
function void DaemonClientImpl::pullImage(boost::string_ref, boost::string_ref)
Dynamic exception type: ServiceException
std::exception:what: std::exception
[tag_errCode*] = 200033
[tag_rspCode*] = 500
[tag_exceptionMessage*] = SERVICE_PULL_IMAGE_FAIL
```

现象: pullImage 失败, 解决办法:

1. 执行命令`docker images`查看是否有成功拉取镜像。
2. 若未拉取镜像, 请尝试手动拉取一个公开镜像, 例如拉取系统应用\$edge-agent的镜像。

应用管理->搜索应用名称(edge_agent)->查看应用



选择应用版本->获取镜像地址。



在边缘节点机器上执行 `docker pull [imageurl]` ;

例如: `docker pull swr.cn-north-4.myhuaweicloud.com/huawei-iot-edge/edgeagent_x86_64:0.3.072.20220901191300`

3. 若拉取连接失败, 请自行检查网络是否正常, 域名(`swr.cn-north-4.myhuaweicloud.com`)是否可以解析成功。
4. 若拉取失败并提示异常如`x509: certificate has expired or is not yet valid`, 可能是边缘节点机器的本地系统时间错误导致。输入`date`命令确定机器本地时间是否正确; 若本地时间错误, 可尝试使用ntp同步系统时间(输入命令`ntp date cn.pool.ntp.org`)。

2.2.5 创建容器失败, connect default network error.

```
2022-10-14T09:58:31.245911 | INFO | createModule | 75 | createModule: name:svs:edge_agent
2022-10-14T09:58:31.629809 | ERROR | createModule | 85 | code:508 body:{"errorCode":"208013","errorDesc":"connect default network error","
2022-10-14T09:58:31.629895 | INFO | reportDeploymentCenter | 79 | reportDeploymentCenter:{"resource":"edgeagent_x86_64","location":"Atlas_36","event":"failed","event_severity":"Ma
30m:ErrorReason| error_code:508| error_msg:install module failed. |}
2022-10-14T09:58:31.955972 | INFO | reportNodeInstallEvent | 152 | report event: msg:{"events":[{"resource":"node_installation","location":"Atlas_36","event":"failed","event_severity":"Ma
jor","event_time":"20221014T095831Z","notify_data":{"host_tag":"DEFAULT","edge_node_id":"Atlas_36","error_message":"install module failed."}]}}
2022-10-14T09:58:32.227779 | INFO | stop | 483 | edgeInstaller stop.
```

可能原因: 机器不支持ipv6, 创建网桥失败。

解决方法: 手动创建ipv4网桥

1. `docker network create iot-edge-bridge --subnet=172.20.0.0/24 --gateway=172.20.0.1`
2. 机器执行: `sh /opt/IoTEdge-Installer/uninstall.sh`
3. console上删除原节点。
4. console上新建一个边缘节点->获取安装命令, 在机器上执行安装命令。

2.2 安装专业版节点, 执行安装命令时失败, 一直提示“edgedaemon not found”?

请检查安装节点的系统版本, 已知CentOS8会触发这个错误。原因是CentOS8缺少edge_daemon的依赖。推荐使用CentOS7.6。

2.3 安装专业版节点失败, 提示“edge_core has existed”, 该如何处理?

在安装专业版节点之前, 先执行卸载IEF节点的命令; 再重新执行安装专业版节点的流程。若仍出现安装失败并提示“edge_core has existed”, 需通过工单联系IEF技术专家协助定位。

2.4 专业版节点所在机器断网/断电一段时间后恢复，节点模块一直处于“部署中”怎么处理？

1. 在IEF控制台检查对应节点的状态是否为在线。
2. 若IEF节点状态为在线，检查IEF节点下容器应用的状态是否为“异常”；可“创建容器应用”查看是否能正常部署应用。
 - 若IEF节点状态为“故障”，或IEF节点无法创建容器应用，则需通过工单联系IEF技术专家协助定位。
 - 若IEF节点可以正常部署容器应用，且部署的应用可正常运行，则需通过工单联系IoTEdge技术专家协助定位。

2.5 专业版节点欠费充值后，部署、升级模块仍然报错？

专业版欠费之后，由于云端缓存的原因，需要等待两小时之后才能正常操作。

2.6 MQTT 设备无法接入边缘节点如何处理？专业版节点所在机器断网/断电一段时间后恢复，节点模块一直处于“部署中”怎么处理？

1. 检查MQTT(S)设备与边缘节点之间的网络是否互通，是否有取消限制边缘节点的7883端口开放（安全组/防火墙）；
2. 校验设备接入参数是否正确，边缘节点提供MQTT(S)接入的端口为7883。
3. 专业版的部署依赖于IEF服务，请检查IEF服务是否正常。

2.7 误删除 IoTDA 上的 edge_node/modbus_server 产品后，在边缘服务创建节点/添加 Modbus 设备失败，该怎么处理？

1. 在IoTDA控制台 -> 资源空间-> 新建资源空间；
2. 在IoT边缘服务控制台注册节点时，“绑定设备接入服务” -> “设备归属资源空间”选择新建的资源空间，即可在新资源空间下创建节点并添加Modbus设备。

2.8 边缘节点离线排查指南

问题描述

在IoTEdge边缘控制台的“边缘节点” > “节点管理页面”，查看节点状态，期望状态是在线，但实际状态为离线。

可能原因

1. 安装边缘节点的机器存在网络限制，包括但不限于安全组未放通、系统防火墙限制、路由器拦截等。
2. 安装边缘节点的机器底层网络不稳定，存在丢包情况。

排查思路

使用SSH工具以root用户登录边缘节点服务器后台系统，执行如下命令：

1. 执行 `cat /proc/sys/net/ipv4/ip_forward`。
若值等于0，则需要添加配置IP转发规则。配置方法可以参考[注册专业版节点步骤5](#)。
2. 对于专业版节点，执行 `docker inspect sys-edge-hub | grep cloud_mqtt_server_ip`，获取到设备接入的ip或域名；
对于基础版节点，首先执行 `ps -ef | grep edge_hub` 获取Edgehub的进程id，然后执行 `cat /proc/{进程id}/environ | tr '\0' '\n' | grep cloud_mqtt_server_ip` 获取到设备接入的ip或域名。
3. 执行 `curl -kv {ip/域名}:8883`，如果未显示Connected，需要检查机器是否存在网络限制。

图 2-1 获取 IP 或域名

