Huawei Cloud EulerOS

最佳实践

文档版本	01
发布日期	2025-11-04

华为云计算技术有限公司

商标声明


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

注意

您购买的产品、服务或特性等应受华为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。

华为云计算技术有限公司

地址： 贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址： https://www.huaweicloud.com/

目 录

1 部署 MySQL

简介

MySQL 是一种开源关系型数据库管理系统。与其他关系型数据库一样，MySQL 将数据存储在由行和列组成的表中。用户可以使用结构化查询语言（通常称为 SQL）定义、操作、控制和查询数据。本教程以MYSQL 8.0.42版本为例，介绍如何在HCE上安装部署MySQL。

准备工作

- 准备两台ECS实例，并分配公网IP或者弹性公网IP（EIP）。
- 实例安全组的入方向规则已放通22、3306端口。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 安装MySQL服务端和客户端。

1. 执行以下命令安装MySQL服务端和客户端：
```
dnf install -y mysql-server mysql-common mysql
```
2. 执行以下命令查看MySQL的版本号：
```
mysql -V
```
如果返回如下类似信息，表示安装成功：
```
mysql  Ver 8.0.42 for Linux on x86_64 (Source distribution)
```

步骤2 配置MySQL。

1. 运行以下命令启动MySQL服务：
```
systemctl start mysqld
```
2. 执行以下命令查看服务状态：
```
systemctl status mysqld
```
如果显示active (running)则表示启动成功。

说明

如果要将mysqld设置为开机自启动，则需要执行如下命令。

systemctl enable mysqld

3. 执行以下命令，对MySQL进行安全配置：

```
mysql_secure_installation
```

根据个人需要和提示信息，自定义选项内容：

```
Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: Y // 是否设置密码组件，建议选Y

There are three levels of password validation policy:

LOW    Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 2  // 选择密码安全策略，建议选2
Please set the password for root here.

New password: // 根据上面选择的密码强度设置密码

Re-enter new password:  // 重新输入密码

Estimated strength of the password: 100
Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : y
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y  // 是否删除匿名用户，建议选y
Success.


Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y  // 是否禁止root账户远程登录，建议选y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.


Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y  // 是否删除测试数据库，建议选y
 - Dropping test database...
Success.

 - Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y  // 是否重新加载权限表，建议选y
Success.
```



```
All done!
```

步骤3 远程连接数据库。

1. 执行以下命令连接数据库：
   ```
   mysql -uroot -p
   ```
   然后输入上述为root用户创建的密码。
2. 输入以下语句创建测试账号和密码：
   ```
   create user 'test'@'%' identified by 'xxxxxx';
   ```
   其中'xxxxxx'为对应密码。
3. 输入以下语句为test账号授权数据库所有权限：
   ```
   grant all privileges on *.* to 'test'@'%';
   ```
4. 输入以下语句刷新权限：
   ```
   flush privileges;
   ```
5. 执行以下语句退出数据库：
   ```
   exit
   ```
6. 在另一台服务器上执行以下命令安装MySQL客户端：
   ```
   dnf install -y mysql-common mysql
   ```
7. 另一台服务器后输入以下命令进行远程连接：
   ```
   mysql -h <MySql server IP> -utest -p
   ```
   或者使用Navicat、VSCode插件进行远程连接。

----结束

⚠ 注意

若连接数据库失败可检查防火墙是否关闭或是否放通MySQL端口，默认端口为3306。同时，以上关于密码、权限相关配置仅用于测试，业务环境请谨慎使用。

2 安装 Docker

简介

Docker是一款开源的应用容器引擎，具有可移植性、可扩展性、高安全性和可管理性等优势。开发者可将应用程序和依赖项打包到一个可移植的容器中，能够快速发布到Linux机器上，实现更高效地构建、部署和管理应用程序。本教程介绍如何在HCE服务器中安装Docker。

准备工作

准备一台ECS实例。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

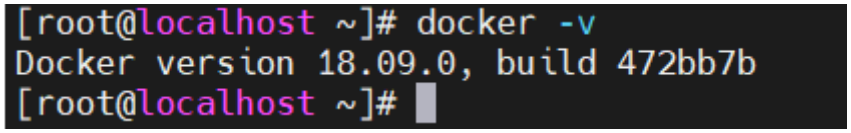
步骤1 安装Docker软件。

1. 执行以下命令安装Docker：
```
dnf install docker
```

步骤2 验证Docker是否安装成功。

1. 执行以下命令启动Docker服务：
```
systemctl start docker
```
2. 执行以下命令后，如果显示Docker版本，则表示Docker安装成功，效果如**图2-1**：
```
docker -v
```

图 2-1 检查 Docker 是否安装成功



----结束

> **⚠ 注意**
>
> 上述方法安装的docker版本为18.09.0，如果需要安装更高版本的Docker，请参考Docker文档进行安装。

Docker 的使用

1. 管理Docker守护进程。
   - 运行守护进程：
     ```
     systemctl start docker
     ```
   - 停止守护进程：
     ```
     systemctl stop docker
     ```
   - 重启守护进程：
     ```
     systemctl restart docker
     ```
2. 管理镜像（以nginx镜像为例，请提前准备好本地镜像文件nginx_latest.tar）。
   - 导入本地镜像：
     ```
     docker load -i <nginx_latest.tar文件路径>
     ```
   - 修改标签：
     ```
     # 将 nginx 镜像的标签从 latest 修改为 v1
     docker tag nginx:latest nginx:v1
     ```
   - 查看已有镜像：
     ```
     docker images
     ```
   - 删除镜像：
     ```
     docker rmi nginx:v1
     ```
3. 管理容器。
   - 启动并进入容器：
     ```
     # 运行以下命令使用指定 ID 的镜像创建容器并进入，其中镜像ID可通过docker images命令获取
     docker run -it <镜像ID> /bin/bash
     ```
   - 退出容器：
     ```
     exit
     ```
   - 以后台方式启动容器：
     ```
     docker run -itd <镜像ID>
     ```
   - 查看所有运行的容器：
     ```
     docker ps
     ```
   - 进入后台运行的容器，其中容器ID可通过docker ps命令获取：
     ```
     docker exec -it <容器ID> /bin/bash
     ```
   - 将容器做成镜像，其中容器ID可通过docker ps命令获取：
     ```
     docker commit <容器ID> [<仓库名>[:<标签>]]
     ```
     例如：
     ```
     docker commit 135****9f757 nginx:v1
     ```

3 搭建 FTP 站点

简介

FTP（File Transfer Protocol，文件传输协议）是一种在计算机之间通过网络传输文件的标准协议。它采用客户端-服务器架构，允许用户从远程服务器上传、下载、重命名、删除、查看文件和目录结构等。FTP有两种传输模式：

- 主动模式（Active）：服务器主动连接客户端的某个端口传输数据。
- 被动模式（Passive）：客户端主动连接服务器开放的端口，适用于防火墙限制较多的网络环境。

