

智能边缘平台

最佳实践

文档版本 01
发布日期 2021-11-30



版权所有 © 华为技术有限公司 2022。保留一切权利。

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

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

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

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

目录

1 通过专线或 VPN 连接 IEF.....	1
2 使用开源 C 语言库连接 MQTT Broker.....	6

1 通过专线或 VPN 连接 IEF

操作场景

线下边缘节点无法通过公网访问IEF时，可以选择通过**云专线（DC）**或**VPN**连接华为云VPC，然后通过**VPC终端节点**在VPC提供私密安全的通道连接IEF，从而使得线下边缘节点在无法访问公网时连接IEF。

连接方案说明

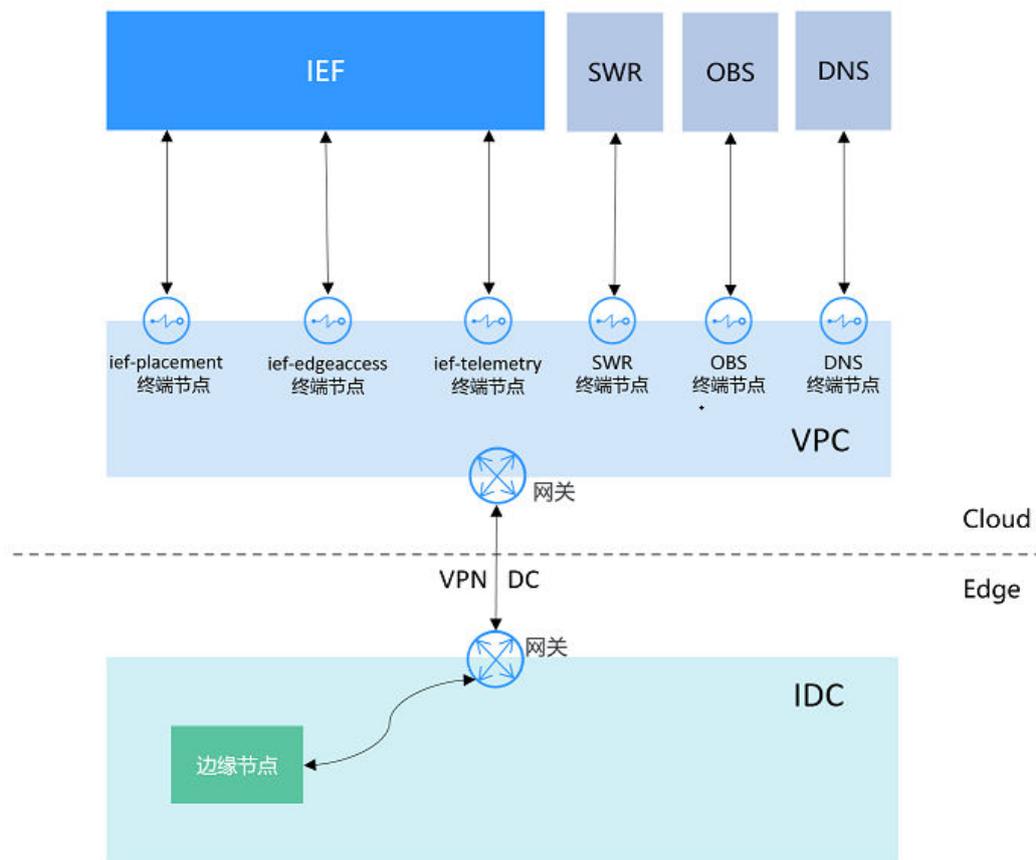
纳管边缘节点部署应用时，需要能够与IEF、SWR、OBS通信，在无法通过公网连接的情况下，可以先通过VPN或专线（DC）与华为云VPC连接，然后通过VPC终端节点服务，让VPC能够在内网访问IEF、SWR和OBS，具体连接方案如**图1-1**所示。

与IEF连接需要创建三个终端节点，分别为如下三个。

- ief-placement：用于边缘节点的纳管和升级。
- ief-edgeaccess：用于边缘节点与IEF发送边云消息。
- ief-telemetry：边缘节点上传监控和日志数据。