```
root@Server-d8830a3f-8cd9-4723-b271-0d592fd8cc39:~# docker inspect sys-edge-hub | grep cloud_mqtt_server_ip
"cloud_mqtt_server_ip":iot-mqtts.cn-north-4.myhuaweicloud.com", cur这个获取到的ip或者域名
root@Server-d8830a3f-8cd9-4723-b271-0d592fd8cc39:~# curl -kv iot-mqtts.cn-north-4.myhuaweicloud.com:8883
* Rebuilt URL to: iot-mqtts.cn-north-4.myhuaweicloud.com:8883/
* Trying 121.36.42.100...
* TCP_NODELAY set
* Connected to iot-mqtts.cn-north-4.myhuaweicloud.com (121.36.42.100) port 8883 (#0)
> GET / HTTP/1.1
> Host: iot-mqtts.cn-north-4.myhuaweicloud.com:8883
> User-Agent: curl/7.58.0
> Accept: */*
* Empty reply from server
* Connection #0 to host iot-mqtts.cn-north-4.myhuaweicloud.com left intact
curl: (52) Empty reply from server
root@Server-d8830a3f-8cd9-4723-b271-0d592fd8cc39:~#
```

4. 执行 `ping {ip/域名}`，一分钟后中止，观察packet loss是否为0，如果不为0需要联系运营商查看网络不稳定的原因。

图 2-2 packet loss 值