FTP支持以下的三种认证模式：

1. 匿名认证
   - 无需用户名和密码，使用anonymous或ftp登录。
   - 通常用于公开资源下载，例如开源软件镜像站。
   - 安全性较低，只能访问受限目录（如 /pub），且通常只读。
2. 本地用户认证
   - 使用Linux系统中的本地用户登录。
   - 通常用于内网资源共享或在开发测试环境中使用。
   - 用户名和密码来源于/etc/passwd和/etc/shadow。权限等同于系统用户，具有一定的操作风险。
3. 虚拟用户认证
   - 用户不是真实的系统用户，独立建立账号数据库。通常设置统一的宿主账户（如vsftpd）进行访问控制。
   - 常结合gdbm或 libdb实现认证。
   - 更安全，权限可精细控制，不依赖系统用户。

vsftpd（very secure FTP daemon）是Linux下的一款小巧轻快、安全易用的FTP服务器软件。本教程介绍如何在HCE上安装并配置vsftpd，主要包括本地用户认证和虚拟用户认证两种方式。

准备工作

- 准备两台ECS实例，并分配公网IP或者弹性公网IP（EIP）。一台作为服务端，一台作为客户端。

- 服务端安全组的入方向规则已放行21端口。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

安装 vsftpd 软件

步骤1 执行如下命令安装vsftpd：

```
dnf install vsftpd
```

步骤2 执行如下命令启动vsftpd服务：

```
systemctl start vsftpd
```

步骤3 执行如下命令查看服务状态：

```
systemctl status vsftpd
```

如果显示**active (running)**则表示启动成功，示例如**图3-1**。

图 3-1 vsftpd 状态示例

```
[root@hce2 ~]# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
     Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; disabled; vendor preset: disabled)
     Active: active (running) since Tue 2023-10-10 15:55:11 CST; 1s ago
    Process: 21033 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=exited, status=0/SUCCESS)
   Main PID: 21034 (vsftpd)
      Tasks: 1 (limit: 198730)
     Memory: 408.0K
     CGroup: /system.slice/vsftpd.service
             └─ 21034 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf
```

说明

如果需要将vsftpd设置为开机自启动，则需要再执行如下命令：

```
systemctl enable vsftpd
```

----结束

配置 FTP 服务器为本地用户认证方式

步骤1 为FTP服务添加用户。

1. 执行如下命令为FTP服务创建一个用户：
   ```
   adduser ftpuser
   ```
2. 执行如下命令为ftp用户设置密码：
   ```
   passwd ftpuser
   ```

步骤2 配置vsftpd服务。

1. 执行如下命令为FTP服务创建目录和文件（可根据个人需要修改目录）：
   ```
   mkdir -p /data/ftp/
   touch /data/ftp/test.txt
   ```
2. 执行如下命令将上述目录的拥有者设置为新用户：
   ```
   chown -R ftpuser:ftpuser /data/ftp/
   ```
3. 打开“/etc/vsftpd/vsftpd.conf”文件，找到如下参数进行配置：
   ```
   # 监听IPv4 sockets。
   listen=YES
   # 根据需要选择是否配置监听IPv6
   listen_ipv6=NO
   ```


在该配置文件的末尾添加下列参数，其中“pasv_address”参数为“准备工作”中的公网IP或者弹性公网IP（EIP）：

```
# 设置本地用户登录后所在目录。
local_root=/data/ftp
#全部用户被限制在主目录。
chroot_local_user=YES
#开启被动模式。
pasv_enable=YES
pasv_address=<FTP服务器公网IP地址>
# 是否允许用户访问其他目录
chroot_list_enable=NO
allow_writeable_chroot=YES
# 如果chroot_list_enable配置为YES，则需要配置该选项；该配置为一个文件，包含哪些用户可以访问其他目录
# chroot_list_file=/etc/vsftpd/chroot_list
# 设置被动模式下，可使用的端口范围，建议把端口范围设置在一段比较高的范围内，有助于提高访问FTP服务器的安全性。
# 可使用的端口范围的最小值，此例中是5000，请根据实际情况修改。
pasv_min_port=5000
# 可使用的端口范围的最大值，此例中是5010，请根据实际情况修改。
pasv_max_port=5010
```

说明

除上述提及的参数，其他参数保持默认值即可。

注意

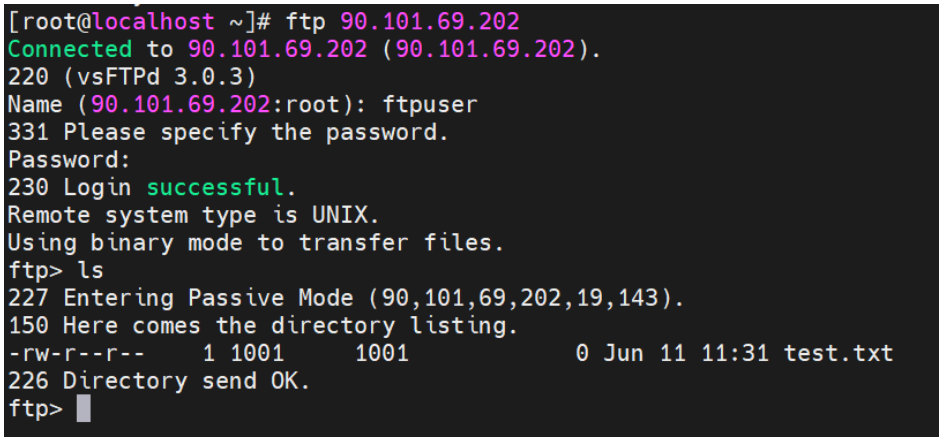
无论是否配置chroot_list_enable，都需要创建“/etc/vsftpd/chroot_list”文件。

4. 执行如下命令重启vsftpd服务：
```
systemctl restart vsftpd
```

步骤3 验证。

1. 执行如下命令查看vsftpd服务所使用的端口：
```
netstat -natp | grep vsftpd
```
默认情况下所使用的端口是21。
2. 在安全组界面，入方向放开21端口，以及配置文件/etc/vsftpd/vsftpd.conf中参数pasv_min_port和pasv_max_port之间的所有端口，此处示例为5000~5010端口。
3. 在客户端执行以下命令安装ftp客户端：
```
dnf install -y ftp
```
4. 在客户端执行以下命令：
```
ftp <FTP服务器公网IP地址>
```
5. 按照提示输入账号和密码。
6. 登录成功后，执行“ls”命令后可以看到test.txt文件表示配置成功。

图 3-2 验证示例



----结束

配置 FTP 服务器为虚拟用户认证方式

步骤1 关闭SELinux。

```
setenforce 0
```

步骤2 创建虚拟用户账号并禁用shell登录：

```
useradd vsftpd -s /bin/false
```

这是虚拟用户的映射账号，用于访问系统资源，但不能登录系统。

步骤3 创建GDBM数据库：

```
gdbmtool /etc/vsftpd/login.pag store ftpuser 123456
```

- ftpuser：虚拟用户名示例。
- 123456：示例密码。
- login.pag：账户密码存储文件。

步骤4 配置 PAM 模块，编辑/etc/pam.d/vsftpd文件，删除已有内容并写入以下内容：

```
auth    required    pam_userdb.so db=/etc/vsftpd/login
account required    pam_userdb.so db=/etc/vsftpd/login
```

