

IoT 边缘

# 用户指南

文档版本 01  
发布日期 2025-01-09



版权所有 © 华为云计算技术有限公司 2025。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云计算技术有限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，华为云计算技术有限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

<b>1 从这里开始</b>	<b>1</b>
1.1 服务控制台总览	1
<b>2 管理边缘节点</b>	<b>3</b>
2.1 注册基础版节点	3
2.1.1 注册基础版节点（Linux 系统）	3
2.1.2 注册基础版节点（鸿蒙系统）	7
2.2 注册专业版节点	15
2.2.1 注册专业版节点（Docker）	15
2.2.2 注册专业版节点（KubeEdge）	22
2.3 节点概览	29
2.4 应用模块	31
2.5 边缘设备	38
2.6 数采配置	41
2.6.1 架构介绍	41
2.6.2 配置流程	43
2.6.3 点位计算	43
2.6.4 最佳实践	48
2.7 数据配置	48
2.7.1 设备数据打印配置	49
2.8 主备配置	52
2.9 远程维护	65
2.10 本地自治	72
2.11 删除边缘节点	72
2.12 重装边缘节点	73
2.13 启用 TPM 硬件加密	76
<b>3 应用管理</b>	<b>80</b>
3.1 应用管理	80
3.1.1 应用管理概述	80
3.1.2 添加业务应用	81
3.1.3 添加驱动应用	91
3.1.4 添加版本	99
3.1.5 部署应用	100

3.1.6 管理应用.....	102
3.1.7 删除应用.....	105
<b>4 设备接入边缘节点.....</b>	<b>107</b>
4.1 接入模式介绍.....	107
4.2 透传网关模式接入.....	109
<b>5 路由转发.....</b>	<b>115</b>
5.1 路由转发概述.....	115
5.2 通道类型概述.....	116
5.2.1 MQTT 类型通道.....	116
5.2.2 IoTDB 类型通道.....	117
5.2.3 InfluxDB V2 类型通道.....	119
5.3 创建通道.....	120
5.4 节点部署 EdgePush 推送应用.....	123
5.5 通道分配.....	125
5.5.1 MQTT 类型通道分配至节点.....	125
5.5.2 IoTDB 类型通道分配至节点.....	126
5.5.3 InfluxDB V2 类型通道分配至节点.....	128
5.6 修改节点的通道详情（可选）.....	129
5.7 将节点的通道信息下发给网关.....	131
<b>6 审计.....</b>	<b>133</b>

# 1 从这里开始

IoT边缘（IoT Edge），是边缘计算在物联网行业的应用。IoT Edge作为物联网边缘“小脑”，在靠近物或数据源头的边缘侧，融合网络、计算、存储、应用核心能力的开放平台，就近提供计算和智能服务，满足行业在实时业务、应用智能、安全与隐私保护等方面的基本需求。

本文介绍用户使用指导。

## 1.1 服务控制台总览

IoT边缘的云端管理界面，支持边缘节点管理、应用管理、数采配置、路由管理、监控运维等。

- 边缘节点管理：网关、服务器、虚拟机，只要满足边缘**硬件规格**要求，就可以部署IoT边缘运行时软件，实现对硬件的远程纳管、监控。
- 边缘应用管理：IoT边缘节点的云端管理界面，支持容器、进程应用一键部署、升级、卸载等。
- 边缘数据采集：在分配的边缘节点上部署数采应用，通过数采配置，下发至边缘节点执行数据采集。
- 边缘数据转发：路由转发配置下发至分配的边缘节点上执行，节点将采集到的数据进行转发。

图 1-1 总览



# 2 管理边缘节点

## 2.1 注册基础版节点

### 2.1.1 注册基础版节点（Linux 系统）

基础版节点的“系统应用”采用进程包部署方式。适用于Linux Core系统，无Docker环境的工业网关。

#### 注册基础版节点

- 步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。
- 步骤2** 选择左侧导航栏“边缘节点 > 节点管理”单击右上角“注册节点”。
- 步骤3** 根据页面提示填写参数，勾选右下角“《华为云服务等级协议》”单击“下一步”配置节点数据。

图 2-1 基础版

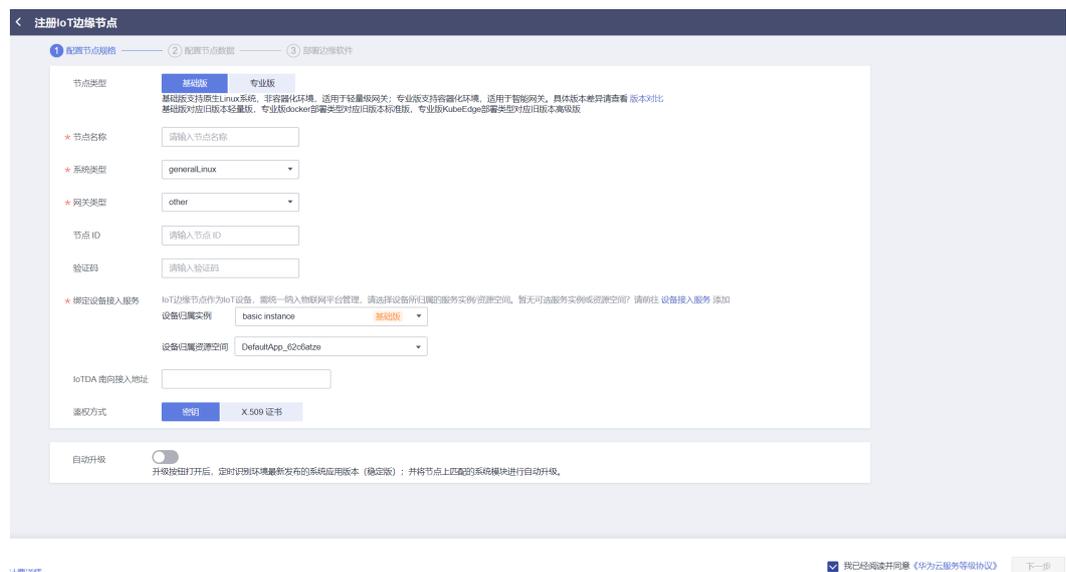


表 2-1 节点规格参数说明

参数名称	参数说明
节点类型	基础版
节点名称	自定义节点名称，如“test”。 <b>说明</b> 节点名称可以在：“节点概览”-“基本信息”中编辑修改。
AI加速卡	<ul style="list-style-type: none"> <li>未搭载：边缘节点未使用AI加速卡时选择。</li> <li>昇腾AI加速卡：支持昇腾AI加速设备（搭载Ascend 310，Ascend 310 Pro处理器的昇腾硬件），边缘节点的服务器搭载相关昇腾AI加速设备时选择。</li> </ul>
昇腾驱动地址	可选，AI加速卡选择昇腾AI加速卡时需要保证安装边缘节点的服务器上已经安装昇腾设备驱动，并根据前端提示填入驱动地址。若该值为空，默认填入Atlas300驱动默认安装地址：/usr/local/Ascend/driver/lib64/。
系统类型	generalLinux。
网关类型	可选。
节点ID	自定义（可选填写）。只允许中文，英文字母、数字、中划线（-），下划线（_），最小长度1，最大长度64。 <b>说明</b> 边缘节点ID自定义(不输入默认生成)。
验证码	输入自定义验证码（可选填写）。验证码，执行安装边缘软件命令时与云端通讯的凭证。
设备归属实例	选择默认归属的IoTDA实例。
设备归属资源空间	选择默认资源空间，或为本实验单独新建一个资源空间，如edge。
IoTDA南向接入地址	选择网关归属的IoTDA实例的南向设备mqtt接入地址，也可以选择自定义的南向接入地址，支持“ip:端口”或“域名:端口”形式。
鉴权方式	密钥：默认生成。 X.509证书：请参考： <a href="#">基于MQTT.fx的X.509证书接入指导</a> 。 <b>警告</b> 鉴权方式为“密钥”的节点，请勿在IoTDA界面重置节点的密钥，重置密钥会导致网关与IoTDA通信异常，且不可恢复。
指纹	证书文件生成的字符串（可选填写）。只能为40位十六进制字符串或64位六进制字符串。
本地路径	输入证书文件和密钥文件在设备上的存储路径，只允许字母、数字、下划线（_）和斜线（/），以/开头并且以.pem或.crt结尾。
自动升级	可选，开启后节点将自动升级系统应用模块。 <b>注意</b> 升级过程中可能导致节点数据上传中断。

步骤4 填写配置节点数据信息后，单击“立即创建”。

图 2-2 配置节点数据

The screenshot shows the '配置节点数据' (Configure Node Data) step in the IoT Edge node registration process. It includes sections for:

- 数据存储路径** (Data Storage Paths): Fields for node data storage root directory (/var/loTEdge/db/), node configuration root directory (/var/loTEdge/config/), and node log root directory (/var/loTEdge/log/).
- 监控运维** (Monitoring and Maintenance): A checked checkbox for '自动部署\$edge\_omagent应用' (Automatically deploy \$edge\_omagent application). Below are log settings: '系统日志' (System logs) and '应用日志' (Application logs) tabs, '云端日志级别' (Cloud log level) set to 'off', '日志文件大小' (Log file size) set to 50, '滚动日志周期' (Log rotation cycle) set to 'Daily', and '滚动日志数量' (Log rotation count) set to 5.
- 离线缓存配置** (Offline Cache Configuration): '上报优先级' (Report priority) set to '实时数据优先' (Real-time data priority), '数据过期时间' (Data expiration time) set to '自定义过期时间' (Custom expiration time) with a value of 7 days (max 14 days), and '缓存大小' (Cache size) set to 2,048 (max 85536M).
- 可靠性配置** (Reliability Configuration): '可靠性级别' (Reliability level) set to '中' (Medium).
- 设备数据格式配置** (Device Data Format Configuration): '设备数据格式' (Device data format) set to 'loTDA1.0规范' (loTDA1.0 specification).

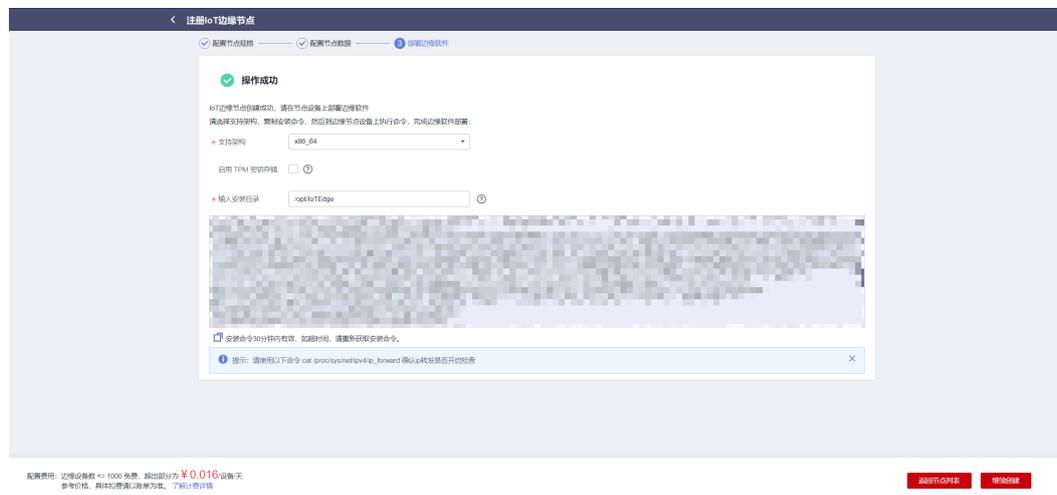
表 2-2 专业版节点信息

参数名称	说明
数据存储路径	默认（允许修改存储路径）。
监控运维	默认勾选“监控运维工具”。
\$edge_omagent应用	负责边缘节点远程监控和运维，提供日志上报、远程SSH、CPU等指标上报、文件上传下载能力。

参数名称	说明
日志配置	<p>系统日志：边缘节点上已部署的系统应用产生的日志。</p> <p>应用日志：边缘节点上已部署的用户定义应用所产生的日志。</p> <ul style="list-style-type: none"> <li>云端日志级别：on, off, trace, debug, info, warn, error, fatal。其中，on表示上传所有日志至云端，off表示不上传日志至云端，其余可选项代表按日志级别采集，选择后将只采集目标级别及以上的日志（模糊匹配）。日志级别的严重程度等级由低到高依次为：trace, debug, info, warn, error, fatal。</li> <li>日志文件大小：日志文件大小限制，单位MB，默认50，取值范围10-1000。某个日志文件如果达到大小限制，则会将文件转储保存至日志目录下。</li> <li>滚动日志周期：日志转储周期，可选项：daily, weekly, monthly, yearly。某个日志文件如果达到转储周期，则会将文件转储保存至日志目录下。日志文件大小和滚动日志周期是同时生效的，满足任何一个条件都会进行日志转储。</li> <li>滚动日志数量：日志文件转储个数，默认5，取值范围1-10。边缘节点保存的转储日志数量如果达到限制，则会删除最早的转储文件。</li> </ul> <p><b>注意</b></p> <ol style="list-style-type: none"> <li>当前“云端日志级别”对日志等级采取模糊匹配的策略，若日志消息中存在与日志级别标识相同的关键字，则也可能被采集。</li> <li>若要开启“云端日志级别”中的trace, debug, info, warn, error, fatal选项，则必须满足以下三个前置条件： <ol style="list-style-type: none"> <li>目标节点是基础版节点或部署类型为Docker的专业版节点。需注意，部署类型为KubeEdge的专业版节点不支持此功能。</li> <li>已安装\$edge_omagen应用，且版本不低于1-1-15-xxx。</li> <li>应用日志需遵循如下规范打印，以便进行过滤： <ul style="list-style-type: none"> <li>日志级别标识需打印为（大小写均可）：trace、debug、info、warn、error、fatal。</li> <li>每条日志消息以YYYY-MM-DD日期格式作为前缀。</li> </ul> </li> </ol> </li> </ol>
离线缓存配置	<p>可对离线节点上报的数据进行缓存，定义缓存周期（可配置为不过期）、缓存大小、上报的优先级、路由转发是否开启离线缓存。仅当上报优先级为“按时间顺序”时支持路由转发开启离线缓存。</p>
可靠性配置	<p>可靠性级别分为“中”和“高”2种。中级别：当应用上报属性和点位的频率超过流控值时，超过部分会被丢弃；高级别：当应用上报属性和点位的频率超过流控值时，通过控制模块的发送速度，降低上报频率，报文不会被丢弃。（注意：高可靠性级别的配置需要定制应用的配合，同时会造成性能的损失）。</p>
设备数据格式配置	<p>设备的数据格式：即单网关节点中设备的标准数据格式，包含IoTDA1.0物模型格式和属性平铺格式（无特殊要求建议选择IoTDA1.0物模型格式）。IoTDA1.0物模型中包含了设备、服务和属性三层结构，具体参考IoTDA子网关属性上报报文格式。属性平铺格式，包含设备、属性2层结构，属性结构可以包含自定义结构的拼接，在json中作为一项平铺展开，具体格式请参考<a href="#">设备数据格式配置</a></p> <p><b>须知</b></p> <p>属性平铺格式功能，需要edge_hub应用1-1-38-xx版本及以上支持</p>

**步骤5** 弹出如下对话框，您可以“返回节点列表”也可以“继续创建”。

图 2-3 基础版注册成功



---结束

## 安装基础版节点

安装“基础版节点”与安装“专业版节点”保持一致，请参照“注册专业版节点”。

### 2.1.2 注册基础版节点（鸿蒙系统）

鸿蒙系统下当前仅支持基础版节点的安装，基础版节点的“系统应用”采用进程包部署方式。适用于Linux Core系统，无Docker环境。以下步骤均使用基于瑞芯微 RK3568 芯片的DAYU200开发板进行节点安装测试。

#### 说明

- 在注册前请检查机器网络是否正常，是否能够连接到公网。
- 在注册之前请检查机器时间是否和北京时间一致，如果偏差过大会导致安装失败。

## 注册基础版节点

**步骤1** 访问[IoT边缘](#)，单击“**管理控制台**”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 节点管理”单击右上角“注册节点”。

**步骤3** 节点类型选择“基础版”，系统类型选择“openHarmony”。

**步骤4** 根据页面提示填写参数，勾选右下角“《华为云服务等级协议》”单击“下一步”配置节点数据。

图 2-4 基础版

注册IoT边缘节点

1 配置节点规格 2 配置节点数据 3 部署边缘软件

节点类型 **基础版** 专业版  
基础版支持原生Linux系统，非麒麟化环境，适用于轻量级网关；专业版支持麒麟化环境，适用于智能网关，具体版本差异请查看 版本对比。  
基础版为默认版本且免费，专业版openEuler等类型均适用版本标准，专业版HuaweiEdge等类型均适用版本高级版。

节点名称 请输入节点名称

系统类型 openHarmony

网关类型 other

节点ID 请输入节点ID

验证码 请输入验证码

绑定设备接入服务 IoT边缘节点作为IoT设备，需统一接入物联网平台管理，请选择设备所归属的服务实例资源空间，若无可选服务实例或资源空间？请前往 设备接入服务 添加。  
设置归属实例 hvstest\_pub\_1otEdge\_1ot IoTDA Instance

设置归属资源空间 hlx\_0413

IoTDA南向接入地址

鉴权方式 **密钥** X.509证书

自动升级  升级按钮打开后，定时识别环境最新发布的系统应用版本（稳定版），并将节点上匹配的系统模块进行自动升级。

部分为¥0.016/设备/天  
准，了解计费详情

我已阅读并同意《华为云服务协议》 [下一步](#)

表 2-3 节点规格参数说明

参数名称	参数说明
节点类型	基础版。
节点名称	自定义节点名称，如“test”。 <b>说明</b> 节点名称可以在：“节点概览”-“基本信息”中编辑修改。
网关类型	可选。
系统类型	generalLinux（标识普通系统，如ubuntu、centos、openEuler），openHarmony（鸿蒙系统）。如果系统为鸿蒙时必须选择openHarmony。
节点ID	自定义（可选填写）。只允许中文，英文字母、数字、中划线（-），下划线（_），最小长度1，最大长度64。 <b>说明</b> 边缘节点ID自定义(不输入默认生成)。
验证码	输入自定义验证码（可选填写）。验证码，执行安装边缘软件命令时与云端通讯的凭证。
设备归属实例	选择默认归属实例。
设备归属资源空间	选择默认资源空间，或为本实验单独新建一个资源空间，如edge。
IoTDA南向接入地址	选择网关归属的IoTDA实例的南向设备mqtt接入地址，也可以选择自定义的南向接入地址，支持“ip:端口”或“域名:端口”形式。

参数名称	参数说明
鉴权方式	密钥：默认生成。 X.509证书：请参考： <a href="#">基于MQTT.fx的X.509证书接入指导</a> 。 <b>警告</b> 鉴权方式为“密钥”的节点，请勿在IoTDA界面重置节点的密钥，重置密钥会导致网关与IoTDA通信异常，且不可恢复。
指纹	证书文件生成的字符串（可选填写）。只能为40位十六进制字符串或64位六进制字符串。
本地路径	输入证书文件和密钥文件在设备上的存储路径，只允许字母、数字、下划线（_）和斜线（/），以/开头并且以.pem或.crt结尾。
自动升级	可选，开启后节点将自动升级系统应用模块。 <b>注意</b> 升级过程中可能导致节点数据上传中断。

**步骤5** 填写配置节点数据信息后，鸿蒙系统因为默认根目录为只读权限，避免数据读写问题，建议节点的根目录配置在/data目录下，单击“立即创建”。

图 2-5 配置节点数据

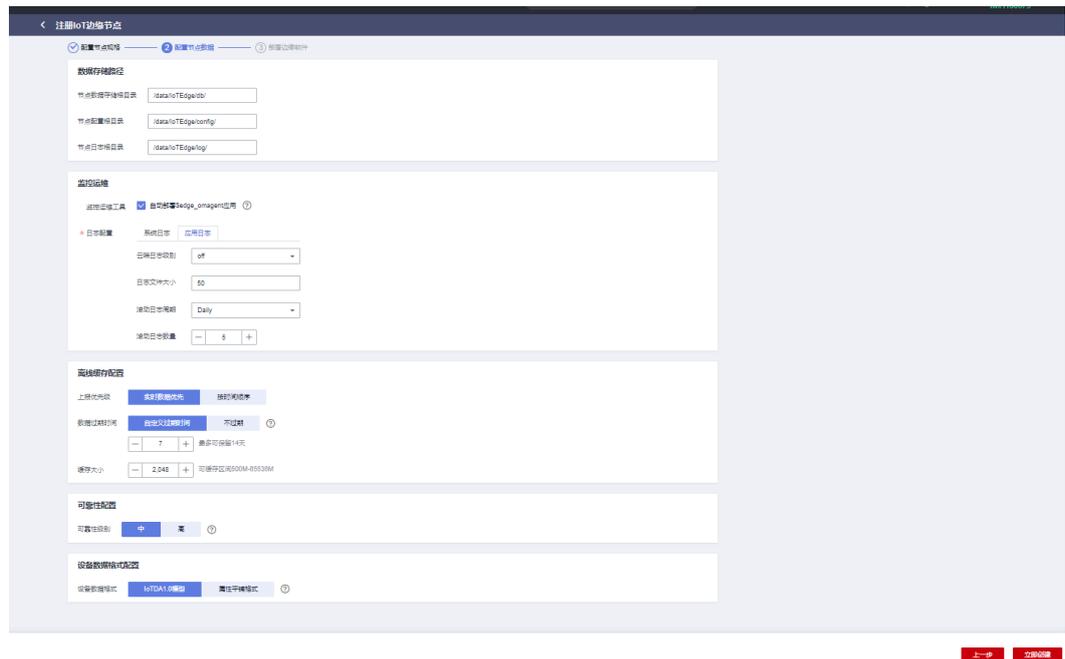


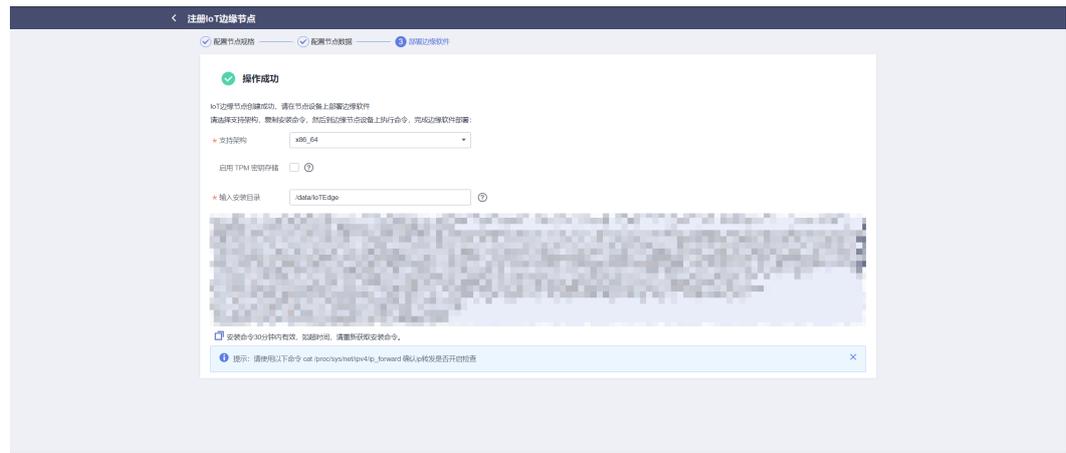
表 2-4 专业版节点信息

参数名称	说明
数据存储路径	默认（允许修改存储路径），鸿蒙系统因为默认根目录为只读权限，避免数据读写问题，建议节点的根目录配置在/data/或子目录下。
监控运维	默认勾选“监控运维工具”。

参数名称	说明
\$edge_omagent应用	负责边缘节点远程监控和运维，提供日志上报、远程SSH（鸿蒙下无此功能）、CPU等指标上报、文件上传下载能力。
日志配置	<p>系统日志：边缘节点上已部署的系统应用产生的日志。</p> <p>应用日志：边缘节点上已部署的用户定义应用所产生的日志。</p> <ul style="list-style-type: none"> <li>• 云端日志级别：on, off, trace, debug, info, warn, error, fatal。其中，on表示上传所有日志至云端，off表示不上传日志至云端，其余可选项代表按日志级别采集，选择后将只采集目标级别及以上的日志（模糊匹配）。日志级别的严重程度等级由低到高依次为：trace, debug, info, warn, error, fatal。</li> <li>• 日志文件大小：日志文件大小限制，单位MB，默认50，取值范围10-1000。某个日志文件如果达到大小限制，则会将文件转储保存至日志目录下。</li> <li>• 滚动日志周期：日志转储周期，可选项：daily, weekly, monthly, yearly。某个日志文件如果达到转储周期，则会将文件转储保存至日志目录下。日志文件大小和滚动日志周期是同时生效的，满足任何一个条件都会进行日志转储。</li> <li>• 滚动日志数量：日志文件转储个数，默认5，取值范围1-10。边缘节点保存的转储日志数量如果达到限制，则会删除最早的转储文件。</li> </ul> <p><b>注意</b></p> <ol style="list-style-type: none"> <li>1. 当前“云端日志级别”对日志等级采取模糊匹配的策略，若日志消息中存在与日志级别标识相同的关键字，则也可能被采集。</li> <li>2. 若要开启“云端日志级别”中的trace, debug, info, warn, error, fatal选项，则必须满足以下三个前置条件： <ol style="list-style-type: none"> <li>1. 目标节点是基础版节点或部署类型为Docker的专业版节点。需注意，部署类型为KubeEdge的专业版节点不支持此功能。</li> <li>2. 已安装\$edge_omagent应用，且版本不低于1-1-15-xxx。</li> <li>3. 应用日志需遵循如下规范打印，以便进行过滤： <ul style="list-style-type: none"> <li>○ 日志级别标识需打印为（大小写均可）：trace、debug、info、warn、error、fatal。</li> <li>○ 每条日志消息以YYYY-MM-DD日期格式作为前缀。</li> </ul> </li> </ol> </li> <li>3. 专业版（Docker）和基础版节点只支持云端日志级别这一个参数，其他参数仅对专业版（KubeEdge）有效。</li> </ol>
离线缓存配置	可对离线节点上报的数据进行缓存，定义缓存周期（可配置为不超过期）跟缓存大小，上报的优先级，路由转发是否开启离线缓存。仅当上报优先级为“按时间顺序”时支持路由转发开启离线缓存。
可靠性配置	可靠性级别分为“中”和“高”2种。中级别：当应用上报属性和点位的频率超过流控值时，超过部分会被丢弃；高级别：当应用上报属性和点位的频率超过流控值时，通过控制模块的发送速度，降低上报频率，报文不会被丢弃。（注意：高可靠性级别的配置需要定制应用的配合，同时会造成性能的损失）。

**步骤6** 弹出如下对话框，您可以“返回节点列表”也可以“继续创建”。

图 2-6 鸿蒙系统基础版注册成功



**注意**

1. 架构选择时, 需要根据硬件架构选择, 此处使用基于RK3568的DAYU200开发板选择arm64, 其余的架构按照实际选择。
2. 鸿蒙系统安装目录需要修改到/data/目录或者子目录下, 避免重启后系统目录变更为只读权限文件系统, 影响系统正常运行。

----结束

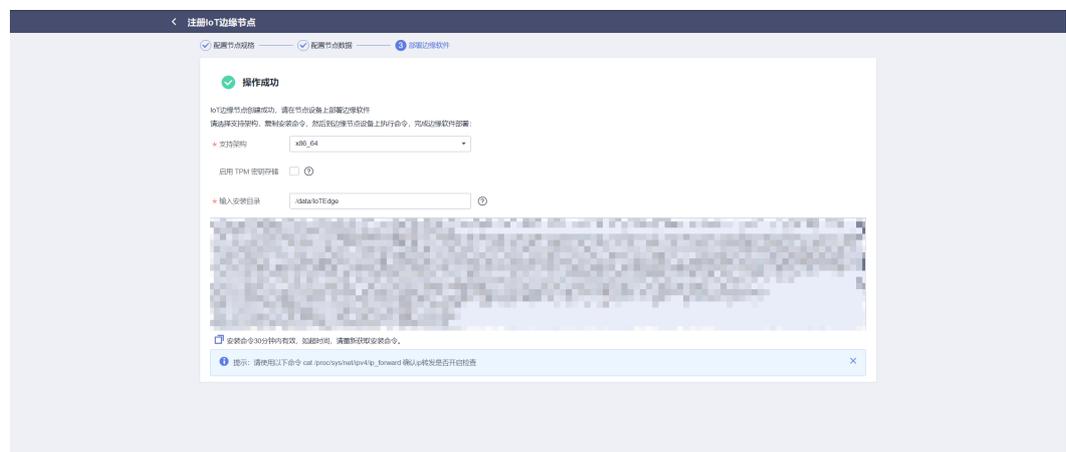
## 安装节点

**说明**

- 在安装前请检查机器网络是否正常, 是否能够连接到公网。
- 在安装之前请检查机器时间是否和北京时间一致, 如果偏差过大会导致安装失败。

**步骤1** 选择支持架构和安装目录之后, 按照图中框选地方获取。

图 2-7 获取安装命令



**步骤2** 鸿蒙操作系统下默认未安装wget以及openssl，因此需要将安装命令拆解为以下几步执行。如图为安装命令的拆分图解

图 2-8 命令拆解

```
cd /tmp && wget "  
https://ioe-installer-cn-north-4.obs.cn-north-4.myhuaweicloud.com/443/edgeRuntime-2022.1222.1216.38-openharmony.tar.gz/1671689782293/edgeRuntime-2022.1222.1216.38-openharmony.tar.gz?AccessKeyId=VGCOR2QMOJGSGU9IM4CE&Expires=1673949121&Signature=pwxh7Zd50k2QpBXSw8R9c1e9pF33D" -O edgeRuntime_signed.tar.gz && tar -zxf edgeRuntime_signed.tar.gz ①  
&& echo "-----BEGIN PUBLIC KEY-----  
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAslw1VOptEpKHfXn64b/  
UuIoiVgUONcyW9vVzFFrNleN3dW6zoQ3BDDIkoGK5kq8/2YTNfh4FoUspDDiOue  
EYrt/cv94iTRtSr0u/NqWJmh2i3n2O6gQynOgee0+rki36G1hbRK2NtL3ixGAcq/  
DWDpCG0kE8E/PillMcNdCKrS3D6TpbbsQeN0z6PY4QojBRHngj+ATjgk57xNGj4 ②  
lJGTibzBKxlm0vBRNlulXRahLAbs5XDytDzseySYpa8sTOMc3WIC72vVXdl55DF  
XpMve5tk+UDkuWZQEUsPBuvkSzrKP3NDFk07pljhIC2DCndu6Dc2X+8d07uaTWN  
WwIDAQAB  
-----END PUBLIC KEY-----" > edge_installer_pub.pem  
&& openssl dgst -sha256 -verify edge_installer_pub.pem -signature edgeRuntime.tar.gz.sig edgeRuntime.tar.gz  
&& tar -zxf edgeRuntime.tar.gz  
&& INSTALL_IOTEDGE_BIN_DIR=/data/IoTEdge sh edge_install.sh iotEdge-south.cn-north-4.myhuaweicloud.com 8943 ③  
81276251938615296 sys_edge_daemon 7d1d857d 05068467bb8026bc2f64c010ble6fb8c cloud true lite DEFAULT && cd -
```

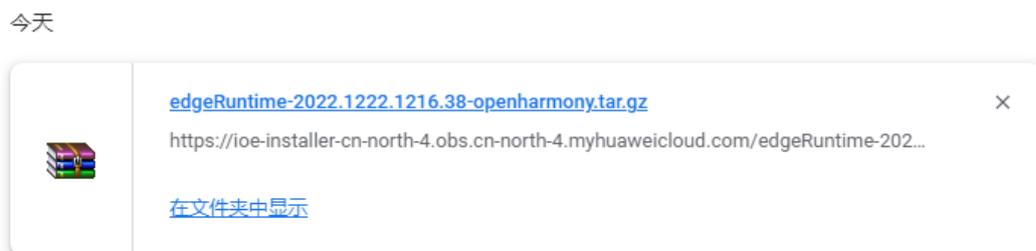
其中第①步为下载安装包，第②步为安装包完整性校验（可以跳过），第③步为执行安装命令。

**步骤3** 根据步骤二的说明获取安装包的下载命令并通过浏览器下载。

安装包下载链接：[https://ioe-installer-cn-north-4.obs.cn-north-4.myhuaweicloud.com/443/edgeRuntime-2022.1222.1216.38-openharmony.tar.gz/1671689782293/edgeRuntime-2022.1222.1216.38-openharmony.tar.gz?AccessKeyId=VGC\\*\\*\\*\\*CE&Expires=1673949121&Signature=pwx\\*\\*\\*\\*3D](https://ioe-installer-cn-north-4.obs.cn-north-4.myhuaweicloud.com/443/edgeRuntime-2022.1222.1216.38-openharmony.tar.gz/1671689782293/edgeRuntime-2022.1222.1216.38-openharmony.tar.gz?AccessKeyId=VGC****CE&Expires=1673949121&Signature=pwx****3D)

浏览器下载：

图 2-9 浏览器下载安装包



**步骤4** 在windows上通过hdc\_std工具，将下载的安装包上传到鸿蒙设备的/data/zh下并查看，上传的目录建议在/data/目录或者子目录下。

图 2-10 上传文件到鸿蒙设备

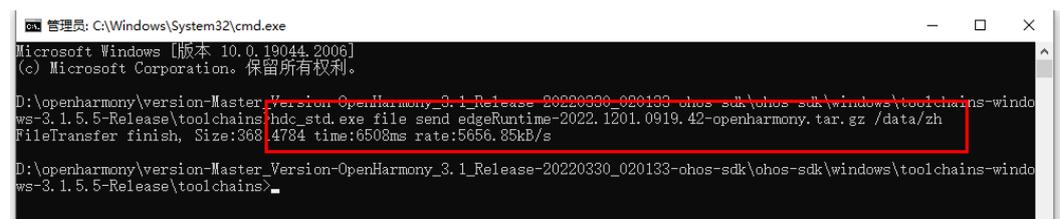


图 2-11 登录鸿蒙设备控制台

```
D:\openharmony\version-Master\Version-OpenHarmony_3.1_Release-20220330_020133-ohos-sdk\ohos-sdk\windows\toolchains-windo
ws-3.1.5.5-Release\toolchains\hdc_std.exe shell
#
#
#
#
```

图 2-12 确定文件成功上传

```
# mount -oremount,rw /
#
# cd /data/zh/
# ls -al
total 72228
drwxrwxrwx 13 root root 4096 2022-12-01 02:09 .
drwxrwx--x 57 system system 4096 2022-11-25 10:01 ..
drwxrwxrwx 5 root root 4096 2022-11-15 16:19 1109
drwxrwxrwx 6 root root 4096 2022-11-11 12:09 1111
drwxrwxrwx 4 root root 4096 2022-11-22 09:43 1121
drwxrwxrwx 4 root root 4096 2022-11-30 03:07 IoTEdge
drwxrwxrwx 5 root root 4096 2017-08-05 09:19 arm32
drwxrwxrwx 5 root root 4096 2022-11-29 09:26 arm64
drwxrwxrwx 4 root root 4096 2022-11-11 11:00 code
drwx----- 3 root root 4096 2022-11-25 07:57 edgeOmsAgent
-rw-r--r-- 1 root root 36814784 2022-12-01 02:10 edgeRuntime-2022.1201.0919.42-openharmony.tar.gz
-rw----- 1 root root 37073681 2022-11-30 01:53 harmony.tar.gz
-rwxr-xr-x 1 root root 5376 2017-08-06 23:57 ldd
drwxr-xr-x 2 root root 4096 2022-11-26 09:46 logs
-rwxr-xr-x 1 root root 7384 2017-08-07 00:19 main
drwxrwxrwx 5 root root 4096 2017-08-14 02:55 openharmony
drwxrwxrwx 2 root root 4096 2017-08-10 22:37 sdk_test
#
```