```
root@Server-d8830a3f-8cd9-4723-b271-0d592fd8cc39:~# ping iot-mqtts.cn-north-4.myhuaweicloud.com
PING iot-mqtts.cn-north-4.myhuaweicloud.com (121.36.42.100) 56(84) bytes of data:
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=1 ttl=51 time=1.74 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=2 ttl=51 time=1.69 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=3 ttl=51 time=1.65 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=4 ttl=51 time=1.63 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=5 ttl=51 time=1.65 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=6 ttl=51 time=1.66 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=7 ttl=51 time=1.65 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=8 ttl=51 time=1.63 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=9 ttl=51 time=1.62 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=10 ttl=51 time=1.63 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=11 ttl=51 time=1.66 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=12 ttl=51 time=1.64 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=13 ttl=51 time=1.66 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=14 ttl=51 time=1.70 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=15 ttl=51 time=1.68 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=16 ttl=51 time=1.64 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=17 ttl=51 time=1.66 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=18 ttl=51 time=1.64 ms
64 bytes from ecs-121-36-42-100.compute.huaweicloud-dns.com (121.36.42.100): icmp_seq=19 ttl=51 time=1.67 ms
^C
--- iot-mqtts.cn-north-4.myhuaweicloud.com ping statistics ---
19 packets transmitted, 19 received, 0% packet loss, time 18026ms
rtt min/avg/max/mdev = 1.629/1.663/1.745/0.038 ms
root@Server-d8830a3f-8cd9-4723-b271-0d592fd8cc39:~#
```

2.9 修改专业版（ Docker ）或基础版节点的云端日志级别，但没有生效？

问题描述

对既有节点的“日志数据配置-云端日志级别”进行修改，例如：从on调整为error，期望由原先的上报所有日志改为只上报error级别及error以上级别的日志，在控制台上修改成功，但等待一段时间后，发现边侧仍在上报所有的日志，修改后的配置并没有生效。

可能原因

日志级别选项中的 trace, debug, info, warn, error, fatal 需要新版的\$edge_omagent应用才能支持（当前仅支持1-1-15-xxx及以上版本）。若节点上运行的是较低版本的\$edge_omagent应用，则修改新选项后不会生效，边侧的\$edge_omagent应用仍将以原有的配置运行。

处理措施

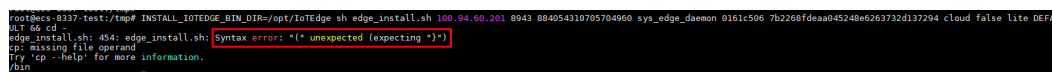
检查节点上运行的\$edge_omagent应用的版本，若低于1-1-15-xxx，则将其升级至新版本。

2.10 ubuntu 系统下节点 shell 命令兼容性导致的安装失败

问题描述

ubuntu系统下安装边缘节点失败，报错如下图所示，主要错误信息为：Syntax error: "(" unexpected (expecting "}")。

图 2-3 安装部署异常