说明

db=/etc/vsftpd/login实际指的是login.pag，无需扩展名。

步骤5 设置vsftpd配置文件，编辑/etc/vsftpd/vsftpd.conf文件，添加以下内容：

```
guest_enable=YES
guest_username=vsftpd
user_config_dir=/etc/vsftpd/user_conf
#开启被动模式。
pasv_enable=YES
pasv_address=<FTP服务器公网IP地址>
# 是否允许用户访问其他目录
chroot_list_enable=NO
allow_writeable_chroot=YES
# 设置被动模式下，可使用的端口范围，建议把端口范围设置在一段比较高的范围内，有助于提高访问FTP服务器的安全性。
# 可使用的端口范围的最小值，此例中是5000，请根据实际情况修改。
pasv_min_port=5000
# 可使用的端口范围的最大值，此例中是5010，请根据实际情况修改。
pasv_max_port=5010
```

步骤6 配置虚拟用户访问权限。

1. 创建用户权限配置目录：
```
mkdir -p /etc/vsftpd/user_conf
```
2. 在/etc/vsftpd/user_conf/ftpuser文件写入以下内容：
```
# 允许写操作（上传、改名、删除等）
write_enable=YES
# 允许本地系统用户（包括虚拟用户）登录 FTP 服务器
local_enable=YES
# 本地用户创建文件/目录时使用的权限掩码
local_umask=077
# 匿名用户创建文件/目录时的权限掩码
anon_umask=022
# 设置虚拟用户或本地用户登录后的根目录，限制用户只能访问该目录及其子目录
local_root=/home/vsftpd/ftp
# 允许匿名用户上传文件
anon_upload_enable=YES
# 允许匿名用户创建目录
anon_mkdir_write_enable=YES
# 允许匿名用户执行其他写操作，如删除或改名文件等
anon_other_write_enable=yes
```

说明

在vsftp中，虚拟用户虽然通过本地用户映射实现，但其权限控制沿用了匿名用户的配置方式，因此需依赖匿名用户相关选项来启用写操作。

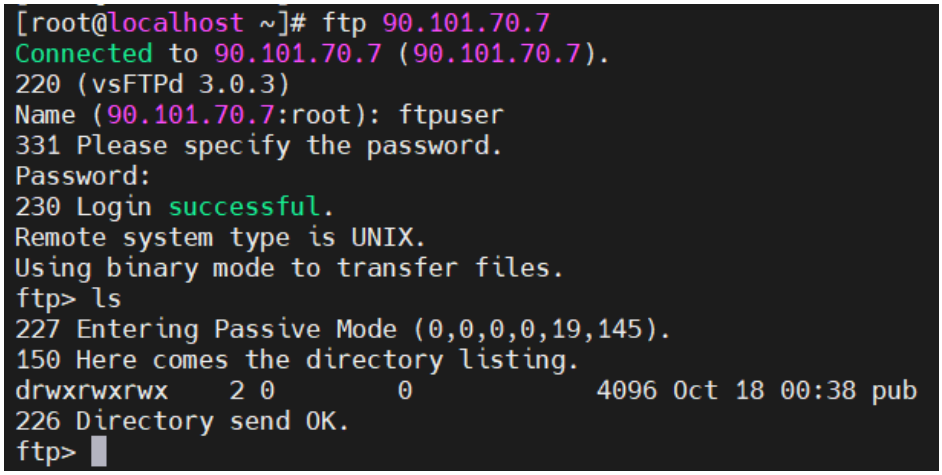
步骤7 准备FTP根目录并赋权：
```
mkdir -p /home/vsftpd/ftp/pub
chmod 777 /home/vsftpd/ftp/pub
```

步骤8 重启vsftpd服务：
```
systemctl restart vsftpd
```

步骤9 验证。

1. 执行如下命令查看vsftpd服务所使用的端口：
```
netstat -natp | grep vsftpd
```
默认情况下所使用的端口是21。
2. 在安全组界面，入方向放开21端口和5000~5010端口。
3. 在客户端执行以下命令安装ftp客户端：
```
dnf install -y ftp
```
4. 在客户端执行以下命令：
```
ftp <FTP服务器公网IP地址>
```
5. 按照提示输入账号和密码。
6. 登录成功后，执行“ls”命令后可以看到pub文件夹表示配置成功。

图 3-3 验证示例



----结束

4 搭建 SFTP

简介

SFTP（Secure File Transfer Protocol）是一种用于安全访问、传输和管理大文件和敏感数据的网络协议，它通过SSH协议运行，能够支持SSH完整的安全和身份验证功能。本教程介绍如何在HCE上部署SFTP服务。

准备工作

- 准备一台ECS实例，并分配公网IP或者弹性公网IP（EIP）。
- 安全组的入方向规则已放行22端口。

操作步骤

1. 执行如下命令创建sftp组：
   ```
   groupadd sftp
   ```
2. 执行如下命令，创建一个用户，用户登录SFTP服务：
   ```
   useradd -g sftp -s /bin/false usftp
   ```
3. 执行如下命令为上述新用户设置密码：
   ```
   passwd usftp
   ```
4. 执行如下命令为sftp用户组创建home目录：
   ```
   mkdir -p /data/sftp/usftp
   ```
5. 执行如下命令修改usftp用户登录时的目录：
   ```
   usermod -d /data/sftp/usftp usftp
   ```
6. 打开/etc/ssh/sshd_config文件，找到以下参数进行配置：
   ```
   # 找到Subsystem配置项替换为以下配置
   Subsystem sftp internal-sftp

   # 在文件末尾添加如下配置：
   Match user usftp  # 匹配usftp用户
   AllowTcpForwarding no # 不允许转发TCP协议
   X11Forwarding no # 不允许进行 X11 转发
   # 使用chroot将用户根目录指定到/data/sftp/%u，%u代表用户名
   ChrootDirectory /data/sftp/usftp
   ForceCommand    internal-sftp # 强制执行内部sftp
   ```
   配置完成后执行以下命令重启sshd服务：
   ```
   systemctl restart sshd
   ```
7. 执行如下命令，创建测试文件：
   ```
   touch /data/sftp/usftp/test.txt
   ```


8. 依次执行如下命令设置目录权限：
```
chown -R root:sftp /data/sftp/usftp
chmod 755 /data/sftp/usftp
```
9. 在另外一台机器上，输入以下命令连接上述SFTP服务：
```
sftp usftp@<SFTP服务公网IP>
```
10. 输入密码后执行**ls**命令查看测试文件：
```
sftp> ls
test.txt
```

5 PostgreSQL 部署

简介

PostgreSQL是一个开源的、高度稳定的数据库系统，支持多种SQL功能，包括外键、子查询、触发器以及用户自定义的数据类型和函数。它进一步增强了SQL语言，提供了一些细致地扩展数据工作负载的功能。它主要用于移动、网络、地理空间和分析应用程序领域，被业界誉为“最先进的开源数据库”。本教程介绍如何在HCE上安装部署PostgreSQL。

准备工作

- 准备两台ECS实例，并分配公网IP或者弹性公网IP（EIP）。一台作为主节点，另一台作为从节点。
- 安全组入方向都已放开5432端口。

须知

5432端口属于高危端口，建议使用前请做好充分的安全评估。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 配置PostgreSQL主节点。