注意执行命令：`mount -oremount,rw /`，该命令必须执行，修改鸿蒙文件系统的可读写权限，以防安装失败。

图 2-13 解压安装包

```
4 tar -zxvf edgeRuntime-2022.1201.0919.42-openharmony.tar.gz
5 tar -zxvf edgeRuntime.tar.gz
#
```

在/data/zh下执行解压命令。至此安装包已经解压在/data/zh。并不一定必须在/data/zh目录下，仅仅因为上传到这个目录下。实际执行时，安装上传的实际目录进行操作。

**步骤5** 上述几步执行完成后就可以进行最后的安装操作。

- 因为前面步骤可能会耗费一些时间，所以此时需要确认安装命令是否过期，避免本次安装以为验证码过期而安装失败，安装命令的有效时间为30min。
- 如果是公网安装，可以提前测试网络是否正常，`ping www.baidu.com`。

命令拆分步骤中的第三步标识了安装命令的获取，安装命令示例：

```
INSTALL_IOTEDGE_BIN_DIR=/data/zh/IoTEdge sh edge_install.sh 100.94.60.201 8943 795632095371472896 sys_edge_daemon 02c61a36 69c459c9224bd0c949520ca962941a cloud false lite DEFAULT && cd -
```

执行安装命令：

图 2-14 执行安装命令

```
ping www.baidu.com
INSTALL_IOTEDGE_BIN_DIR=/data/zh/IoTEdge sh edge_install.sh 100.94.60.201 8943 795632095371472896 sys_edge_daemon 02c61a36 69c459c9224bd0c949520ca962941a cloud false lite DEFAULT && cd -
```

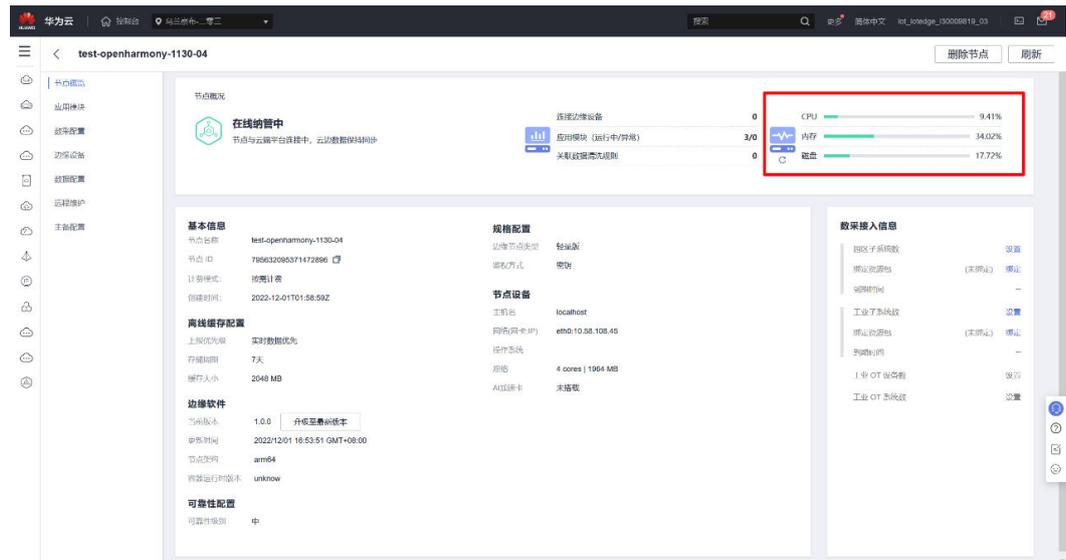
安装成功：



图 2-18 部署\$edge\_omagent 应用



图 2-19 指标采集



----结束

## 2.2 注册专业版节点

### 2.2.1 注册专业版节点（ Docker ）

专业版提供物联网行业所需的边缘管理的基本功能。

## 说明

- 在注册前请检查机器软硬件配置，如磁盘、内存和Docker版本，切换root用户执行安装命令。
  - 请充分确认以免安装失败或者安装后服务不可用。
  - Docker版本必须高于17.06，推荐使用18.06.3版本。  
(请勿使用18.09.0版本Docker，该版本存在严重bug，详见[这里](#)；如果已使用此版本，请尽快升级。Atlas 500小站预置的Docker 18.09.0.60软件版本已经修改该问题。)  
Docker安装方法请参见[这里](#)。
- 可以选择使用开源的Docker CE或者使用Docker企业版（Docker EE），使用Docker EE时，需要支付Docker额外费用，更多Docker EE相关细节请参考[Docker官方文档](#)。

## 须知

Docker安装完成后，请将Docker进程配置为开机启动，避免系统重启后Docker进程未启动引起的系统异常。

## 注册专业版节点

- 访问[IoT边缘](#)，单击“管理控制台”进入IoT边缘控制台。
- 选择左侧导航栏“边缘节点 > 节点管理”单击右上角“注册节点”。
- 根据页面提示填写参数，勾选右下角“《华为云服务等级协议》”单击“下一步”配置节点数据。

图 2-20 专业版

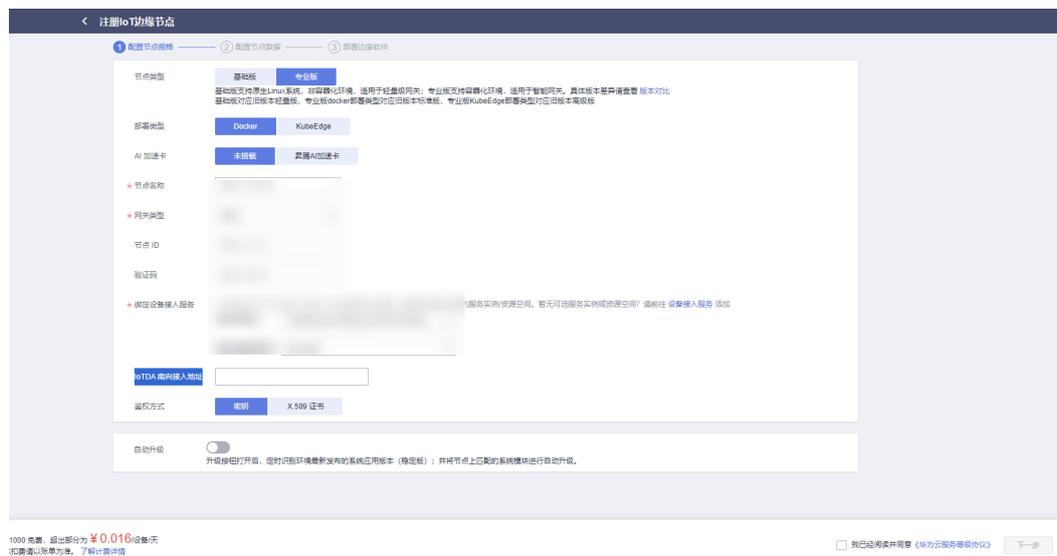


表 2-5 节点规格参数说明

参数名称	参数说明
节点类型	专业版。

参数名称	参数说明
部署类型	Docker。
AI加速卡	<ul style="list-style-type: none"> <li>未搭载：边缘节点未使用AI加速卡时选择。</li> <li>昇腾AI加速卡：支持昇腾AI加速设备，边缘节点的服务器搭载相关昇腾AI加速设备时选择。</li> </ul>
昇腾驱动地址	可选，AI加速卡选择昇腾AI加速卡时需要保证安装边缘节点的服务器上已经安装昇腾设备驱动，并根据前端提示填入驱动地址。若该值为空，默认填入Atlas300驱动默认安装地址： <code>/usr/local/Ascend/driver/lib64/</code> 。
节点名称	自定义节点名称，如“test”。 <b>说明</b> 节点名称可以在：“节点概览” - “基本信息”中编辑修改。
网关类型	other。
节点ID	自定义（可选填写）。只允许中文，英文字母、数字、中划线（-），下划线（_），最小长度1，最大长度64。 <b>说明</b> 边缘节点ID自定义(不输入默认生成)。
验证码	输入自定义验证码（可选填写）。验证码，执行安装边缘软件命令时与云端通讯的凭证。
设备归属实例	选择默认归属实例。
设备归属资源空间	选择默认资源空间，或为本实验单独新建一个资源空间，如edge。
IoTDA南向接入地址	选择网关归属的IoTDA实例的南向设备mqtt接入地址，也可以选择自定义的南向接入地址，支持“ip:端口”或“域名:端口”形式。
鉴权方式	密钥：默认生成。 X.509证书：请参考： <a href="#">基于MQTT.fx的X.509证书接入指导</a> 。 <b>警告</b> 鉴权方式为“密钥”的节点，请勿在IoTDA界面重置节点的密钥，重置密钥会导致网关与IoTDA通信异常，且不可恢复。
指纹	证书文件生成的字符串（可选填写）。只能为40位十六进制字符串或64位六进制字符串。
本地路径	输入证书文件和密钥文件在设备上的存储路径，只允许字母、数字、下划线（_）和斜线（/），以/开头并且以.pem或.crt结尾。
自动升级	可选，开启后节点将自动升级系统应用模块。 <b>注意</b> 升级过程中可能导致节点数据上传中断。

**步骤4** 填写配置节点数据信息后，单击“立即创建”。

图 2-21 配置节点数据

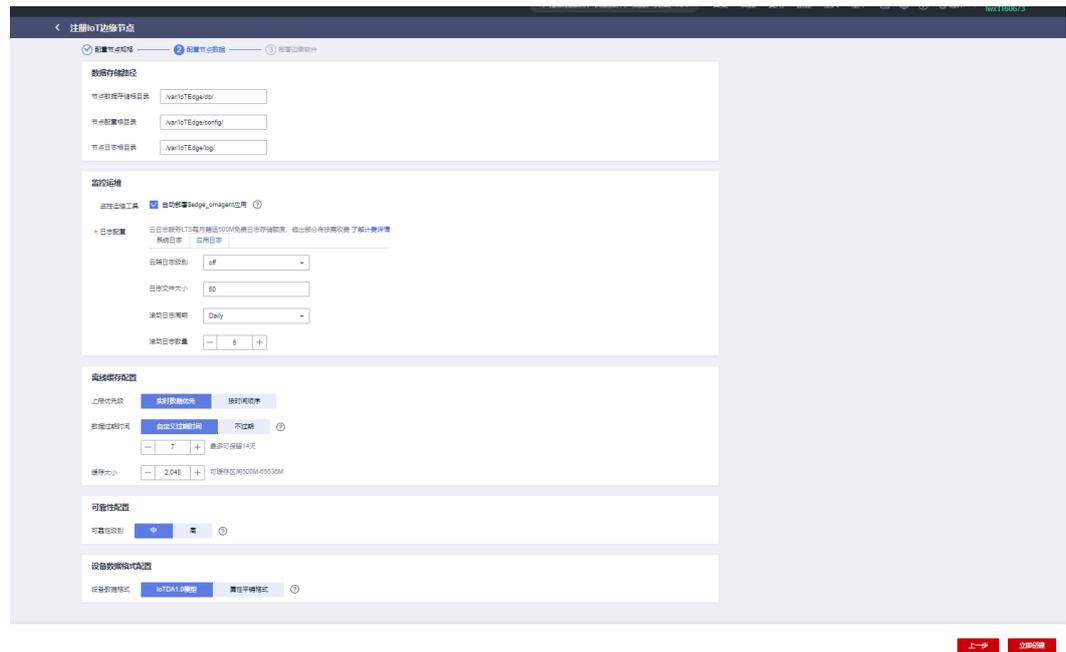


表 2-6 专业版节点信息

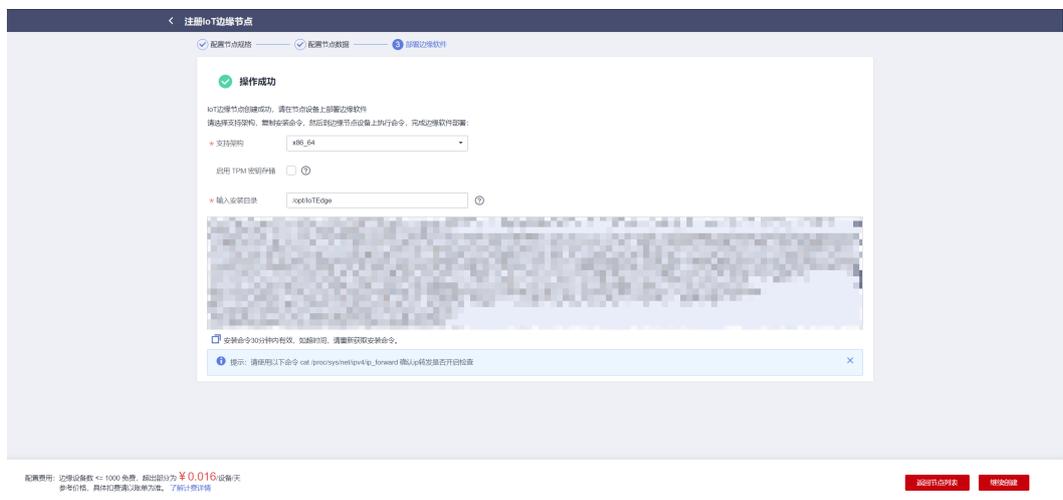
参数名称	说明
数据存储路径	默认（允许修改存储路径）。
监控运维	默认勾选“监控运维工具”。
\$edge_omagent应用	负责边缘节点远程监控和运维，提供日志上报、远程SSH、CPU等指标上报、文件上传下载能力。

参数名称	说明
日志配置	<p>系统日志：边缘节点上已部署的系统应用产生的日志。</p> <p>应用日志：边缘节点上已部署的用户定义应用所产生的日志。</p> <ul style="list-style-type: none"> <li>• 云端日志级别：on, off, trace, debug, info, warn, error, fatal。其中，on表示上传所有日志至云端，off表示不上传日志至云端，其余可选项代表按日志级别采集，选择后将只采集目标级别及以上的日志（模糊匹配）。日志级别的严重程度等级由低到高依次为：trace, debug, info, warn, error, fatal。</li> <li>• 日志文件大小：日志文件大小限制，单位MB，默认50，取值范围10-1000。某个日志文件如果达到大小限制，则会将文件转储保存至日志目录下。</li> <li>• 滚动日志周期：日志转储周期，可选项：daily, weekly, monthly, yearly。某个日志文件如果达到转储周期，则会将文件转储保存至日志目录下。日志文件大小和滚动日志周期是同时生效的，满足任何一个条件都会进行日志转储。</li> <li>• 滚动日志数量：日志文件转储个数，默认5，取值范围1-10。边缘节点保存的转储日志数量如果达到限制，则会删除最早的转储文件。</li> </ul> <p><b>注意</b></p> <ol style="list-style-type: none"> <li>1. 当前“云端日志级别”对日志等级采取模糊匹配的策略，若日志消息中存在与日志级别标识相同的关键字，则也可能被采集。</li> <li>2. 若要开启“云端日志级别”中的trace, debug, info, warn, error, fatal选项，则必须满足以下三个前置条件： <ol style="list-style-type: none"> <li>1. 目标节点是基础版节点或部署类型为Docker的专业版节点。需注意，部署类型为KubeEdge的专业版节点不支持此功能。</li> <li>2. 已安装\$edge_omagent应用，且版本不低于1-1-15-xxx。</li> <li>3. 应用日志需遵循如下规范打印，以便进行过滤： <ul style="list-style-type: none"> <li>○ 日志级别标识需打印为（大小写均可）：trace、debug、info、warn、error、fatal。</li> <li>○ 每条日志消息以YYYY-MM-DD日期格式作为前缀。</li> </ul> </li> </ol> </li> </ol> <p><b>说明</b></p> <p>专业版（部署类型为Docker）和基础版节点只涉及日志级别这一个参数，其他参数只对专业版（部署类型为KubeEdge）有效。</p>
离线缓存配置	<p>可对离线节点上报的数据进行缓存，定义缓存周期（可配置为不过期）跟缓存大小、上报的优先级、路由转发是否开启离线缓存。仅当上报优先级为“按时间顺序”时支持路由转发开启离线缓存。</p>
可靠性配置	<p>可靠性级别分为“中”和“高”2种。中级别：当应用上报属性和点位的频率超过流控值时，超过部分会被丢弃；高级别：当应用上报属性和点位的频率超过流控值时，通过控制模块的发送速度，降低上报频率，报文不会被丢弃。（注意：高可靠性级别的配置需要定制应用的配合，同时会造成性能的损失）。</p>

参数名称	说明
设备数据格式配置	<p>设备的数据格式：即单网关节点中设备的标准数据格式，包含IoTDA1.0物模型格式和属性平铺格式。IoTDA1.0物模型中包含了设备、服务和属性三层结构，具体参考IoTDA网关子设备属性上报报文格式。属性平铺格式，包含设备、属性2层结构，属性结构可以包含自定义结构的拼接，在json中作为一项平铺展开，具体格式请参考<a href="#">步骤6设备数据格式配置</a>。</p> <p><b>须知</b> 属性平铺格式功能，需要edge_hub应用1-1-38-xx版本及以上支持</p>

**步骤5** 弹出如下对话框，您可以“返回节点列表”也可以“继续创建”。

图 2-22 注册成功



### 说明

安装部署节点前，请确认机器的IP转发已开启并且已开启自启动，以及7883和8943端口开放。其中，IP转发规则配置如下：

图 2-23 IP 转发规则配置



1. 检查机器ip\_forward in/proc文件系统的值。  

```
cat /proc/sys/net/ipv4/ip_forward
```

若值等于0，则需要添加配置IP转发规则。
2. 检查配置文件/etc/sysctl.conf是否包含net.ipv4.ip\_forward = 1。  
如果配置文件中没有net.ipv4.ip\_forward参数项，请新增一行net.ipv4.ip\_forward = 1。  

```
vim /etc/sysctl.conf
```

按“i”进入编辑模式，修改完成后按“Esc”退出编辑模式，然后按“Shift” + “;”组合键，输入wq并回车，保存修改并退出。
3. 重新加载此文件的值。  

```
sysctl -p
```
4. 检查7883和8943端口是否开放，如果是华为云服务器，请参考：[安全组](#)。

---结束

## 安装专业版节点

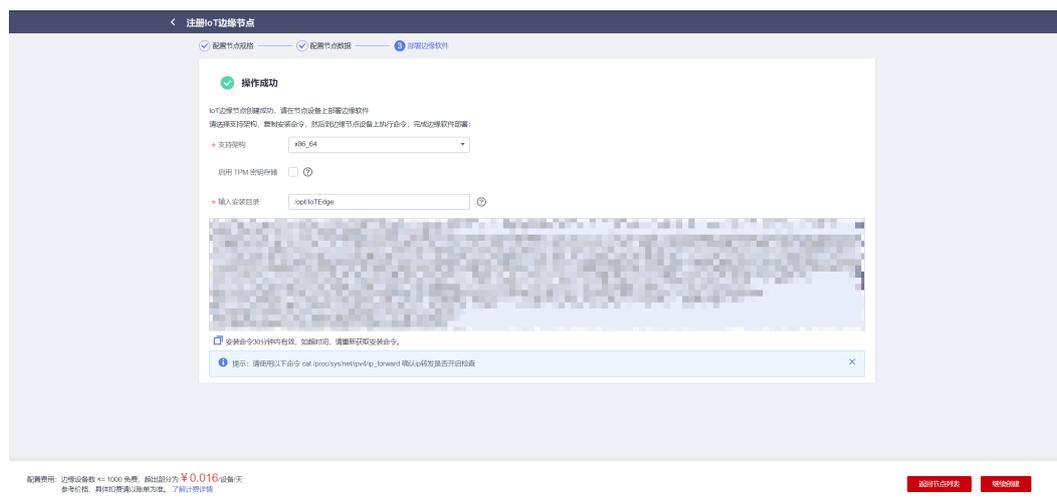
**步骤1** 在“节点管理”的节点列表选择您需要安装的节点，单击右侧“安装”，获得提示。

图 2-24 安装

节点名称	节点类型	资源包类型	资源包ID	状态	主机名/网络	操作
	专业版	--	--	未安装		删除 <b>安装</b>
	专业版	--	--	离线	172.30.7.64,172.17.0.1,1...	删除
	专业版	--	--	已安装		删除

**步骤2** 选择支持架构和安装目录，然后单击  复制安装命令，使用SSH工具以root用户登录边缘节点服务器后台系统，执行安装命令。

图 2-25 提示



### 说明

安装命令的选择需要与购买的ecs的架构保持一致。

图 2-26 执行安装

```

/usr/bin/systemctl
Removed symlink /etc/systemd/system/multi-user.target.wants/edge daemon.service.
2024-05-28T14:43:54.673289 | INFO | start read base path config.
2024-05-28T14:43:54.677487 | INFO | read log_base_path:/var/iotEdge/Log/
2024-05-28T14:43:54.705099 | INFO | read db_base_path:/var/iotEdge/db/
2024-05-28T14:43:54.718931 | INFO | read config_base_path:/var/iotEdge/config/
2024-05-28T14:43:54.738484 | INFO | end read base path config.
2024-05-28T14:43:54.752397 | INFO | delete edgeRuntime data.
2024-05-28T14:43:54.837031 | INFO | uninstall edgeRuntime finish.
[root@master001 ~]# cd /tmp && wget https://iae-installer-203.obs.cn-north-7.ulanhq.huawei.com:443/edgeRuntime-daemon60-2024.0229.1619.26.x86_64/1709195169125/edgeRuntime-daemon60-2024.0229.1619.26.x86_64.tar.gz && tar -xzf 1709195169125/edgeRuntime-daemon60-2024.0229.1619.26.x86_64.tar.gz && mv edgeRuntime-daemon60-2024.0229.1619.26.x86_64.tar.gz /opt/iotEdge && cd /opt/iotEdge && sh ./install.sh 100.94.60.201 8943 zzzzzz sys_edge_daemon 304cd980 7622268fdea045248e6263732d137294 cloud
-----BEGIN PUBLIC KEY-----
MIIEDjANBgkqhkiG9w0BAQFAAQCAQAAIIBCCkCAQEA...
-----END PUBLIC KEY-----
> edge_installer_pub.pem 66 openssl dgst -sha256 -verify edge_installer_pub.pem -signature edgeRuntime.tar.gz.sig edgeRuntime.tar.gz && cd /opt/iotEdge && sh ./install.sh 100.94.60.201 8943 zzzzzz sys_edge_daemon 304cd980 7622268fdea045248e6263732d137294 cloud
false standard DEFAULT unset 66 cd
-2024-05-28 14:44:12 - https://iae-installer-203.obs.cn-north-7.ulanhq.huawei.com/edgeRuntime-daemon60-2024.0229.1619.26.x86_64/1709195169125/edgeRuntime-daemon60-2024.0229.1619.26.x86_64.tar.gz && tar -xzf 1709195169125/edgeRuntime-daemon60-2024.0229.1619.26.x86_64.tar.gz && mv edgeRuntime-daemon60-2024.0229.1619.26.x86_64.tar.gz /opt/iotEdge && cd /opt/iotEdge && sh ./install.sh 100.94.60.201 8943 zzzzzz sys_edge_daemon 304cd980 7622268fdea045248e6263732d137294 cloud
Resolving iae-installer-203.obs.cn-north-7.ulanhq.huawei.com [iae-installer-203.obs.cn-north-7.ulanhq.huawei.com]... 100.95.181.129
Connecting to iae-installer-203.obs.cn-north-7.ulanhq.huawei.com [iae-installer-203.obs.cn-north-7.ulanhq.huawei.com] [100.95.181.129]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 27088609 (26M) [application/gzip]
Saving to: 'edgeRuntime_signed.tar.gz'

100%===== 27,088,609 93.1M/s in 0.3s

2024-05-28 14:44:17 (93.1 MB/s) - 'edgeRuntime_signed.tar.gz' saved [27088609/27088609]
Verified OK

```

图 2-27 安装命令执行成功

```

2024-05-28T14:44:29.581321 | INFO | reportDeploymentState | 81 | reportDeploymentState, deploymentState:DeploymentState[ node_id:zzzzz, deploy_id:1716878669308, deploy_state:RUNNING, reason:(zero) ]
2024-05-28T14:44:31.632185 | INFO | saveBasePathInfo | 78 | config base path start.
2024-05-28T14:44:31.632291 | INFO | saveLogPathInfo | 83 | config log_path:/var/iotEdge/Log/, db_path:/var/iotEdge/db/, config_path:/var/iotEdge/config/
2024-05-28T14:44:31.632457 | INFO | saveBasePathInfo | 95 | config base path success.
2024-05-28T14:44:31.632535 | INFO | addAgentCustomConfigs | 261 | add custom configs start.
2024-05-28T14:44:31.632603 | INFO | addAgentCustomConfigs | 285 | add custom configs end.
2024-05-28T14:44:31.632654 | INFO | pullImage | 319 | start pull image, deploy type is cloud
2024-05-28T14:44:35.898319 | INFO | createModule | 94 | module created, CreateRespBody: ModuleCreateResp[ moduleId:sys_edge_agent, moduleName:sys-edge-agent, appId:sedge_agent, appVersion: test-tpm-2-x86, containerId:4b3e43cee7eac9bc282b994377abfbf48a284389a36856f3cac0338743096, processId:(zero), type:docker, ]
2024-05-28T14:44:42.390915 | INFO | startModule | 160 | start module moduleId:sys_edge_agent
2024-05-28T14:44:42.390934 | INFO | listModule | 250 | listModule
2024-05-28T14:44:42.440775 | INFO | checkModulesDo | 340 | start check modules, time:1
2024-05-28T14:44:42.440909 | INFO | printModulesAndReport | 484 | ***success:1 failed:0
2024-05-28T14:44:42.491163 | INFO | reportDeploymentState | 81 | reportDeploymentState, deploymentState:DeploymentState[ node_id:zzzzz, deploy_id:1716878669308, deploy_state:RUNNING, reason:(zero) ]
2024-05-28T14:44:42.542192 | INFO | reportEvent | 158 | Begin reporting event: { "resource": "node.installation", "location": "zzzzzz", "event": "succeeded", "event_severity": "Info", "event_time": "20240528T044422", "notify_data": { "host_tag": "DEFAULT", "edge_node_id": "zzzzzz" } }
2024-05-28T14:44:42.592933 | INFO | reportModuleStatus | 158 | reportModuleStatus, modulesInfo:ModuleStatusInfo[ host_tag:DEFAULT, deploy_id:1716878669308, modules:[ModuleStatusInfo[ module_id:sys_edge_agent, app_id:sedge_agent, app_version:test-tpm-2-x86, container_state:RUNNING, liveness_state:(zero), update_time:1716878669308 ], ] ]
2024-05-28T14:44:43.043089 | INFO | checkBridgeNetwork | 556 | Begin checking bridge network
2024-05-28T14:44:50.524137 | INFO | install | 494 | install edge finish.
2024-05-28T14:44:50.524559 | INFO | stop | 545 | edgeinstaller stop.
root

```

**步骤3** 单击“我知道了”，等待边缘节点的状态变为“在线”，表示该节点已安装并连接成功。

**步骤4** 您可以单击“节点名称”，查看该节点详细信息，关于节点详情描述可参见[节点概览](#)。

----结束

## 2.2.2 注册专业版节点 (KubeEdge)

基于[智能边缘平台 \(IEF\)](#) 的容器应用部署和管理能力，提供跨节点的资源调度以及故障迁移，边云一致的容器生态和服务协同，共享边缘市场。

### 注册专业版节点

**步骤1** 访问[IoT边缘](#)，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 节点管理”单击右上角“注册节点”。

**步骤3** 根据页面提示填写参数，勾选右下角“《华为云服务等级协议》”单击“下一步”配置节点数据。

图 2-28 专业版

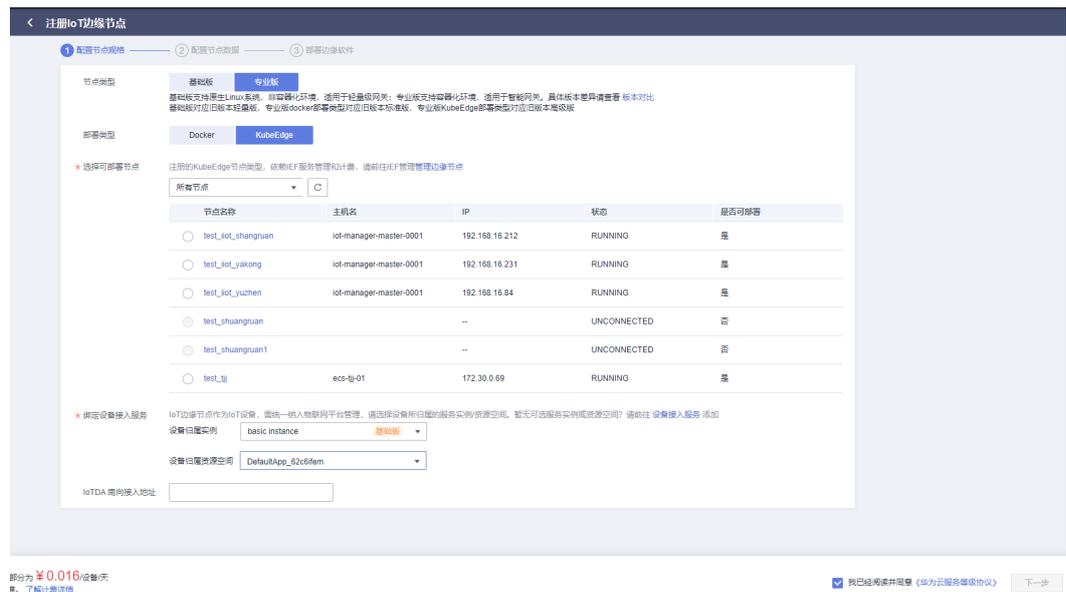


表 2-7 节点规格参数说明

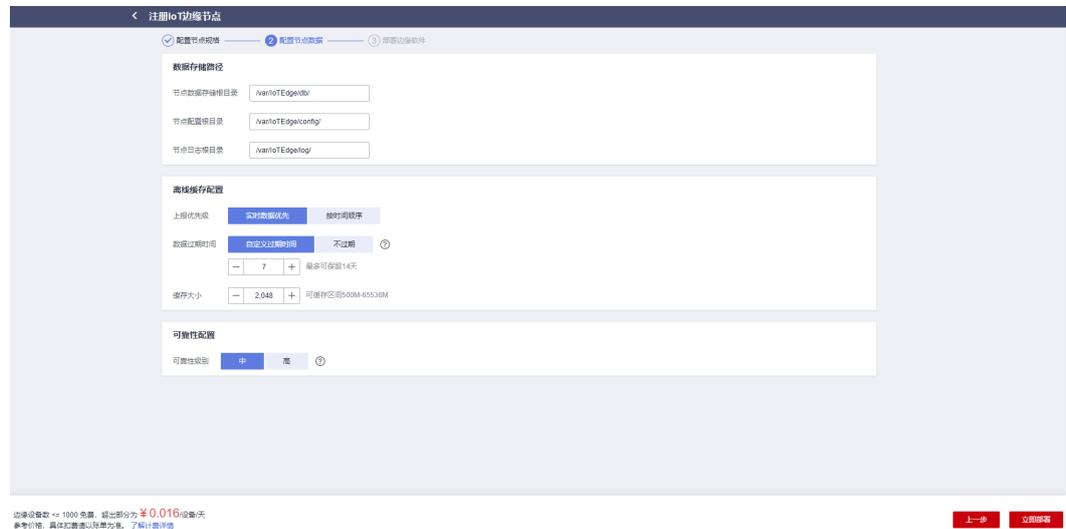
参数名称	参数说明
节点类型	专业版。
部署类型	KubeEdge.
节点名称	选择可部署节点。
AI加速卡	<ul style="list-style-type: none"> <li>● 未搭载：边缘节点未使用AI加速卡时选择。</li> <li>● AI加速卡：支持昇腾310芯片的设备（如Atlas 300AI加速卡或Atlas 500智能小站），如果使用昇腾310芯片，请选择“AI加速卡”。</li> <li>● Nvidia GPU：如果您的边缘节点搭载了Nvidia GPU显卡，请选择“Nvidia GPU”。</li> </ul>
设备归属实例	选择默认归属实例。
设备归属资源空间	选择默认资源空间，或为本实验单独新建一个资源空间，如edge。
IoTDA南向接入地址	选择网关归属的IoTDA实例的南向设备mqtt接入地址，也可以选择自定义的南向接入地址，支持“ip:端口”或“域名:端口”形式。

**警告**

请勿在IoTDA界面重置节点的密钥，重置密钥会导致网关与IoTDA通信异常，且不可恢复。

**步骤4** 填写配置节点数据信息后，单击“立即创建”。

**图 2-29** 配置节点数据

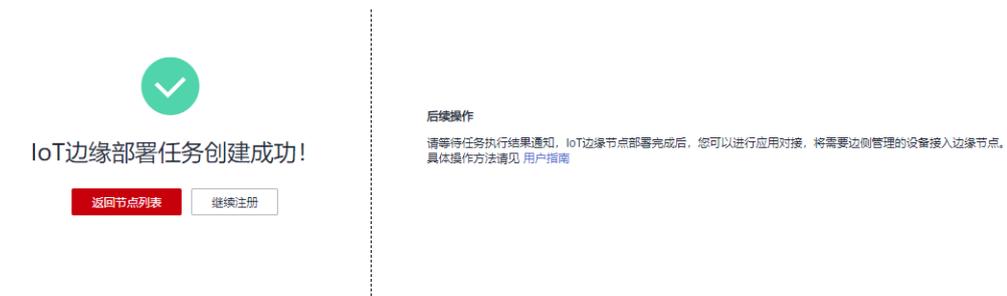


**表 2-8** 节点信息

参数名称	说明
数据存储路径	默认（允许修改存储路径）。
离线缓存配置	可对离线节点上报的数据进行缓存，定义缓存周期（可配置为不过期）跟缓存大小和上报的优先级有关。专业版节点（KubeEdge）不支持路由转发离线缓存。
存储周期	选择存储天数0~7天，默认为0。最多可保留14天。
缓存大小	可缓存区间500M-8192M。

**步骤5** 弹出如下对话框，部署任务创建成功，您可以“返回节点列表”也可以“继续注册”。

**图 2-30** 节点部署成功



----结束

## 安装专业版节点

**步骤1** 使用root用户登录边缘节点服务器后台系统，使用SFTP工具或移动存储设备，将边缘节点的安装工具和配置文件的执行命令复制到节点的目录下。

图 2-31 复制到节点目录中

```
[root@edgevicemanager-beta-0002 edge-installer]# cd /root
[root@edgevicemanager-beta-0002 ~]# ll
total 0
[root@edgevicemanager-beta-0002 ~]# ll
total 18928
-rw----- 1 root root 19367451 Nov  6 10:44 edge-installer_1.0.9_x86_64.tar.gz
-rw----- 1 root root    10443 Nov  6 10:44 edgeNodeTest.tar.gz
[root@edgevicemanager-beta-0002 ~]#
```

**步骤2** 解压installer安装包

例：`sudo tar -zxvf edge-installer_1.0.9_x86_64.tar.gz -C /opt`

其中edge-installer\_1.0.9\_x86\_64.tar.gz为安装包文件名，请下载实际文件名为准。

图 2-32 解压安装包

```
[root@edgevicemanager-beta-0002 ~]# sudo tar -zxvf edge-installer_1.0.9_x86_64.tar.gz -C /opt
edge-installer/
edge-installer/installer
edge-installer/encrypt_tool
edge-installer/conf/
edge-installer/conf/reboot.json
edge-installer/conf/logging.yaml
edge-installer/conf/upgrade.json
edge-installer/conf/script/
edge-installer/conf/script/uninstall_ief.sh
edge-installer/conf/script/get_arch.sh
edge-installer/conf/script/parse_user_config.sh
edge-installer/conf/config.yaml
edge-installer/conf/bootstrap.json
edge-installer/edgectl
edge-installer/version
```

**步骤3** 解压配置文件到/opt/IEF/Cert

例：`sudo mkdir -p /opt/IEF/Cert; sudo tar -zxvf node_name.tar.gz -C /opt/IEF/Cert`