与SWR连接需要创建一个终端节点，与OBS通信需要创建OBS和DNS两个终端节点（OBS只能通过域名访问，需要通过DNS动态解析OBS的地址才能访问到）。

图 1-1 通过专线或 VPN 连接 IEF



操作步骤

步骤1 在华为云上创建VPC。

创建VPC的方法请参见[创建虚拟私有云和子网](#)。

您也可以使用已有VPC。

须知

VPC网段不能与IDC的网段重复。

步骤2 使用DC或VPN连接VPC。

具体连接方法请参见[如下链接](#)。

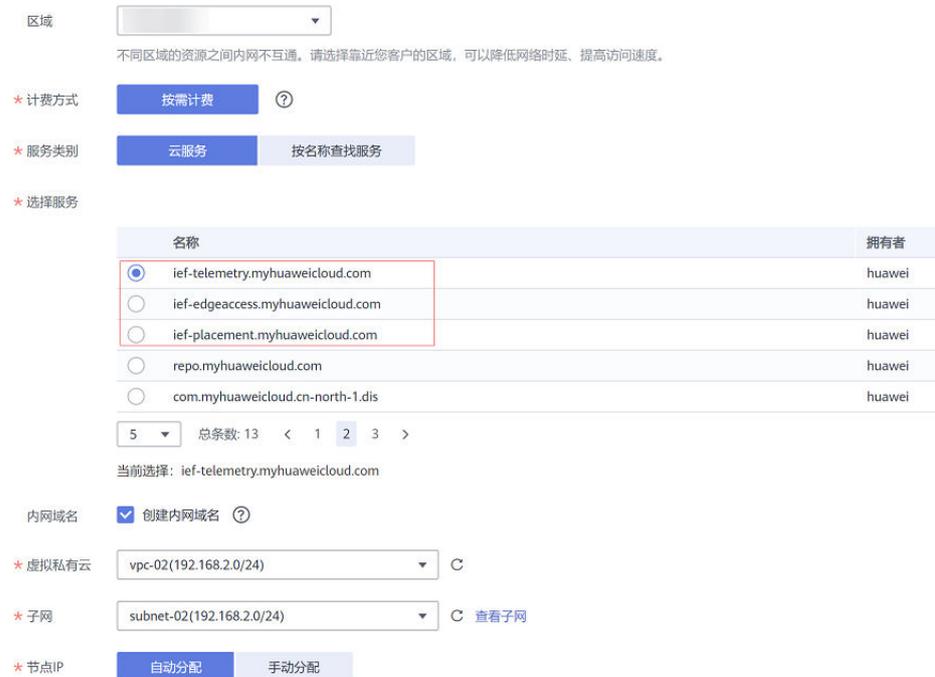
- VPN: https://support.huaweicloud.com/intl/zh-cn/qs-vpn/zh-cn_topic_0133627788.html
- DC: https://support.huaweicloud.com/intl/zh-cn/qs-dc/zh-cn_topic_0145790541.html

步骤3 创建IEF终端节点，使得边缘节点能够与IEF连接。

共需要创建三个终端节点，分别为ief-placement、ief-edgeaccess和ief-telemetry。具体创建步骤如下。

1. 登录**VPCEP控制台**，单击右上角的“购买终端节点”。
2. 选择IEF的终端节点和虚拟私有云，如下图所示。

图 1-2 创建 IEF 终端节点



区域

不同区域的资源之间内网不互通。请选择靠近您客户的区域，可以降低网络时延、提高访问速度。

* 计费方式 **按需计费** ?

* 服务类别 **云服务** 按名称查找服务

* 选择服务

名称	拥有者
<input checked="" type="radio"/> ief-telemetry.myhuaweicloud.com	huawei
<input type="radio"/> ief-edgeaccess.myhuaweicloud.com	huawei
<input type="radio"/> ief-placement.myhuaweicloud.com	huawei
<input type="radio"/> repo.myhuaweicloud.com	huawei
<input type="radio"/> com.myhuaweicloud.cn-north-1.dis	huawei

5 总条数: 13 < 1 2 3 >

当前选择: ief-telemetry.myhuaweicloud.com

内网域名 创建内网域名 ?