1. 执行如下命令安装服务端、客户端和相关组件：
```
dnf install -y postgresql postgresql-contrib postgresql-server
```
2. 执行如下命令初始化数据库：
```
postgresql-setup --initdb --unit postgresql
```
3. 依次执行如下命令启动服务并查看服务状态：
```
systemctl start postgresql
systemctl status postgresql
```
如果显示**active (running)**则表示服务启动成功。

说明

如果需要将PostgreSQL服务设置为开机自启动，则需要执行以下命令。

```
systemctl enable postgresql
```

4. 执行以下命令登录postgres账号：
```
su - postgres
```
5. 执行以下语句进入psql终端：
```
psql
```
6. 输入以下语句创建账号，并配置密码和权限：
```
CREATE ROLE replica login replication ENCRYPTED PASSWORD 'xxxxxx';
```
其中账号为**replica**，权限**(login和replication)**为登录和备份权限，密码为**xxxxxx**。
7. 输入以下语句查看创建的账号：
```
SELECT usename FROM pg_user;
```
如果返回结果如图5-1所示，则表示账户创建成功。

图 5-1 账户创建成功

```
postgres=# SELECT usename FROM pg_user;
 usename
----------
 postgres
 replica
(2 rows)
```

8. 输入以下语句查看创建的权限：
```
SELECT rolname FROM pg_roles;
```
如果返回结果如图5-2所示，则表示权限创建成功。

图 5-2 权限创建成功

```
postgres=# SELECT rolname FROM pg_roles;
          rolname
---------------------------
 pg_monitor
 pg_read_all_settings
 pg_read_all_stats
 pg_stat_scan_tables
 pg_read_server_files
 pg_write_server_files
 pg_execute_server_program
 pg_signal_backend
 postgres
 replica
(10 rows)
```

9. 执行以下语句退出psql终端：
```
\q
```
10. 执行以下语句退出postgres用户：
```
exit
```
11. 编辑“/var/lib/pgsql/data/pg_hba.conf”文件配置replica用户白名单：

    找到IPv4 local connections部分，添加以下信息：
```
host    all             all             <从节点的IPv4网段>          md5
host    replication     replica         <从节点的IPv4网段>          md5
```

    图 5-3 添加信息

    将自己部署的从节点IPv4网段信息填入上述对应位置。
12. 编辑“/var/lib/pgsql/data/postgresql.conf”文件，找到如下参数，进行配置：
```
listen_addresses = '*'  #监听的内网 IP 地址
max_connections = 100   #最大连接数，从库的 max_connections 必须要大于主库的
wal_level = hot_standby  #启用热备模式
synchronous_commit = on  #开启同步复制
max_wal_senders = 32    #同步最大的进程数量
wal_sender_timeout = 60s #流复制主机发送数据的超时时间
```
13. 最后执行以下命令重启服务：
```
systemctl restart postgresql
```

步骤2 配置PostgreSQL从节点。

1. 执行以下命令在从节点安装服务端、客户端和相关组件：
```
dnf install -y postgresql postgresql-contrib postgresql-server
```
2. 执行以下命令在从节点创建备份目录：
```
pg_basebackup -D /var/lib/pgsql/data -h <主库IP> -p 5432 -U replica -X stream -P -R
```
    具体参数含义如下：

    -D：指定备份目录。

    -h：指定主库的 IP 地址或主机名。

    -p：指定主库的端口号。

    -U：指定用于连接数据库的用户名。

    -X：指定WAL文件复制方式，此处stream表示流复制。

    -P：显示进度信息。

    -**R**：表示自动写replication的配置信息。

    如果显示如图5-4所示，则表示备份成功。

    图 5-4 备份成功

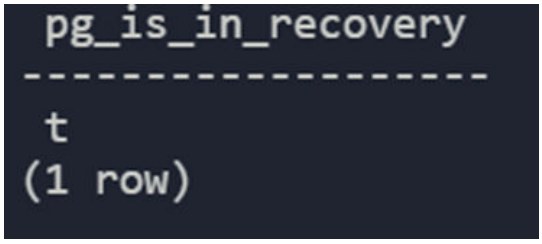
    此时，会在“/var/lib/pgsql/data”目录生成standby.signal文件，并在postgresql.auto.conf文件自动写入主库连接信息。
3. 执行以下命令修改目录所属的用户组：
```
chown -R postgres.postgres /var/lib/pgsql/data
```

4. 最后执行以下命令重启服务：
```
systemctl restart postgresql
```

步骤3 验证配置是否成功。

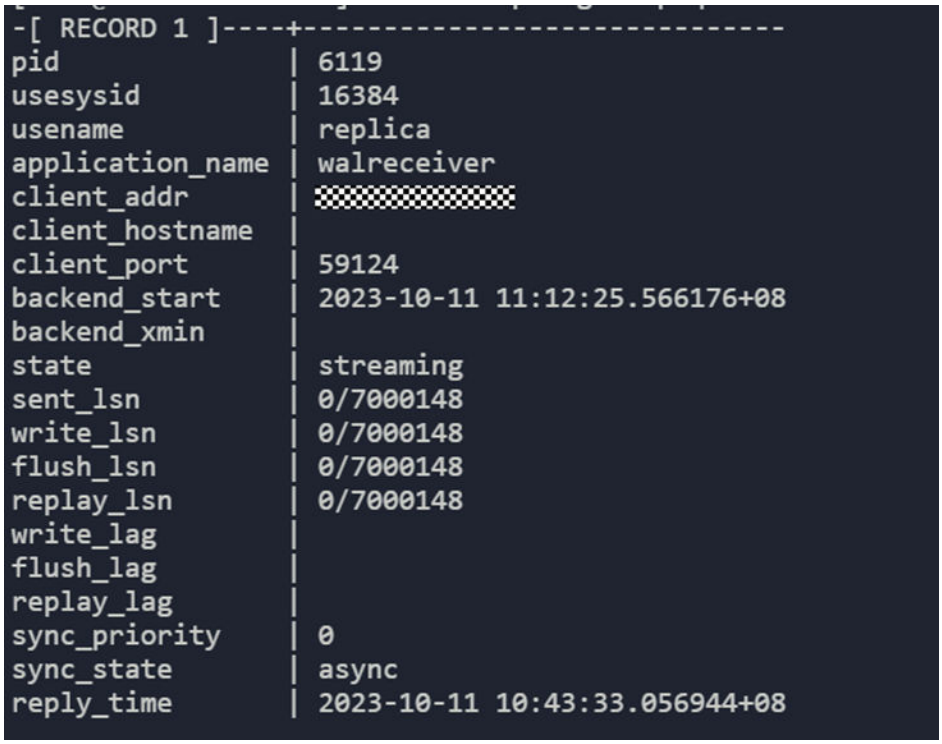
1. 在从节点执行以下命令验证从节点是否配置成功：
```
sudo -u postgres psql -c "SELECT pg_is_in_recovery()"
```
如果显示如图5-5所示，表示从节点配置成功。

图 5-5 从节点配置成功



2. 在主节点执行如下命令获取从节点信息：
```
sudo -u postgres psql -x -c "SELECT * FROM pg_stat_replication" -d postgres
```
如果显示如图5-6所示，则表示配置成功。