```
root@ecs-8337-test:/tmp# INSTALL_IOTEDGE_BIN_DIR=/opt/IoTEdge sh edge_install.sh 100.94.60.201 8943 884054310705704960 sys_edge_daemon 0161c506 7b2268fdeaa045248e6263732d137294 cloud false lite DEFA
RT 66 cd
edge_install.sh: 454: edge_install.sh: Syntax error: "(" unexpected (expecting "")")"
cp: missing file operand
Try 'cp --help' for more information.
/bin
```

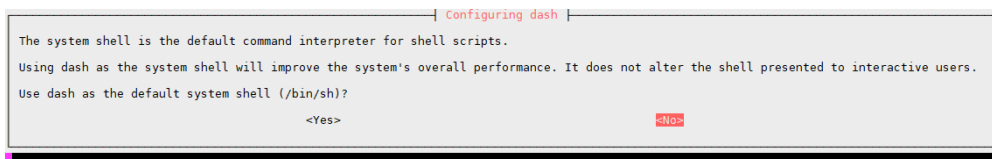
原因

出现该问题的原因为ubuntu系统下默认的shell为dash，安装脚本中使用的/bin/sh在ubuntu下默认指向/bin/dash，dash对于命令()不支持，所以导致安装执行失败。dash是不同于bash的shell，它主要是为了执行脚本而出现，而不是交互，它速度更快，但功能相比bash要少很多，语法严格遵守POSIX标准。

规避方法

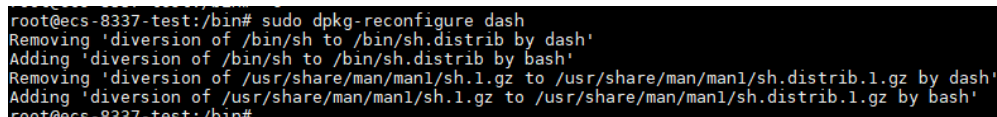
1. 修改ubuntu的默认shell，执行以下命令。
sudo dpkg-reconfigure dash
2. 交互界面中选择“No”。

图 2-4 修改 ubuntu 默认 sh



3. 执行完，出现如下提示表示已经切换为bash，如下图：

图 2-5 切换 bash 成功



3 子设备管理

- 3.1 边缘节点和子设备均在线，为什么子设备上报的数据在IoTDA平台查看不到？
- 3.2 Modbus设备与MQTT设备在数据上报流程有什么区别？
- 3.3 边缘节点断网离线后，为何其子设备仍然在线？
- 3.4 如何让边端的设备数据不上云？
- 3.5 集成ModuleSDK后，上报数据成功后，设备状态显示为未激活，如何上报子设备状态？
- 3.6 添加边缘设备的什么时候填写模块ID？

3.1 边缘节点和子设备均在线，为什么子设备上报的数据在IoTDA 平台查看不到？

1. 查看EdgeHub日志（专业版节点日志路径：`/var/loTEdge/log/sys_edge_hub/log/edge-hub-running.log`；专业版节点日志路径：`/var/IEF/app/log/sys_edge_hub/log/edge-hub-running.log`），触发子设备数据上报时，查看日志内容是否有异常信息。
2. 若日志提示“`No services`”，为该设备上报的数据与设备产品的`servicId`对应不上。当前边缘节点不感知产品的更新。若新建一个空白的设备产品，为边缘节点创建子设备，然后修改设备产品新增服务和属性，则此时设备上报数据中的服务属性在边缘节点无感知，即会出现“`No services`”的异常。
3. 当进行[基于IoT边缘实现模拟MQTT设备的快速接入](#)实践，添加边缘设备时，误操作填写了模块ID后，导致数据无法上报，删除边缘设备后重新添加。模块ID的设置代表设备接入点为该模块；空代表设备接入点为设备自身，相当于网关。

3.2 Modbus 设备与 MQTT 设备在数据上报流程有什么区别？

MQTT设备主动上报设备数据给边缘节点；Modbus设备数据上报由边缘节点周期性采集，采集周期由添加Modbus子设备时配置的“`period`”参数所指定。

3.3 边缘节点断网离线后，为何其子设备仍然在线？

在IoTDA平台上，边缘节点为直连设备，边缘节点的子设备为非直连设备。IoTDA只检测直连设备与平台的连接状态；子设备的状态由网关管理，需要网关调用修改子设备状态变更接口来更新子设备的状态。

所以当边缘节点突然断网离线时，未能更新子设备的状态。因此出现边缘节点离线，其子设备仍在线的情况。

3.4 如何让边端的设备数据不上云？

在边缘节点详情页 -> 模块管理 -> 数据流转配置，不配置消息目标为“云端”的数据流转规则即可。

为防止设备数据丢失，建议检查是否已开启边缘节点的数据存储功能，或者通过配置数据流转规则将数据流转到自定义应用进行存储。

3.5 集成 ModuleSDK 后，上报数据成功后，设备状态显示为未激活，如何上报子设备状态？

集成ModuleSDK后，目前边缘非直连设备，不能动态获取设备状态，只能根据自身业务是否正常，主动上报设备状态，来更新设备状态。

代码解析：

片段一：初始化构造函数，初始化设备状态数据。

```
public class ModbusDriver implements GatewayCallback {  
    /**  
     * 驱动客户端，与边缘Hub建立MQTT连接  
     */  
    private DriverClient driverClient;  
  