* 虚拟私有云 vpc-02(192.168.2.0/24) C

* 子网 subnet-02(192.168.2.0/24) C 查看子网

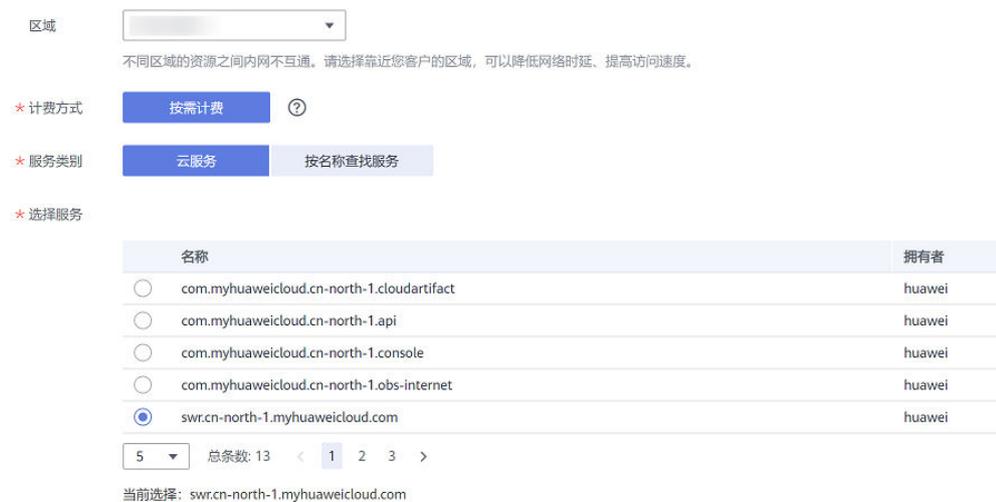
* 节点IP **自动分配** 手动分配

3. 单击“立即购买”，确认信息无误后单击“提交”，完成创建。

步骤4 创建SWR终端节点，使得边缘节点能够从SWR拉取容器镜像。

创建方法与**创建IEF终端节点**相同。

图 1-3 创建 SWR 终端节点



区域

不同区域的资源之间内网不互通。请选择靠近您客户的区域，可以降低网络时延、提高访问速度。

* 计费方式 **按需计费** ?

* 服务类别 **云服务** 按名称查找服务

* 选择服务

名称	拥有者
<input type="radio"/> com.myhuaweicloud.cn-north-1.cloudartifact	huawei
<input type="radio"/> com.myhuaweicloud.cn-north-1.api	huawei
<input type="radio"/> com.myhuaweicloud.cn-north-1.console	huawei
<input type="radio"/> com.myhuaweicloud.cn-north-1.obs-internet	huawei
<input checked="" type="radio"/> swr.cn-north-1.myhuaweicloud.com	huawei

5 总条数: 13 < 1 2 3 >

当前选择: swr.cn-north-1.myhuaweicloud.com

步骤5 创建DNS和OBS终端节点，使得边缘节点能够访问OBS。

具体方法请参见**访问OBS**。

步骤6 给边缘节点添加hosts配置。

查询IEF和SWR的终端节点IP地址，共4个IP地址，配置到边缘节点的“/etc/hosts”文件中。

图 1-4 查询终端节点 IP 地址



打开“/etc/hosts”文件，在文件末尾加入如下配置，使得访问IEF和SWR的域名指向终端节点的IP地址。

须知

此处IP地址和域名需要根据实际情况修改，IP地址为上面步骤查询到的地址，不同区域的域名不相同，具体请参见[域名地址](#)。

```
192.168.2.20 ief2-placement.cn-north-1.myhuaweicloud.com
192.168.2.142 ief2-edgeaccess.cn-north-1.myhuaweicloud.com
192.168.2.106 ief2-telemetry.cn-north-1.myhuaweicloud.com
192.168.2.118 swr.cn-north-1.myhuaweicloud.com
```

步骤7 注册并纳管边缘节点，具体步骤请参见[边缘节点概述](#)。

---结束

域名地址

说明

铂金版ief-edgeaccess有单独的地址，请在IEF控制台“总览”页面查询，云端接入域名的取值即为edgeaccess域名。

区域	名称	域名
华北-北京一	ief-placement	ief2-placement.cn-north-1.myhuaweicloud.com
	ief-edgeaccess	ief2-edgeaccess.cn-north-1.myhuaweicloud.com
	ief-telemetry	ief2-telemetry.cn-north-1.myhuaweicloud.com
	swr	swr.cn-north-1.myhuaweicloud.com
华北-北京四	ief-placement	ief2-placement.cn-north-4.myhuaweicloud.com
	ief-edgeaccess	ief2-edgeaccess.cn-north-4.myhuaweicloud.com