图 5-6 配置成功



----结束

注意

以上密码、权限相关配置仅用于测试，业务环境请谨慎使用。

6 Redis 一主多从部署

简介

Redis（Remote Dictionary Server）是一个使用C语言编写的开源、基于内存、分布式、可选持久性的键值对存储数据库。Redis 是一个功能丰富的存储系统，适用于多种场景，包括缓存、会话存储、排行榜、实时分析等。它有广泛的应用，并且拥有活跃的社区支持。本教程介绍如何在HCE上安装部署Redis。

准备工作

- 准备两台ECS，并分配公网IP或者弹性公网IP（EIP）；其中一台作为Redis 主节点，一台作为从节点。
- 安全组入方向都已放开6379端口。

须知

6379端口属于高危端口，建议使用前请做好充分的安全评估。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 安装Redis软件及配置。

1. 在两台ECS上分别执行如下命令安装Redis：
```
dnf install redis
```
2. 在两台ECS上分别执行如下命令启动服务：
```
systemctl start redis
```

说明

如果需要将Redis服务设置为开机自启动，则需要执行如下命令：
```
systemctl enable redis
```

3. 执行如下命令查看服务状态：
```
systemctl status redis
```


如果显示**active (running)**，则表示服务启动成功。

4. 在主节点，编辑“/etc/redis.conf”文件，找到如下属性进行配置：
```
bind <主节点IP>     # 填写主节点IP
requirepass ******* # 配置密码
```
5. 在从节点，编辑“/etc/redis.conf”文件，找到如下属性进行配置：
```
bind <从节点IP>                  # 填写从节点IP
requirepass *******              # 配置密码
slaveof <主节点IP> <主节点port>
masterauth <主节点密码>
```
6. 在两台ECS上分别执行如下命令重启服务：
```
systemctl restart redis
```

步骤2 验证。

1. 在主节点输入以下命令连接Redis：
```
redis-cli -h <主节点IP>
auth <密码>
```
2. 然后输入以下命令查看节点信息，示例如下：
```
127.0.0.1:6379> info replication
# Replication
role:master
connected_slaves:1
slave0:ip=x.x.x.x,port=6379,state=online,offset=4382,lag=0
master_replid:5d68ccf7722f461cc5f004c7e96fd7c506990508
master_replid2:0000000000000000000000000000000000000000
master_repl_offset:4382
second_repl_offset:-1
repl_backlog_active:1
repl_backlog_size:1048576
repl_backlog_first_byte_offset:1
repl_backlog_histlen:4382
127.0.0.1:6379>
```

----结束

⚠ 注意

以上密码等配置以及Redis部署架构仅用于测试演示，业务环境请谨慎使用。

7 Tomcat 安装

简介

Tomcat是一个免费的，开放源代码的Web应用服务器，是Apache软件基金会项目中的一个核心项目，是一款比较流行的web应用服务器。本教程介绍如何在HCE上安装部署Tomcat。

准备工作

- 准备一台ECS，并分配公网IP或者弹性公网IP（EIP）。
- 安全组入方向已放开8080端口。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 软件安装。

1. 执行以下命令安装Java：
   ```
   dnf install java-1.8.0-openjdk
   ```
2. 执行如下命令验证是否安装成功：
   ```
   java -version
   ```
3. 执行如下命令安装Tomcat：
   ```
   dnf install tomcat
   ```
   Tomcat会被安装到“/usr/share/tomcat”目录。

步骤2 配置软件。

1. 在“/usr/lib/jvm/”目录下查找java路径，如在该目录下存在文件“java-1.8.0-openjdk-xxx”（其中“xxx”为具体版本号），则可以得到java的jre路径为“/usr/lib/jvm/java-1.8.0-openjdk-xxx/jre”。
2. 编辑“/etc/profile”配置环境变量，新增如下内容，其中“JAVA_HOME”参数需要根据步骤**步骤2.1**找到的jre路径进行填充：
   ```
   JAVA_HOME={jre路径}
   PATH=$PATH:$JAVA_HOME/bin
   CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
   export JAVA_HOME CLASSPATH PATH
   ```


3. 执行以下命令激活上述环境变量：
```
source /etc/profile
```
4. 清空“/usr/share/tomcat/conf/server.xml”文件，粘贴以下内容：
```
<?xml version="1.0" encoding="UTF-8"?> <Server port="8006" shutdown="SHUTDOWN"> <Listener
className="org.apache.catalina.core.JreMemoryLeakPreventionListener"/> <Listener
className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener"/> <Listener
className="org.apache.catalina.core.ThreadLocalLeakPreventionListener"/> <Listener
className="org.apache.catalina.core.AprLifecycleListener"/> <GlobalNamingResources> <Resource
name="UserDatabase" auth="Container"  type="org.apache.catalina.UserDatabase"
description="User database that can be updated and saved"
factory="org.apache.catalina.users.MemoryUserDatabaseFactory"  pathname="conf/tomcat-
users.xml"/> </GlobalNamingResources> <Service name="Catalina"> <Connector port="8080"
protocol="HTTP/1.1"  connectionTimeout="20000"  redirectPort="8443"  maxThreads="1000"
minSpareThreads="20"  acceptCount="1000"  maxHttpHeaderSize="65536"  debug="0"
disableUploadTimeout="true"  useBodyEncodingForURI="true"  enableLookups="false"
URIEncoding="UTF-8"/> <Engine name="Catalina" defaultHost="localhost"> <Realm
className="org.apache.catalina.realm.LockOutRealm"> <Realm
className="org.apache.catalina.realm.UserDatabaseRealm"   resourceName="UserDatabase"/> </
Realm> <Host name="localhost" appBase="/data/wwwroot/default" unpackWARs="true"
autoDeploy="true"> <Context path="" docBase="/data/wwwroot/default" debug="0"
reloadable="false" crossContext="true"/> <Valve
className="org.apache.catalina.valves.AccessLogValve" directory="logs"
prefix="localhost_access_log." suffix=".txt" pattern="%h %l %u %t %s %b" /> </Host> </Engine> </
Service> </Server>
```
保存并退出，可根据需要，自定义上述配置。
5. 执行以下命令创建上述配置中appbase和docbase填入的目录：
```
mkdir -p /data/wwwroot/default
```
6. 然后执行以下命令将上述目录所属用户设置为tomcat：
```
chown -R tomcat.tomcat /data/wwwroot/
```
7. 新建“/usr/share/tomcat/bin/setenv.sh”文件， 输入以下内容配置JVM内存参数：
```
JAVA_OPTS='-Djava.security.egd=file:/dev/./urandom -server -Xms256m -Xmx496m -
Dfile.encoding=UTF-8'
```
8. 执行以下命令启动Tomcat服务：
```
systemctl start tomcat
```
9. 执行如下命令查看Tomcat服务状态：
```
systemctl status tomcat
```
如果显示**active (running)**则表示服务启动成功。

步骤3 验证Tomcat安装是否成功。

1. 执行以下命令，新增测试页面：
```
echo Tomcat test > /data/wwwroot/default/index.jsp
```
2. 然后在浏览器输入http://<Tomcat服务公网IP>:8080进行访问，如果显示如图7-1则表示安装成功：