其中node\_name.tar.gz为配置文件名，请下载实际文件名为准。

图 2-33 解压配置文件

```
pi@raspberrypi:~/wzw $ sudo mkdir -p /opt/IEF/Cert; sudo tar -zxvf edgeNodeTest1.tar.gz -C /opt/IEF/Cert
NwVITLPnIX_private_cert.crt
NwVITLPnIX_private_cert.key
system/sys_ca.crt
system/sys_private_cert.crt
system/sys_private_cert.key
user_config
NwVITLPnIX_ca.crt
```

**步骤4** 执行安装命令，当最后一行提示“install\_success”时，即代表安装成功，稍后可在“边缘节点”页查看节点状态。

```
cd /opt/edge-installer; sudo ./installer -op=install
```

图 2-34 执行安装命令

```
2020-04-22 16:28:04.843 +08:00 INFO execution/execution.go:36 precheck install taskName: 3_edge_logger
2020-04-22 16:28:04.843 +08:00 INFO execution/execution.go:40 task: precheck, cmdString: /opt/IEFpack/2.20.0/packages/3_edge_logger/setup.sh precheck install
2020-04-22 16:28:04.916 +08:00 INFO execution/execution.go:175 Install taskName: 3_edge_logger
2020-04-22 16:28:04.916 +08:00 INFO execution/execution.go:175 task: install, cmdString: /opt/IEFpack/2.20.0/packages/3_edge_logger/setup.sh install
2020-04-22 16:28:11.345 +08:00 INFO execution/execution.go:189 Install Success!
2020-04-22 16:28:11.345 +08:00 INFO execution/execution.go:189 Verification taskName: 3_edge_logger
2020-04-22 16:28:11.345 +08:00 INFO execution/execution.go:111 task: verification, cmdString: /opt/IEFpack/2.20.0/packages/3_edge_logger/setup.sh verification
2020-04-22 16:28:11.555 +08:00 INFO deploy/bootstrap.go:89 install succeed!
2020-04-22 16:28:11.568 +08:00 INFO deploy/bootstrap.go:96 install process end
2020-04-22 16:28:11.568 +08:00 INFO deploy/bootstrap.go:103 install status install_success
2020-04-22 16:28:13.692 +08:00 INFO httpclient/httpsclient.go:83 proxy: <nil>, err:<nil>
2020-04-22 16:28:25.926 +08:00 INFO deploy/deploy.go:134 report version: [2.20.0] status [install_success] finished.
```

**步骤5** 如果安装过程异常或失败，请参见[边缘节点纳管失败如何处理](#)。

### ⚠ 注意

安装完IEF相关程序后，边缘节点会显示未安装状态，需等待10分钟左右，IEF相关配置初始化完成，才会拉起边缘节点相关应用，使边缘节点在线。

----结束

## 运维管理（专业版）

边缘节点在智能边缘平台（IEF）注册并纳管后，您可以在智能边缘平台（IEF）和应用运维管理（AOM）的控制台上对边缘节点进行运维管理。

- 智能边缘平台（IEF）通过纳管用户的边缘节点，提供将云上应用延伸到边缘的能力，联动边缘和云端的数据，同时，在云端提供统一的设备/应用监控、日志采集等运维能力，为企业提供完整的边缘计算解决方案。
- 应用运维管理（Application Operations Management，简称AOM）是云上应用的一站式立体化运维管理平台，可以实时监控IoT边缘服务中的边缘节点的各项指标、日志及事件等数据共同分析应用健康状态，提供灵活的告警及丰富的数据可视化功能，帮助您及时发现故障，全面掌握应用、资源及业务的实时运行状况。

在IEF查看边缘节点监控信息和日志

**步骤1** 登录[智能边缘平台（IEF）管理控制台](#)。

**步骤2** 选择左侧导航栏“边缘资源 > 边缘节点”进入页面，单击相应的边缘节点名称。

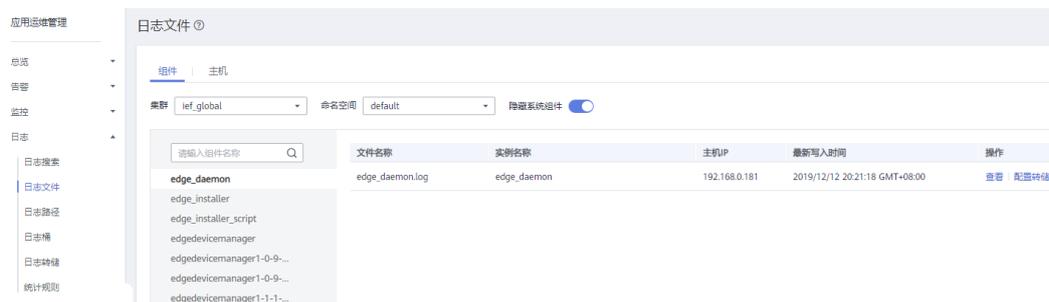
**步骤3** 选择“监控”，查看边缘节点监控信息。

**步骤4** 单击“查看日志”，可查看节点日志文件。

图 2-35 监控



图 2-36 查看日志



----结束

在AOM查看日志

**步骤1** 登录AOM管理控制台。**步骤2** 选择左侧导航栏“日志 > 日志文件 > 组件”，搜索应用名称“log”，单击日志文件右侧“查看”，即可查看详细日志。

图 2-37 查看详细日志



----结束

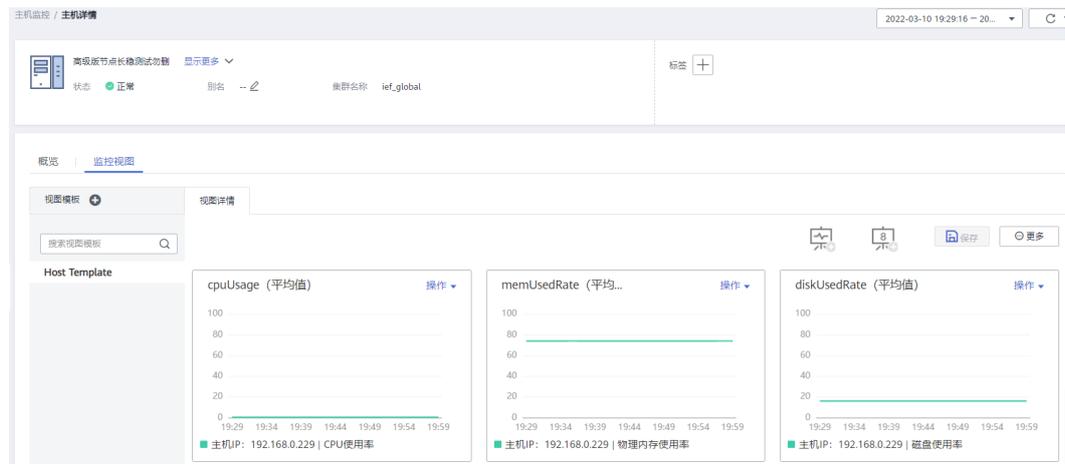
在AOM查看节点监控信息

**步骤1** 登录AOM管理控制台。**步骤2** 选择左侧导航栏“监控 > 主机监控”。

图 2-38 主机监控

**步骤3** 单击节点名称，在监控视图Tab页下，您可以设置监控节点使用的资源，如CPU、内存的使用率等。

图 2-39 监控视图



---结束

在AOM查看应用监控信息

步骤1 登录AOM管理控制台。

步骤2 选择左侧导航栏“监控 > 应用监控”中选择要监控的应用。

步骤3 单击列表中应用名称，在应用详情页面以应用维度对服务进行批量管理与监控。

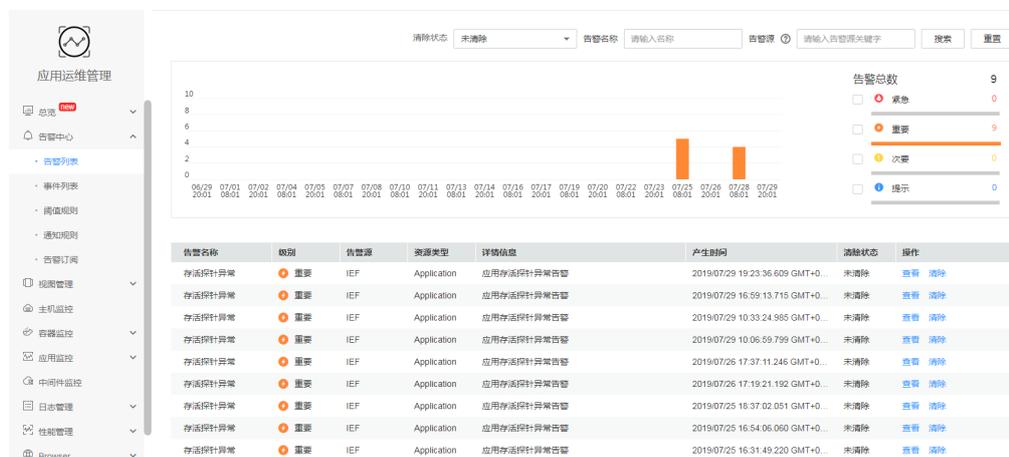
---结束

在AOM查看边缘节点的告警

IEF为每个边缘节点预置了4个告警规则，这4类告警会自动上报到AOM。

- Docker安装异常告警：边缘节点配置Docker使能时，查询docker信息失败时上报告警。
- 应用存活探针异常告警：应用配置存活探针，探针检测到异常时上报告警。
- 设置GPU信息失败告警：边缘节点配置GPU使能时，查询GPU信息失败时上报告警。
- 申请GPU资源失败告警：部署GPU应用，申请GPU资源失败时上报告警。

图 2-40 查看告警



## 2.3 节点概览

您可以查看边缘节点列表中节点状态及详细信息。在节点详情页，选择“节点概览”页签。

**步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 节点管理”进入页面。

图 2-41 边缘节点列表



**步骤3** 选择您的边缘节点，单击“节点名称”进入节点详情页。

图 2-42 节点详细信息



表 2-9 节点状态

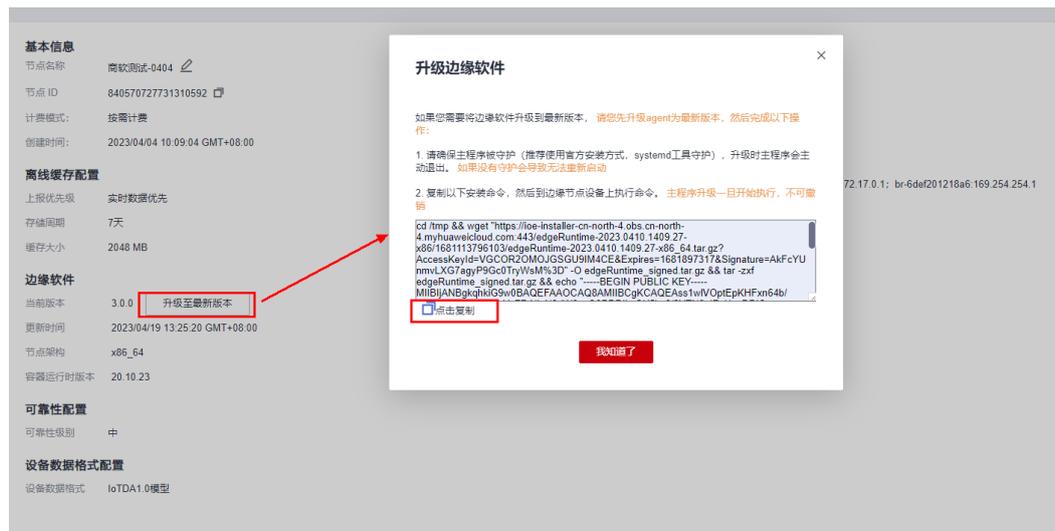
所有状态	说明
未安装	表示待安装边缘软件。
已安装	表示已安装边缘软件。
在线	表示节点在线纳管中。
离线	表示节点离线纳管中。
删除中	表示节点正在删除中。

## 须知

查看节点的CPU、内存、磁盘信息，需要部署\$edge\_omagent应用  
关于部署边缘应用的更多使用描述，可参见[部署应用](#)。

**步骤4** 边缘软件显示底层软件版本，单击升级至最新版本可以获取升级命令，复制命令到边缘服务器上执行，执行完毕后可以升级底层软件到最新版本。

图 2-43 获取升级命令升级底层软件版本



**步骤5** 专业版或者基础版节点，若搭载昇腾AI加速卡，且已安装，可以选择查看昇腾设备的详细信息。

图 2-44 昇腾 AI 加速卡详细信息

## 查看昇腾AI加速卡

昇腾驱动地址 [/usr/local/Ascend/driver/lib64/444/555/](#)

设备ID	型号	产品类型	使用情况	健康状态	设备故障描述信息
Ascend310-0	D310	Atlas 300I Mo...	user_AI001: 2 AI Core	Healthy	--
Ascend310-1	D310	Atlas 300I Mo...	--	Healthy	--

**步骤6** 设备数据格式配置，支持“loTDA1.0模型”和“属性平铺格式”2种数据格式，在创建节点时选择。

- 节点的设备数据格式为“loTDA1.0模型”，节点的设备数据格式如下：

```
{
  "devices": [
    {
      "device_id": "device1",
      "services": [
        {
          "service_id": "Motor",
          "properties": {
            "Current": 10.111,

```

```

    "Voltage": 20
  },
  "event_time": "2006-01-02T15:04:05.000Z"
}
]
}
]
}

```

- 节点的设备数据格式为“属性平铺格式”，节点的设备数据格式如下：

```

{
  "devices": [
    {
      "device_id": "device1",
      "properties": {
        "Motor/Current": 10.111,
        "Motor/Voltage": 20
      },
      "event_time": "2006-01-02T15:04:05.000Z"
    }
  ]
}

```

---结束

“数采接入信息”展示边缘节点通过数采应用所接入的行业子系统/设备的情况。如物联网工程师通过数采应用“\$industry\_dc\_bsi”对接10个ERP/MES子系统接入点后，应在“工业子系统数”中通过“设置”填写数量为10；然后购买工业子系统采集服务并与节点进行绑定。

表 2-10 数采接入信息

参数名称	说明
园区子系统数	节点下园区子系统数采应用对接的园区子系统数量。通过“设置”填写对接的园区子系统数量后，需“绑定”对应类型的采集服务。在“到期时间”可查看所绑定的采集服务的到期时间。
工业子系统数	节点下工业子系统数采应用对接的工业子系统数量。通过“设置”填写对接的工业子系统数量后，需“绑定”对应类型的采集服务。在“到期时间”可查看所绑定的采集服务的到期时间。
工业OT设备数	节点下工业OT设备数采应用对接的工业OT设备数量。
工业OT系统数	节点下工业OT系统数采应用对接的工业OT系统数量。

表 2-11 数采应用与数据采集服务的对应关系

数采应用名称	数据采集服务名称
\$industry_dc_bsi	工业子系统数据采集服务

## 2.4 应用模块

在节点详情页的“模块管理”页签，可部署边缘应用、配置业务应用、数据流转配置。

- 关于部署边缘应用的更多使用描述，可参见[部署应用](#)。
- 关于数据流转配置的更多使用描述，可参见[数据流转配置](#)。

图 2-45 部署边缘应用



模块ID	模块名称	所属应用	版本	应用类型	实例状态	操作
sys_edge_agent	sys_edge_agent	Sedge_agent	1.0-0-standard-v05	系统必选	运行中	升级
sys_edge_hub	sys_edge_hub	Sedge_hub	1.0-0-standard-v05	系统必选	运行中	升级
user_test_app_multployment	user_test_app_multployment	test_app_multployment	1.0.0	用户定义	运行中	删除 升级

## 配置业务应用

业务应用支持应用自定义配置，应用部署成功之后，根据应用是否关联业务模板提供不同的配置页面。下面简单介绍如何使用。应用模板配置以及应用和模板如何关联的详细使用可以参照[基于IoT边缘实现业务应用自定义配置](#)。

图 2-46 应用配置入口位置



模块ID	模块名称	所属应用	版本	应用类型	实例状态	操作
sys_edge_hub	sys_edge_hub	Sedge_hub	2023-04-13-1-standard-v05	系统必选	运行中	升级
sys_edge_agent	sys_edge_agent	Sedge_agent	1-1-22-v05	系统必选	运行中	升级
user_test_app_config	user_test_app_config	test_app_config	1.0.0	用户定义	运行中	停止 删除 升级 配置

根据应用是否关联业务模板呈现不同的配置页面。

## 关联业务模板的应用配置

如果对应的应用关联了业务应用模板，则呈现如下配置参数页面：

图 2-47 配置页面展示



图 2-48 添加自定义配置数据

✕

### 配置参数

模块 ID	所属应用	版本
user_test_app_config	test_app_config	1.0.0

\* 业务模块

连接信息

\* 服务端点

配置信息 ?

∨ 配置信息1 🗑

\* 设备id

\* 加密数据

∨ 配置信息2 🗑

\* 设备id

\* 加密数据

⊕ 添加配置信息

删除配置参数

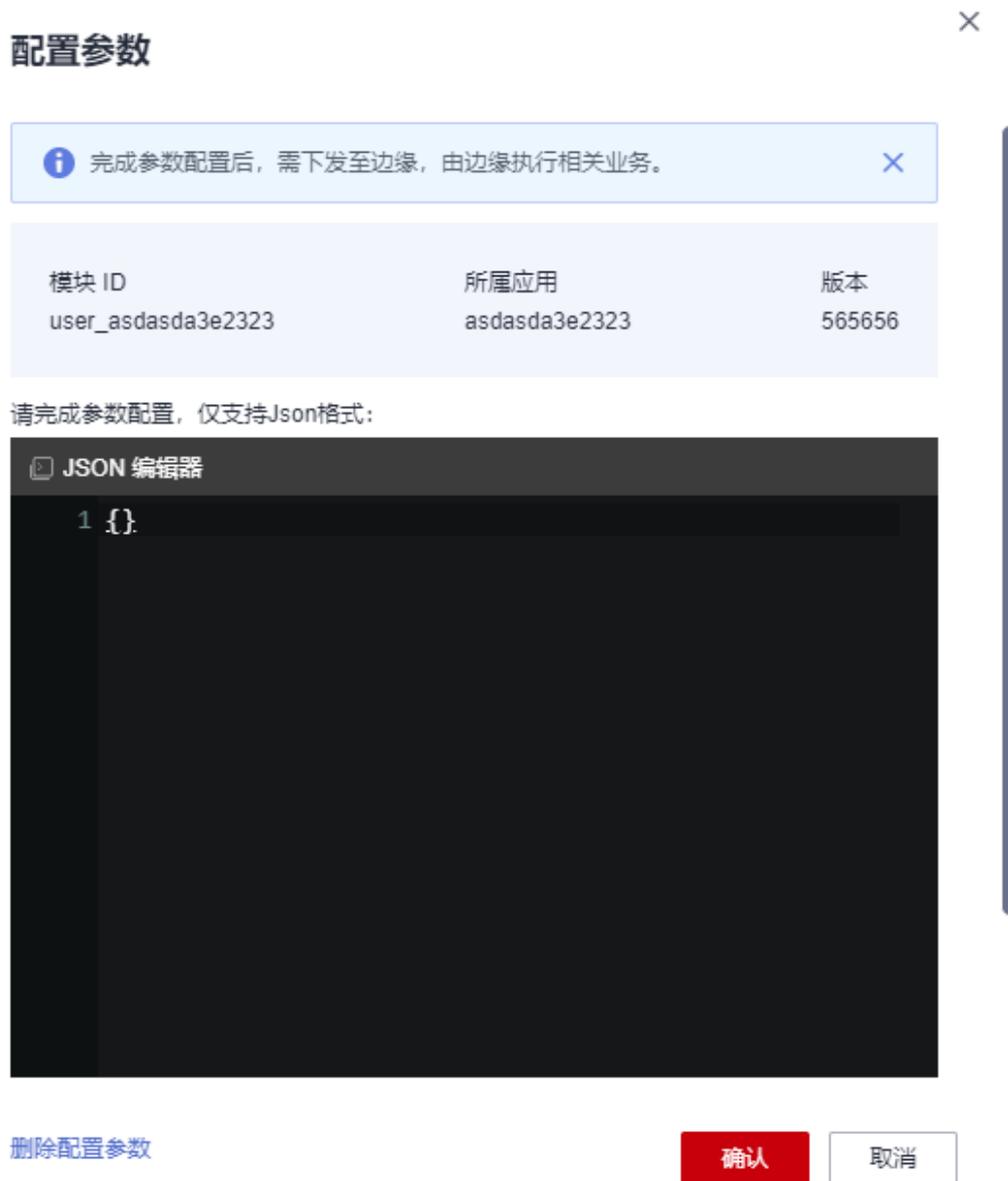
确认取消

这里在配置信息中添加两个配置项。单击确认进行配置保存并同时下发至边缘模块。

### 未关联业务模板的应用配置

如果该应用没有关联业务模板，则呈现如下图所示的配置页面：

图 2-49 未关联业务模板的应用配置



#### 须知

1. 如果未关联业务模板，应用的配置数据为增量修改，本次修改会和之前的配置数据按key进行比对，不一致进行修改，本次新增key为新增数据。
2. 如果需要删除之前的配置项，需要将对应的配置项置为null。
3. 如果需要删除所有配置数据，可以直接单击“删除配置配置参数”进行删除。
4. 系统应用如果没有关联业务应用配置模板不支持应用配置。
5. 驱动应用不支持此种配置方式。

## 删除所有配置数据

如果需要删除所有配置数据，则可以单击如下图示中按钮进行删除。

图 2-50 删除配置参数



删除后的配置数据为空，如下结构：

```
{}
```

## 数据流转配置

您可以根据需要配置数据流转的来源和目标，让消息按照规定的路径转发至对应端点，提高数据的安全性。

系统默认数据流转规则是从“设备端”到“云端”，即边缘节点默认将设备数据上报到云端IoTDA云服务。

### ⚠ 注意

- 只有配置了消息输入输出端点的应用才能够使用数据流转功能，在模块间流转设备数据。
- 删除模块或升级模块后可能会导致数据流转失效，请及时手动调整规则。

## 操作步骤

- 步骤1** 访问[IoT边缘](#)，单击“管理控制台”进入IoT边缘控制台。
- 步骤2** 选择左侧导航栏“边缘节点 > 节点管理”进入页面。
- 步骤3** 选择您的边缘节点，单击“节点名称”进入节点详情页。
- 步骤4** 选择“应用模块 > 模块管理”页签，单击数据流转配置。
- 步骤5** 依次单击“配置规则 > 添加规则”，根据[表2-12](#)添加您的规则。

图 2-51 数据流转配置



### 📖 说明

默认流转规则default表示数据的一种默认流向，所有设备端的数据都流转到云端（IoTDA）。

表 2-12 添加规则

参数名称	说明
规则名称	数据流转规则的名称。
消息类型	数据流转规则所流转的数据内容，有“设备数据”和“自定义消息”两种类型。“设备数据”即边缘设备的数据，可能为物模型1.0格式或属性平铺格式，取决于边缘节点的设备数据格式；“自定义消息”即消息体为模块自行义，边缘网关不解析，仅根据流转规则进行透传。
消息来源	发送消息的模块及所属应用的输出端点。
消息目标	接收消息的模块及所属应用的输入端点。

**步骤6** 单击“保存”，完成数据流转配置。

对于不需要使用的流转规则可以删除。删除后，该消息规则不再生效，系统不会再对发送到源端点指定资源的消息进行转发处理。

----结束

## 资源监控

### 须知

使用资源监控的功能，需要部署\$edge\_omagent应用  
关于部署边缘应用的更多使用描述，可参见[部署应用](#)。

### 操作步骤

**步骤1** 访问[IoT边缘](#)，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 节点管理”进入页面。

**步骤3** 选择您的边缘节点，单击“节点名称”进入节点详情页。

**步骤4** 在右上角则可查看节点服务器的CPU、内存、磁盘信息。



----结束

## 2.5 边缘设备

在节点详情页的“边缘设备”页签，可查看或者添加边缘设备。

关于子设备管理的更多使用描述，可参见[设备接入边缘节点](#)。

图 2-52 边缘设备



## 批量导入设备

**步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 在左侧导航栏选择“边缘节点 > 节点管理”，选择之前创建的边缘节点，单击节点进入详情页。

**步骤3** 在左侧导航栏选择“边缘设备 > 批量注册”。

图 2-53 边缘设备批量注册



**步骤4** 单击“查看任务列表”可以显示当前账号下的所有设备任务。

图 2-54 边缘设备批量注册任务列表



**步骤5** 单击“批量注册”在“批量添加设备”弹窗中输入任务名称，单击“下载模板文件”并填写设备信息，单击“添加文件”然后单击“确认”。

图 2-55 批量注册



表 2-13 点位模板文件说明

参数名称	说明
node_id	必填，设备ID
product_id	必填，产品ID，必须为MQTT协议的产品
device_name	设备名
description	设备描述

参数名称	说明
gateway_id	创建直连edge设备时，不需要填该字段，创建二级子设备时，该字段为父设备id。
password	密码

**步骤6** 设备批量任务添加成功。

**图 2-56** 批量注册列表



**说明**

1. 单个资源空间默认最多创建的批量设备任务数为20。
2. 单个资源空间最多同时处理的批量设备任务数为10。
3. 批量任务老化，已完成的批量任务在30天后自动老化删除，包含设备批量注册、批量创建点位等批量执行的任务。
4. 单个批量设备任务(一个execl文件)最多可以添加设备数为10000。
5. 处于执行中的任务不能删除。
6. 设备所对应产品必须是MQTT协议的产品。

----结束

## 批量删除设备

**步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 在左侧导航栏选择“边缘节点 > 节点管理”，选择之前创建的边缘节点，单击节点进入详情页。

**步骤3** 在左侧导航栏选择“边缘设备 > 批量删除”。

**图 2-57** 边缘设备批量删除



**步骤4** 单击“查看任务列表”可以显示当前账号下的所有设备任务。

**图 2-58** 边缘设备批量删除任务列表



**步骤5** 单击“批量删除”，在“批量删除设备”弹窗中输入任务名称，单击“下载模板文件”并填写设备信息，单击“添加文件”后单击“确认”。

图 2-59 批量删除



表 2-14 模板文件说明

参数名称	说明
device_id	必填，设备ID（注意：逐行删除设备，如有子设备，子设备需在父设备前）

**步骤6** 设备批量任务添加成功。

图 2-60 批量删除列表

### 说明

1. 单个资源空间默认最多创建的批量设备任务数为20。
2. 单个资源空间最多同时处理的批量设备任务数为10。
3. 批量任务老化，已完成的批量任务在30天后自动老化删除，包含设备批量注册、批量创建点位等批量执行的任务。
4. 单个批量设备任务(一个excel文件)最多可以删除设备数为10000，预计耗时50分钟。
5. 处于执行中的任务不能删除。
6. 逐行删除设备，如有子设备，子设备需在父设备前。

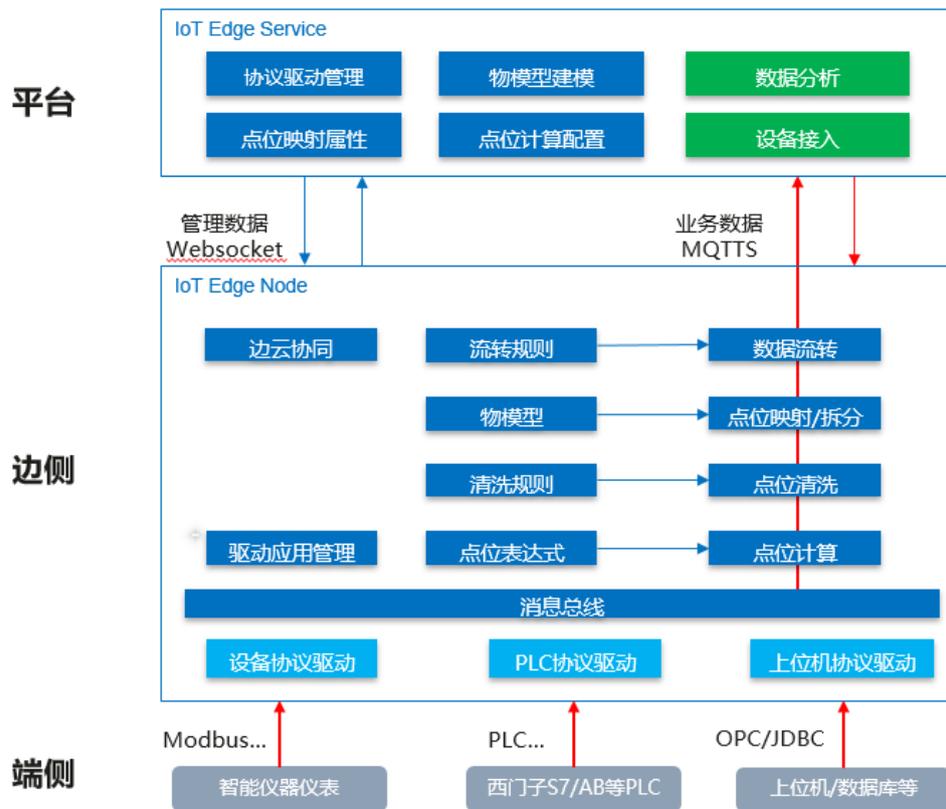
----结束

## 2.6 数采配置

### 2.6.1 架构介绍

点位计算、点位表达式、点位清洗等边缘计算。

图 2-61 架构图



## 竞争力描述

**数采驱动热插拔:** 支持数采驱动自定义、下发、升级、监控等管理，热部署替换，无需重启。

**原始数据校正:** 支持点位表达式，在边缘侧执行线性计算，对采集到的数据进行校正。

**重复数据冗余清洗:** 在数据采集频率高，冗余数据过多时，自动筛选掉重复数据，对跳变的波动数据过滤清洗。

**点位动态映射物模型:** 原始点位与物模型属性，支持动态配置其映射关系，即改即用；数据上行自动将点位转成应用可读的属性，数据下行自动将属性转成点位反写。

**低时延反向控制:** 支持百毫秒级下发修改对应点位，实现实时反向控制。

## 使用限制

- 跨数据源设备不支持反控。

## 2.6.2 配置流程

图 2-62 数据采集实施步骤



- 步骤1 创建边缘网关（专业版），请参考 [边缘节点](#)。
- 步骤2 创建产品，选择Other协议（自定义协议要用Other）。
- 步骤3 设备建模，产品中添加自定义service和属性。
- 步骤4 添加子设备，请参考[设备接入边缘节点](#)。
- 步骤5 添加驱动应用，发布版本，请参考[添加驱动应用](#)。
- 步骤6 边缘节点部署相应的驱动应用，请参考[部署应用](#)。
- 步骤7 创建数采连接，请参见[OT数采配置](#)。
- 步骤8 实时值跟踪，点采集值按钮，主动查询点位值，请参见[查看采集结果](#)。

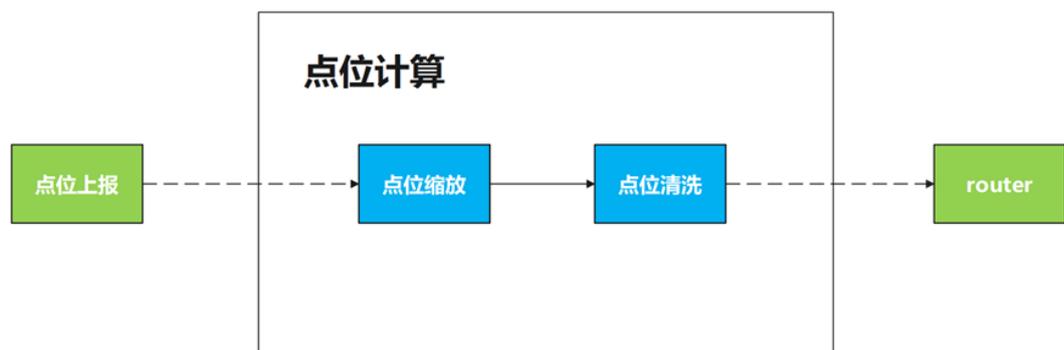
----结束

## 2.6.3 点位计算

### 业务流程

点位计算业务流程如[图1 流程图](#)所示，先进行点位缩放得到真实值，再用真实值进行点位清洗得到上报值。

图 2-63 流程图



## 点位缩放

对采集数据做规整，减少应用对数据处理和适配的工作量，如从PLC采集上来的原始数据，需要经过计算后才能表达真实含义。具体操作步骤请参见[点位缩放](#)。

### 概述

点位缩放，提供通过线性表达式将采集值转换为真实值的功能。支持公式 $Y = \text{ratio} * X + \text{base}$ ，使用线性表达式，通过配置ratio、base、accuracy来对采集值进行处理。

公式：

$$Y = \text{ratio} * X + \text{base}$$

- **ratio**: 缩放的倍率，浮点型，取值范围[-10000, +10000]，最大支持小数点后7位
- **base**: 基准值，浮点型，取值范围[-10000, +10000]，最大支持小数点后7位
- **accuracy**: 缩放后结果的精度，精确到小数点后几位，直接截取，不四舍五入。整数类型，取值范围[-1, 8]。-1或界面不输入，表示不对小数位进行处理

### 场景示例

- 场景1:

机器的基准温度是100°C，但采集器上报的值只为1、2、3、4、5，即可以使用表达式进行计算，得到需要的值。

$$\text{设ratio} = 1, \text{base} = 100, Y = X + 100$$

表 2-15 温度转换表

X	Y
1	101
2	102
3	103

- 场景2: 精度截断场景

$$\text{设ratio} = 0.11, \text{base} = 100, \text{accuracy} = 2$$

表 2-16

X	Y	精度处理后
10	101.1	101.1
100	111	111
100.11	111.0121	111.01

- 场景3:

驱动采集点位值的单位为W（瓦），平台接收属性的单位为KW（千瓦），可以使用点位缩放进行单位换算，公式为： $Y=0.001X$

设ratio = 0.001, base = 0

表 2-17 功率换算表

X (W)	Y (KW)
100	0.1
1000	1
20000	20

## 点位清洗

点位清洗，在边侧提供对设备上报的点位进行数据去重、数据波动抑制的功能，适用于降低冗余数据、上云带宽的OT数采场景。

点位清洗规则是针对模块的每个点位，为可选项，使用前需要配置“静默时间窗”和“偏差”这两个参数。具体操作步骤请参见[点位清洗](#)。

### 概述

数据上报：当满足上报条件，数据才会上报。

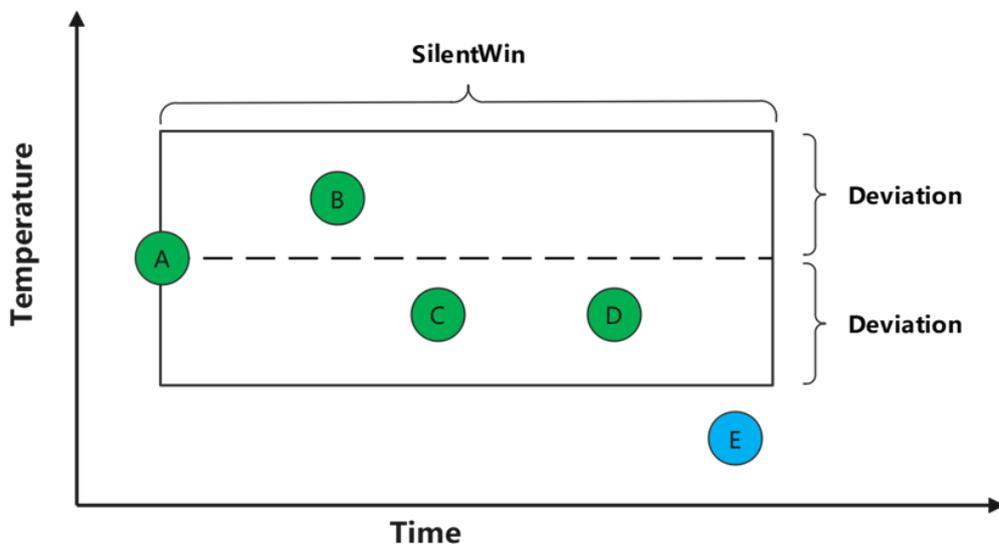
上报条件有两种：

- 条件1：当前上报值与前一次上报值的差值超过偏差的区间
- 条件2：当前上报与上一次上报的时间差，超过静默时间窗

**偏差 (Deviation)**：震荡的偏差区间值，在该区间的值将视作正常的波动，数据将不进行上报，取值范围[0~100]，类型浮点型。如偏差值为10时，表示在±10%（[-10%，+10%]）的区间内值将视为正常的波动。

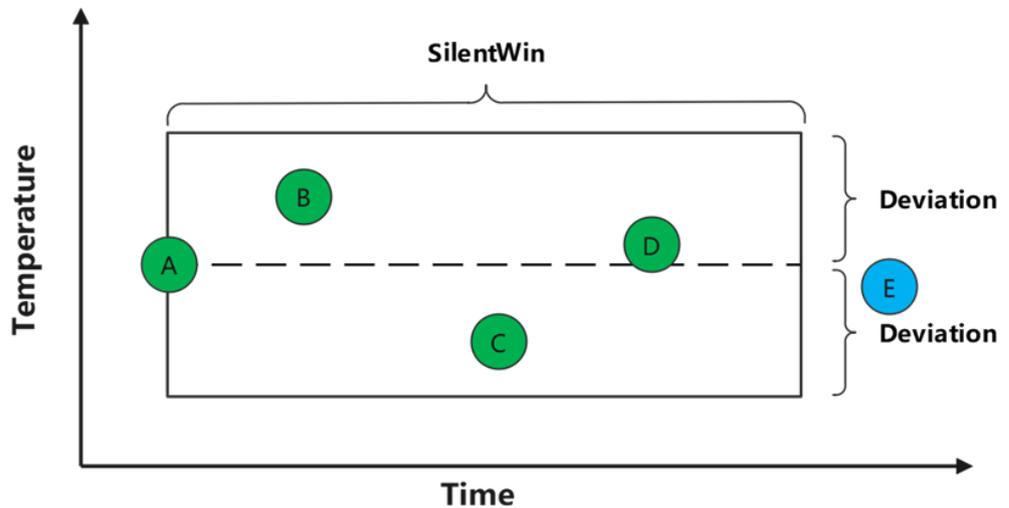
**静默窗口 (SilentWin)**：静默时间窗口，在静默时间窗口内没有触发上报条件的数据，将不会上报。取值范围(0, 86400]，单位秒，当值为0时，表示时间窗口无穷大。

示例说明1（超出偏差范围）：



在第0s，值A上报，开启点位清洗规则。在静默时间窗（SilentWin）内，上报的值为A、B、C、D、E。其中值B、C、D到达时都未满足上报条件，不进行上报。值E到达， $|Data_F - Data_A| > Deviation$ ，触发上报条件1，值E将进行上报。

示例说明2（超出静默时间窗）：



在第0s，值A上报，开启点位清洗规则。在静默时间窗（SilentWin）内，上报的值为A、B、C、D、E。其中值B、C、D到达时都未满足上报条件，不进行上报。值E到达， $Time_E - Time_A > SilentWin$ ，触发上报条件2，值E将进行上报。

样例数据示例：

SilentWin = 0（无穷大），Deviation = 1%（所有数据都未清洗掉，都上报）

时间	值（温度）	上报数据
0s	100°C	100°C
10s	200°C	200°C
20s	300°C	300°C
30s	400°C	400°C

SilentWin = 25s，Deviation = 0%（超过时区范围，将当前值上报）

时间	值（温度）	上报数据
0s	100°C	100°C
10s	100°C	-
20s	100°C	-
30s	100°C	100°C(30s)

**场景示例****场景1：变更检测**

SilentWin = 35s, Deviation = 0%

在35秒的时间窗口内，相同的值将不进行上报。在值有变化之后，将会上报当前值。示例如下：

- 0秒上报值36°C，30s达成条件1，将37°C进行上报。重置时间窗口、上报条件。
- 70秒的值为37°C，达成条件2：70 - 30 = 40s > 35s，将37°C(70s)上报，重置时间窗口、上报条件。

时间	值 (温度)	上报数据	达成条件
0s	36°C	36°C	-
10s	36°C	-	-
20s	36°C	-	-
30s	37°C	37°C	条件1
40s	37°C	-	-
50s	37°C	-	-
60s	37°C	-	-
70s	37°C	37°C(70s)	条件2

**场景2：波动抑制**

SilentWin = 60s, Deviation = 10%

时间	值 (温度)	上报数据
0s	100°C	100°C
10s	96°C	-
20s	95°C	-
25s	110°C	-
30s	120°C	120°C
40s	116°C	-
50s	117°C	-
60s	121°C	-
100s	130°C	130°C

## 点位阈值

该功能开启后需要配置点位上报数据的最大值、最小值，端侧发生设备数据上报时，超出最大值和最小值范围的异常值会被忽略，不进行上报。具体操作步骤，请参考[点位阈值](#)。

### 应用场景

- 该功能应用于设备上报数据存在过大过小的不稳定异常值需要忽略时的场景。
- 该功能针对数据源下的每个点位设置，仅支持整型和浮点型的数据类型的点位配置。

### 场景示例

创建点位时，设置点位阈值的最小值为0，最大值为10。

表 2-18 上报结果举例

设备上报值	是否上报	描述
0	是	点位阈值端点值放通。
10	是	点位阈值端点值放通。
5	是	点位阈值范围内放通。
15	否	不在点位阈值范围内，作为异常值过滤。

## 2.6.4 最佳实践

在节点详情页的“数采配置”页签，可对第三方应用的配置项进行管理。

### 基于 IoT 边缘实现 OT 数采

OT数采配置，提供了在IoT边缘控制台简单配置，实现OT数采（DCS、PLC、OPCUA等）。

基于IoT边缘实现OT数采更多描述，请参见[基于IoT边缘实现OT数采（OPCUA协议）](#)。

## 2.7 数据配置

在节点详情页的“数据配置”页签，可查看该边缘节点的日志配置信息。

关于日志配置的更多使用描述，专业版节点可参见[表2 专业版节点信息](#)和[云日志服务（LTS）](#)；专业版节点可参见[应用运维管理（AOM）](#)。

图 2-64 日志配置



## 2.7.1 设备数据打印配置

用户在创建边缘网关后，设备数据打印配置默认为关闭状态，此时边缘网关不会在本地记录设备上报的点位数据。用户可以开启或修改该配置对日志进行打印。在数据丢失敏感的高可靠场景下，用户可以通过开启设备数据打印配置来检查边缘网关是否上报数据成功，当点位数据丢失时，方便排查定位上报全链路中的异常点。

当用户启动该配置后，在网关机器上的日志根路径下，系统模块hub会增加一个payload日志子目录，用来记录网关成功上报的点位数据。边缘网关上报的点位数据会在网关机器或者虚机上保存一份副本，形式为.gz后缀的压缩文档。例如：日志根路径为/var/loTEdge/log，那么记录数据的文件位于/var/loTEdge/log/sys\_edge\_hub/log/payload下。

该配置包含三个参数：disk\_quota、age、state。

表 2-19 参数说明

参数名称	参数描述
disk_quota	磁盘配额，用于限制设备数据记录消耗磁盘空间的最大值，防止数据量过大，记录数据消耗太多磁盘资源，影响其他应用。数据记录文件超出限制时，边缘网关会主动清理最旧的文件。用户可按照实际情况配置，不可超出实际资源的最大值。
age	老化时间，用于老化数据记录文件，如果旧的数据记录文件超出老化极限，在重新写入新压缩文件时，会清理旧的压缩文件。
state	配置的开关，用于标识配置是否需要生效。

## 操作步骤

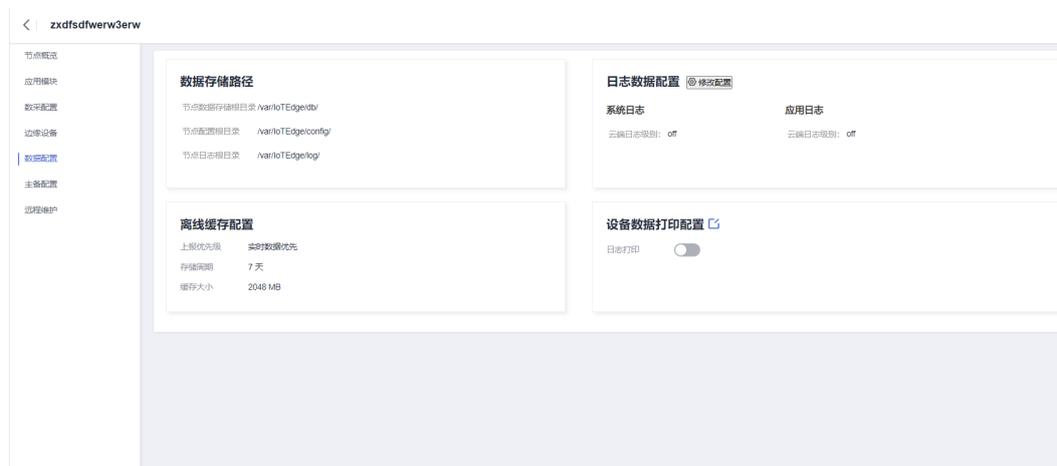
**步骤1** 选择需要配置的网关，单击网关名称进入“节点详情”页面。

图 2-65 选择边缘网关



步骤2 进入“节点详情”后，单击“数据配置”页签。

图 2-66 数据配置页签



步骤3 开启“设备数据打印配置”开关，填写磁盘配额和老化时间参数，单击“确认”，此时开始记录网关成功上报的点位数据。

### 说明

磁盘配额：限制记录数据的文件占用的磁盘空间，范围是1024~5120之间的整数，单位为MB。  
老化时间：限制记录数据的文件的保存时间，范围是1~7之间的整数，单位为天。

### 注意

磁盘配额需要根据实际机器资源填写，否则在网关上报数据量较大时，存在磁盘写满，影响机器上其他程序运行的风险。

图 2-67 打开设备数据打印配置开关

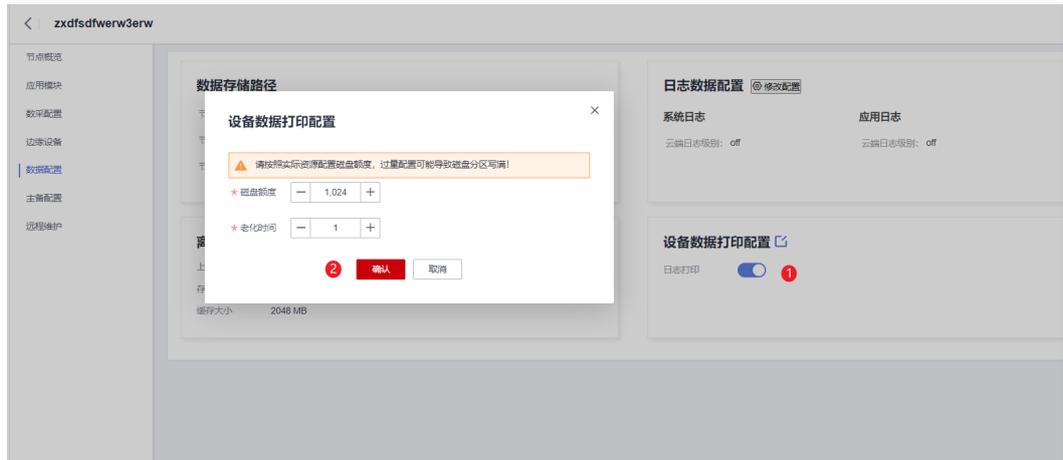


图 2-68 开启配置开关后的页面



步骤4 关闭设备数据打印配置开关。

图 2-69 开关关闭页面



### 说明

配置开关重新开启可以复用关闭前的配置值。

步骤5 编辑修改参数，单击“确定”。

### 说明

新配置代替旧配置生效期间，记录功能会短暂关闭，可能会导致数据漏记，不推荐频繁修改。

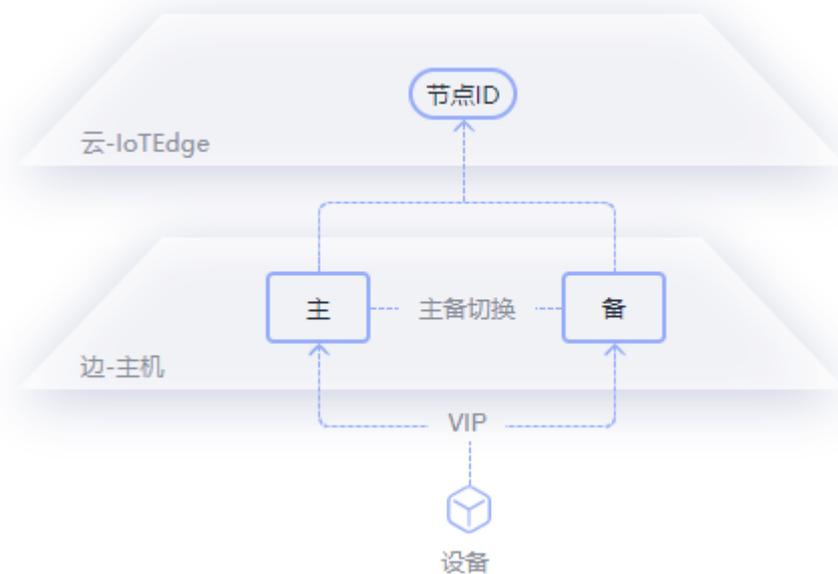
图 2-70 编辑配置页面



---结束

## 2.8 主备配置

图 2-71 构架图



通过主备配置，保证主机数据的可靠性和相关业务的可持续性。

[主] 执行边缘所有业务，[备] 下载所有镜像且仅执行edge\_agent、edge\_keepalive应用，在主节点发生宕机，网络故障等情况下进行主备倒换，且原主节点恢复后不会进行抢占。

## 使用约束

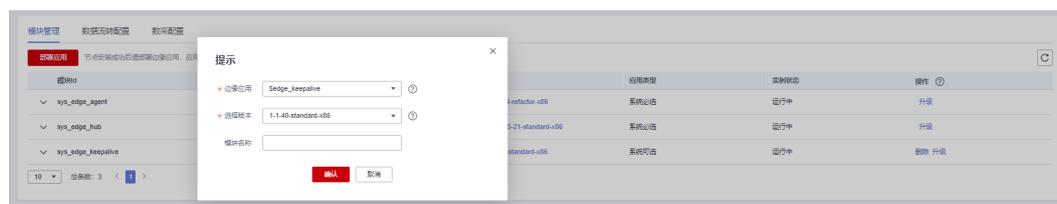
1. 主备配置时，所绑定的硬件网关上的网卡，需要在同一个网络内，保证心跳正常。
2. 主备网关之间需要设置正确的iptables，防止心跳网络异常。
3. 建议排查防火墙设置，如果防火墙没有关闭需要增加vrrp放通策略。
4. 专业版kubeEdge部署类型的节点不支持主备配置。
5. 如果需要删除主备配置，建议先在后安装的备主机上进行卸载操作(执行卸载命令 `sh /opt/loTEdge-Installer/uninstall.sh`)，然后删除主备配置，保留第一个安装的主机。如果后安装的备主机如果没有进行卸载操作，删除主备配置默认会将应用停止掉，影响该节点的运行。
6. 设置错误的网卡名称会导致模块edge-keepalive工作在故障状态，导致主备不能够正常工作。
7. 当前edge-keepalive升级会导致主备切换，所以不建议对edge-keepalive进行升级操作。
8. 主备情况下，主节点主机状态必须离线才支持重装，节点不一定离线。
9. 备节点离线才支持重装。备节点重装时，主节点必须在线。
10. 主备情况下，主节点重装不支持修改设备密码以及设置验证码，只能后台随机生成。
11. 主备切换时长（切换成功并且业务恢复）与网关上安装的应用数量、应用启动时间、点位数有关。
12. 在主备切换期间，没有办法处理数据，数据是丢失的。

## 部署应用

**步骤1** 选择您的边缘节点，单击“节点名称”进入节点详情页。

**步骤2** 在左侧导航栏选择“应用模块 > 模块管理”页签，单击左上角“部署应用”，部署 `$edge_keepalive`应用：

图 2-72 部署应用



- 边缘应用： `$edge_keepalive`
- 选择版本： 1-1-40-standard-x86（根据当前环境中应用版本为准）

**步骤3** 选择左侧导航栏“应用模块 > 模块管理”页签，查看已部署边缘应用。

图 2-73 模块管理

模块ID	模块名称	所属应用	版本	应用类型	实例状态	操作
sys_edge_keepalive	sys_edge_keepalive	Sedge_keepalive	1.1.40-standard-x86	系统可选	运行中	删除 升级

----结束

## 添加主备配置

**步骤1** 选择您的边缘节点，单击“节点名称”进入节点详情页。

**步骤2** 在左侧导航栏选择“主备配置”单击“主备配置”。

图 2-74 主备配置



**步骤3** 信息填写后，单击“确认”。

图 2-75 网络配置

### 主备配置 ×

**i** 提示：如果主备配置的网卡子网掩码小于24，请确保在同一局域网下，其他主备节点的“浮动IP地址”最后一段，不与此主备节点“浮动IP地址”的最后一段相同。否则，会导致冲突的2套主备节点出现异常，主备频繁切换。（“浮动IP地址”的最后一段：如浮动IP地址为192.168.0.88，则浮动IP地址的最后一段为“88”） ×

★ 主网卡名称

★ 备网卡名称

★ 浮动IP地址

IPv6 地址

确认
取消

表 2-20 主备配置

参数名称	描述
主网卡名称	表示主节点上虚拟IP所绑定使用的网卡名字，如eth0、eth1等，注意名称不要写错，写错会导致vip绑定失败。
备网卡名称	表示备节点上虚拟IP所绑定使用的网卡名字，如eth0、eth1等。
浮动虚拟IP	表示对外提供的一个虚拟IP（IPv4），供设备连接使用，具体虚拟IP按照自身业务规划设定。
IPv6地址	表示对外提供的一个IPv6地址，供设备连接使用，具体按照自身业务规划设定。

**步骤4** 查看ip信息。使用ip a查看主节点ip信息，如果当前设置均没有错误的话，在主节点上机器上已经可以查到已绑定的虚拟IP，例如当前设置的vip为172.30.0.201，主节点绑定的网卡为eth0。

图 2-76 查看 IP 信息

```
[root@ecs-tjj-01 ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether fa:16:3e:7c:6b:c7 brd ff:ff:ff:ff:ff:ff
    inet 172.30.0.69/24 brd 172.30.0.255 scope global noprefixroute dynamic eth0
        valid_lft 58032sec preferred_lft 58032sec
    inet 172.30.0.201/24 brd 172.30.0.255 scope global secondary eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fe7c:6bc7/64 scope link
        valid_lft forever preferred_lft forever
3: br-614f66ff0128: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:3d:96:ff:fa brd ff:ff:ff:ff:ff:ff
    inet 172.20.0.1/16 brd 172.20.255.255 scope global br-614f66ff0128
        valid_lft forever preferred_lft forever
    inet6 fe80::42:3dff:fe96:fffa/64 scope link
```

### 说明

查看网卡名称可以使用ifconfig查看，如下：

图 2-77 查看网卡名称

```
[root@ecs-tjj-01 ~]# ifconfig
br-614f66ff0128: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.20.0.1 netmask 255.255.0.0 broadcast 172.20.255.255
    inet6 fe80::42:3dff:fe96:fffa prefixlen 64 scopeid 0x20<link>
    ether 02:42:3d:96:ff:fa txqueuelen 0 (Ethernet)
    RX packets 1219 bytes 193144 (188.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1151 bytes 647295 (632.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:5e:bf:4b:69 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.30.0.69 netmask 255.255.255.0 broadcast 172.30.0.255
    inet6 fe80::f816:3eff:fe7c:6bc7 prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:7c:6b:c7 txqueuelen 1000 (Ethernet)
    RX packets 4756635 bytes 2942640862 (2.7 GiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 7458021 bytes 981313731 (935.8 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

----结束

## 安装备节点

**步骤1** 选择您的边缘节点，单击“节点名称”进入节点详情页。

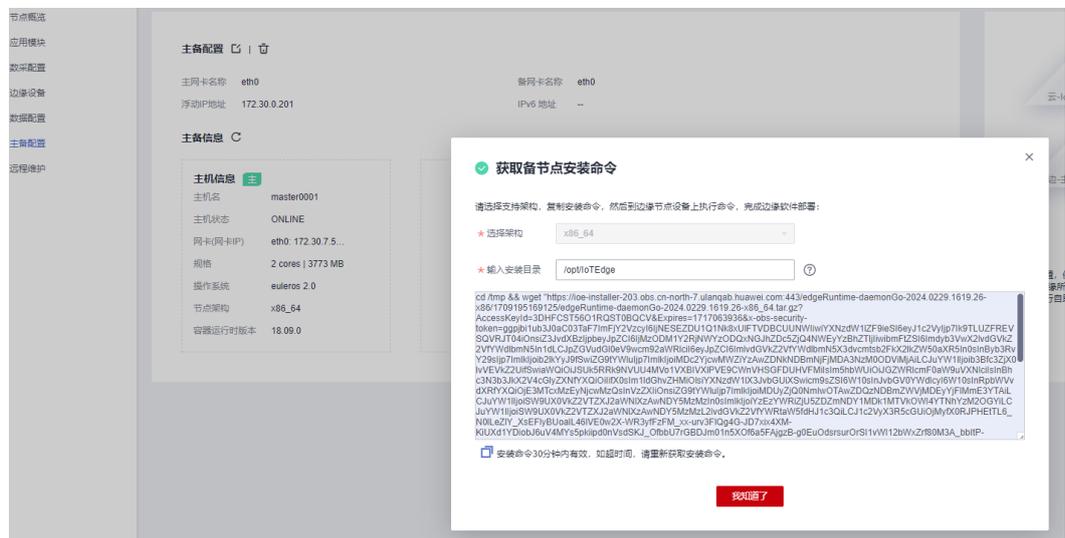
**步骤2** 在左侧导航栏选择“主备配置”页面获取“获取备节点安装命令”。

### 注意

主节点和备节点设置绑定的网卡之间网络一定需要能够正常访问，否则会出现主备异常。

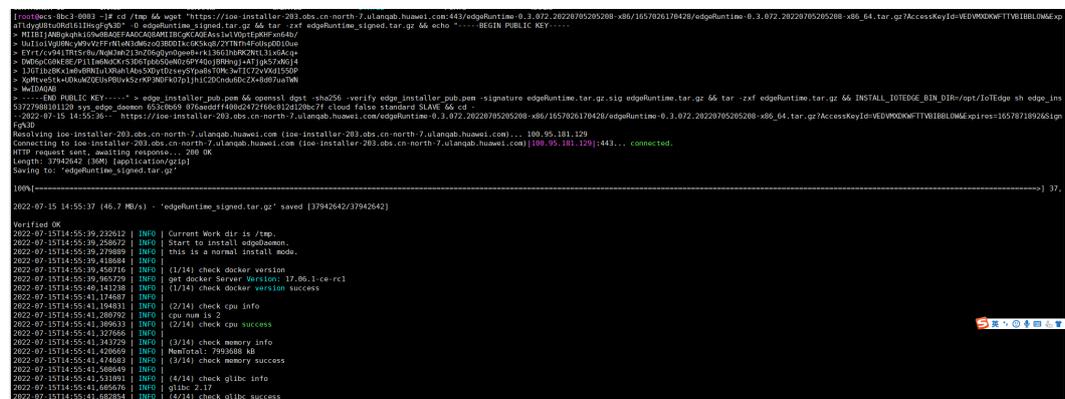
登录边缘节点主备用机器，检查主备机时间偏差，保持在30S以内，否则可能会出现备节点安装失败。

图 2-78 安装备节点



**步骤3** 登录边缘节点备用机器，执行安装命令。

图 2-79 执行安装命令



如下表示备节点安装成功。

图 2-80 安装成功



备节点默认只运行edge\_agent以及edge\_keepalive模块，其余模块为created状态。

图 2-81 运行状态

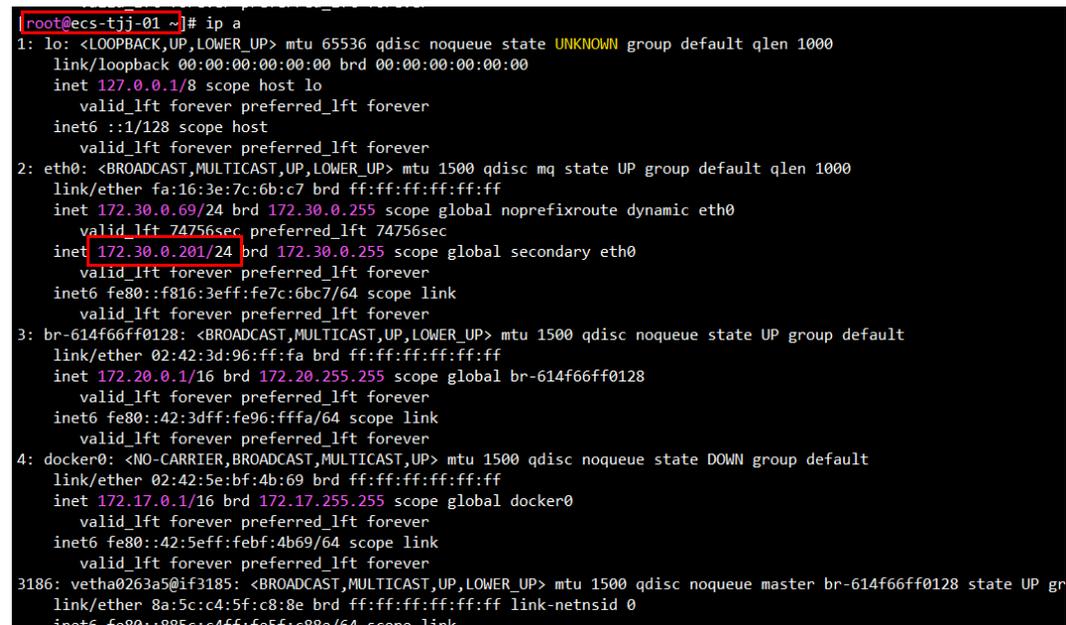


----结束

### 确认当前虚拟 ip 绑定情况

查看主节点机器，当前主节点eth0网卡绑定了vip，为工作节点。

图 2-82 查看 IP 信息



查看备节点机器的ip列表，eth0网卡没有绑定vip

图 2-83 查看网卡名称

```

[ecs-8bc3-0003 ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP qlen 1000
    link/ether fa:16:3e:73:2d:f9 brd ff:ff:ff:ff:ff:ff
    inet 172.30.0.144/24 brd 172.30.0.255 scope global dynamic eth0
        valid_lft 52998sec preferred_lft 52998sec
    inet6 fe80::f816:3eff:fe73:2df9/64 scope link
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
    link/ether 02:42:14:d0:2d:ed brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 scope global docker0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:14ff:fed0:2ded/64 scope link
        valid_lft forever preferred_lft forever
107: br-fc161b9bbbb0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP
    link/ether 02:42:9a:2a:bd:55 brd ff:ff:ff:ff:ff:ff
    inet 172.20.0.1/24 scope global br-fc161b9bbbb0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:9aff:fe2a:bd55/64 scope link
        valid_lft forever preferred_lft forever
1239: veth946cdb2@if1238: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-fc161b9bbbb0 state UP
    link/ether 1a:a0:12:93:ae:97 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::18a0:12ff:fe93:ae97/64 scope link
    
```

查看console的主备配置节点，可以看到工作节点为机器 ecs-tjj-01(主机信息后的绿色主标签表示工作节点)

图 2-84 查看主备状态



至此，主备配置已结束。

## 简单测试

模拟当前工作节点发生故障，不再往备节点发送心跳包。

- 停止主节点sys\_edge\_keepalive来模拟宕机情况

图 2-85 模拟宕机

```

[ecs-tjj-01 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED        STATUS        PORTS          NAMES
f56947806af        image              "/opt/iot/edge_kee..." 18 minutes ago Up 18 minutes          sys-edge-keepalive
816ff8f08a47        100.95.181.176:5300/huawei-iot-edge/edgehub-go:2022.0712.1051.01 "/opt/iot/edge_hub/..." 42 minutes ago Up 42 minutes          0.0.0.0:7883->7883/tcp, 0.0.0.0:8943->8943/tcp
91f83043166        registry.cn-huawe... sar-cn-north-3.myhuaweicloud.com/huawei-iot-edge/edgeaccess:2022.0022.0027.001 "/opt/iot/edge_acc..." 43 minutes ago Up 43 minutes          0.0.0.0:7000->7000/tcp
823184765af        sar-cn-north-3.myhuaweicloud.com/huawei-iot-edge/edgeagent_x86_64-0.3.072.20220715114248 "/bin/sh -c 'sh /opt..." 43 minutes ago Up 43 minutes          sys-edge-agent
f56947806af        [ecs-tjj-01 ~]# docker stop f56947806af
f56947806af

[ecs-tjj-01 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED        STATUS        PORTS          NAMES
823184765af        sar-cn-north-3.myhuaweicloud.com/huawei-iot-edge/edgeagent_x86_64-0.3.072.20220715114248 "/bin/sh -c 'sh /opt..." 44 minutes ago Up 44 minutes          sys-edge-agent
f56947806af        [ecs-tjj-01 ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND                  CREATED        STATUS        PORTS          NAMES
0b788226c6a        100.95.181.176:5300/huawei-iot-edge/edgekeepalive:2022.0425.1134.55 "/opt/iot/edge_kee..." 1 second ago    Up less than a second          sys-edge-keepalive
c8189602771        registry.cn-huawe... sar-cn-north-3.myhuaweicloud.com/huawei-iot-edge/edgehub-go:2022.0022.0027.001 "/opt/iot/edge_acc..." 8 seconds ago   Created          sys-edge-access
5ea3771a852        100.95.181.176:5300/huawei-iot-edge/edgehub-go:2022.0712.1051.01 "/opt/iot/edge_hub/..." 20 seconds ago  Created          sys-edge-hub
823184765af        sar-cn-north-3.myhuaweicloud.com/huawei-iot-edge/edgeagent_x86_64-0.3.072.20220715114248 "/bin/sh -c 'sh /opt..." 44 minutes ago Up 44 minutes          sys-edge-agent
    
```

停止后查看应用状态，sys\_edge\_hub已进入created状态。查看浮动ip情况，发现原主节点vip已移除。

图 2-86 查看 IP 信息

```
root@ecs-tjj-01 ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether fa:16:3e:7c:6b:c7 brd ff:ff:ff:ff:ff:ff
    inet 172.30.0.69/24 brd 172.30.0.255 scope global noprefixroute dynamic eth0
        valid_lft 73792sec preferred_lft 73792sec
    inet6 fe80::f816:3eff:fe7c:6bc7/64 scope link
        valid_lft forever preferred_lft forever
3: br-614f66ff0128: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:3d:96:ff:fa brd ff:ff:ff:ff:ff:ff
    inet 172.20.0.1/16 brd 172.20.255.255 scope global br-614f66ff0128
        valid_lft forever preferred_lft forever
    inet6 fe80::42:3dff:fe96:fffa/64 scope link
        valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:5e:bf:4b:69 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:5eff:feb9:4b69/64 scope link
        valid_lft forever preferred_lft forever
vetha0263a5@if3185: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-614f66ff0128 state UP
    link/ether 8a:5c:c4:5f:c8:8e brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::885c:c4ff:fe5f:c88e/64 scope link
        valid_lft forever preferred_lft forever
```

- 查询查看备节点浮动ip和模块运行状态。

图 2-87 查看运行状态

```
root@ecs-8bc3-0003 ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP qlen 1000
    link/ether fa:16:3e:7c:6b:c7 brd ff:ff:ff:ff:ff:ff
    inet 172.30.0.14/24 brd 172.30.0.255 scope global dynamic eth0
        valid_lft 3028sec preferred_lft 3028sec
    inet 172.30.0.203/24 brd 172.30.0.255 scope global secondary eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fe7c:20f9/64 scope link
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
    link/ether 02:42:5e:bf:4b:69 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 scope global docker0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:5eff:feb9:4b69/64 scope link
        valid_lft forever preferred_lft forever
vetha0263a5@if3185: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP
    link/ether 8a:5c:c4:5f:c8:8e brd ff:ff:ff:ff:ff:ff
    inet 172.20.0.1/16 scope global br-fc93b02600
        valid_lft forever preferred_lft forever
    inet6 fe80::42:5eff:feb9:4b69/64 scope link
        valid_lft forever preferred_lft forever
```

浮动ip已经绑定到备节点eth0网卡上，且业务模块都在原备节点上正常运行。

- 在console查看主备信息，工作节点已经切换到主机名为ecs-8bc3-0003的备用机器上。

图 2-88 查看主备状态

主备配置  | 

主网卡名称	eth0	备网卡名称	eth0
浮动IP地址	172.30.0.201		

主备信息 

**主机信息**

主机名	ecs-tij-01
主机状态	ONLINE
网卡(网卡IP)	eth0: 172.30.0.6...
规格	4 cores   7820 MB
操作系统	centos 8
节点架构	x86_64
容器运行时版本	18.09.8

**主机信息** 主

主机名	ecs-8bc3-0003
主机状态	ONLINE
网卡(网卡IP)	eth0: 172.30.0.1...
规格	2 cores   7806 MB
操作系统	euleros 2.0
节点架构	x86_64
容器运行时版本	17.06.1-ce-rc1

至此，简单主备切换测试已经完成。

## 备节点异常重装

当备节点因为一些异常原因不能够继续使用时，需要更换服务器来继续提供备节点的功能支持，需要进行备节点的重新安装动作。

---

### 注意

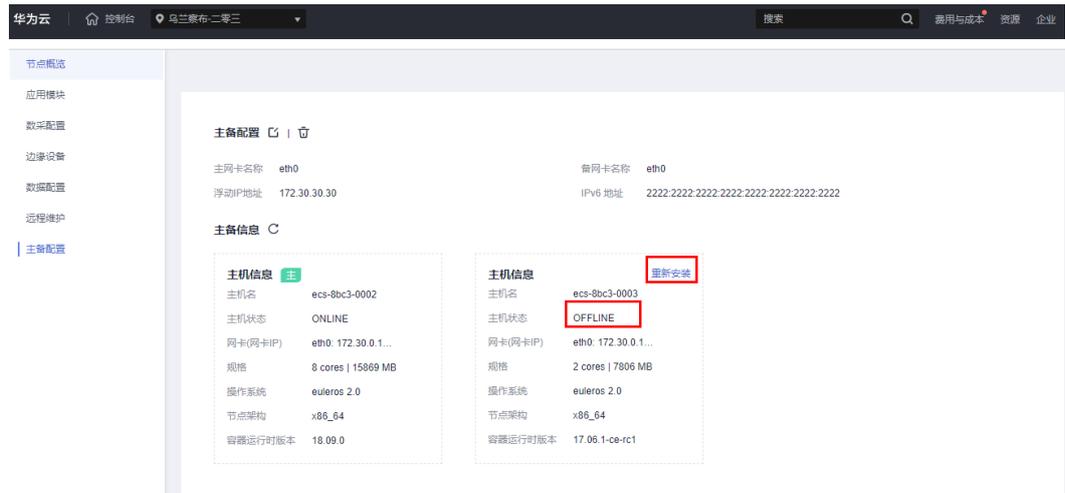
前提条件：

备节点主机状态已经离线且不会继续进行工作，如下图所示。

手动进行原有异常备节点的卸载动作，以防后续恢复正常后影响当前主备的正常工作。

---

图 2-89 重装备节点



**步骤1** 在主备配置页签，单击备节点“重新安装”字样，获取安装命令。



请确定原有备节点主机状态已经离线且不会继续进行工作。

图 2-90 获取备节点安装命令



**步骤2** 登录新的备节点机器，执行备节点的重装命令。

图 2-91 执行安装命令