区域	名称	域名
	ief-telemetry	ief2-telemetry.cn-north-4.myhuaweicloud.com
	swr	swr.cn-north-4.myhuaweicloud.com
华南-广州	ief-placement	ief-placement.cn-south-1.myhuaweicloud.com
	ief-edgeaccess	ief-edgeaccess.cn-south-1.myhuaweicloud.com
	ief-telemetry	ief-telemetry.cn-south-1.myhuaweicloud.com
	swr	swr.cn-south-1.myhuaweicloud.com
华东-上海一	ief-placement	ief-placement.cn-east-3.myhuaweicloud.com
	ief-edgeaccess	ief-edgeaccess.cn-east-3.myhuaweicloud.com
	ief-telemetry	ief-telemetry.cn-east-3.myhuaweicloud.com
	swr	swr.cn-east-3.myhuaweicloud.com
华东-上海二	ief-placement	ief2-placement.cn-east-2.myhuaweicloud.com
	ief-edgeaccess	ief2-edgeaccess.cn-east-2.myhuaweicloud.com
	ief-telemetry	ief2-telemetry.cn-east-2.myhuaweicloud.com
	swr	swr.cn-east-2.myhuaweicloud.com

2 使用开源 C 语言库连接 MQTT Broker

操作场景

MQTT是一种发布/订阅模式的消息协议，能够在硬件性能低下的远程设备以及网络状况糟糕的情况下工作。

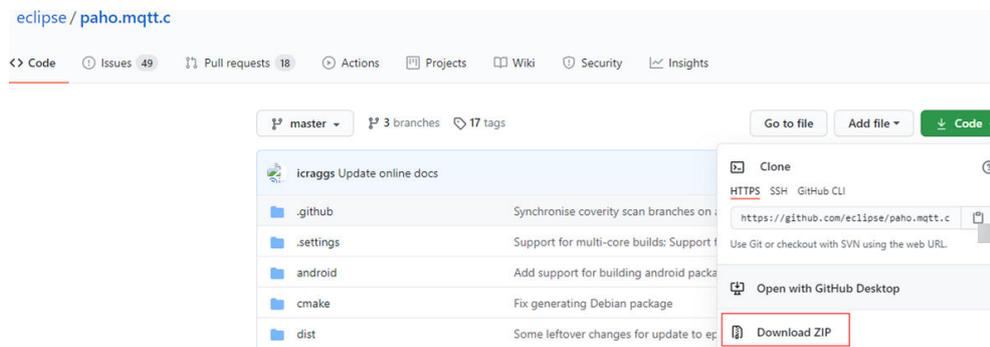
本文介绍一种开源的C语言库Eclipse Paho C Client Library连接使用IEF的内置MQTT Broker。

操作步骤

步骤1 准备一台Linux机器，下载源码。

git clone https://github.com/eclipse/paho.mqtt.c.git

或直接在<https://github.com/eclipse/paho.mqtt.c>页面下载zip包，然后解压。



步骤2 安装编译依赖工具。

Ubuntu系统执行如下命令。

apt-get install build-essential gcc make cmake cmake-gui cmake-curses-gui

apt-get install fakeroot fakeroot devscripts dh-make lsb-release

apt-get install libssl-dev

apt-get install ninja-build

CentOS系统执行如下命令。

yum install build-essential gcc make cmake cmake-gui cmake-curses-gui

```
yum install fakeroot fakeroot devscripts dh-make lsb-release
```

```
yum install openssl-devel
```

```
yum install ninja-build
```

步骤3 修改示例程序代码。

在源码的“src/samples/paho_cs_pub.c”文件中，增加如下行。

```
ssl_opts.enableServerCertAuth = 0;
```

```
94         ssl_opts.CApath = opts.copath;
95         ssl_opts.keyStore = opts.cert;
96         ssl_opts.trustStore = opts.cafile;
97         ssl_opts.privateKey = opts.key;
98         ssl_opts.privateKeyPassword = opts.keypass;
99         ssl_opts.enabledCipherSuites = opts.ciphers;
100        ssl_opts.enableServerCertAuth = 0;
101        conn_opts.ssl = &ssl_opts;
```

在“src/samples/paho_cs_sub.c”文件中增加如下行。

```
91         ssl_opts.CApath = opts.copath;
92         ssl_opts.keyStore = opts.cert;
93         ssl_opts.trustStore = opts.cafile;
94         ssl_opts.privateKey = opts.key;
95         ssl_opts.privateKeyPassword = opts.keypass;
96         ssl_opts.enabledCipherSuites = opts.ciphers;
97        ssl_opts.enableServerCertAuth = 0;
98        conn_opts.ssl = &ssl_opts;
```

步骤4 编译示例程序。

```
mkdir /tmp/build.paho
```

```
cd /tmp/build.paho
```

```
cmake -GNinja -DPAHO_BUILD_STATIC=TRUE -DPAHO_BUILD_SHARED=FALSE
-DPAHO_WITH_SSL=TRUE -DPAHO_BUILD_SAMPLES=TRUE {paho.mqtt.c目录}
```

```
ninja package
```