    ScheduledExecutorService scheduler = Executors.newScheduledThreadPool(1);  
    //ONLINE:设备在线。  
    //OFFLINE:设备离线。  
    //ABNORMAL:设备异常。  
    //INACTIVE:设备未激活。  
    //FROZEN:设备冻结。  
    static HashMap<String, String> dataMap = new HashMap() {  
        private static final long serialVersionUID = 821550418990759366L;  
  
        {  
            this.put("0", "ONLINE");//在线  
            this.put("1", "OFFLINE");//离线  
            this.put("2", "ABNORMAL");//异常  
            this.put("3", "INACTIVE");//未激活  
            this.put("4", "FROZEN");//冻结  
        }  
    };  
    public ModbusDriver() throws Exception {  
        this.driverClient = DriverClient.createFromEnv();  
    }  
}
```

片段二：调用上报子设备状态API，循环更新子设备状态。（可根据自身业务调整，此代码只用于演示功能）


```
public static void main(String[] args) throws Exception {
    ModbusDriver modbusDriver = new ModbusDriver();
    modbusDriver.start();
}

public void start() throws Exception {
    // 设置回调, 打开客户端
    this.driverClient.setGatewayCallback(this);
    this.driverClient.open();
    this.scheduler.scheduleAtFixedRate(new Runnable() {
        @Override
        public void run() {
            UpdateSubDevicesStatusEvent updateSubDevicesStatusEvent = new
UpdateSubDevicesStatusEvent();
            DeviceStatus deviceStatus = new DeviceStatus();
            //设备Id (添加边缘设备时设置)
            deviceStatus.setDeviceId("scmj19961031b");
            int j = new Random().nextInt(4);
            //随机获取设备状态上报
            deviceStatus.setStatus(ModbusDriver.dataMap.get(String.valueOf(j)));
            updateSubDevicesStatusEvent.setDeviceStatuses(Collections.singletonList(deviceStatus));
            try {
                ModbusDriver.this.driverClient.updateSubDevicesStatus(String.valueOf(new
Random().nextInt(100000)), updateSubDevicesStatusEvent);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }, 0, 5, TimeUnit.SECONDS);
}
```

3.6 添加边缘设备的什么时候填写模块 ID?

默认情况下不填写模块ID, 当应用集成了ModuleSDK后, 需配置流转规则, 添加边缘设备需填写添加第三方应用的模块id, 数据才能进行上报到平台。

4 应用管理

- 4.1 在边缘节点下部署自定义应用，实例异常如何处理？
- 4.2 部署自定义应用时，数据存储卷挂载没有按照配置进行挂载的原因？
- 4.3 自定义应用重启后报错"The module bind failed from cloud"?
- 4.4 在节点部署应用后，节点由“在线”状态变为“离线”状态，或者导致其他应用“实例异常”？
- 4.5 自定义应用报错could not execute statement?
- 4.6 创建应用版本时，如何添加自定义命令？
- 4.7 部署/升级应用出现"依赖的应用软件包没有部署"或"依赖的应用软件包版本不匹配"

4.1 在边缘节点下部署自定义应用，实例异常如何处理？

1. 检查自定义应用所使用的SWR镜像是否设置为“公开”（在云服务“SWR” --> 我的镜像 --> (选中镜像)镜像详情中查看镜像类型为“私有/公开”，通过编辑按钮可更改类型）。若设置为“私有”，需将类型修改为“公开”。
2. 在边缘节点上查看是否有自定义容器运行(执行命令`docker ps -a`)，若容器为异常退出，需检查自定义应用模块的配置信息是否正确，如检查容器镜像/运行命令是否正确，容器规格申请配置是否过大等。
3. 若自定义的容器没运行，查看是否已成功拉取镜像(执行命令`docker images`)；若镜像没拉取到边端，需检查边缘节点的网络状况是否稳定。

4.2 部署自定义应用时，数据存储卷挂载没有按照配置进行挂载的原因？

检查自定义应用版本中的数据存储类型，若类型为LOG/DB/CONFIG时，会给主机目录默认添加前缀："/var/loTEdge/{log|db|config}"。

若需要将容器目录直接挂载到主机某个目录，本地卷的类型需选择为“其他”。

4.3 自定义应用重启后报错"The module bind failed from cloud"?

请在添加应用时配置CONFIG存储卷。未配置CONFIG卷，应用重启后丢失配置会导致此项错误。

4.4 在节点部署应用后，节点由“在线”状态变为“离线”状态，或者导致其他应用“实例异常”？

请检查节点内存占用情况，当节点内存不足时会导致docker容器被停止。\$edge_hub容器停止会导致节点“离线”，其他容器停止会导致“实例异常”。

4.5 自定义应用报错 could not execute statement?

请检查部署应用配置的LOG/DB/CONFIG挂载磁盘是否容量充足。当磁盘已满将只能读取，无法存储，从而导致导致边缘组件报错。

4.6 创建应用版本时，如何添加自定义命令？

方法一：

1. 通过docker run 命令将应用容器运行起来；