```

[root@ecs-8bc3-0003 ~]#
[root@ecs-8bc3-0003 ~]# cd /tmp && wget "https://ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com:443/edgeRuntime-0.3.072.20220831104732_x86_1661914937140/edgeRuntime-0.3.072.20220831104732_x86_64.tar.gz?accesskeyid=qwefw3j0qj10T06AVX6Espiras=166720831865signature=ThyCAbUGVfVf1l9uq6E5AjoFPk2BnE3D" -O edgeRuntime_signed.tar.gz && echo "-----BE
GIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAACQAM1BCCgCAQEAsIwVlVpTepMhFmG4b/
Uu1tCVWUWkcW9wVzFmleE3m0zQ8D01kCCK8q9ZyTRf8r4uIsU0D1u0e
> EYrt/cv941TRTSrdu/nqM2m213mZ06gYm0qea9rkk36ghhRkNtL3ixGAc+
> D0dpcCCKeE/p1im0NCKrs3dE7p0C0eN0z6P/4q0j0rWmjYATjgk5Yw0c4
> 1IOT4y8Xkmp9B8tU1lY8ah4bsCS0jDZssy0pas3T0m3uTfCZvWd3550P
> XpMvve3tk+U0kUwZQe5PBuVksZrKP3NDfK07p1jhiCZDcndu6CZX48D07uaTWN
> WtDARqB
-----END PUBLIC KEY-----" > edge_installer_pub.pem && openssl dgst -sha256 -verify edge_installer_pub.pem -signature edgeRuntime.tar.gz.sig edgeRuntime.tar.gz &&
6 INSTALL_IOTEDGE_BIN_DIR=/opt/IoTEdge sh edge_install.sh 100.04.00.201 8943 765074272428666880 sys_edge_daemon dd718df0 60b459ef692f4bd9bd58dca9029dia cloud false standard SLAVE && cd
-2022-10-31 15:55:34 - https://ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com/edgeRuntime-0.3.072.20220831104732_x86_1661914937140/edgeRuntime-0.3.072.20220831104732_x86_64.tar.gz?access
keyid=qwefw3j0qj10T06AVX6Espiras=166720831865signature=ThyCAbUGVfVf1l9uq6E5AjoFPk2BnE3D
Resolving ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com (ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com)... 100.95.181.129
Connecting to ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com [100.95.181.129]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 38013328 (36M) [application/gzip]
Saving to: 'edgeRuntime_signed.tar.gz'

100%[=====] 38,013,328  66.7MB/s  in 0.5s

2022-10-31 15:55:35 (66.7 MB/s) - 'edgeRuntime_signed.tar.gz' saved [38013328/38013328]

Verified OK
2022-10-31T15:55:39.257042 INFO | Current Work dir is /tmp.
2022-10-31T15:55:39.284087 INFO | Start to install edgeDaemon.
2022-10-31T15:55:39.305200 INFO | this is a normal install mode.
2022-10-31T15:55:39.428777 INFO |
2022-10-31T15:55:39.451038 INFO | (1/14) check docker version
2022-10-31T15:55:42.353075 INFO | get docker Server Version: 17.06.1-ce-rc1
2022-10-31T15:55:42.930050 INFO | (1/14) check docker version success
2022-10-31T15:55:43.553837 INFO |
2022-10-31T15:55:43.578052 INFO | (2/14) check cpu info
2022-10-31T15:55:43.655081 INFO | cpu num is 4
2022-10-31T15:55:43.673832 INFO | (2/14) check cpu success
2022-10-31T15:55:43.692162 INFO |
2022-10-31T15:55:43.709658 INFO | (3/14) check memory info
2022-10-31T15:55:43.783078 INFO | MemTotal: 7993688 kb
2022-10-31T15:55:43.822254 INFO | (3/14) check memory success
2022-10-31T15:55:43.867021 INFO |
2022-10-31T15:55:43.896016 INFO | (4/14) check glibc info
2022-10-31T15:55:43.974065 INFO | glibc: 2.17
2022-10-31T15:55:44.047176 INFO | (4/14) check glibc success
2022-10-31T15:55:44.070779 INFO |
2022-10-31T15:55:44.083204 INFO | (5/14) start verify duplicate install iot edge lite
2022-10-31T15:55:44.108670 INFO | (5/14) check edge duplicate install success

```

如下表示备节点安装成功。

图 2-92 安装成功

```

2022-10-31T15:55:59.913791 INFO | reportDeploymentState | 79 | reportDeploymentState, deploymentState:DeploymentState[ node_id:765674272428666880, deploy_id:1667180929380, deploy_state:PENDING
MG, reason:{zero} ]
2022-10-31T15:56:00.896813 INFO | saveBasePathInfo | 74 | config base path start
2022-10-31T15:56:01.869617 INFO | saveBasePathInfo | 79 | config log_path:/var/IoTEdge/log/,db_path:/var/IoTEdge/db/, config_path:/var/IoTEdge/config/
2022-10-31T15:56:01.987079 INFO | saveBasePathInfo | 91 | config base path success.
2022-10-31T15:56:01.997139 INFO | addAgentCustomConfigs | 206 | addCustomConfigs start.
2022-10-31T15:56:01.987208 INFO | addAgentCustomConfigs | 230 | addCustomConfigs end.
2022-10-31T15:56:01.997258 INFO | pullImage | 385 | start pull image, deploy type is cloud
2022-10-31T15:56:03.637679 INFO | createModule | 175 | create Module, name:sys_edge_agent
2022-10-31T15:56:05.591787 INFO | createModule | 93 | module created, CreateRespBody: ModuleCreateResp[ moduleId:sys_edge_agent, moduleName:sys_edge_agent, appId:edge_agent, appVersion:1-537-
1-1mout-866, containerId:8bc3-421b927e9892ec58f32021928793f3a395fabab, processId:{zero}, type:docker, ]
2022-10-31T15:56:05.591954 INFO | startModule | 159 | startModule moduleId:sys_edge_agent
2022-10-31T15:56:13.461757 INFO | listModule | 245 | listModule
2022-10-31T15:56:13.483091 INFO | checkModule | 208 | start check modules, time:1
2022-10-31T15:56:13.584123 INFO | listModule | 245 | listModule
2022-10-31T15:56:13.696931 INFO | printModulesInReport | 349 | **SUCCESS** failed:0
2022-10-31T15:56:13.697967 INFO | reportDeploymentState | 74 | reportDeploymentState, deploymentState:DeploymentState[ node_id:765674272428666880, deploy_id:1667180929380, deploy_state:RUNNING

```

步骤3 查看console界面确认备节点已在线。

图 2-93 查看主备状态

The screenshot shows a console interface with several tabs: '节点概述', '应用模块', '数据采集', '边缘设备', '数据配置', '远程维护', and '主备配置'. The '主备配置' tab is selected, showing '主备配置' and '主备信息'. Under '主备信息', there are two columns of node details. The first column shows the primary node 'ecs-8bc3-0002' with status 'ONLINE'. The second column shows the secondary node 'ecs-8bc3-0003' with status 'ONLINE', which is highlighted with a red box.

至此备节点重装完成。

----结束

主节点异常重装

当主节点因为一些异常原因不能够继续使用时，需要更换服务器来继续提供主节点的功能支持，需要进行主节点的重新安装动作。

**注意**

主节点主机状态已经离线且不会继续进行工作，如下图所示。

手动进行原有异常主节点的卸载动作，以防后续恢复正常后影响当前主备的正常工作。

图 2-94 重装主节点



**步骤1** 在主备配置页签，单击主节点“重新安装”字样，获取安装命令。

**注意**

请确定原有主节点主机状态已经离线且不会继续进行工作。

图 2-95 获取主节点安装命令



**步骤2** 登录新的主节点机器，执行主节点的重装命令。

图 2-96 执行安装命令

```
2022-10-31 16:01:46 | INFO | uminstall edgeRuntime finish.
[ecs-8bc3-0002] # cd /tmp && wget "https://ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com/443/edgeRuntime/0.3.072.20220831104732-x86_64/1661914937140/edgeRuntime.0.3.072.20220831104732-x86_64.tar.gz?AccessKeyId=QWTFW3J0H1DVT06AVX&Expires=166720716465&Signature=bcG8yYf28mSKvMk0QWmuh8Q4h30" -O edgeRuntime_signed.tar.gz && tar -xzf edgeRuntime_signed.tar.gz && echo "-----BEGIN PUBLIC KEY-----
MIIEIjANBgkqhkiG9w0BAQEFAAOCAQAMIIBcQKAgEAsIuVpTEpKHFm64b/
UuioiVgUeNcyW9VzFFmLeN3dWzq38001kCkKq8/2YTNFh4F0Usp0D10ue
EVRT/cv94I/TRTSfou/Ngm2h213Z0ZegYmQge0rRk136G1hRkZNL1ixGkAg
DnGpccKKEEcp1TeHdKvS36T1pKcCwE9v0v0d0j8RwmgjAT1gK57MgY4
13GTlb2Kx1m0vBRNtULxrah1Abs5XdytDzsay5YpA8TOM3wTIC72vXkD15SDP
XPMVes1k+UDkZQZEUspBUvKsZrKP3NDFK07p1jhiC2CDuDuDCZKx8d7ua1W
W4D1u0v8
-----END PUBLIC KEY-----" > edge_installer_pub.pem && openssl dgst -sha256 -verify edge_installer_pub.pem -signature edgeRuntime.tar.gz.sig edgeRuntime.tar.gz && tar -xzf edgeRuntime.tar.gz &&
INSTALL_IOEDGE_BIN_DIR=/opt/IOEdge sh edge_install.sh 100.94.50.201 8943 76567427242866880 sys-edge_daemon fileds37 6084596927483089d8d86826d1a cloud false standard0 DEFAULT 66 cd -
2022-10-31 16:07:08 | https://ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com/443/edgeRuntime/0.3.072.20220831104732-x86_64/1661914937140/edgeRuntime.0.3.072.20220831104732-x86_64.tar.gz?Access
KeyId=QWTFW3J0H1DVT06AVX&Expires=166720716465&Signature=bcG8yYf28mSKvMk0QWmuh8Q4h30
Resolving ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com (ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com)... 180.95.181.129
Connecting to ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com (ioe-installer-203.obs.cn-north-7.ulanqab.huawei.com)|180.95.181.129|:443... connected.
HTTP request sent, awaiting response... 200 OK
length: 38013328 (36M) [application/gzip]
Saving to: 'edgeRuntime_signed.tar.gz'

===== 38,013,328 59.2MB/s in 0.6s

2022-10-31 16:07:09 (59.2 MB/s) - 'edgeRuntime_signed.tar.gz' saved [38013328/38013328]

Verified OK
2022-10-31 16:07:10.892895 | INFO | Current work dir is /tmp.
2022-10-31 16:07:10.894334 | INFO | Start to install edgeDaemon.
2022-10-31 16:07:10.895968 | INFO | This is a normal install mode.
2022-10-31 16:07:10.904607 | INFO |
2022-10-31 16:07:10.905776 | INFO | (1/14) check docker version
2022-10-31 16:07:11.121544 | INFO | get docker Server Version: 18.06.0
2022-10-31 16:07:11.130493 | INFO | (1/14) check docker version success
2022-10-31 16:07:12.133803 | INFO |
2022-10-31 16:07:12.135456 | INFO | (2/14) check cpu info
2022-10-31 16:07:12.142271 | INFO | cpu num is 8
2022-10-31 16:07:12.144130 | INFO | (2/14) check cpu success
2022-10-31 16:07:12.146280 | INFO |
2022-10-31 16:07:12.147734 | INFO | (3/14) check memory info
```

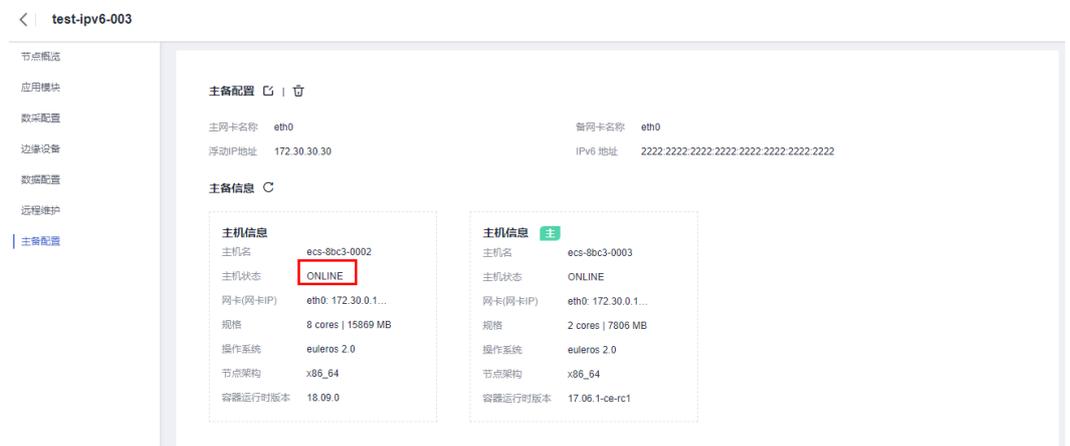
如下表示主节点安装成功。

图 2-97 安装成功

```
NG_reason: {zero}
2022-10-31 16:07:30.200419 | INFO | saveBasePathInfo | 74 | config base path start.
2022-10-31 16:07:30.200518 | INFO | saveBasePathInfo | 79 | config log_path: /var/IOEdge/log/, db_path: /var/IOEdge/db/, config_path: /var/IOEdge/config/
2022-10-31 16:07:30.200751 | INFO | saveBasePathInfo | 91 | config base path success.
2022-10-31 16:07:30.200800 | INFO | addAgentCustomConfigs | 200 | addCustomConfigs start.
2022-10-31 16:07:30.200856 | INFO | addAgentCustomConfigs | 230 | addCustomConfigs end.
2022-10-31 16:07:30.200894 | INFO | pullImage | 305 | start pull image, deploy type is cloud
2022-10-31 16:07:30.218242 | INFO | createModule | 175 | create Module, name: sys-edge-agent
2022-10-31 16:07:30.266906 | INFO | createModule | 93 | module created, CreateRespBody: ModuleCreateResp{ moduleId: sys-edge-agent, moduleName: sys-edge-agent, appId: sedge_agent, appVersion: 1-537-
timeout: 480, containerId: B92255Sec5FApa3672b577fbc82ed9d3d096fa51f982774a401fa26, processId: (zero), type: docker, }
2022-10-31 16:07:30.460153 | INFO | startModule | 159 | startModule moduleId: sys-edge-agent
2022-10-31 16:07:35.964158 | INFO | listModule | 245 | listModule
2022-10-31 16:07:35.964158 | INFO | listModule | 245 | listModule
2022-10-31 16:07:36.067258 | INFO | listModule | 245 | listModule
2022-10-31 16:07:36.170038 | INFO | printModulesAndReport | 349 | **SUCCESS! failed!0
2022-10-31 16:07:36.170159 | INFO | reportDeploymentState | 79 | reportDeploymentState, deploymentState: DeploymentState{ node_id: 76567427242866880, deploy_id: 1667203313840, deploy_state: RUNNING
```

步骤3 查看console界面确认主节点已在线。

图 2-98 查看主备状态



至此，主节点重装完成。

---结束

## 2.9 远程维护

远程维护是IoT边缘网关提供的Web版客户端工具，可帮助您使用SSH协议进行远程访问主机或下载edge应用模块日志。

## 前提条件

- 专业版和基础版边缘节点均支持远程维护功能。
- 节点状态为未安装/升级中/删除中时，无法使用远程登录功能。
- 远程主机安装了SSH服务端程序，并正常监听22端口，仅下载模块日志的话可以忽略这条。
- 节点已经部署\$edge\_omagent模块且运行正常。

### 注意

- 输入的远程主机账号密码错误，则无法登录节点。
- 节点状态是离线时，可能会因无法正常建立通道，导致功能不可用。
- 只支持一个用户在线登录，新用户登录会造成老用户下线。
- 页签切换，会造成客户端通道关闭，需要重新登录。
- 当应用\$edge\_agent大于1-1-6版本时，不再支持远程SSH功能，如需使用，则需部署\$edge\_omagent应用1-1-11或以上版本。

## 操作步骤

**步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 节点管理”进入页面。

**步骤3** 选择您的边缘节点，单击“节点名称”进入节点详情页。

**步骤4** 选择“远程维护”页签，单击“节点登录”按钮。

图 2-99 节点登录



**步骤5** 弹出登录SSH服务端弹窗，输入远程访问主机的账号密码后单击“确认”，例如：  
root/123456。

图 2-100 节点登录



登录

\* 用户名

\* 密码

确认 取消

**步骤6** 若账户密码正确，则登录成功，界面上回显远程主机登录成功的返回信息。同时可以单击进行目录树展开。

图 2-101 登录成功

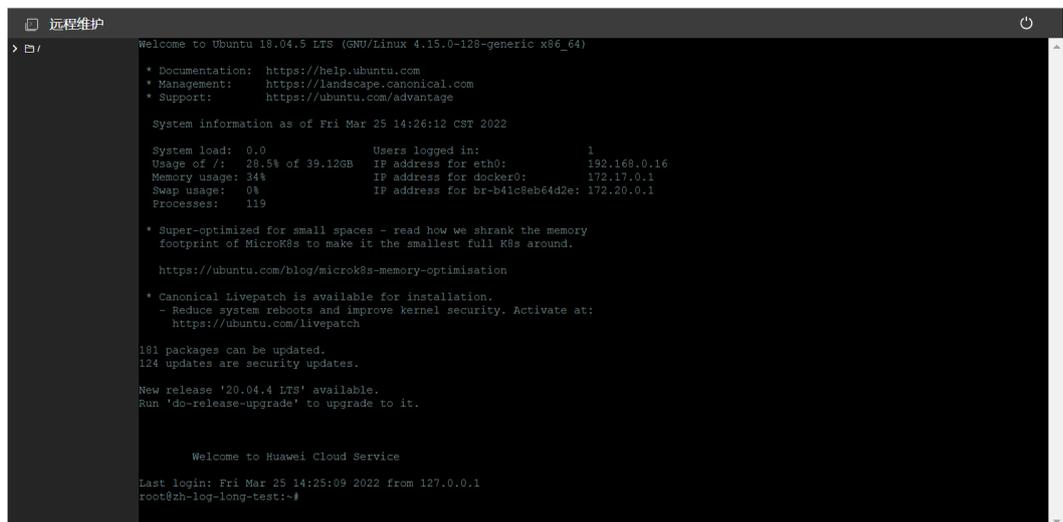
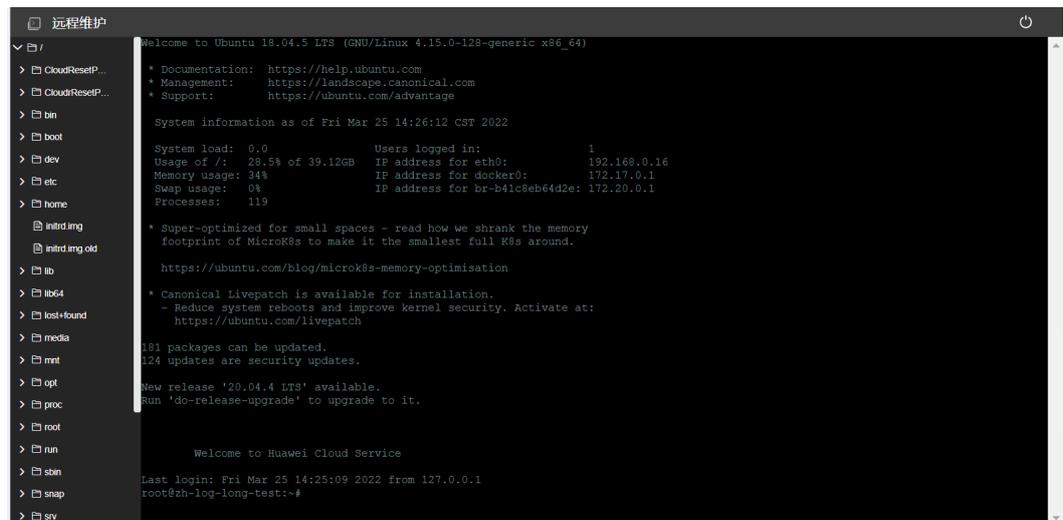


图 2-102 目录树展示



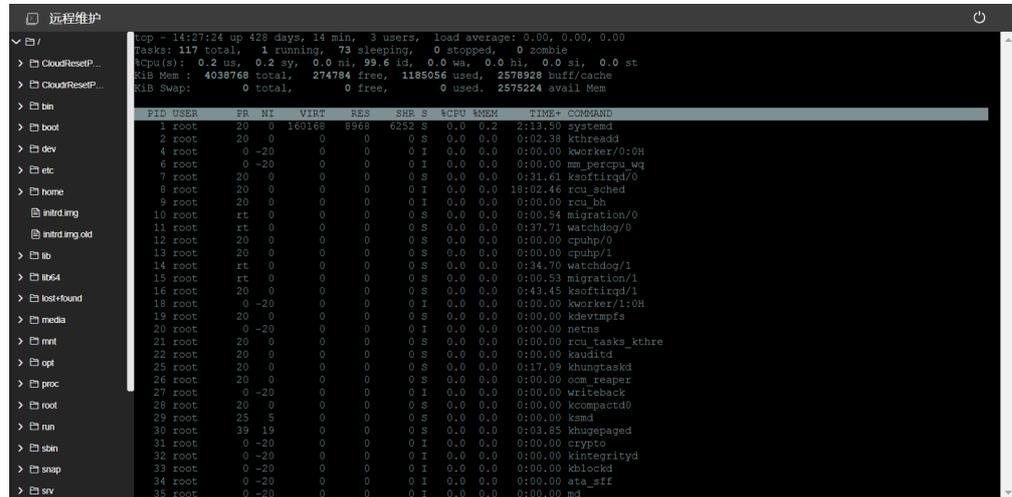
**步骤7** 界面操作，您可以通过命令行的方式与主机进行交互。

示例如下：

1. 查看系统当前运行状况，执行命令：

> top

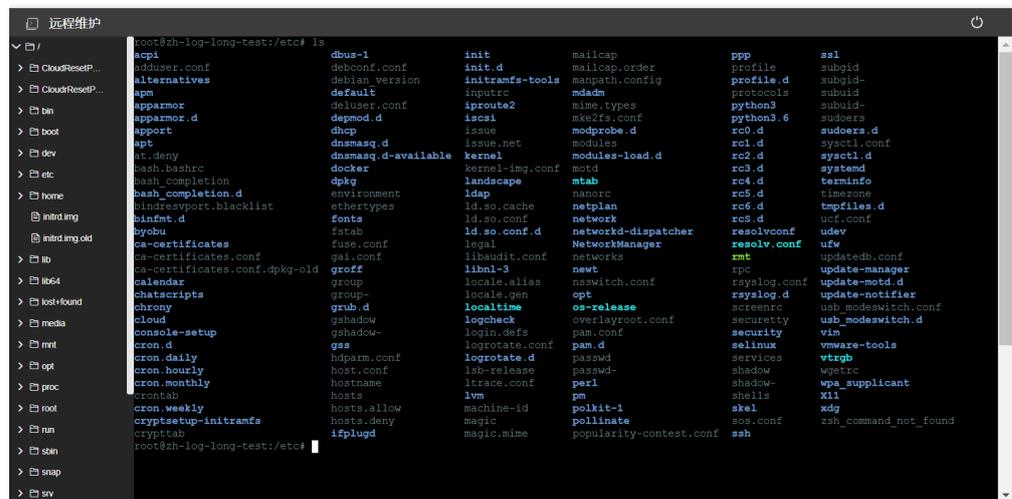
图 2-103 top 回显



2. 查看 /etc 目录下文件和目录，执行命令：

> ls

图 2-104 ls 回显



### 步骤8 文件上传下载功能

**注意**

1. 只允许对文件进行下载。
2. 只允许在文件夹上进行文件的上传，表示将文件上传到远程主机对应的文件夹下。
3. 上传下载的文件大小最大为10M。
4. 文件上传下载所操作的目录受创建节点时所配置的根目录限制，如下图所示。



5. 使用文件上传下载功能，需要部署\$edge\_omagent大于1-1-11的版本。

在指定文件夹或者文件单击鼠标右键，可以展示文件上传下载功能，如下图：

图 2-105 文件文件上传下载功能展示

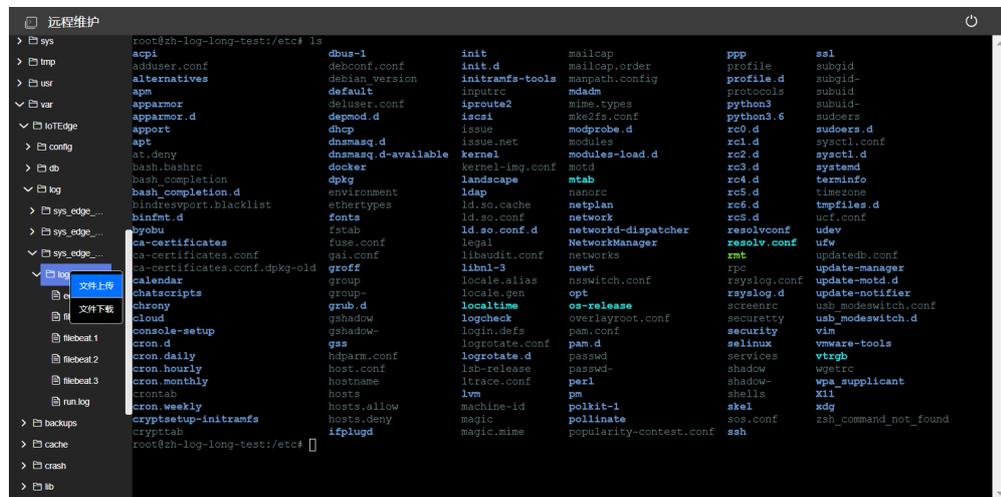


图 2-106 上传文件

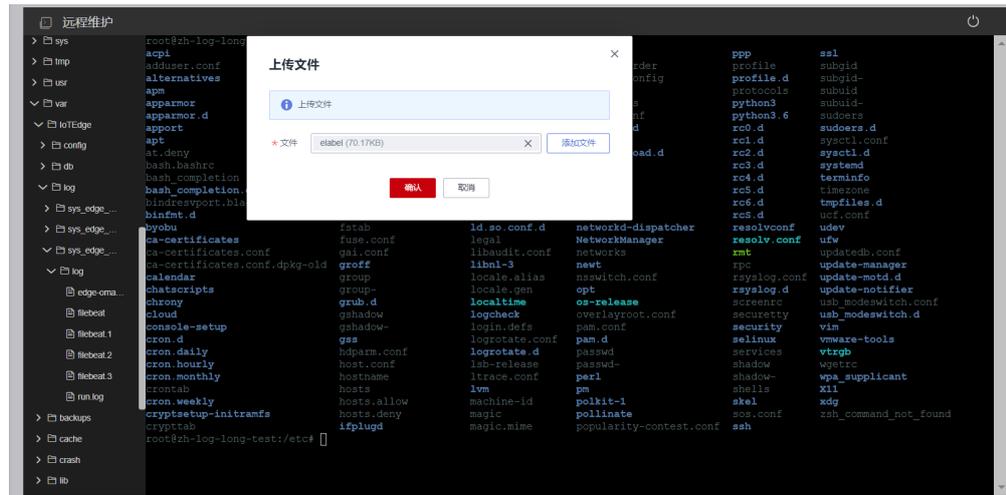
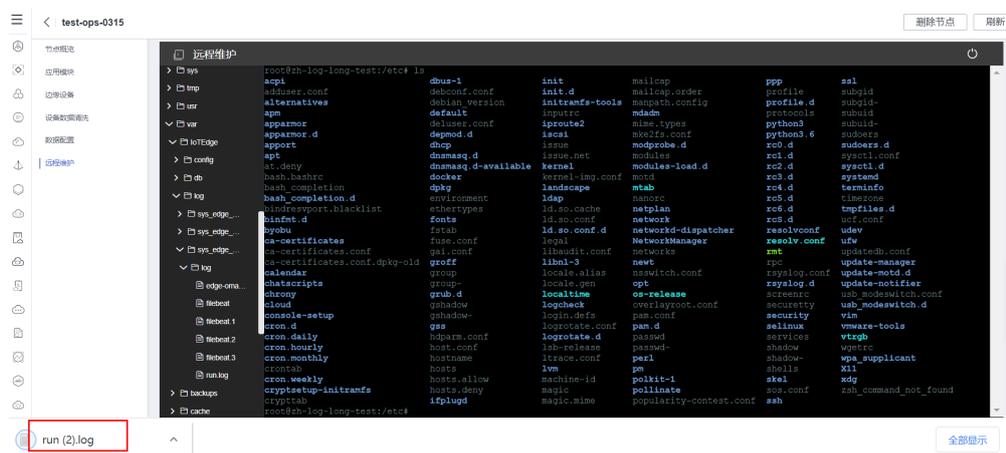


图 2-107 下载文件



**步骤9** 退出节点登录。

您可以直接关闭浏览器、刷新界面、单击右上角关闭按钮、或在操作界面输入“exit”命令即可退出节点登录。

**步骤10** 返回“远程维护”页签，单击“下载日志”。

图 2-108 下载日志



**步骤11** 选择日志文件，单击“确认”。

图 2-109 选择下载文件

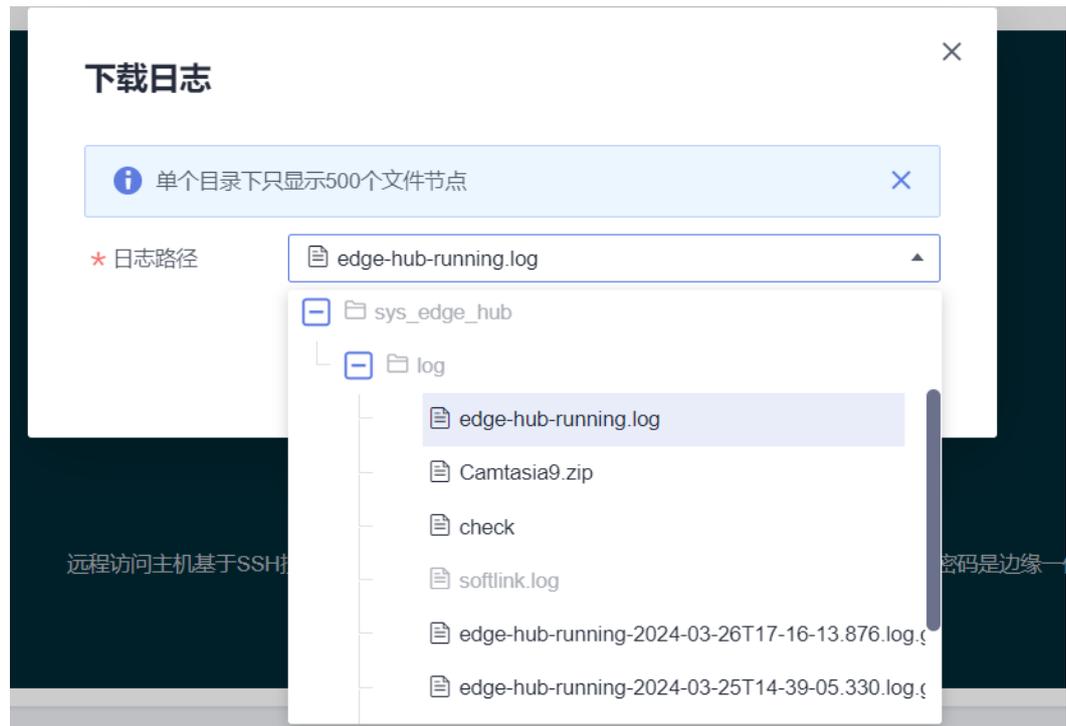


图 2-110 单击确认下载文件



**步骤12** 浏览器下载远程文件到本地下载目录。

----结束

**⚠ 危险**

远程主机需要安装了SSH服务端程序，并正常监听22端口，使用过程中需要输入账号和密码，请用户妥善保存登录信息。

## 2.10 本地自治

IoT边缘通过架构设计使其能够在网络受限时仍然实现边缘节点自治，具体体现在以下四个方面：

1. **边缘节点持续运行**：即使边缘节点无法连接到云端服务，也可以保证已部署的数据采集、数据处理和数据推送等边缘模块持续运行。
2. **边缘模块自我恢复**：在网络断开的情况下，如果边缘节点上的任何模块发生异常退出，系统可以自动重新启动它，确保服务的稳定性。
3. **离线数据边侧缓存**：当节点无法将数据上报到云端时，节点会将数据缓存在本地。数据不会因为暂时的网络问题而丢失，在网络恢复后重新尝试上报。
4. **自动恢复离线操作**：当网络恢复后，边缘节点会自动连接到IoT边缘服务。节点离线期间进行的所有管理操作，将在节点恢复在线后下发执行。管理操作包括但不限于：部署、升级或删除应用模块；管理数据流转规则；管理数据采集配置；管理离线缓存配置；管理日志采集开关。

以上这些架构设计使边缘节点能够在网络不稳定的环境中保持运行，并在网络恢复后，自动同步离线期间的管理操作，从而实现在网络断开时的本地自治。

## 2.11 删除边缘节点

删除边缘节点可以参考本章内容操作步骤。

### 操作步骤

**步骤1** 访问**IoT边缘**，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 节点管理”进入页面，查看所有的边缘节点。

图 2-111 删除节点



**步骤3** 选择需要删除的节点，单击右侧“删除”按钮，请您仔细阅读提示并进行如下操作。

图 2-112 专业版删除提示



**步骤4** （可选）解绑节点上关联的边缘设备。

#### 📖 说明

要将边缘节点上的所有子设备都删除完毕后，才允许删除边缘节点。

**步骤5** 卸载节点上的边缘软件。

复制删除提示中的命令，使用SSH工具，连接到边缘节点的设备上以root用户执行命令，删除边缘节点上的软件和配置文件。

**步骤6** 单击“确认”，等待边缘节点删除完成即可。

#### 📖 说明

删除专业版节点（KubeEdge），只删除了IoT边缘平台的节点，并未删除智能边缘平台（IEF）的节点。

#### ---结束

#### ⚠️ 危险

删除节点会同步删除在IOT边缘创建的节点下部署的应用、模块、函数、证书。删除操作无法恢复，请谨慎操作！

## 2.12 重装边缘节点

### 使用约束

- 重新安装边缘节点时，边缘节点需要处于离线状态。
- 边缘节点类型是基础版或专业版节点（Docker），不支持专业版节点（KubeEdge）。
- 含有主备配置的节点，主节点重装参考[主节点异常重装](#)，备节点重装请参考[备节点异常重装](#)。

### 操作步骤

**步骤1** 在“边缘节点”的节点列表选择您需要重新安装的节点，进入节点详情页面。

图 2-113 边缘节点列表

节点名称	节点类型	资源包类型	资源包ID	状态
	专业版	--	--	离线
TEST_SS	专业版	--	--	离线
	专业版	--	--	离线

**步骤2** 进入节点概览页面，单击右上角重新安装，进入重新安装流程。

图 2-114 节点概览



**步骤3** 仔细阅读橙色方框内的注意事项，确认需要重新安装边缘节点后，单击获取安装命令，然后单击  拷贝安装命令，使用SSH工具以root用户登录新的边缘节点服务器后台系统，执行重装命令。

#### 注意

单击获取安装命令后，节点的密钥和验证码会立即被重置，会影响当前边缘节点的业务运行。

图 2-115 重新安装提示

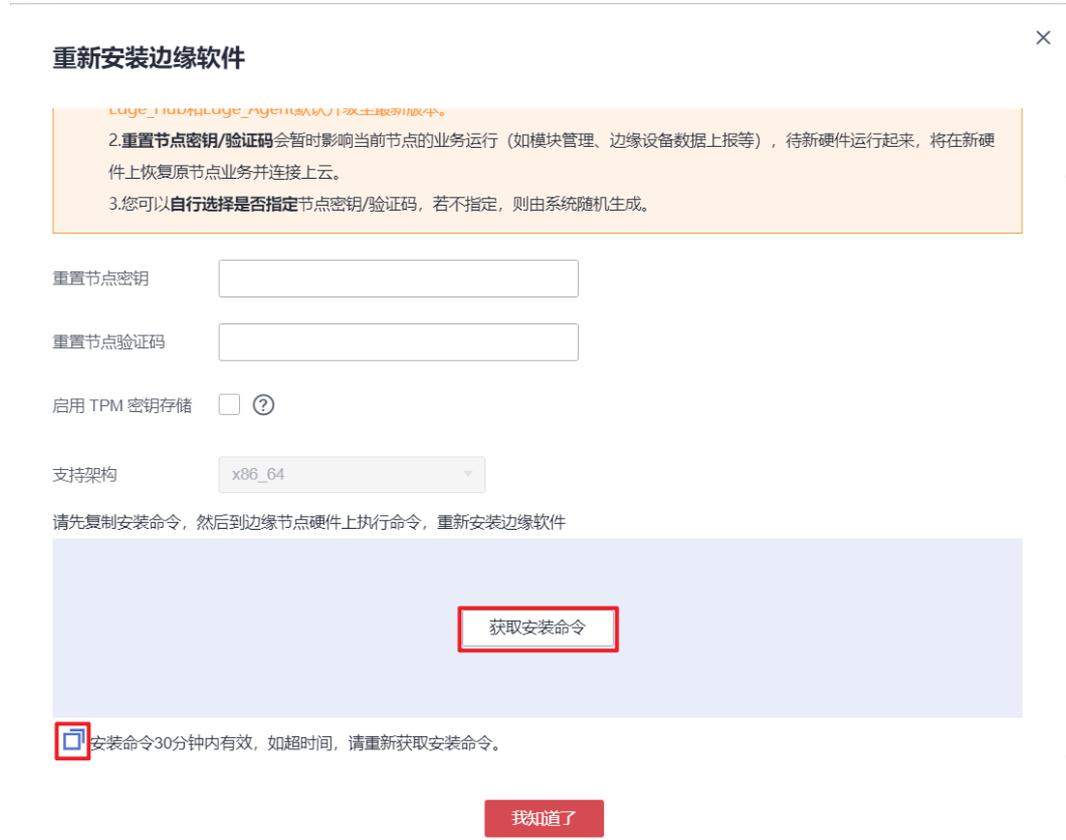


图 2-116 执行重装

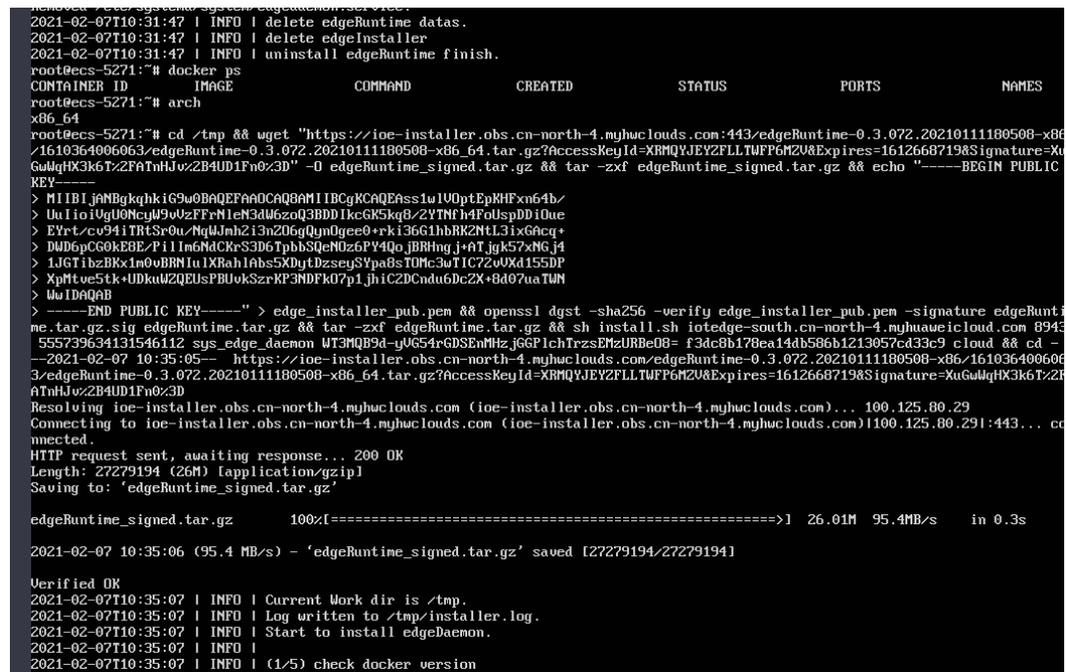


图 2-117 重装命令执行成功