图 7-1 安装成功

----结束

⚠ 注意

以上相关配置仅用于测试，业务环境请谨慎使用。

8 Kafka 安装

简介

Kafka是一个拥有高吞吐、可持久化、可水平扩展、支持流式数据处理等多种特性的分布式消息流处理中间件，采用分布式消息发布与订阅机制，在日志收集、流式数据传输、在线/离线系统分析、实时监控等领域有广泛的应用。本教程介绍如何在HCE上安装部署Kafka。

准备工作

- 准备一台ECS，并分配公网IP或者弹性公网IP（EIP）。
- 安全组入方向已放开9092端口。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 安装kafka软件。

执行以下命令安装软件：

```
dnf install kafka
```

执行结束后，Kafka会被安装到/opt/kafka目录。

步骤2 软件配置。

1. 编辑“/opt/kafka/config/server.properties”文件，找到如下属性进行修改：
   ```
   listeners=PLAINTEXT://<内网IP>:9092
   advertised.listeners=PLAINTEXT://<公网IP>:9092
   ```
2. 新建“/lib/systemd/system/zookeeper.service”文件，并输入以下内容：
   ```
   [Unit]
   Description=Zookeeper service
   After=network.target

   [Service]
   Type=simple
   Environment="PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
   User=root
   Group=root
   ExecStart=/opt/kafka/bin/zookeeper-server-start.sh /opt/kafka/config/zookeeper.properties
   ```



```
ExecStop=/opt/kafka/bin/zookeeper-server-stop.sh
Restart=on-failure
SuccessExitStatus=143

[Install]
WantedBy=multi-user.target
```

3. 新建“/lib/systemd/system/kafka.service”文件，并输入以下内容：

```
[Unit]
Description=Apache Kafka server (broker)
After=network.target  zookeeper.service

[Service]
Type=simple
Environment="PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
User=root
Group=root
ExecStart=/opt/kafka/bin/kafka-server-start.sh /opt/kafka/config/server.properties
ExecStop=/opt/kafka/bin/kafka-server-stop.sh
Restart=on-failure
SuccessExitStatus=143

[Install]
WantedBy=multi-user.target
```

4. 依次输入以下命令，启动Kafka和Zookeeper服务：

```
systemctl daemon-reload
systemctl start zookeeper
systemctl start kafka
```

步骤3 验证。

1. 使用以下命令创建Topic：

```
 /opt/kafka/bin/kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic test
```

2. 使用以下命令查看刚才创建的Topic：

```
/opt/kafka/bin/kafka-topics.sh --list --zookeeper localhost:2181
```

如果部署正常，则会输出**test**。

----结束

⚠ 注意

以上相关配置仅用于测试，业务环境请谨慎使用。

9 Gnome 桌面安装

简介

GNOME 是一种由自由开源软件组成的图形桌面环境，它为Linux 和其他类Unix 操作系统提供一个功能完善、操作简单、界面友好的用户界面。本教程介绍如何在HCE上安装Gnome图形化界面，以及中文支持。

准备工作

准备一台ECS主机。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 安装Gnome桌面。

执行如下命令安装Gnome桌面：

```
dnf install gnome-initial-setup gnome-terminal
```

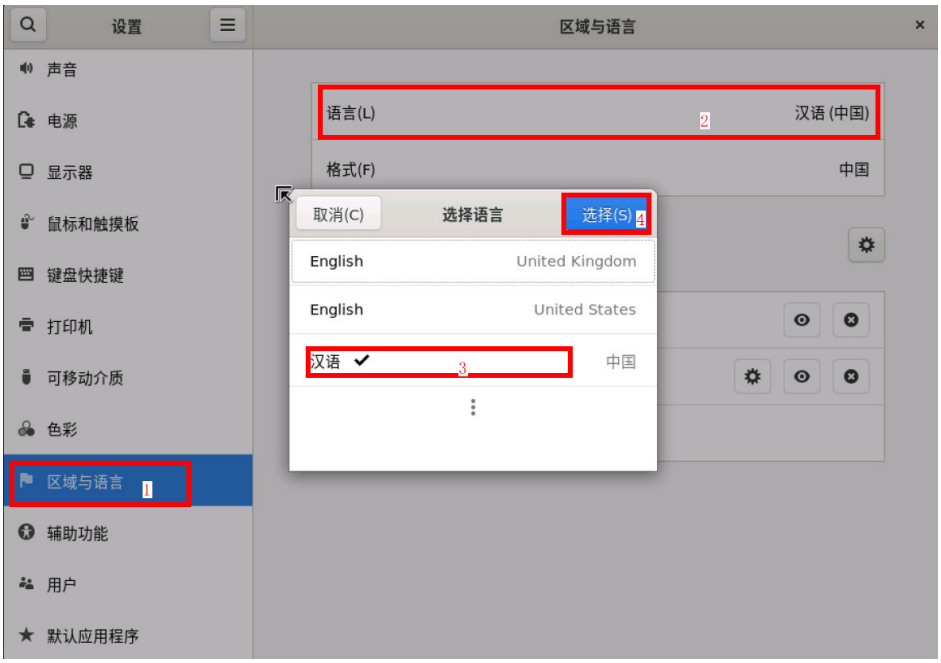
步骤2 设置为默认图形化桌面启动。

1. 执行以下命令将图形化界面设置为默认启动：
   ```
   systemctl set-default graphical.target
   ```
2. 之后执行如下命令重启：
   ```
   reboot
   ```

步骤3 设置中文字体（可选）。

1. 如果需要设置中文字体，则需要先打开终端执行以下命令，安装依赖包：
   ```
   sudo dnf install google-noto-sans-cjk-ttc-fonts google-noto-serif-cjk-ttc-fonts
   ```
2. 然后编辑“/etc/profile”文件，在文件末尾添加以下内容：
   ```
   export LC_ALL="zh_CN.UTF-8"
   ```
3. 重启机器，待重启之后，找到“设置 > 区域和语言”，单击“语言”，选择“汉语”，步骤如下图：

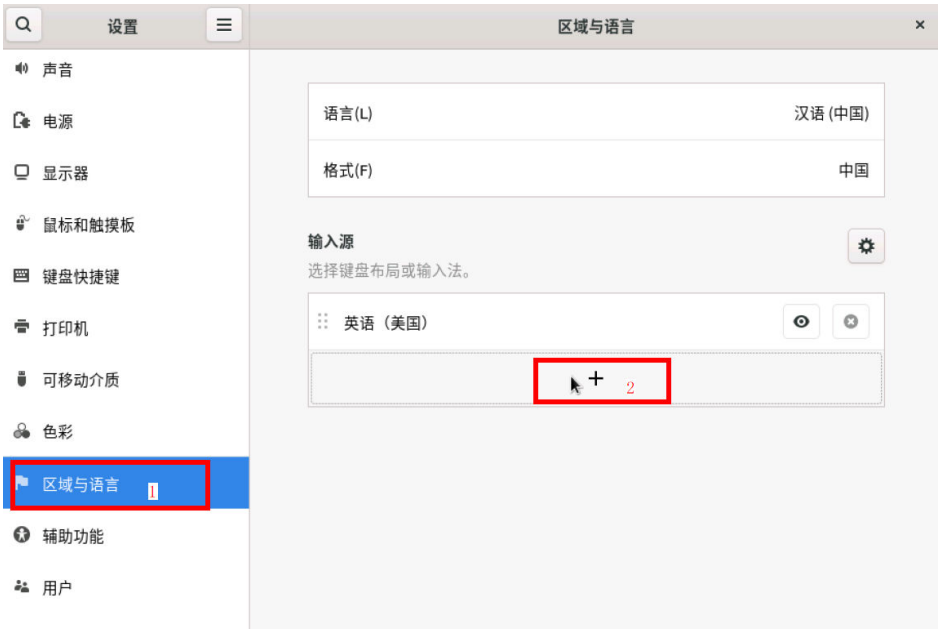
图 9-1 设置中文字体



步骤4 安装中文输入法（可选）。

1. 如果需要安装中文输入法，则需要先执行以下命令安装依赖包：
   ```
   sudo dnf install ibus ibus-libpinyin
   ```
2. 重启机器，待重启之后，找到设置 > 区域和语言 > 输入源，单击加号按钮，步骤如下图：