```
docker run [OPTIONS] IMAGE [COMMAND] [ARG...]
```

2. 将[COMMAND]填入到运行命令处，[ARG...]填入到命令参数中，参数有多个时，参考输入框中的提示，以换行进行分隔。

3. [OPTIONS] 中相关配置根据具体配置内容在高级配置中选填。

方法二：

1. 通过docker run 命令将应用容器运行起来；

2. 使用docker ps -a 查看运行的应用容器ID；

3. 通过docker inspect [CONTAINER ID]查看容器详细信息；将Cmd中的内容填至参数列表，将Entrypoint中的内容填至运行命令处，为null则不填

```
    ],  
    "Cmd": [  
      "/opt/iot/edge_rule/scripts/start.sh",  
      "run"  
    ],  
    "ArgsEscaped": true,  
    "Image": "iot-edge-agent",  
    "Volumes": {  
      "/opt/iot/edgeDaemon/socket/": {},  
      "/opt/iot/edge_rule/db": {},  
      "/opt/iot/edge_rule/logs": {},  
      "/opt/iot/edge_rule/secret": {}  
    },  
    "WorkingDir": "",  
    "Entrypoint": null,  
    "OnBuild": null,  
    "Labels": {  
      "MAINTAINER": "= EDGE",  
      "iot-edge-module-label": "iot-edge-module-label"  
    }  
  }  
}
```

将应用版本部署到边缘之后如果出现容器无法运行、退出等问题，需要保证其他的配置正确。（如配置了挂载卷，但是没有设置访问权限）。

4.7 部署/升级应用出现"依赖的应用软件包没有部署"或"依赖的应用软件包版本不匹配"

部署/升级应用时，若出现"依赖的应用软件包没有部署"或"依赖的应用软件包版本不匹配"问题，请检查节点是否部署\$edge_agent应用并且\$edge_agent已经升级到最新版本。

5 账号与权限

5.1 在IoT边缘添加边缘设备或子设备，出现提示：“该用户没有权限 IOTDA.000022”

5.1 在 IoT 边缘添加边缘设备或子设备，出现提示：“该用户没有权限 IOTDA.000022”

创建委托

1. 进入控制台，单击“用户名 > 统一身份认证”，进入统一身份认证服务控制台。
2. 在左侧导航栏选择“委托”，单击页面右上角“创建委托”。
3. 填写参数信息。
 - 委托名称：“oce_admin_trust”
 - 委托类型：“云服务”
 - 云服务：IoTEdge
 - 持续时间：永久
4. 单击“下一步”。

添加策略

5. 搜索“Tenant Administrator”并勾选，单击“下一步”。

设置最小授权范围

6. 选择指定区域项目资源，勾选“cn-north-4 [华北-北京四]”。
7. 单击“确定”，完成委托创建。

6 Module-SDK

6.1 Module-SDK与EdgeHub如何选择对应版本?

6.1 Module-SDK 与 EdgeHub 如何选择对应版本?

IoT边缘云服务，专业版和基础版大于1-1-22版本的EdgeHub，不再支持集成Java SDK小于2.1.2版本的应用、不再支持集成C SDK编译日期小于2021-07-07 14:32:56版本的应用，专业版不受影响。