```

2022-04-07T10:24:38.192629 | INFO | reportDeploymentState | 65 | reportDeploymentState, deploymentState:DeploymentState{ node_id:ss47, deploy_id:1649298329566, deploy_state:PEMGOING, reason:(zero),
2022-04-07T10:24:40.263105 | INFO | saveBasePathInfo | 70 | config base path start.
2022-04-07T10:24:40.263367 | INFO | saveBasePathInfo | 73 | config log path:/var/loTEdge/log/,db_path:/var/loTEdge/db/, config_path:/var/loTEdge/config/
2022-04-07T10:24:40.263505 | INFO | saveBasePathInfo | 87 | config base path success.
2022-04-07T10:24:40.263634 | INFO | addAgentCustomConfig | 207 | addCustomConfig start.
2022-04-07T10:24:40.263882 | INFO | addAgentCustomConfig | 231 | addCustomConfig end.
2022-04-07T10:24:40.263725 | INFO | pullImage | 395 | start pull image, deploy type is cloud
2022-04-07T10:24:40.925910 | INFO | createModule | 75 | create Module, name:sys-edge-agent
2022-04-07T10:24:47.092738 | INFO | createModule | 93 | module created, CreateRespBody: ModuleCreateSp{ moduleId:sys-edge-agent, moduleName:sys-edge-agent, appId:edge-agent, appVersion:1-0-409-active-x
36, containerId:31a59f6a4ec7c66f752171ca3d199992008a8f84922c558642e9a645602, processId:(zero), type:docker,}
2022-04-07T10:24:47.092904 | INFO | startModule | 159 | startModule moduleId:sys-edge-agent
2022-04-07T10:24:53.699519 | INFO | listModule | 245 | listModule
2022-04-07T10:24:53.894377 | INFO | checkModules | 288 | start check modules, time:1
2022-04-07T10:24:53.894578 | INFO | listModule | 245 | listModule
2022-04-07T10:24:53.909072 | INFO | printModulesAndReport | 350 | ***success! failed:0
2022-04-07T10:24:53.909264 | INFO | reportDeploymentState | 65 | reportDeploymentState, deploymentState:DeploymentState{ node_id:ss47, deploy_id:1649298329566, deploy_state:RUNNING, reason:(zero),
2022-04-07T10:24:53.909701 | INFO | reportModuleStatus | 112 | reportModuleStatus, modulesInfo:ModuleStatusInfo{ host_tag:DEFAULT, deploy_id:1649298329566, modules:[ModuleStatusInfo{ module_id:sys-ed
ge-agent, app_id:edge-agent, app_version:1-0-409-active-x06, container_state:RUNNING, liveness_state:(zero), update_time:1649298329564,}]},}
2022-04-07T10:24:54.027991 | INFO | install | 429 | install edge finish.
2022-04-07T10:24:54.028293 | INFO | stop | 480 | edgeInstaller stop.
^C
root@zxw951241-edge-node-test ~#
    