图 9-2 新增输入源

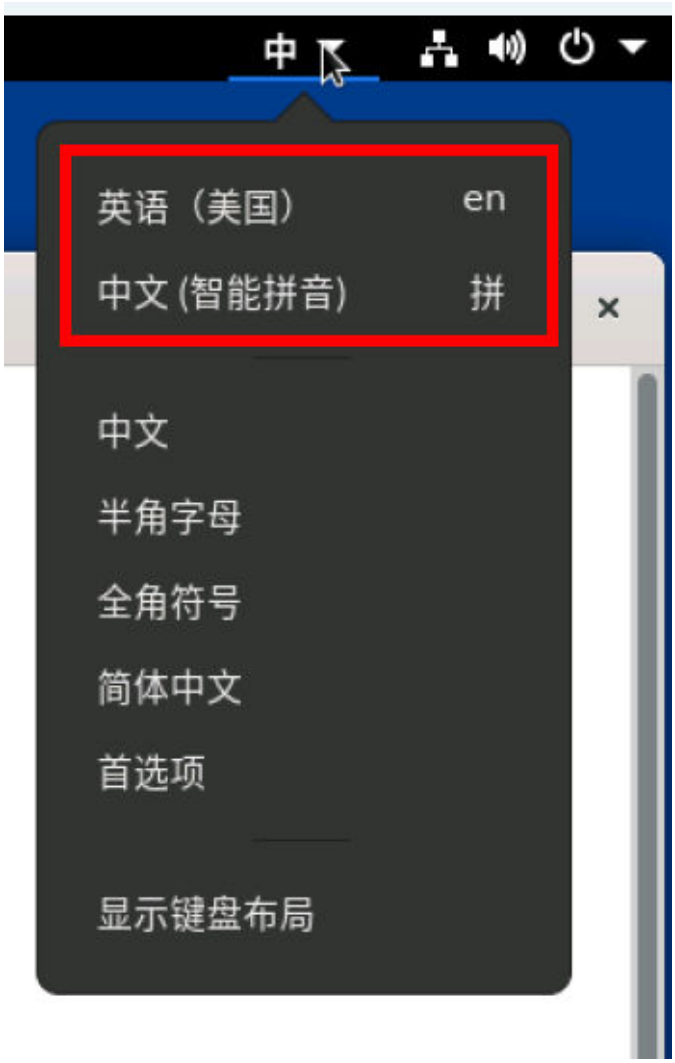


说明

HCE 3.0的位置有变化，设置 > 键盘 > 输入源

3. 最后依次选择“汉语”->“中文（智能拼音）”，最后单击“添加”按钮；添加之后可在状态栏进行输入法切换：

图 9-3 中文输入法安装成功



----结束

10 Apache 安装

简介

Apache HTTP Server（简称Apache）是Apache软件基金会的一个开源网页服务器，可以在大多数操作系统中运行，由于其能跨平台、安全性高而被广泛使用，是最流行的Web服务器软件之一，其拥有的特性包括支持FastCGI、支持SSL、集成Perl处理模块等。本教程介绍如何在HCE中部署Apache。

准备工作

- 准备一台ECS，并分配公网IP或者弹性公网IP（EIP）。
- 安全组入方向已放开80端口。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 安装Apache软件。

1. 执行如下命令安装Apache服务：
   ```
   dnf install httpd httpd-devel
   ```
   执行完后，Apache服务配置文件路径为：“/etc/httpd/conf/httpd.conf”。
2. 执行以下命令查看Apache版本号：
   ```
   httpd -v
   ```
   输出内容示例如下：
   ```
   [root@localhost system]# httpd -v
   Server version: Apache/2.4.51 (Unix)
   Server built:   Feb  9 2022 09:00:41
   ```
3. 执行如下命令启动Apache服务：
   ```
   systemctl start httpd
   ```
   如果输出显示**active (running)**，则表示启动成功。

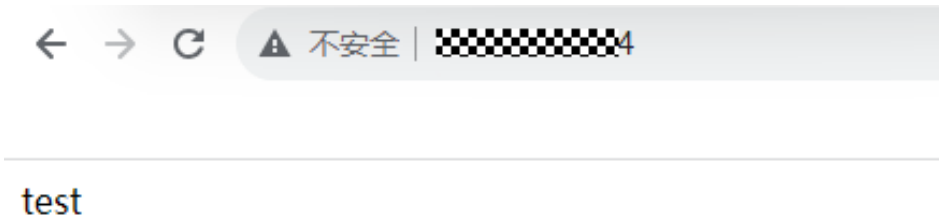
   说明

   如果要将Apache服务设置为开机自启动，则需要依次执行以下命令：
   ```
   systemctl enable httpd
   systemctl daemon-reload
   ```

步骤2 验证。

1. 执行以下命令创建测试页面：
```
echo test > /var/www/html/index.html
```
2. 在浏览器中输入http://<公网IP>，如果显示如图10-1，则表示部署成功：

图 10-1 部署成功



----结束

11 qperf 编译支持 IPV6

简介

目前HCE 2.0中的qperf版本为0.4.9，不支持IPV6，如果要使用IPV6，需要从社区获取0.4.11及以上版本。

本教程介绍如何在HCE 2.0上下载qperf 0.4.11源码并编译。

准备工作

准备一台ECS实例，并分配公网IP或者弹性公网IP（ EIP ）。

前置条件

已完成本机yum源的配置，可参考**HCE的REPO源配置与软件安装**进行yum源的配置。

操作步骤

步骤1 执行以下命令下载qperf 0.4.11源码文件：

```
wget https://github.com/linux-rdma/qperf/archive/refs/tags/v0.4.11.tar.gz
```

步骤2 执行以下命令解压下载文件：

```
tar -xf qperf-0.4.11.tar.gz
```

解压后得到目录**qperf-0.4.11**。

步骤3 执行以下命令安装依赖：

```
dnf install gcc make automake
```

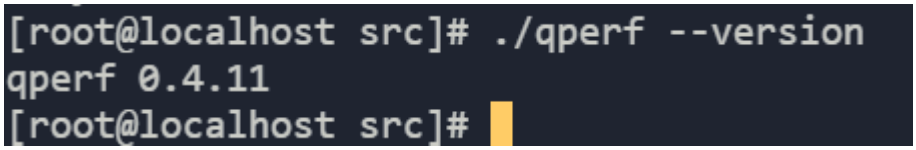
步骤4 进入qperf-0.4.11目录，依次执行以下命令进行编译：

```
./cleanup
./autogen.sh
./configure
make
```