其中 {paho.mqtt.c目录} 为 paho.mqtt.c 源码所在的目录，如“/root/work/paho.mqtt.c”。

步骤5 进入编译之后的目录，将编译生成的二进制文件“paho_cs_pub”和“paho_cs_sub”拷贝至边缘节点上。

```
cd /tmp/build.paho/src/samples/
```

```
(base) root@cci-clustermanager-xsw:/tmp/build.paho/src/samples# ls
CMakeFiles          MQTTAsync_publish  MQTTAsync_subscribe MQTTClient_publish_async paho_c_pub  paho_cs_sub
cmake_install.cmake MQTTAsync_publish_time MQTTClient_publish  MQTTClient_subscribe  paho_cs_pub  paho_c_sub
```

步骤6 下载边缘节点证书。

1. 登录 IEF 控制台，在左侧选择“边缘资源 > 边缘节点”，在右侧单击边缘节点名称，进入边缘节点详情页。选择“证书”页签，单击“添加证书”。

图 2-1 添加证书



2. 在弹出的窗口中输入证书名称，单击“确定”。
3. 将下载好的证书，拷贝至边缘节点，并解压。

步骤7 运行示例程序。

以发布消息到指定topic为示例，查看“paho_cs_pub”命令指导。

```
(base) root@cci-clustermanager-xsw:/tmp/build.paho/src/samples# ./paho_c_pub
Eclipse Paho MQTT C publisher

Library information:
Product name: Eclipse Paho Asynchronous MQTT C Client Library
Version: 1.3.8
Build level: 2021-01-20T14:05:21Z
OpenSSL version: OpenSSL 1.1.0l 10 Sep 2019
OpenSSL flags: compiler: gcc -DDSO_DLFCN -DHAVE_DLFCN_H -DNDEBUG -DOPENSSL_THREADS -DOPENSSL_NO_STATIC_ENGI
BN_ASM_MONT -DOPENSSL_BN_ASM_MONT5 -DOPENSSL_BN_ASM_GF2m -DSHA1_ASM -DSHA256_ASM -DSHA512_ASM -DRC4_ASM -DM
H_ASM -DECP_NISTZ256_ASM -DPADLOCK_ASM -DPOLY1305_ASM -DOPENSSLDIR="/usr/local/ssl/" -DENGINESSDIR="/us
OpenSSL build timestamp: built on: reproducible build, date unspecified
OpenSSL platform: platform: linux-x86_64
OpenSSL directory: OPENSSLDIR: "/usr/local/ssl"

Usage: paho_c_pub [topicname] [-t topic] [-c connection] [-h host] [-p port]
[-q qos] [-i clientid] [-u username] [-P password] [-k keepalive_timeout]
[-V MQTT-version] [--quiet] [--trace trace-level]
[-r] [-n] [-m message] [-f filename]
[--maxdatalen len] [--message-expiry seconds] [--user-property name value]
[--will-topic topic] [--will-payload message] [--will-qos qos] [--will-retain]
[--cafile filename] [--capath dirname] [--cert filename] [--key filename]
[--keypass string] [--ciphers string] [--insecure]

-t (--topic)      : MQTT topic to publish to
-c (--connection) : connection string, overrides host/port e.g wss://hostname:port/ws. Use this option
                  rather than host/port to connect with TLS and/or web sockets. No default.
-h (--host)      : host to connect to. Default is localhost.
-p (--port)      : network port to connect to. Default is 1883.
-q (--qos)       : MQTT QoS to publish with (0, 1 or 2). Default is 0.
-V (--MQTTversion) : MQTT version (31, 311, or 5). Default is 311.
--quiet          : do not print error messages.
--trace          : print internal trace ("error", "min", "max" or "protocol").
```

发布消息示例如下：

```
./paho_cs_pub -c ssl://127.0.0.1:8883 -q 0 -m "xxx" -t "aaa" --cert /root/
mqtt_cert/xOEMIsYVpw_private_cert.crt --key /root/mqtt_cert/
xOEMIsYVpw_private_cert.key
```

这条命令向内置MQTT Broker名为“aaa”的Topic发送了内容为“xxx”的消息，其中“127.0.0.1:8883”为边缘节点内置MQTT Broker的地址，“/root/mqtt_cert/xOEMIsYVpw_private_cert.crt”和“/root/mqtt_cert/xOEMIsYVpw_private_cert.key”为边缘节点证书。

----结束