```

**步骤4** 单击“我知道了”，等待边缘节点的状态变为“在线”，表示该节点已重新安装并连接成功。

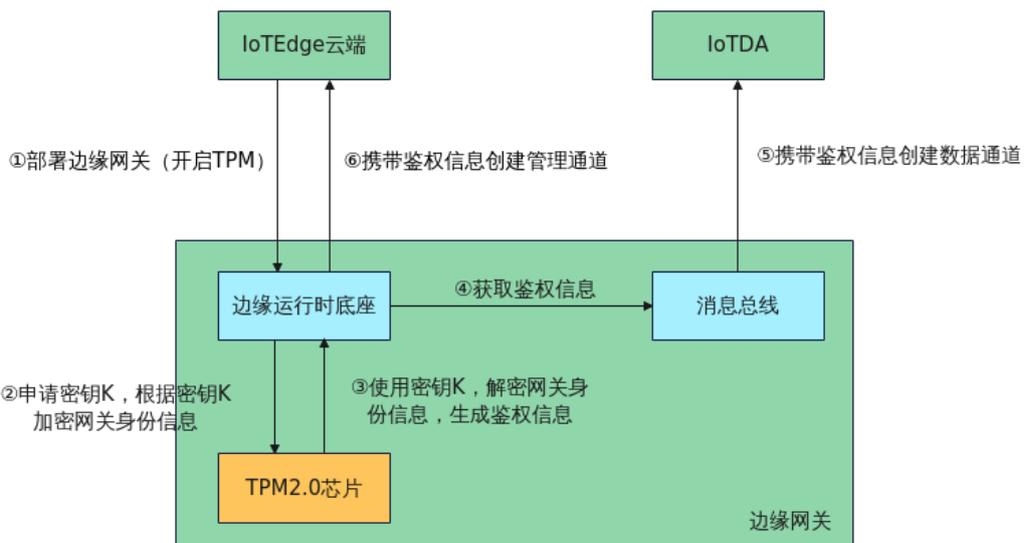
----结束

## 2.13 启用 TPM 硬件加密

TPM(Trusted Platform Module)安全芯片是指符合TPM（可信平台模块）标准的安全芯片，它能有效地保护服务器，防止非法用户访问敏感数据。

通过安装节点时，选定开启TPM加密功能，实现对节点侧密钥、鉴权凭证等敏感数据的硬件加密。

图 2-118 开启 TPM 硬件场景下边缘网关访问云端



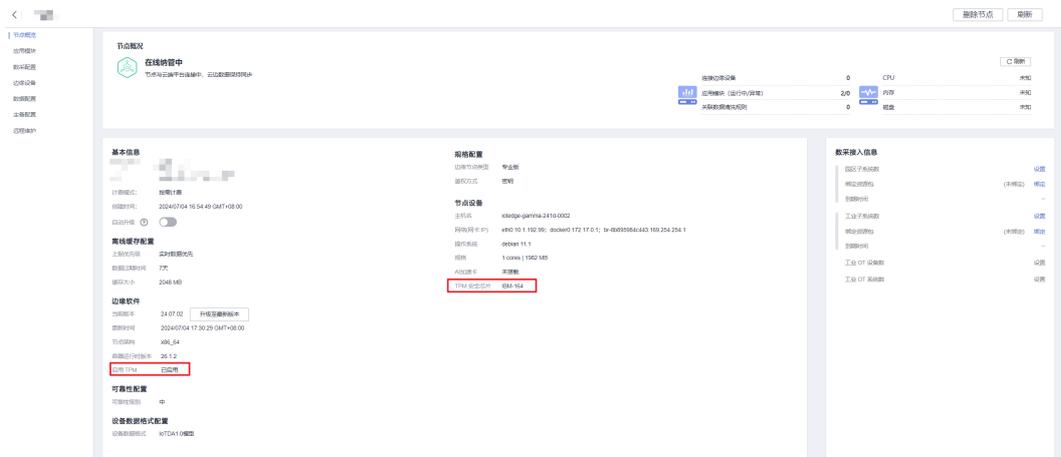
开启TPM硬件加密功能后，网关运行时底座将向TPM芯片申请根密钥，该根密钥用于加密敏感信息（如网关身份密钥）。当网关上的应用需要使用到这些敏感信息时，再经过TPM解密后使用。引入TPM芯片后，提升边缘网关的安全等级，软硬件协同实现边缘本地敏感数据的加密存储，保障用户数据的高安全、高隐私。

### 使用约束

1. 该功能仅适用于装有TPM2.0安全芯片的硬件网关。
2. 仅安装节点时可选择是否开启TPM硬件加密，节点安装完成后无法更改。
3. 部署类型为KubeEdge的专业版边缘节点不支持开启TPM硬件加密。
4. 开启TPM硬件加密的节点后将无法进行主备配置。



图 2-121 节点详细信息

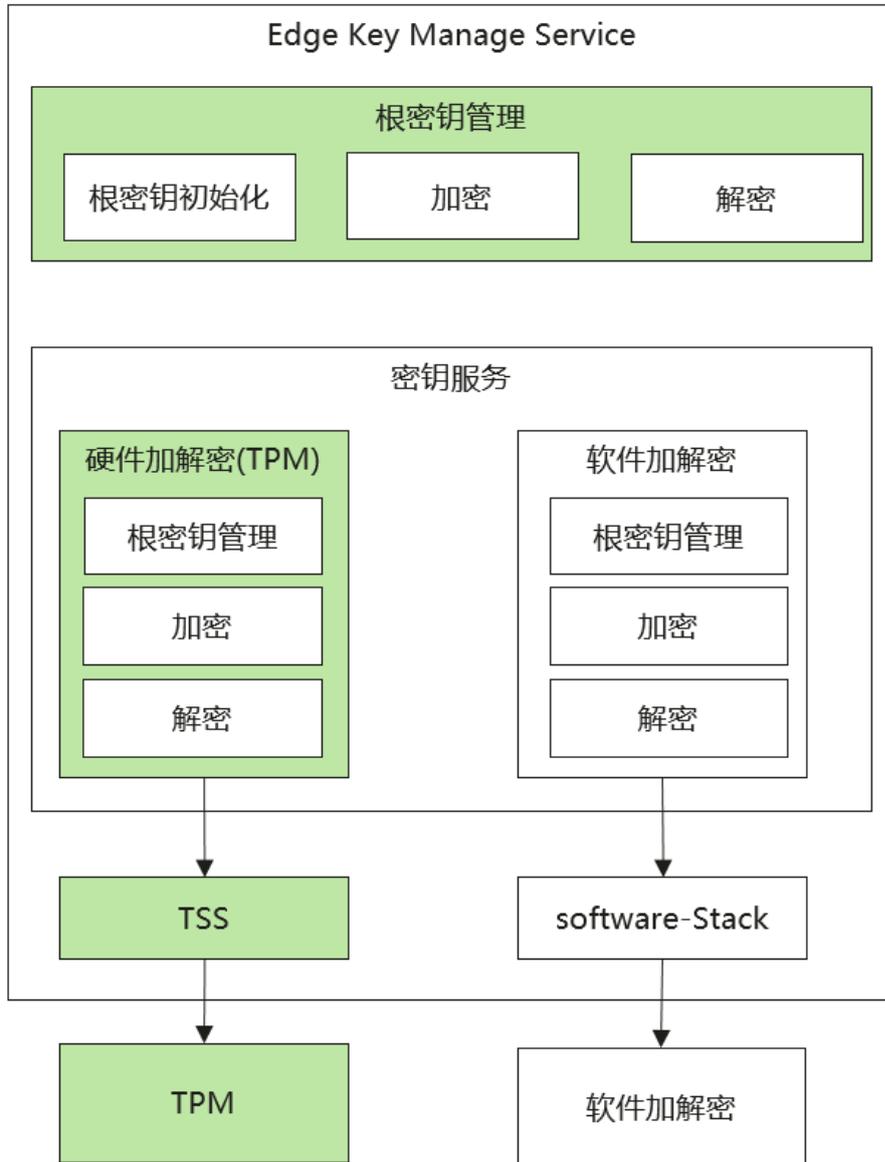


----结束

## TPM 硬件加密与本地软件加解密对比

使用软件加解密，密钥材料放置于磁盘中。而在带TPM安全芯片的系统中，一些关键的密钥是存在TPM芯片内部，这时攻击者只有攻破TPM才有可能攻破系统的防护，攻克一块芯片比一块硬盘成本和难度更高。TPM芯片可以生成、存储和保护密钥。它提供了一个安全的环境来生成密钥对，并将私钥存储在TPM的安全存储器中。私钥无法被直接访问，只能通过TPM提供的安全接口进行使用。经TPM加密的数据只有此TPM芯片才能解密，从而达到数据与TPM芯片绑定来保护数据的目的。简而言之，TPM硬件加解密与本地软件加解密对比，具有更高的安全性、保密性。

图 2-122 边缘密钥服务体系



# 3 应用管理

## 3.1 应用管理

### 3.1.1 应用管理概述

应用管理是IoT边缘提供的边缘应用管理能力，通过将系统预置的应用或用户自定义的应用部署到边缘节点，可在边侧进行协议解析、数据处理、本地子系统集成等能力拓展。当前IoT边缘支持容器化部署和安装包部署的方式，您可以制作镜像上传到[容器镜像服务\(SWR\)](#)，也可以选择上传安装包到[对象存储服务\(OBS\)](#)。系统预置的应用如表3-1所示。

表 3-1 系统预置应用

应用名称	应用类型	说明
\$edge_hub	系统必选	边缘节点上的处理中心，负责设备和通信管理。
\$edge_agent	系统必选	在边缘节点上管理边缘应用，包括部署、升级、运行监控。
\$edge_access	系统可选	扩展边缘节点的协议接入能力，当前支持Modbus, OPC-UA协议接入。不推荐使用，推荐使用OT数采（以 <a href="#">基于IoT边缘实现OT数采（OPCUA协议）</a> 为例，可开发自定义驱动应用）。
\$edge_rule	系统可选	负责边缘设备数据的计算和处理，需与数据清洗配合使用。
\$edge_apigw	系统可选	边缘节点的API网关，负责为边缘应用提供路由转发，正反向代理等能力，需与路由管理配合使用。
\$industry_dc_bsi	系统可选	负责采集MES、ERP等子系统数据，对接到异构系统。系工业子系统数采应用，若想部署该应用，需购买工业子系统数采集成/采集服务。
\$edge_omagent	系统可选	负责边缘节点远程监控和运维，提供日志上报、远程ssh、指标上报、文件上传下载能力。

应用名称	应用类型	说明
\$edge_keeplive	系统可选	边缘节点主备切换模块,保证节点的高可用。
\$ot_dc_opcua	系统可选	边缘扩展协议接入。
\$ot_dc_yk	系统可选	负责边缘OT数采。
\$edge_push	系统可选	提供外部推送能力。
\$edge_schedule	系统可选	边缘定时调度模块。
\$edge_npu_plugin	系统可选	边缘管理NPU设备分配挂载应用。

### 3.1.2 添加业务应用

添加您的自定义边缘应用。

#### 操作步骤

**步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 应用管理”进入页面，选择“业务应用”，单击“添加应用”。

图 3-1 添加业务应用

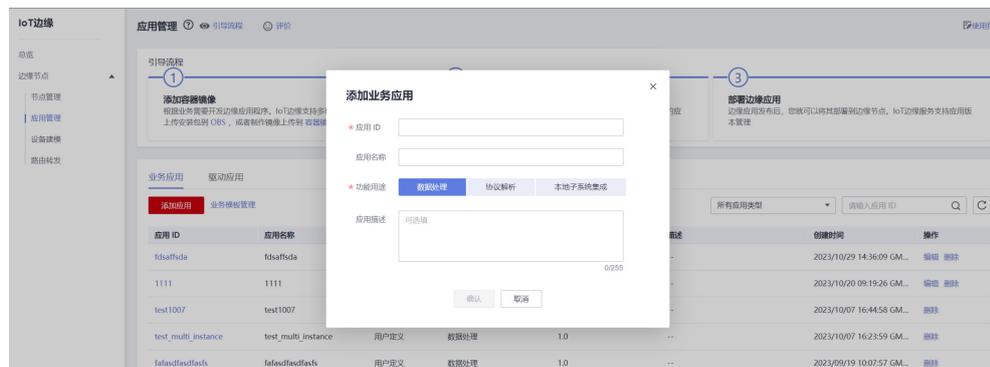


表 3-2 参数说明

参数名称	参数说明
应用ID	自定义应用ID。
应用名称	自定义边缘应用的名称。

参数名称	参数说明
功能用途	数据处理：提供设备数据处理能力； 协议解析：提供设备接入能力； 本地子系统集成：提供IT子系统集成能力。

**步骤3** 添加完成后，单击“前往：应用版本配置”。

**图 3-2** 应用添加成功



**步骤4** 软件部署配置。

1. 选择部署方式，参考表3-3填写基本信息。

**图 3-3** 软件部署配置



表 3-3 部署配置

参数名称	说明
部署方式	<p>容器化部署：在边缘节点上应用通过Docker容器运行。添加容器化部署的应用需先确保容器镜像已上传到<a href="#">容器镜像服务(SWR)</a>，且设置其类型为“公开”，例如<a href="#">图3-4</a>所示。</p> <p>安装包部署：在边缘节点是以进程形式运行。添加安装包部署的应用需先确保安装包已按照打包规范进行压缩打包，并上传到<a href="#">对象存储服务(OBS)</a>。</p>
容器镜像	部署方式为容器化部署时，参考 <a href="#">表3-4</a> 填写。
手动输入	输入镜像地址：统一在 <a href="#">容器镜像服务(SWR)</a> 中管理。
单击选择	<p>选择镜像：选择需要部署的镜像，单击“确认”。</p> <ul style="list-style-type: none"> <li>- 我的镜像：展示了您在容器镜像服务中创建的所有镜像。</li> <li>- 他人共享：展示了其他用户共享的镜像。</li> </ul> <p>版本：选择需要部署的镜像版本。</p>
安装包地址	部署方式为安装包部署时，填写您此前上传到对象存储服务OBS中的安装包地址。

2. 选择容器规格，根据您的需要选择。

图 3-4 容器规格配置

容器规格

CPU配额  申请预留  限制使用

内存配额  申请预留  限制使用

AI加速卡配额

表 3-4 选择镜像输入方式

参数名称	说明
手动输入	输入镜像地址：统一在 <a href="#">容器镜像服务(SWR)</a> 中管理。
单击选择	<p>选择镜像：选择需要部署的镜像，单击“确认”。</p> <ul style="list-style-type: none"> <li>- 我的镜像：展示了您在容器镜像服务中创建的所有镜像。</li> <li>- 他人共享：展示了其他用户共享的镜像。</li> </ul> <p>版本：选择需要部署的镜像版本。</p>

表 3-5 AI 加速卡配额指定参数说明

参数名称	说明
不申请配额	应用无需使用AI加速卡
申请GPU配额	应用需要Nvidia GPU，指定需要使用的GPU显存大小，应用部署时根据该配置分配资源。暂仅专业版节点支持，在专业版基础版节点部署模块时该字段不生效。
申请NPU配额	应用需要使用NPU，指定NPU的型号，及使用数量，应用部署时根据节点资源情况进行分配。

## 3. 高级配置，按照如下步骤进行配置。

图 3-5 高级配置



## a. 运行命令（只有容器化部署的方式可以配置）

图 3-6 运行命令



- 运行命令：输入可执行的命令，例如“/run/start”。
  - 若可执行命令有多个，多个命令之间用换行进行分隔，且末尾不能有空行。
- 命令参数：输入控制容器运行命令参数，例如-port=8080。
  - 若参数有多个，多个参数以换行分隔，且末尾不能有空行。

## 说明

启动命令和参数复杂时，建议编写启动脚本，打包到镜像中，如start.sh，启动命令设置为 ./shart.sh。

若启动命令已经定义在dockerfile中，则该项可以不进行配置。

## b. 选项配置

打开勾选项 ，开启特权模式，容器将具有其宿主机几乎所有能力，包括一些内核特性和所有设备访问。容器在特性模式下运行会存在一些安全风险，默认关闭该选项。

### c. 环境变量

图 3-7 环境变量



容器运行环境中设定的一个变量。可以在部署应用时修改。

单击  添加环境变量，输入变量名称和变量值，添加环境变量。

#### 说明

IoTEdge平台不会对用户输入的环境变量进行加密。

如果用户配置的环境变量涉及敏感信息，用户需要自行加密后再填入，并在应用中自己完成解密过程。

IoTEdge平台服务不提供任何加解密工具，如果您需要设置加密密文，可以使用其他平台的加解密工具。

### d. 数据存储

卷是指容器运行过程中使用的存储卷。

将主机某个目录挂载到容器中。主机目录是一种持久化存储，应用删除后 hostPath 里面的内容依然存在于边缘节点本地硬盘目录中，如果后续重新创建应用，挂载后依然可以读取到之前写入的内容。

图 3-8 数据存储



**说明**

- 请不要挂载在系统目录下，如“/”、“/var/run”等，会导致容器异常。建议挂载在空目录下，若目录不为空，请确保目录下无影响容器启动的文件，否则文件会被替换，导致容器启动异常，应用创建失败。
- 挂载高危目录的情况下，建议使用低权限账号启动，否则可能会造成宿主机高危文件被破坏。
- 本地卷的类型选择LOG/DB/CONFIG时，会给主机目录默认添加前缀：“/var/IoTEdge/{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”。
- 当挂载的目录磁盘满，会导致节点异常，无法使用，请及时清理。

e. **健康检查**

健康检查是指容器运行过程中根据用户需要定时检查容器健康状况或是容器中负载的健康状况。

- 应用存活探针：应用存活探针用于探测容器是否正常工作，不正常则重启实例。当前支持发送HTTP请求和执行命令检查，通过检测容器响应是否正常。
- 应用业务探针：应用业务探针用于探测业务是否就绪，如果业务还未就绪，就不会将流量转发到当前实例。

**图 3-9 健康检查****表 3-6 检查方式**

参数名称	说明
HTTP请求检查	<p>向容器发送HTTP GET请求，如果探针收到2xx或3xx，说明容器是健康的。</p> <p>例如图3-10的配置，会在容器启动10秒（延迟时间）后，发送HTTP GET请求到“http://{实例IP}/health:8080”，如果在2秒（超时时间）内没有响应则视为检查失败；如果请求响应的状态码为2xx或3xx，则说明容器是健康的。</p> <p><b>说明</b> 这里无需填写主机地址，默认直接使用实例的IP（即往容器发送请求），除非您有特殊需求。</p>

参数名称	说明
执行命令检查	探针执行容器中的命令并检查命令退出的状态码，如果状态码为0则说明健康。 例如图3-11的配置，会在容器启动10秒（延迟时间）后，在容器中执行“cat /tmp/healthy”命令，如果在2秒（超时时间）内没有响应，则视为检查失败；如果命令成功执行并返回0，则说明容器是健康的。
TCP检查	例如图3-12的配置，检查TCP端口是否正常可以访问。如果可以，说明容器是正常的。

图 3-10 HTTP 请求检查

\* 检查方式  不配置  HTTP请求检查  执行命令检查  TCP检查

\* 路径/端口

主机地址

HTTP协议  HTTP  HTTPS

\* 延迟时间 ②  秒

\* 探测超时时间  秒

图 3-11 执行命令检查

应用存活探针 检查容器是否正常，不正常则返回实例

\* 检查方式  不配置  HTTP请求检查  执行命令检查  TCP检查

\* 执行命令

\* 延迟时间 ②  秒

\* 探测超时时间  秒

图 3-12 TCP 检查

\* 检查方式  不配置  HTTP请求检查  执行命令检查  TCP检查

\* 端口

\* 延迟时间 ②  秒

\* 探测超时时间  秒

步骤5 第二步“运行配置”。

图 3-13 端点和部署配置

**添加边缘应用**

① 软件部署配置 ———— ② 运行配置 ———— ③ 配置确认

**端点配置**  
例如，EdgeHub使用MQTT Broker做消息总线，输入端点代表模块接收消息的Topic，输出端点代表发送消息的Topic（端点仅代表而非真正的MQTT Topic）。

输入端点

输出端点

**部署配置**

重启策略:  总是重启  失败时重启  不重启

当应用实例退出时，无论是正常退出还是异常退出，系统都会重新拉起应用实例。

网络类型:  主机网络  端口映射

使用宿主机（边缘节点）的网络，即容器与主机间不做网络隔离，使用同一个IP。

## 1. 端点配置。

边缘节点上各模块之间的通讯采用MQTT作为消息总线，其中系统模块EdgeHub作为MQTT Broker提供模块间的数据流转功能。应用版本中的输入/输出端点，即为模块对接消息总线的设备数据入口和出口点，用于配置数据流转规则。

- 输入端点：模块从消息总线中接收设备数据的入口点。应用程序集成边缘SDK时需设置输入端点的监听/回调，以接收设备数据并进行处理。
- 输出端点：模块往消息总线中发送设备数据的出口点。应用程序集成边缘SDK时，从输出端点往消息总线发送设备数据。

**说明**

只有配置了消息输入输出端点的应用实例化的模块才能够使用设备数据流转功能。

建议输入/输出端点配置成设备数据格式，即iotda\_1.0或flat\_json，用以关联接收到/发送的设备数据格式。

## 2. 部署配置。

表 3-7 部署配置

参数名称	说明
重启策略	<p>根据您的需要选择。</p> <ul style="list-style-type: none"> <li>- 总是重启：当应用实例退出时，无论是正常退出还是异常退出，系统都会重新拉起应用实例。</li> <li>- 失败时重启：当应用实例异常退出时，系统会重新拉起应用实例，正常退出时，则不再拉起应用实例。</li> <li>- 不重启：当应用实例退出时，无论是正常退出还是异常退出，系统都不再重新拉起应用实例。</li> </ul>

参数名称	说明
网络类型	<p>容器访问支持主机网络和端口映射两种方式。</p> <ul style="list-style-type: none"> <li>- <b>主机网络</b> 使用宿主机（边缘节点）的网络，即容器与主机间不做网络隔离，使用同一个IP。</li> <li>- <b>端口映射</b> 容器网络虚拟化隔离，容器拥有单独的虚拟网络，容器与外部通信需要与主机做端口映射。配置端口映射后，流向主机端口的流量会映射到对应的容器端口。例如容器端口80与主机端口8080映射，那主机8080端口的流量会流向容器的80端口。</li> </ul>

**步骤6** 第三步配置确认，参考表3-8 填写基本信息

图 3-14 应用版本基本配置

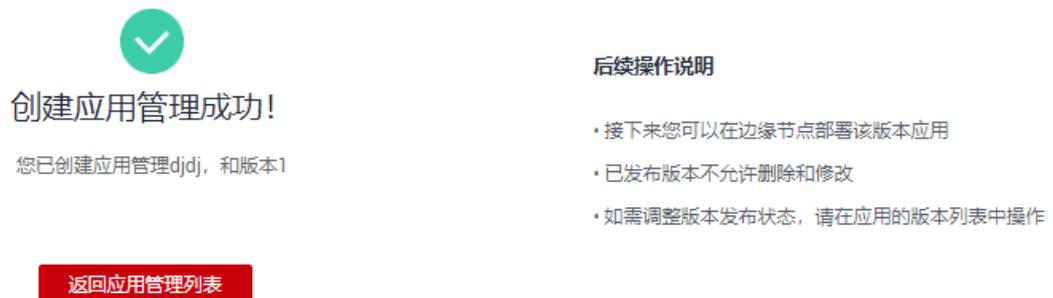
表 3-8 应用版本基本配置

参数名称	说明
应用ID	自定义边缘应用ID。
应用名称	自定义边缘应用的名称。

参数名称	说明
功能用途	您可以根据实际情况选择： 数据处理：提供设备数据处理能力。 协议解析：提供设备接入能力。 本地子系统集成：提供IT子系统集成能力。 网关管理：扩展边缘节点的协议接入能力，当前支持Modbus，OPC-UA协议接入。 混合应用：包含数据处理、协议转换、子系统接入、网关管理、数据采集其中两个及以上功能的应用。 数据采集：提供OT数采能力。
应用描述	输入对应用的描述信息，方便与其他应用进行区分。
SDK版本	集成边缘SDK的版本号。
版本	自定义您的应用版本号。可选择是否“立即发布”。
支持架构	选择应用可支持的架构。有x86_64、arm32、arm64的架构，支持多选。
支持多部署	应用版本支持在一个边缘节点上部署多个实例。
业务模板	业务应用使用的配置模板也可以不关联模板。

**步骤7** 单击“确认添加”，完成应用的创建；单击“立即发布”，完成应用创建并发布新版本。

图 3-15 创建成功



**步骤8** 单击“返回应用管理列表”。

您可以看到应用类型为“用户定义”，区别于系统预置的应用。

图 3-16 应用管理列表



----结束

### 3.1.3 添加驱动应用

添加您的自定义边缘应用。

#### 操作步骤

- 步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。
- 步骤2** 选择左侧导航栏“边缘节点 > 应用管理”进入页面，选择驱动应用，单击“添加应用”。

图 3-17 添加驱动应用

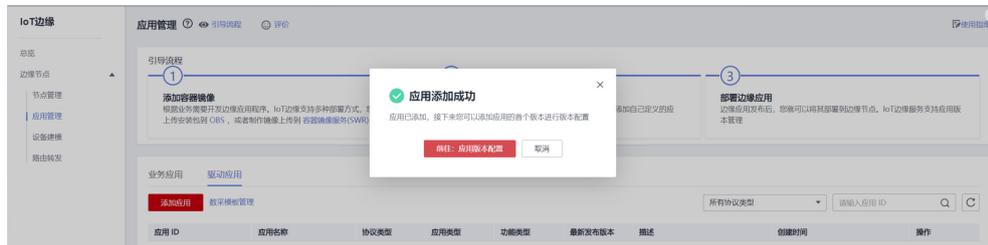


表 3-9 添加驱动应用

参数名称	说明
驱动ID	自定义，不支持中文，必填，长度不能小于4个字符。
驱动名称	自定义，必填，长度不能小于4个字符。
协议类型	应用集成的协议类型，如无，选择Other。
驱动用途	您可以根据实际情况选择 数据采集：提供OT数采能力。 混合应用：包含数据处理、协议转换、子系统接入、网关管理、数据采集其中两个及以上功能的应用。
应用描述	(可选填) 描述应用的功能，以及用途。

**步骤3** 添加完成后，单击“前往：应用版本配置”。

**图 3-18** 添加成功



**步骤4** 第一步“软件部署配置”。

1. 选择部署方式，参考表3-10填写基本信息。

**图 3-19** 软件部署配置



**表 3-10** 部署配置

参数名称	说明
部署方式	<p>容器化部署：在边缘节点上应用通过Docker容器运行；添加容器化部署的应用需先确保容器镜像已上传到<b>容器镜像服务(SWR)</b>，且设置其类型为“公开”，例如<b>图3-20</b>所示。</p> <p>安装包部署：在边缘节点是以进程形式运行；添加安装包部署的应用需先确保安装包已按照打包规范进行压缩打包，并上传到<b>对象存储服务(OBS)</b>。</p>
安装包部署	部署方式为安装包部署时，填写您此前上传到对象存储服务OBS中的安装包地址。

参数名称	说明
容器镜像	部署方式为容器化部署时，参考表3-11填写。

表 3-11 选择镜像输入方式

参数名称	说明
手动输入	输入镜像地址：统一在容器镜像服务(SWR)中管理。
单击选择	<p>选择镜像：选择需要部署的镜像，单击“确认”。</p> <ul style="list-style-type: none"> <li>我的镜像：展示了您在容器镜像服务中创建的所有镜像。</li> <li>他人共享：展示了其他用户共享的镜像。</li> </ul> <p>版本：选择需要部署的镜像版本。</p>

2. 选择容器规格，根据您的需要选择。

图 3-20 容器规格配置



3. 高级配置，按照如下步骤进行配置。

图 3-21 高级配置



- a. 运行命令（只有容器化部署的方式可以配置）

图 3-22 运行命令



- 运行命令：输入可执行的命令，例如“/run/start”。
  - 若可执行命令有多个，多个命令之间用换行进行分隔，且末尾不能有空行。
- 命令参数：输入控制容器运行命令参数，例如-port=8080。
  - 若参数有多个，多个参数以换行分隔，且末尾不能有空行。

### 说明

启动命令和参数复杂时，建议编写启动脚本，打包到镜像中，如start.sh，启动命令设置为 ./shart.sh。

若启动命令已经定义在dockerfile中，则该项可以不进行配置。

若应用部署实例异常，可参考配置指导：[https://support.huaweicloud.com/iotedge\\_faq/iotedge\\_04\\_0029.html](https://support.huaweicloud.com/iotedge_faq/iotedge_04_0029.html)

### b. 选项配置

打开勾选项 ，容器开启特权模块，将拥有访问主机设备的权限，可以访问主机上的设备（如GPU、FPGA），容器在特性模式下运行会存在一些安全风险，默认关闭该选项。

### c. 环境变量

图 3-23 环境变量



容器运行环境中设定的一个变量。可以在部署应用时修改。

单击 添加环境变量，输入变量名称和变量值，添加环境变量。

### 说明

IoTEdge平台不会对用户输入的环境变量进行加密。

如果用户配置的环境变量涉及敏感信息，用户需要自行加密后再填入，并在应用中自己完成解密过程。

IoTEdge平台服务不提供任何加解密工具，如果您需要设置加密密文，可以使用其他平台的加解密工具。

#### d. 数据存储

卷是指容器运行过程中使用的存储卷。

将主机某个目录挂载到容器中。主机目录是一种持久化存储，应用删除后 hostPath 里面的内容依然存在于边缘节点本地硬盘目录中，如果后续重新创建应用，挂载后依然可以读取到之前写入的内容。

图 3-24 数据存储



#### 说明

- 请不要挂载在系统目录下，如“/”、“/var/run”等，会导致容器异常。建议挂载在空目录下，若目录不为空，请确保目录下无影响容器启动的文件，否则文件会被替换，导致容器启动异常，应用创建失败。
- 挂载高危目录的情况下，建议使用低权限账号启动，否则可能会造成宿主机高危文件被破坏。
- 本地卷的类型选择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”。
- 当挂载的目录磁盘满，会导致节点异常，无法使用，请及时清理。

#### e. 健康检查

健康检查是指容器运行过程中根据用户需要定时检查容器健康状况或是容器中负载的健康状况。

- 应用存活探针：应用存活探针用于探测容器是否正常工作，不正常则重启实例。当前支持发送HTTP请求和执行命令检查，通过检测容器响应是否正常。
- 应用业务探针：应用业务探针用于探测业务是否就绪，如果业务还未就绪，就不会将流量转发到当前实例。

图 3-25 健康检查



表 3-12 检查方式

参数名称	说明
HTTP请求检查	<p>向容器发送HTTP GET请求，如果探针收到2xx或3xx，说明容器是健康的。</p> <p>例如图3-26的配置，会在容器启动10秒（延迟时间）后，发送HTTP GET请求到“http://{实例IP}/health:8080”，如果在2秒（超时时间）内没有响应则视为检查失败；如果请求响应的状态码为2xx或3xx，则说明容器是健康的。</p> <p><b>说明</b> 这里无需填写主机地址，默认直接使用实例的IP（即往容器发送请求），除非您有特殊需求。</p>
执行命令检查	<p>探针执行容器中的命令并检查命令退出的状态码，如果状态码为0则说明健康。</p> <p>例如图3-27的配置，会在容器启动10秒（延迟时间）后，在容器中执行“cat /tmp/healthy”命令，如果在2秒（超时时间）内没有响应，则视为检查失败；如果命令成功执行并返回0，则说明容器是健康的。</p>
TCP检查	<p>例如图3-28的配置，检查TCP端口是否正常可以访问。如果可以，说明容器是正常的。</p>

图 3-26 HTTP 请求检查

\* 检查方式: 不配置 | **HTTP请求检查** | 执行命令检查 | TCP检查

\* 路径/端口: /health | 8080

主机地址: 请输入主机地址

HTTP协议: **HTTP** | HTTPS

\* 延迟时间 ①: - 10.5 + 秒

\* 探测超时时间: - 2 + 秒

图 3-27 执行命令检查

应用存活探针 检查容器是否正常，不正常则重启实例

\* 检查方式: 不配置 | HTTP请求检查 | **执行命令检查** | TCP检查

\* 执行命令: cat /tmp/healthy

\* 延迟时间 ①: - 0 + 秒

\* 探测超时时间: - 1 + 秒

图 3-28 TCP 检查

\* 检查方式: 不配置 | HTTP请求检查 | 执行命令检查 | **TCP检查**

\* 端口: 8080

\* 延迟时间 ①: - 0 + 秒

\* 探测超时时间: - 1 + 秒

## 步骤5 第二步“运行配置”。

图 3-29 端点和部署配置

The screenshot shows a configuration interface for adding an edge application. It has a progress bar at the top with three steps: 1. Software Deployment Configuration, 2. Run Configuration (current), and 3. Configuration Confirmation. The 'Endpoint Configuration' section includes a note about MQTT Broker usage and two input fields for 'Input Endpoint' and 'Output Endpoint', each with a 'Add Endpoint' button. The 'Deployment Configuration' section has a 'Restart Strategy' section with three options: 'Always Restart' (selected), 'Restart on Failure', and 'Do Not Restart'. Below this is a 'Network Type' section with two options: 'Host Network' (selected) and 'Port Mapping'. A note explains that 'Host Network' means the container and host share the same IP.

## 1. 端点配置。

EdgeHub与其他模块之间的通讯采用MQTT作为消息总线，其中MQTT作为Broker可与各模块进行数据流转。

- 输入端点：模块订阅的指定正则MQTT Topic的后缀，EdgeHub向该模块发送数据时则根据该模块配置的输入端点构造MQTT Topic并发送数据。
- 输出端点：从模块端向EdgeHub发送数据时使用的MQTT Topic后缀；即EdgeHub接收该模块配置的输出端点对应的MQTT Topic的数据。

## 📖 说明

只有配置了消息输入输出端点的应用实例化的模块才能够使用数据流转功能。

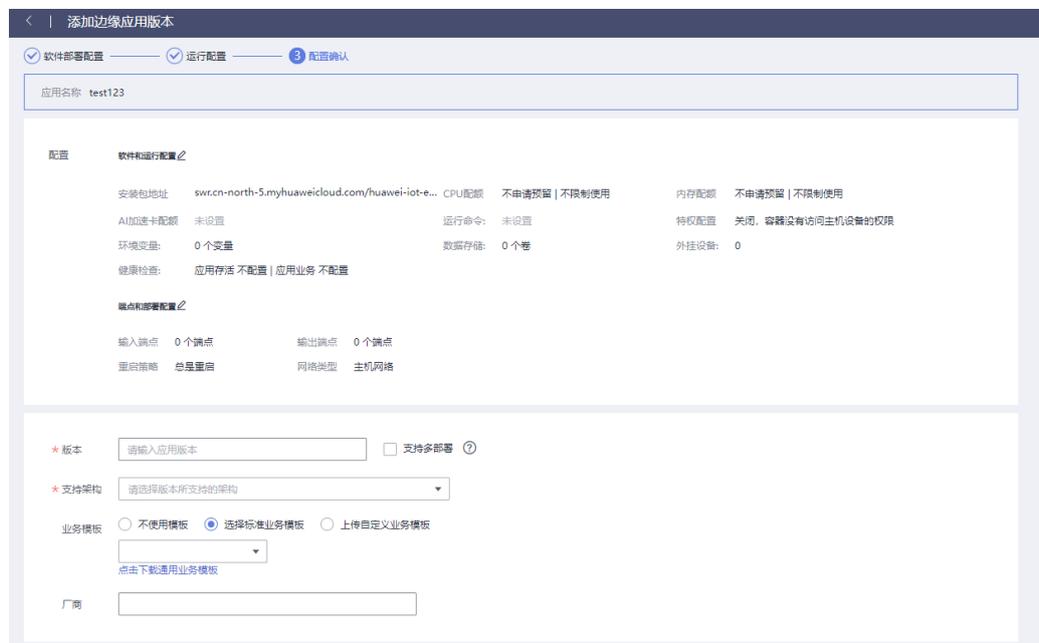
## 2. 部署配置。

表 3-13 部署配置

参数名称	说明
重启策略	<p>根据您的需要选择。</p> <ul style="list-style-type: none"> <li>- 总是重启：当应用实例退出时，无论是正常退出还是异常退出，系统都会重新拉起应用实例。</li> <li>- 失败时重启：当应用实例异常退出时，系统会重新拉起应用实例，正常退出时，则不再拉起应用实例。</li> <li>- 不重启：当应用实例退出时，无论是正常退出还是异常退出，系统都不再重新拉起应用实例。</li> </ul>

参数名称	说明
网络类型	<p>容器访问支持主机网络和端口映射两种方式。</p> <ul style="list-style-type: none"> <li>- <b>主机网络</b> 使用宿主机（边缘节点）的网络，即容器与主机间不做网络隔离，使用同一个IP。</li> <li>- <b>端口映射</b> 容器网络虚拟化隔离，容器拥有单独的虚拟网络，容器与外部通信需要与主机做端口映射。配置端口映射后，流向主机端口的流量会映射到对应的容器端口。例如容器端口80与主机端口8080映射，那主机8080端口的流量会流向容器的80端口。</li> </ul>

**步骤6** 第三步配置确认，参考表3-14 填写基本信息



**表 3-14** 应用版本基本配置

参数名称	说明
应用名称	自定义边缘应用的名称。
版本	自定义您的应用版本号。可选择是否“立即发布”。
支持架构	选择应用可支持的架构。有x86_64、arm32、arm64的架构，支持多选。
支持多部署	应用版本是否支持在一个边缘节点上部署多个实例。
数采模板	驱动应用OT数采使用的模板，数采模板配置可参考 <a href="#">系统数据源模板导入</a> 。

**步骤7** 单击“确认添加”，完成应用的创建；单击“立即发布”，完成应用创建并发布新版本。

图 3-30 创建成功



**步骤8** 单击“返回应用管理列表”。

您可以看到应用类型为“用户定义”，区别于系统预置的应用。

图 3-31 应用管理列表

应用 ID	应用名称	协议类型	应用类型	功能类型	最新发布版本	描述	创建时间	操作
			用户定义	数据采集	--	--	2023/05/29 17:36:47 GMT+0...	编辑 删除
			用户定义	数据采集	--	--	2023/05/15 16:25:19 GMT+0...	编辑 删除
			系统可选	数据采集	1-0-1-x86-s...	Modbus-TCP	2023/02/09 15:18:17 GMT+0...	
			系统可选	数据采集	1-0-0-stand...	iec104数采驱动	2022/09/17 15:23:54 GMT+0...	
			系统可选	数据采集	1-1-2-stand...	负责接入监控视频设备	2022/09/06 19:45:02 GMT+0...	
			系统可选	数据采集	1-0-10-stan...	边缘扩展协议接入...	2022/02/18 14:08:47 GMT+0...	
			系统可选	数据采集	1-0-5-stand...	负责边缘OT数采	2021/12/13 20:54:43 GMT+0...	
			系统可选	数据采集	1-0-01-x86	负责边缘OT数采	2021/12/13 20:36:33 GMT+0...	

----结束

### 3.1.4 添加版本

IoT边缘服务支持创建多个应用版本，方便您管理边缘应用。

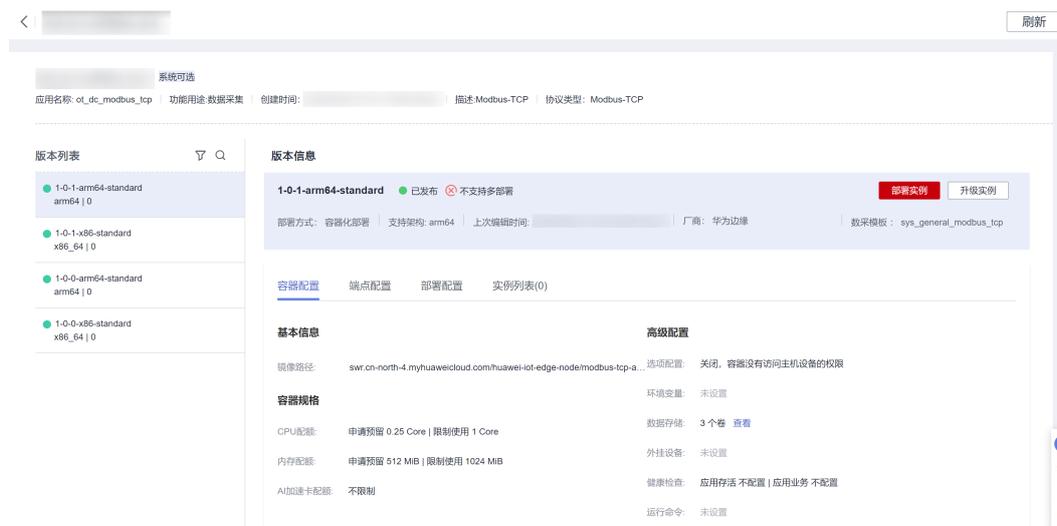
#### 操作步骤

**步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 应用管理”进入页面。

**步骤3** 选择需要添加版本的“应用名称”，进入应用详情页。

图 3-32 应用详情



**步骤4** 单击左侧版本列表的  添加版本按钮，添加边缘应用版本。  
可参考**操作步骤**填写基本信息，完成新应用版本的创建。

图 3-33 添加版本



----结束

## 3.1.5 部署应用

在边缘节点安装成功后可部署边缘应用。

### 操作步骤

- 步骤1** 访问**IoT边缘**，单击“管理控制台”进入IoT边缘控制台。
- 步骤2** 选择左侧导航栏“边缘节点 > 节点管理”进入页面。
- 步骤3** 选择您的边缘节点，单击“节点名称”进入节点详情页。

**步骤4** 在左侧导航栏选择“应用模块 > 模块管理”页签，单击“部署应用”。

图 3-34 部署边缘应用



**步骤5** 根据页面提示填写参数信息后，单击“确认”。

图 3-35 部署应用

×

**提示**

功能用途 数据处理 协议解析 本地子系统集成  
网关管理 混合应用 数据采集

\* 边缘应用  ?

\* 选择版本  ?

模块名称

表 3-15 部署应用

参数名称	说明
功能用途	您可以根据实际情况选择： 数据处理：提供设备数据处理能力。 协议解析：提供设备接入能力。 本地子系统集成：提供IT子系统集成能力。 网关管理：扩展边缘节点的协议接入能力，当前支持Modbus，OPC-UA协议接入。 混合应用：包含数据处理、协议转换、子系统接入、网关管理、数据采集其中两个及以上功能的应用。 数据采集：提供OT数采能力。
边缘应用	在应用管理中创建的应用名称。
选择版本	在应用管理中创建并发布的应用版本。
模块名称	自定义。

**步骤6** 弹出“操作成功”对话框，再单击“确认”返回部署边缘应用列表。

图 3-36 确认



**步骤7** 单击“刷新”，当应用的实例状态由“部署中”转为“运行中”表示部署成功。

模块ID	模块名称	所属应用	版本	应用类型	实例状态	操作
sys_edge_agent	sys_edge_agent	Sedge_agent	1-0-0-standard-v85	系统必备	运行中	升级
sys_edge_hub	sys_edge_hub	Sedge_hub	1-0-0-standard-v85	系统必备	运行中	升级
user_test_app_multiployment	user_test_app_multiployment	Test_app_multiployment	1.0.0	用户定义	运行中	删除 升级

#### ----结束

只有应用版本是多部署的时候，且运行配置的网络类型是端口映射后，可在部署应用时，或应用后添加端口映射。

运行配置，请参考[端点和部署配置](#)。

#### ⚠ 注意

- 专业版（Docker）默认部署sys\_edge\_hub和sys\_edge\_agent，专业版（KubeEdge）默认部署sys\_edge\_hub。
- 专业版和基础版默认部署\$edge\_omagent，在注册节点过程中可选择是否自动部署。
- 只有已发布的应用版本允许被部署。
- 如果应用添加时配置了支持多模块部署 同一个节点下是允许部署多次的。
- 应用支持的架构和边缘节点架构相同才能够部署成功。
- 如果应用需要AI加速卡，边缘节点没有AI加速卡将部署失败。
- 部署应用模块支持升级操作，可选择高版本也可选择低版本，目前只有Agent应用升级失败会回退到原版本。

## 3.1.6 管理应用

您可以管理IoT边缘应用，在添加应用完成后可以对应用版本进行编辑、发布、拷贝、删除、部署实例、升级实例等操作。

### 查看应用详情

**应用管理列表：**所有的边缘应用会显示在边缘应用列表中。

您可以查看应用的应用类型、部署方式、功能类型、最新发布版本、描述、创建时间、操作等。

### 须知

若应用包含版本，则不允许删除此应用。若需删除应用，请先删除该应用的版本。

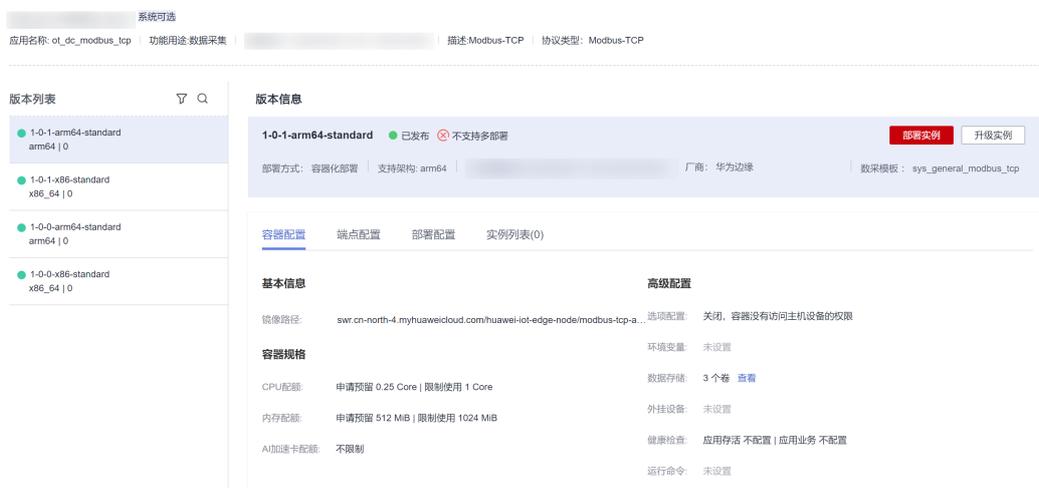
图 3-37 应用管理列表



**应用详情：**在应用列表中，单击“应用名称”进入应用详情，可以查看应用最新版本的

详情。

图 3-38 版本详情



## 查看版本详情

在应用详情中，单击左侧“应用名称”，则可查看版本详情，如图2所示。

 说明

**已发布：**只有已发布的版本才可以部署实例，已发布的版本不允许编辑和删除。

**未发布：**未发布的版本可以进行编辑、发布等操作。

**已下线：**已下线的版本只能被拷贝和删除。

**部署实例：**可对在线节点进行应用批量部署。

**升级实例：**可对已部署在在线节点的应用进行批量升级。

表 3-16 参数说明

参数名称	说明
发布版本	单击“发布版本”后，版本状态由“未发布”变为“已发布”后，可进行部署实例，且该版本所属的应用均不能被删除。
拷贝版本	复制现有版本配置信息，方便快速创建一个新的版本。
下线版本	可将已发布的版本下线处理。
删除版本	对于不需要使用的版本，可以删除。 <b>说明</b> 未发布的版本以及下线状态（且没有部署实例的）版本允许删除。 若下线的版本已经部署到某个节点，则无法删除。
部署实例	选择边缘节点部署实例，部署完成后，您可以在版本详情下的“实例列表”中查看实例状态。

## 批量部署/升级实例

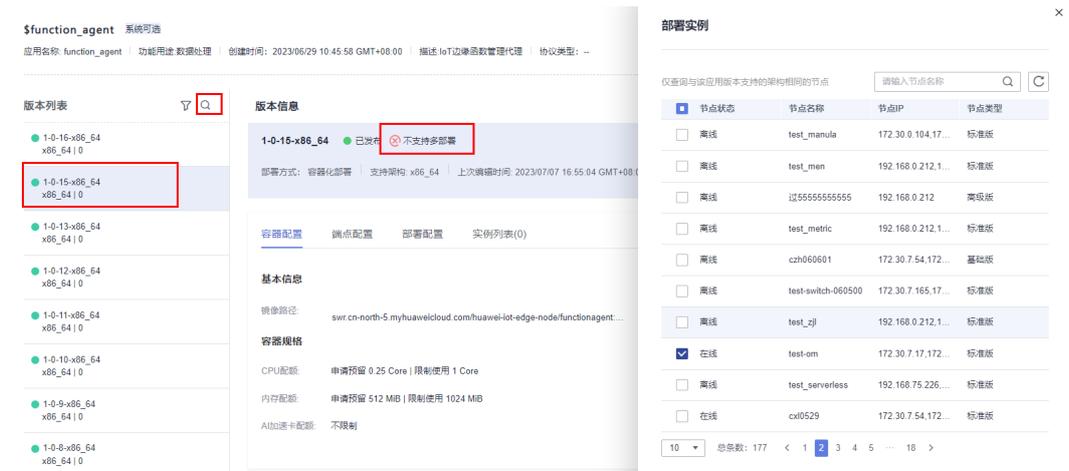
若需要在多个边缘节点上部署同一版本的应用实例，可以进行批量部署或升级应用实例。

 注意

支持多部署的应用版本，不支持批量部署或批量升级实例操作。

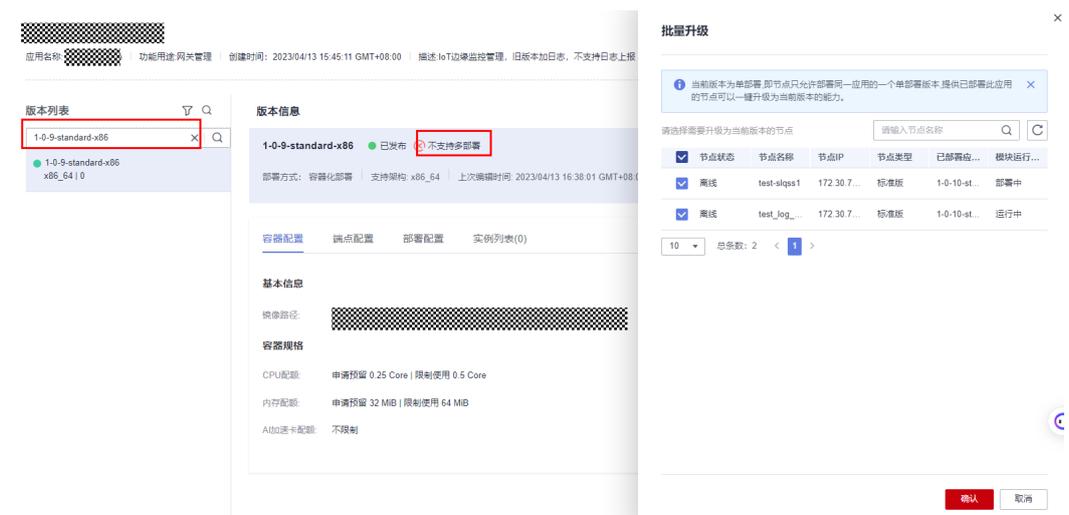
**批量部署：**在应用列表中，单击“应用名称”，进入应用详情，筛选需要部署的应用版本，单击部署实例，展示部署的节点列表，若应用版本不支持多实例部署，则可多选想要部署的节点进行批量部署。

图 3-39 批量部署应用



**批量升级：**在应用列表中，单击“应用名称”，进入应用详情，筛选想要升级的目标应用版本，单击升级实例，可以查询到可以升级到该应用版本的节点列表，若应用版本不支持多实例部署，可以多选想要升级的节点列表，确认进行升级操作。

图 3-40 批量升级应用



## 应用安全启动

通过IoTEdge管理控制台添加的应用，部署到边缘节点时，在边缘节点上会自动校验应用包签名，进行完整性校验防应用篡改，同时，校验下发部署方式是否云边协同通道，当完整性校验通过且下发通道合法时，才会正常启动应用，否则禁止启动。

### 3.1.7 删除应用

您可以管理IoT边缘应用，在添加应用完成后可以对应用版本进行删除操作。

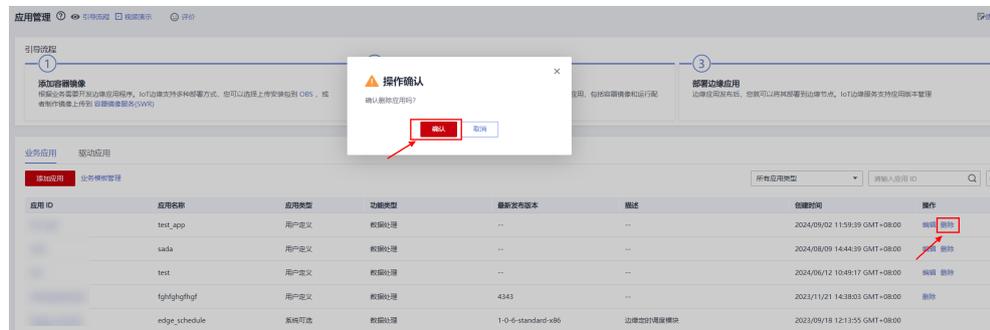
#### 操作步骤

**步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。

**步骤2** 选择左侧导航栏“边缘节点 > 应用管理”进入页面。

**步骤3** 选择需要删除应用后面操作的“删除”按钮。

图 3-41 删除应用



----结束

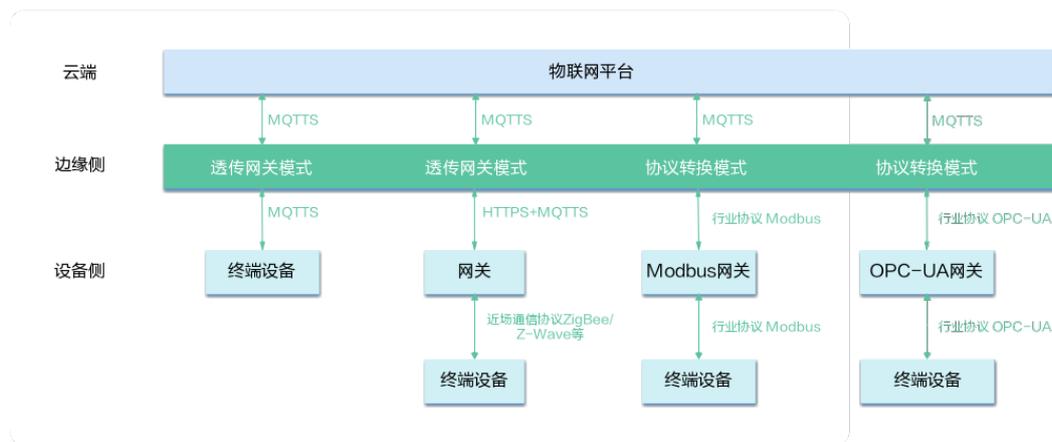
# 4 设备接入边缘节点

## 4.1 接入模式介绍

IoT边缘应用部署完成后，边缘节点即作为云端物联网平台在设备侧的延伸。边缘节点可以为设备就近提供计算和管理服务，其优势有低时延业务本地管理、与云端断链时可在本地控制和执行规则等。设备接入边缘节点，通过边缘节点将数据上传到云端物联网平台。边缘节点提供两种模式接入设备：

- **透传网关模式：**边缘节点支持通用MQTT协议接入，MQTT协议的设备或网关可以直连云端物联网平台，也可以通过边缘节点连接云端物联网平台。这种模式下，边缘节点作为透传网关，将设备或网关上报的数据透传至云端物联网平台。

图 4-1 模式介绍



边缘节点当前支持两种模式接入设备：

接入模式	适用设备类型	场景描述
透传网关模式	针对采用MQTTS协议直接接入边缘节点的设备。	设备通过集成IoT Device SDK、IoT Device SDK Tiny或者实现原生MQTTS协议，将数据发送给边缘节点，边缘节点直接透传给云端物联网平台。

接入模式	适用设备类型	场景描述
	针对未实现TCP/IP协议栈，无法直接同物联网平台通信，希望利用网关连接边缘节点的设备。	设备作为网关子设备，网关直连边缘节点，边缘节点透传网关数据至云端物联网平台 <ul style="list-style-type: none"> <li>• 网关通过集成IoT Device SDK、IoT Device SDK Tiny接入边缘节点。</li> <li>• 设备通过近场通信协议接入网关，确保设备和网关支持协议相同。</li> </ul>

### 须知

IoTEdge透传IoTDA报文，支持能力如下“表1 IoTEdge支持透传IoTDA报文能力详情”所示。其中，对于表中“4.6 网关新增子设备请求”添加的设备，只能在IoTDA控制台查看设备信息，无法在IoTEdge控制台管理设备信息。

表 4-1 IoTEdge 支持透传 IoTDA 报文能力详情

报文类型	是否支持
1 设备命令	支持
1.1 平台命令下发	支持
2 设备消息	支持
2.1 设备消息上报	支持
2.2 平台消息下发	支持
3 设备属性	支持
3.1 设备属性上报	支持
3.2 网关批量设备属性上报	支持
3.3 平台设置设备属性	支持
3.4 平台查询设备属性	支持
3.5 设备侧获取平台的设备影子数据	支持
4 网关与子设备管理	支持
4.1 平台通知网关子设备新增	支持
4.2 平台通知网关子设备删除	支持
4.3 网关同步子设备列表	支持
4.4 网关更新子设备状态	支持
4.5 网关更新子设备状态响应	支持
4.6 网关新增子设备请求	支持

报文类型	是否支持
4.7 网关新增子设备请求响应	支持
4.8 网关删除子设备请求	支持
4.9 网关删除子设备请求响应	支持
5 软固件升级	不支持
5.1 平台下发获取版本信息通知	不支持
5.2 设备上报软固件版本	不支持
5.3 平台下发升级通知	不支持
5.4 设备上报升级状态	不支持
6 文件上传/下载管理	不支持
6.1 设备上报获取文件上传URL请求	不支持
6.2 平台下发文件上传临时URL	不支持
6.3 设备上报文件上传结果	不支持
6.4 设备上报获取文件下载URL请求	不支持
6.5 平台下发文件下载临时URL	不支持
6.6 设备上报文件下载结果	不支持
7 设备时间同步	不支持
7.1 设备时间同步请求	不支持
7.2 设备时间同步响应	不支持
8 设备信息上报	不支持
8.1 设备信息上报	不支持
9 设备日志收集	不支持
9.1 平台下发日志收集通知	不支持
9.2 设备上报日志内容	不支持
10 远程配置	不支持
10.1 平台下发配置通知	不支持
10.2 设备上报配置响应	不支持

## 4.2 透传网关模式接入

本模式适用于两种场景的设备：

- 支持TCP/IP协议栈的设备，设备可以直接与物联网平台进行通信，但希望[直连边缘节点](#)实现近端管理的设备。
- 不支持TCP/IP协议栈的设备，无法直接同物联网平台通信，需要[通过网关连接边缘节点](#)实现近端管理的设备。

### 须知

IoTEdge支持透传IoTDA报文的能力详情，如表[表4-1](#)。

## 设备直连边缘节点

设备通过原生MQTT协议或集成[IoT Device SDK](#)、[IoT Device SDK Tiny](#)直连边缘节点，边缘节点将设备上报的数据透传给物联网平台，并将平台下发的命令透传给设备。

请参考以下流程完成设备接入：

**步骤1** 在[设备接入服务](#)完成设备的产品模型（也称Profile）、编解码插件的开发，并进行自助测试。

对于设备通过边缘节点上报至物联网平台的数据，云端物联网平台要能够解析设备上报的数据。

具体请参考“设备接入服务”的[开发指南](#)中[产品开发的](#)指导。

- **Profile**是用来描述设备能力的文件，通过JSON的格式定义了设备的基本属性、上报数据和下发命令的消息格式。定义Profile，即在物联网平台构建一款设备的抽象模型，在设备上报数据时，平台能够根据定义的Profile理解该款设备支持的属性信息。
- **编解码插件**开发要根据设备上报数据的格式来判断，如果设备上报的“数据格式”为“二进制码流”，则该产品模型下需要进行编解码插件开发。

**步骤2** 进行设备接入物联网平台的设备侧开发。

1. 根据设备自身情况选择合适的接入方式，可以选择原生MQTT协议接入、或者集成[IoT Device SDK](#)、[IoT Device SDK Tiny](#)接入。具体请参考“设备接入服务”的[开发指南](#)中[设备侧开发](#)的指导。
  - a. 采用原生MQTT协议接入，请参考“设备接入服务”的[开发指南](#)中[使用MQTT Demo接入](#)的指导。
  - b. 通过集成SDK接入，请参考“设备接入服务”的[开发指南](#)中[IoT Device SDK使用指南](#)的指导。
2. 将设备接入的IP地址修改为需要连接的边缘节点本地的IP地址。

### 说明

通过MQTT协议的设备接入边缘节点的端口地址是7883，并且需要下载[证书文件](#)进行认证。

**步骤3** 在“IoT边缘控制台”添加设备。

1. 访问[IoT边缘](#)，单击“管理控制台”进入IoT边缘控制台。
2. 选择左侧导航栏的“边缘节点 > 节点管理”，选择您的边缘节点名称，进入节点详情页。

- （可选）选择“应用模块 > 模块管理”页签，部署边缘应用。  
为了保证设备能够正常在线，请您先部署步骤**步骤2**中的应用，直连场景下不需要部署其他应用，具体操作请参考**部署应用**。
- 选择左侧导航栏“边缘设备”页签，单击左上角的“添加边缘设备”按钮，按照**表4-2**填写参数后，单击“确认”。

图 4-2 添加边缘设备

提示

接入设备需统一纳入物联网平台管理，并和当前边缘节点归属同一个设备接入服务实例资源空间。  
归属服务实例 IoTDA默认基础版  
归属资源空间

\* 所属产品  MQTT C  
没有可选产品？请前往设备接入服务增加自定义产品，并定义设备功能前往添加产品

\* 设备标识码

\* 设备名称  ?

模块 ID  ?

password

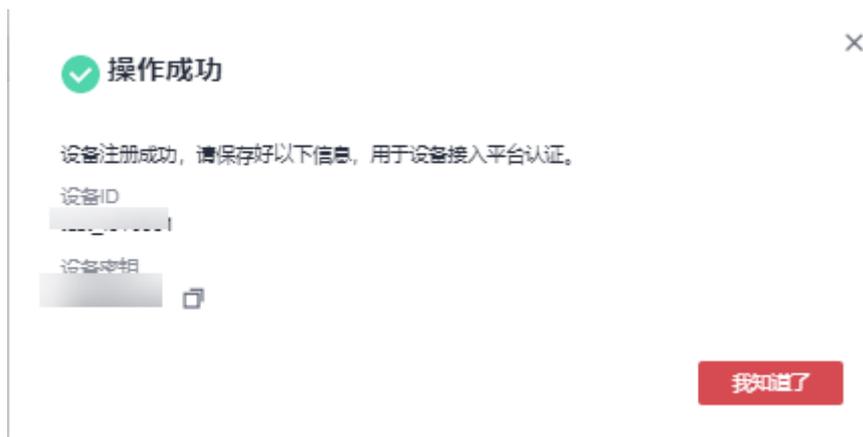
确认 取消

表 4-2 添加边缘设备

参数名称	说明
所属产品	选择要添加的产品名称。 只有在设备接入服务里创建了产品，此处才可以选择具体的产品。如没有，请先 <a href="#">创建产品</a> 。
设备标识码	设备唯一物理标识，如IMEI、MAC地址等，用于设备在接入物联网平台时携带该标识信息完成接入鉴权。 MQTT设备：通过注册成功后生成的“设备ID”（与设备标识码一一对应）和“设备密钥”接入平台。
设备名称	自定义。
模块ID	不填。 <a href="#">添加边缘设备的什么时候填写模块ID?</a>
password	自定义设备密钥。

- 添加设备成功，请保存好设备ID和密钥，用于设备接入平台认证。

图 4-3 设备 ID 和密钥



### 说明

参考[使用MQTT.fx调测](#)文档，使用MQTT.fx工具接入IoT边缘平台。

- 完成添加设备后，可以在设备列表中查看设备状态，并管理边缘设备。也可以单击“设备ID”跳转至“设备接入服务控制台”进行设备管理。

表 4-3 操作说明

操作名称	说明
配置	查看边缘设备的原有配置，或者修改边缘设备的配置。
删除	删除边缘设备。 <b>说明</b> 若您已在“设备接入服务控制台”删除边缘设备，请前往“IoT边缘控制台”继续删除该设备，否则“IoT边缘服务”将继续对此设备计费。
管理子设备	查看边缘设备详情，以及注册子设备，具体操作可参考 <a href="#">网关与子设备管理</a> 。

**步骤4** 将设备连接到边缘节点上。

**步骤5** 验证设备上报的数据能在“IoT边缘控制台”中查看。

- 设备上电，并向物联网平台上报数据。
- 访问[IoT边缘](#)，单击“管理控制台”进入IoT边缘控制台。
- 选择左侧导航栏的“边缘节点 > 节点管理”，选择您的边缘节点名称，进入节点详情页。
- 选择“边缘设备”页签，单击“管理子设备”查看对应设备的状态。如果状态为“在线”，则表示设备已经成功接入物联网平台。

图 4-4 管理子设备



- 返回“边缘设备”页签，单击对应的“设备ID”进入设备详情页，在详情页中查看“最近上报数据”，如果能正常解析和显示对应的数据，则表示设备上报数据成功。

#### 📖 说明

如果需要查看所有上报的历史数据，则可以在设备详情的“历史数据”中进行查看。

----结束

## 设备通过网关连接边缘节点

设备通过一些简单的近场通信协议如ZigBee、ZWave、Bluetooth或者是其他的一些非IP有线方式传输协议如串口、并口等接入网关，网关通过集成IoT Device SDK连接边缘节点，边缘节点透传网关上报的数据，以及将平台的命令下发给网关。

请参考以下流程完成设备接入：

- 步骤1** 在**设备接入服务**完成网关、设备的产品模型（也称Profile）、编解码插件的开发，并进行自助测试。

对于通过边缘节点上报至物联网平台的网关和设备的数据，云端物联网平台要能够解析。

- **Profile**是用来描述设备能力的文件，通过JSON的格式定义了设备的基本属性、上报数据和下发命令的消息格式。定义Profile，即在物联网平台构建一款设备的抽象模型，在设备上报数据时，平台能够根据定义的Profile理解该款设备支持的属性信息。
- **编解码插件**开发要根据设备上报数据的格式来判断，如果设备上报的“数据格式”为“二进制码流”，则该产品模型下需要进行编解码插件开发。

具体操作请参照“设备接入服务”的**开发指南**中**产品开发**的指导。

- 步骤2** 进行网关接入物联网平台的设备侧开发。

1. 在网关集成IoT Device SDK接入平台。目前IoT Device SDK支持C、Java两个版本。实际开发中，请根据开发时使用的语言、平台，选用合适的SDK集成。具体请参照“设备接入服务”**开发指南**中**IoT Device SDK使用指南**的指导。
2. 将网关接入的IP地址修改为需要连接的边缘节点本地的IP地址。

#### 📖 说明

通过MQTT协议的网关接入边缘节点的端口地址是7883。

- 步骤3** 在“IoT边缘控制台”添加网关。物联网平台中，网关也是作为一个设备注册。具体请参照**步骤3**。

- 步骤4** 将网关连接到边缘节点上。

- 步骤5** 将子设备连接到网关。

- 步骤6** 验证设备上报的数据能在“IoT边缘控制台”中查看。

1. 设备上电，并向物联网平台上报数据。
2. 访问**IoT边缘**，单击“管理控制台”进入IoT边缘控制台。
3. 选择左侧导航栏的“边缘节点 > 节点管理”，选择您的边缘节点名称，进入节点详情页。
4. 选择“边缘设备”页签，单击对应的“设备ID”跳转至“设备接入服务控制台”进入设备详情页。

5. 选择“子设备”页签，在设备列表中查看对应设备的状态。如果状态为“在线”，则表示设备已经成功接入物联网平台。

**图 4-5 查看设备状态**

### 说明

子设备的状态表示子设备接入网关的状态，由网关上报到物联网平台进行状态的刷新；如果网关不能正常上报子设备的状态信息到物联网平台，则展示的子设备状态不会刷新。例如：某子设备通过网关接入到物联网平台，子设备状态为在线状态，如果此时网关与物联网平台断开连接，则网关不能上报子设备的状态到物联网平台，该子设备的状态会一直显示在线。

6. 单击“查看”具体的子设备，可以查看子设备的[设备详情](#)、[历史上报数据](#)、[查看操作记录](#)。

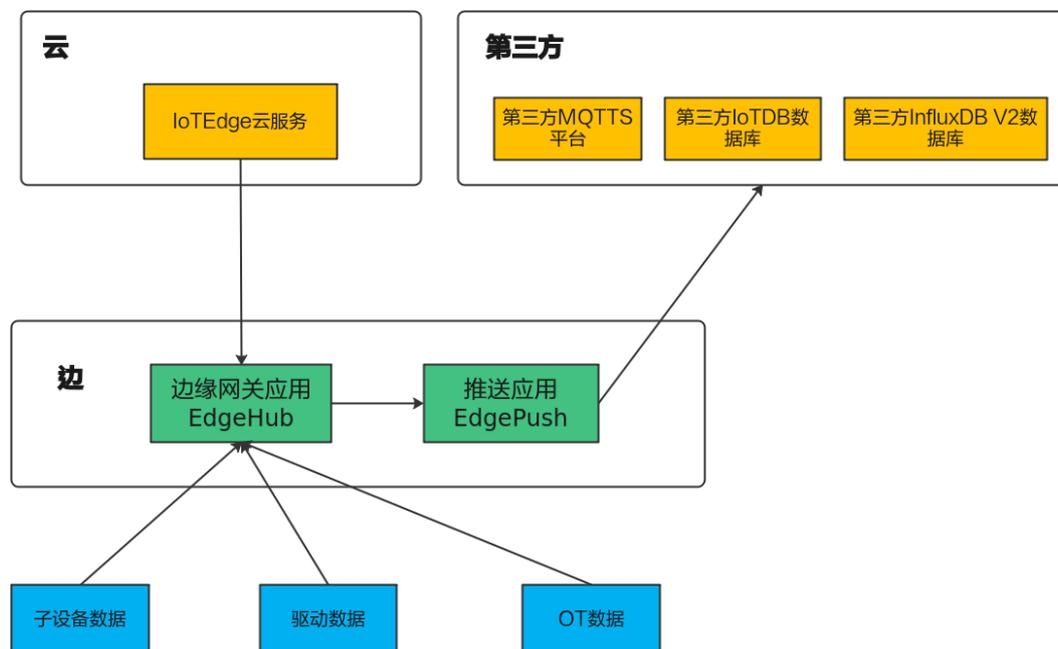
----结束

# 5 路由转发

## 5.1 路由转发概述

边缘节点在本地作为设备网关，除了收集边缘设备上报的设备数据外，还支持对接第三方云平台、外部实体等其他应用。通过在边缘节点部署\$edge\_push业务应用，将业务数据转发给云端应用或后端服务以实现业务需求。路由转发可配置本地设备数据转发的接收端，当前支持MQTT、IoTDB和InfluxDB V2数据端对接。路由转发支持将推送失败的数据进行离线缓存。

图 5-1 路由转发流程图



当前支持将子设备数据、驱动数据以及OT数据，推送给第三方MQTT平台、IoTDB数据库以及InfluxDB V2数据库，具体格式请参考[使用步骤](#)。

## 使用限制

- 单用户最大配置10个通道，每个通道最多分配100个节点，单通道流量最大100TPS。
- 专业版节点KubeEdge不支持离线缓存。
- 注册节点时，仅当上报优先级为“按时间顺序”才能支持路由转发开启离线缓存。

## 使用步骤

步骤1 [创建MQTT/loTDB/InfluxDB V2通道](#)

步骤2 [在节点部署EdgePush推送应用](#)

步骤3 [将通道分配到“数据需要推送”的节点](#)

步骤4 [按需修改节点的通道详情](#)

步骤5 [将节点的通道信息下发给网关](#)

----结束

## 5.2 通道类型概述

### 5.2.1 MQTT 类型通道

MQTT通道类型需要填写的字段，如下所示：

表 5-1 MQTT 通道参数

字段	含义
接收端名称	通道的名称
通道类型	MQTT
接收地址	第三方平台MQTT Broker的地址（只支持tls通道），例：ssl://127.0.0.1:7883
<b>连接信息</b>	
鉴权方式	当前只支持用户密码类型
ClientId	MQTT鉴权的ClientId（创建通道时，不可填。通道分配到节点后，可自定义修改，默认为节点ID。）
用户名	MQTT鉴权的用户名
密码	MQTT鉴权的密码
信任证书	MQTT Broker的CA证书（可选项，未上传证书表示不校验MQTT Broker的服务端证书）；是否校验域名（在取消勾选时，只会校验服务端证书，不会校验证书中的域名）

字段	含义
推送信息	
Topic	MQTT Broker数据推送Topic, 例: \$oc/devices/gateway
服务质量	MQTT QOS, 默认qos 0
描述	通道的描述信息

- 当节点的设备数据格式为“**IoTDA1.0物模型**”时, MQTT通道, 推送的数据格式为IoTDA 1.0的物模型格式, 报文格式如下:

```
{
  "devices": [{
    "device_id": "device1",
    "services": [{
      "service_id": "Motor",
      "properties": {
        "Current": 10.111,
        "Voltage": 20
      },
      "event_time": "2006-01-02T15:04:05.000Z"
    }
  ]
}
```

- 当节点的设备数据格式为“**属性平铺格式**”时, MQTT通道, 推送的数据格式为属性平铺格式, 报文格式如下:

```
{
  "devices": [
    {
      "device_id": "device1",
      "properties": {
        "Motor/Current": 10.111,
        "Motor/Voltage": 20
      },
      "event_time": "2006-01-02T15:04:05.000Z"
    }
  ]
}
```

## 5.2.2 IoTDB 类型通道

### 须知

该通道只支持设备数据格式为“IoTDA1.0物模型”的节点。

IoTDB类型通道需要填写的字段, 如下所示:

表 5-2 IoTDB 通道参数

字段	含义
接收端名称	通道的名称。
通道类型	数据库。
数据库类型	IoTDB。
接收地址	第三方平台IoTDB的地址， 例： 127.0.0.1:6667。
<b>连接信息</b>	
用户名	IoTDB鉴权的用户名。
密码	IoTDB鉴权的密码。
<b>推送信息</b>	
存储组	IoTDB写入数据的存储（需要在IoTDB授权对应用户写该存储的权限），例： edge。
描述	通道的描述信息。

在IoTDB时序数据库中，所有存储组的固定前缀为“root.”，如参数“存储组”为“edge”时，写入的数据库的存储组为“root.edge”。

**注意**

使用IoTDB “1.1.x” 版本，需要确保部署的是“1-0-5-standard-xx”及以上版本的edge\_push。请注意，“1-0-5-standard-xx”及以上版本的edge\_push，不支持“1.1.x”以下版本的IoTDB。

写入数据库的数据格式为华为云IoTDA 1.0的数据格式，例如IoTDA 1.0的报文如下：

```
{
  "devices": [{
    "device_id": "device1",
    "services": [{
      "service_id": "Motor",
      "properties": {
        "Current": 10.111,
        "Voltage": 20
      },
      "event_time": "2006-01-02T15:04:05.000Z"
    }
  ]
}
```

则写入数据库格式为：

```
root.edge.device1.Motor.Current => 10.111
root.edge.device1.Motor.Voltage => 20
```

## 5.2.3 InfluxDB V2 类型通道

### 须知

该通道只支持设备数据格式为“IoTDA1.0物模型”的节点。

InfluxDB V2类型通道需要填写的字段，如下所示：

表 5-3 InfluxDB V2 通道参数

字段	含义
接收端名称	通道的名称。
通道类型	数据库。
数据库类型	InfluxDB V2。
接收地址	第三方平台InfluxDB V2的地址， 例： https://127.0.0.1:8086或http://127.0.0.1:8086。
<b>连接信息</b>	
Token	InfluxDB v2用来做鉴权和校验身份的token。
信任证书	InfluxDB v2的CA证书（可选项，未上传证书表示不校验MQTT Broker的服务端证书）；是否校验域名（在取消勾选时，只会校验服务端证书，不会校证书中的域名）。
<b>推送信息</b>	
组织	InfluxDB V2中的organization。
桶	InfluxDB V2中的bucket。
描述	通道的描述信息。

在InfluxDB V2中bucket下measurement的概念，写入数据库的数据为华为云IoTDA 1.0物模型格式，

例如IoTDA 1.0的报文如下，该物模型的产品名为“ElectricalMachinery”：

```
{
  "devices": [{
    "device_id": "device1",
    "services": [{
      "service_id": "Motor",
      "properties": {
        "Current": 10.111,
        "Voltage": 20
      },
      "event_time": "2006-01-02T15:04:05.000Z"
    }
  ]
}
```

```
]
}
```

写入数据库时，是以“产品名”为measurement写入数据库，同时以设备ID和服务ID为tag，上述报文写入格式如下：

```
measurement: ElectricalMachinery
tags: {"device": "device1", "service": "Motor"} fields: {"Current": 10.111, "Voltage": 20}
```

## 5.3 创建通道

路由转发提供管理数据接收端通道的能力，包括基本的新增、删除、修改以及分配等功能。

### 操作步骤

- 步骤1** 访问[IoT边缘](#)，单击“**管理控制台**”进入IoT边缘控制台。
- 步骤2** 选择左边导航栏“**边缘节点 > 路由转发**”进入页面，单击右上角“**添加数据接收端点**”，添加通道；

当前通道支持MQTT、IoTDB、InfluxDB V2三种通道类型。

MQTT通道界面如下图所示，参数详情见“[MQTT类型通道](#)”，单击“**确认**”创建。

## 添加接收端信息

×

\* 接收端点名称

通道  MQTT  数据库

\* 接收地址

**连接信息**

鉴权方式  用户密码

\* 用户名

\* 密码  

信任证书 [添加信任证书](#)

校验域名

**推送信息**

\* Topic

\* 服务质量   0  1  2

IoTDB通道界面如下图所示，参数详情见“[IoTDB类型通道](#)”，单击“确认”创建。

### 添加接收端信息

×

* 接收端点名称	<input type="text" value="IoTDB"/>
通道	<input type="radio" value="MQTT"/> <input checked="" type="radio" value="数据库"/>
数据库类型	<input checked="" type="radio" value="IoTDB"/> <input type="radio" value="InfluxDB_V2"/>
* 接收地址	<input type="text" value="127.0.0.1:6663"/>
连接信息	
* 用户名	<input type="text" value="kzxkzx"/>
* 密码	<input type="password" value="....."/>
推送信息	
数据格式	<input checked="" type="radio" value="物模型格式"/>
* 存储组	<input type="text" value="edge"/>
描述	<input type="text"/>

InfluxDB V2通道界面如下图所示，参数详情见“[InfluxDB V2类型通道](#)”，单击“确认”创建。

### 添加接收端信息 ×

★ 接收端点名称

通道

数据库类型

★ 接收地址

连接信息

★ token

信任证书 [添加信任证书](#)

校验域名

推送信息

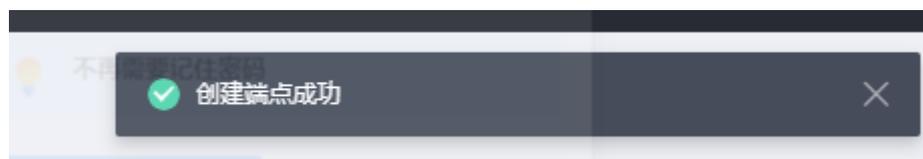
数据格式

★ 组织

★ 桶

★ 桶前缀

### 步骤3 创建成功



----结束

## 5.4 节点部署 EdgePush 推送应用

EdgePush用于将业务数据转发给第三方云平台、外部实体等其他服务。系统会根据路由转发是否开启离线缓存创建不同的默认数据流转规则，将设备数据流转到EdgePush模块。

- 路由转发未开启离线缓存，节点上首次部署EdgePush模块时，系统会创建默认数据流转规则(规则名称为default\_to\_push)。

- 路由转发开启离线缓存，将节点的通道信息下发给网关时，系统会创建对应通道的数据流转规则(规则名称格式为default\_to\_push\_channel\_{channel\_id}，{channel\_id}为通道的唯一标识)，数据转发失败会进行重试。

## 操作前提

路由转发功能需要1-1-21以上的edgehub版本支持。

## 操作步骤

- 步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。
- 步骤2** 选择左侧导航栏“边缘节点 > 节点管理”进入页面，单击相应的边缘节点名称。
- 步骤3** 选择“应用模块 > 模块管理”，单击“部署应用”。
- 步骤4** 选择“数据处理”，选择应用\$edge\_push，选择版本，单击“确认”。

图 5-2 数据处理



- 步骤5** 部署成功。



步骤6 选择“数据流转配置”，查看默认数据流转规则。

图 5-3 数据流转

规则名称	消息类型	消息来源	消息目标	规则是否可用
default	设备数据	设备端	云端	是
default_to_push	设备数据	设备端	sys_edge_push_flat_json	是

### 说明

删除或修改该默认数据流转规则会影响EdgePush接收到的设备数据，进而影响节点的路由转发功能，请谨慎操作。

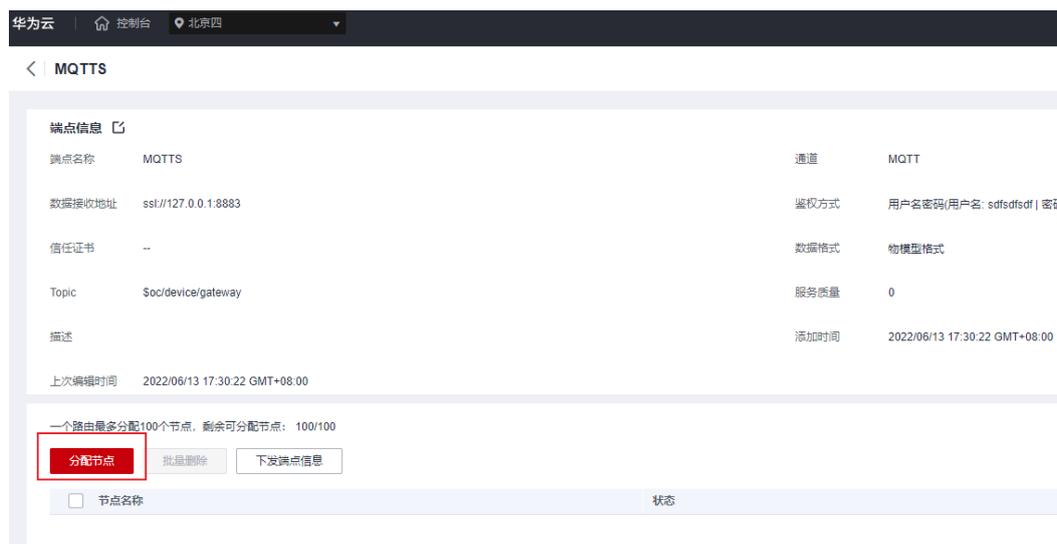
----结束

## 5.5 通道分配

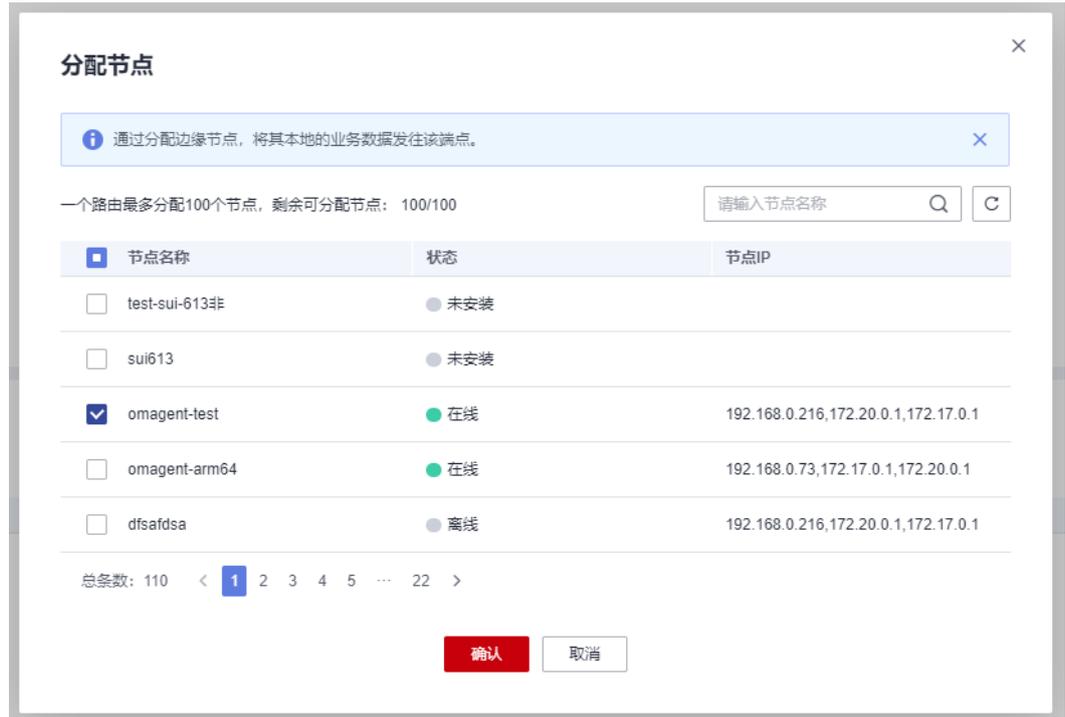
### 5.5.1 MQTT 类型通道分配至节点

#### 操作步骤

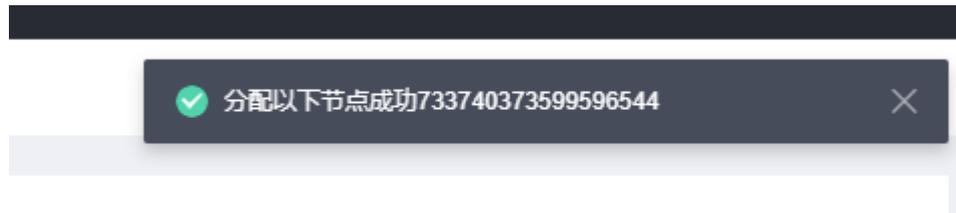
- 步骤1 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。
- 步骤2 在左边导航栏选择“边缘节点 > 路由转发”“进入页面，单击创建通道创建的“MQTT”类型通道的端点名称。
- 步骤3 进入通道详情界面，单击“分配节点”。



- 步骤4 进入分配节点界面，勾选节点部署EdgePush推送应用步骤部署edgepush的“节点”，单击“确认”。



步骤5 右上角显示分配成功。

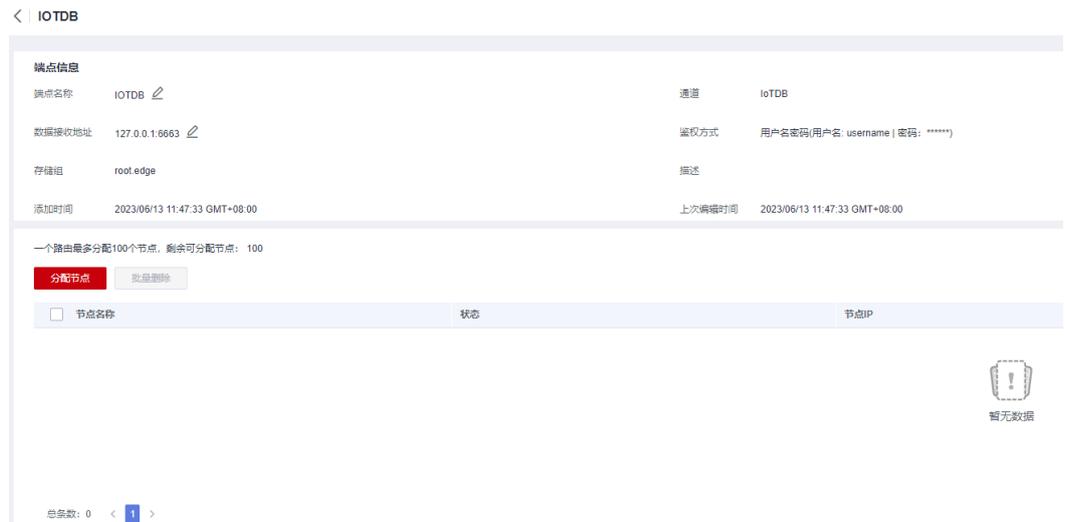


----结束

## 5.5.2 IoTDB 类型通道分配至节点

### 操作步骤

- 步骤1 访问[IoT边缘](#)，单击“[管理控制台](#)”进入IoT边缘控制台。
- 步骤2 在左边导航栏选择“边缘节点 > 路由转发”“进入页面，单击[创建通道](#)创建的“IoTDB”类型通道的端点名称。
- 步骤3 进入通道详情界面，单击“分配节点”。



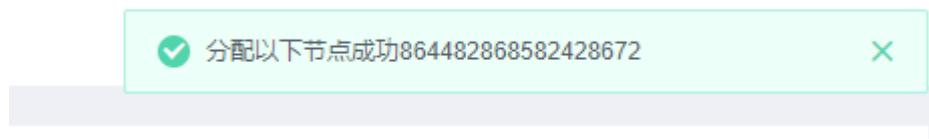
**步骤4** 进入分配节点界面，勾选**节点部署EdgePush推送应用**步骤部署edgepush的“节点”，单击“确认并下发”。

### 须知

IoTDB暂不开放按节点修改通道，所以在分配节点时，单击“确认并下发”按钮后，通道信息就已下发到边缘节点。



**步骤5** 右上角显示分配成功。

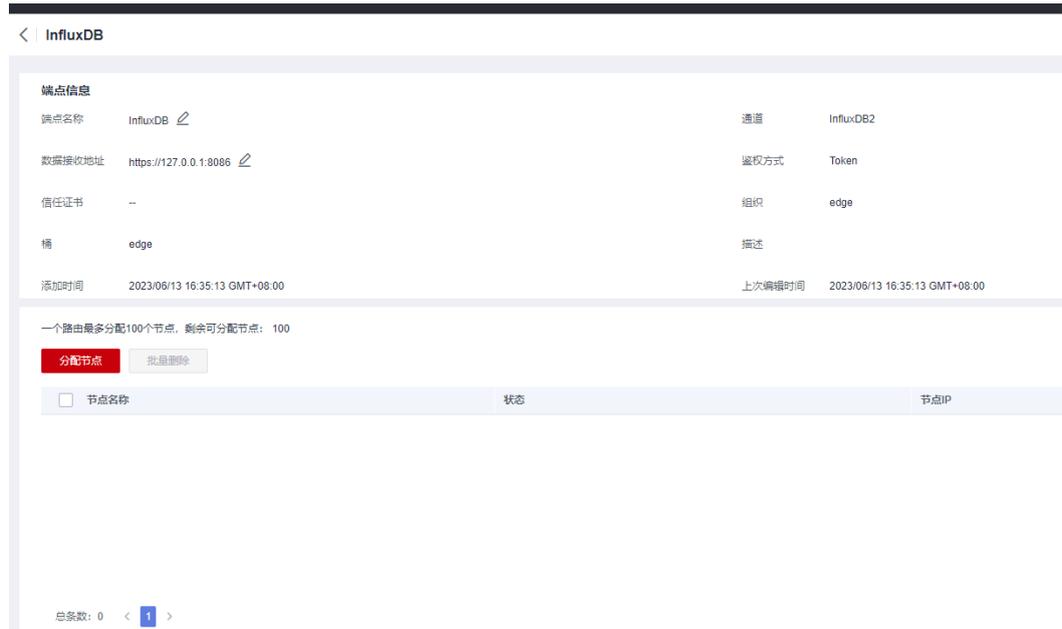


----结束

## 5.5.3 InfluxDB V2 类型通道分配至节点

### 操作步骤

- 步骤1** 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。
- 步骤2** 在左边导航栏选择“边缘节点 > 路由转发”“进入页面，单击创建通道创建的“InfluxDB”类型通道的端点名称。
- 步骤3** 进入通道详情界面，单击“分配节点”。



- 步骤4** 进入分配节点界面，勾选节点部署EdgePush推送应用步骤部署edgepush的“节点”，单击“确认并下发”。

### 须知

InfluxDB暂不开放按节点修改通道，所以在分配节点时，单击“确认并下发”按钮后，通道信息就已下发到边缘节点。



步骤5 右上角显示分配成功。



----结束

## 5.6 修改节点的通道详情（可选）

### 须知

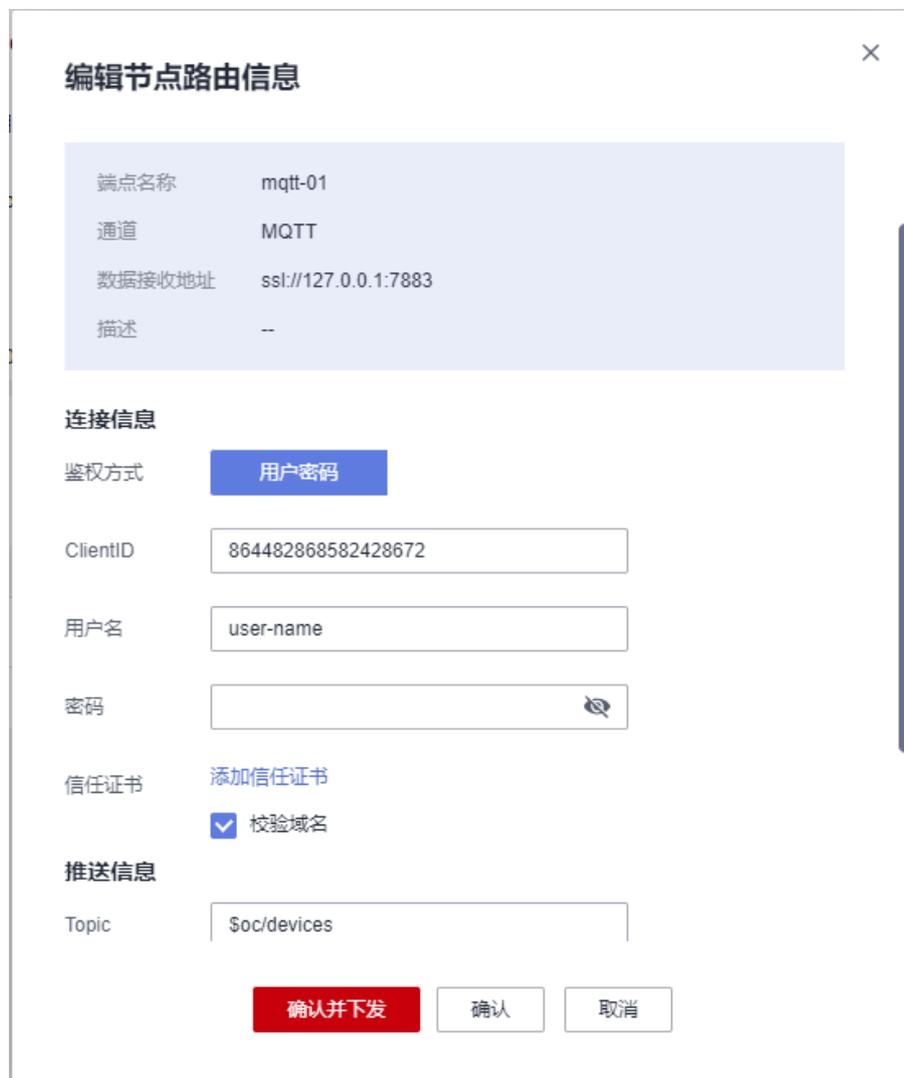
当前只支持“MQTT”类型通道，按节点修改路由信息。

### 操作步骤

- 步骤1 访问IoT边缘，单击“管理控制台”进入IoT边缘控制台。
- 步骤2 在左边导航栏选择“边缘节点 > 路由转发”“进入页面，单击创建通道创建的“MQTT”类型通道的端点名称。
- 步骤3 进入通道详情界面，选择分配的节点，单击“修改路由”按钮。



**步骤4** 按需修改节点通道信息，其中ClientId默认为节点ID。



**步骤5** 修改完成后，单击“确认”按钮，表示只保存修改，不下发到边缘网关，节点的通道状态为“待下发”。



----结束

## 5.7 将节点的通道信息下发给网关

### 操作步骤

- 步骤1** 访问IoT边缘, 单击“管理控制台”进入IoT边缘控制台。
- 步骤2** 在左边导航栏选择“边缘节点 > 路由转发”“进入页面, 单击创建通道创建的“MQTT”类型通道的端点名称。
- 步骤3** 进入通道详情界面, 选择分配的节点, 勾选状态为“待下发”的节点, 单击“下发”按钮。



- 步骤4** 界面弹出下发确认界面, 检查选择的节点是否正确, 单击“确认”按钮。



步骤5 通道的节点状态变为“已下发”。



----结束

# 6 审计

## 概述

云审计服务（Cloud Trace Service，以下简称CTS），是公有云安全解决方案中专业的日志审计服务，提供对各种云资源操作记录的收集、存储和查询功能，可用于支撑安全分析、合规审计、资源跟踪和问题定位等常见应用场景。

查看审计日志的详细操作请参考[查看审计日志](#)。

## 支持云审计的关键操作

平台提供了云审计服务。开通云审计服务后，通过云审计服务，您可以记录与云服务器相关的操作事件，便于日后的查询、审计和回溯。

表 6-1 云审计服务支持的 IoT 边缘云服务操作列表

操作名称	资源类型	事件名称
创建远程运维连接	node	createRemoteOps
创建委托授权	agency	createAgency
创建数据清洗规则	cleanRule	createCleanRule
更新数据清洗规则	cleanRule	updateCleanRule
删除数据清洗规则	cleanRule	deleteCleanRule
复制数据清洗规则	cleanRule	copyCleanRule
创建节点数据清洗规则	cleanRule	createNodeCleanRule
批量删除节点数据清洗规则	cleanRule	batchDeleteNodeCleanRule
删除节点数据清洗规则	cleanRule	deleteNodeCleanRule
发布数据清洗规则	cleanRule	publishCleanRule
创建应用	application	createApplication
删除应用	application	deleteApplication

操作名称	资源类型	事件名称
创建应用版本	application	createApplicationVersion
更新应用版本	application	updateApplicationVersion
删除应用版本	application	deleteApplicationVersion
更新应用版本状态	application	updateApplicationVersionStatus
创建边缘节点	node	createNode
更新边缘节点	node	updateNode
删除边缘节点	node	deleteNode
添加设备	device	createDevice
更新设备	device	updateDevice
删除设备	device	deleteDevice
生成modbus协议设备接入码	device	createDevicieAccessCode
创建外部实体	externals	createExternal
更新外部实体	externals	updateExternal
删除外部实体	externals	deleteExternal
创建边缘模块	module	createModule
更新边缘模块	module	updateModule
删除边缘模块	module	deleteModule
配置边缘节点路由	route	configNodeRoute
生成边缘节点安装命令	node	createInstallCommand
在IEF节点上部署边缘节点	node	deployNode
上传协议映射文件	product	uploadProtocolMappingsFile
用户批量上传/导入南向3rdIA配置项	itApplication	batchImportConfigs
创建&更新南向3rdIA配置项信息	itApplication	createAndUpdateConfig
删除南向3rdIA配置项	itApplication	deleteConfig
创建/更新北向NA	northApplication	createAndUpdateNA
删除北向NA	northApplication	deleteNA
向边缘节点授权北向NA	northApplication	authorizeNA

操作名称	资源类型	事件名称
绑定设备方式创建边缘节点	bindNode	Create Node From Bind Device
批量加密	batchEncrypt	Batch Encrypt
批量解密	batchDecrypt	Batch Decrypt
更新节点数采接入信息	node	createImplementationInfo
采集服务绑定边缘节点	resource	resourceBindNode
采集服务解绑边缘节点	resource	resourceUnbindNode