步骤5 验证编译结果：

编译得到的二进制在**qperf-0.4.11/src**目录下，进入该目录后执行**./qperf --version**，得到类似如下回显，表示运行成功：

图 11-1 运行成功



说明

HCE3.0中qperf为0.4.11，不涉及上述问题

步骤6 （可选）移动qperf到/usr/bin目录下，并验证qperf命令。

在**qperf-0.4.11/src**目录下执行以下命令：

```
cp qperf /usr/bin/
chmod 755 /usr/bin/qperf
qperf --version
```

----结束

12 网卡中断绑核

简介

网卡中断绑核是一种优化网络性能的技术，它可以将网卡的数据处理任务分配到特定的CPU核心上，从而提高网络数据的处理效率和吞吐量。本文介绍如何在HCE上配置网卡中断绑核。

准备工作

准备一台ECS实例，并分配公网IP或者弹性公网IP（EIP）。

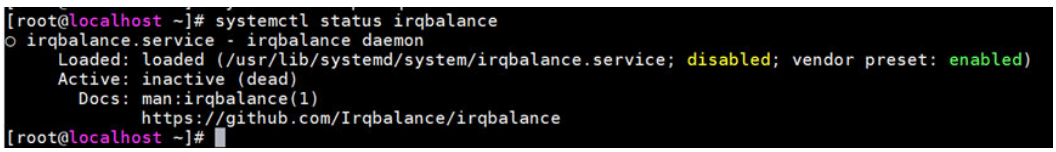
前置条件

请确认irqbalance中断均衡服务已关闭，具体操作步骤如下：

步骤1 执行以下命令查看irqbalance中断均衡服务是否关闭：

```
systemctl status irqbalance
```

如果显示Active: inactive，如下图所示，则表示服务关闭：



步骤2 如果显示Active: active，则表示服务处于运行状态，需要执行以下命令关闭irqbalance中断均衡服务：

```
systemctl stop irqbalance
```

步骤3 （可选）如果需要禁用开机启动，则执行以下命令永久关闭irqbalance中断均衡服务：

```
systemctl disable irqbalance
```

----结束

操作步骤

步骤1 执行以下命令查看网卡队列数，本例中有两个网卡队列，每个队列分别对应一个输入和输出中断：

```
ethtool -l eth0
```

说明

eth0为网卡名，请根据实际情况进行替换，以下步骤均以eth0举例。

```
[root@localhost ~]# ethtool -l eth0
Channel parameters for eth0:
Pre-set maximums:
RX:             n/a
TX:             n/a
Other:          n/a
Combined:       2
Current hardware settings:
RX:             n/a
TX:             n/a
Other:          n/a
Combined:       2
[root@localhost ~]#
```

步骤2 执行lscpu查看当前CPU个数，本例有4个CPU核心：

```
[root@localhost ~]# lscpu
Architecture:             x86_64
  CPU op-mode(s):         32-bit, 64-bit
  Address sizes:          46 bits physical, 48 bits virtual
  Byte Order:             Little Endian
CPU(s):                   4
  On-line CPU(s) list:    0-3
Vendor ID:                GenuineIntel
```

步骤3 执行以下命令查看当前网卡队列对应中断：

```
cat /proc/interrupts | grep virtio0 | awk '{print $1 $(NF)}'
```

本例中eth0的数据输入使用的中断号分别是25和27，数据输出使用的中断号分别是26和28：

```
[root@localhost ~]# cat /proc/interrupts | grep virtio0 | awk '{print $1 $(NF)}'
24:virtio0-config
25:virtio0-input.0
26:virtio0-output.0
27:virtio0-input.1
28:virtio0-output.1
[root@localhost ~]#
```

说明

网卡和virtio的对应关系可以通过以下命令进行查看：

```
ethtool -i eth0 | grep bus-info | awk -F "bus-info:" '{print $2}' | xargs -I {} ls /sys/bus/pci/drivers/virtio-pci/{} | grep virtio
```

```
[root@localhost ~]# ethtool -i eth0 | grep bus-info | awk -F "bus-info:" '{print $2}' | xargs -I {} ls /sys/bus/pci/drivers/virtio-pci/{} | grep virtio
virtio0
[root@localhost ~]#
```

步骤4 执行以下命令查看当前网卡中断绑定情况：

```
cat /proc/irq/{25,26,27,28}/smp_affinity_list
```

本例中25,26,27,28号中断均绑定在CPU3上：

```
[root@localhost ~]# cat /proc/irq/{25,26,27,28}/smp_affinity_list
3
3
3
3
[root@localhost ~]#
```

步骤5 手动绑定网卡中断：

执行以下命令将eth0的数据输入中断分别绑定到CPU0和CPU1：

```
echo 0 > /proc/irq/25/smp_affinity_list
echo 1 > /proc/irq/27/smp_affinity_list
```

执行以下命令将eth0的数据输出中断分别绑定到CPU2和CPU3：

```
echo 2 > /proc/irq/26/smp_affinity_list
echo 3 > /proc/irq/28/smp_affinity_list
```

步骤6 验证网卡中断绑定是否成功：

执行以下命令查看网卡中断绑定情况：

```
cat /proc/irq/{25,26,27,28}/smp_affinity_list
```

```
[root@localhost ~]# cat /proc/irq/{25,26,27,28}/smp_affinity_list
0
2
1
3
[root@localhost ~]#
```

显示已按照预期情况绑定。

----结束

13 程序绑核

简介

进程绑核（又称 CPU 亲和性，CPU Affinity）是一种将进程或线程 “绑定” 到特定 CPU 核心上运行的技术，通过限制进程在指定核心间调度，实现对进程运行位置的精确控制。

准备工作

准备一台ECS主机。

前置条件

- 适用场景：CPU调度频繁、跨numa和跨片带来性能下降的场景。
- 设置效果：设置CPU到固定核心上，不会发生CPU调度、跨numa和跨片。

操作步骤

步骤1 taskset命令，将进程绑定CPU。

启动时绑核：

```
taskset -c {cpu_lists}  + 运行的程序
```

运行中绑核：

```
taskset -cp {cpu_lists}  + 程序的pid
```

步骤2 如果是numa架构，也可选择numactl命令进行绑定。

绑核：

```
numactl -C {cpu_list} + 运行的程序
#-C cpus或--physcpubind=cpus
```

绑numa范围的核：

```
numactl -N {numa_node_list} + 运行的程序
# -N nodes或--cpunodebind=nodes
```

----结束

说明

- {cpu_list}：示例0，或者1~13,16等。
- {numa_node_list}：示例+0，或者+0,+2等多个numa，或者0-2也可以。
- 查看程序当前运行的cpu核亲和性：taskset -cp {pid}。
- numactl其它参数：
  - -m {numa_node}绑内存，如-m +0，绑定内存到numa0，-m +0,+1则绑定到numa0和numa1，或者0-2也可以，表示numa0/1/2。
  - -i <nodes>设置内存交织，如-i 0,1则numa0和numa1的内存交织，-i all表示所有numa内存交织。