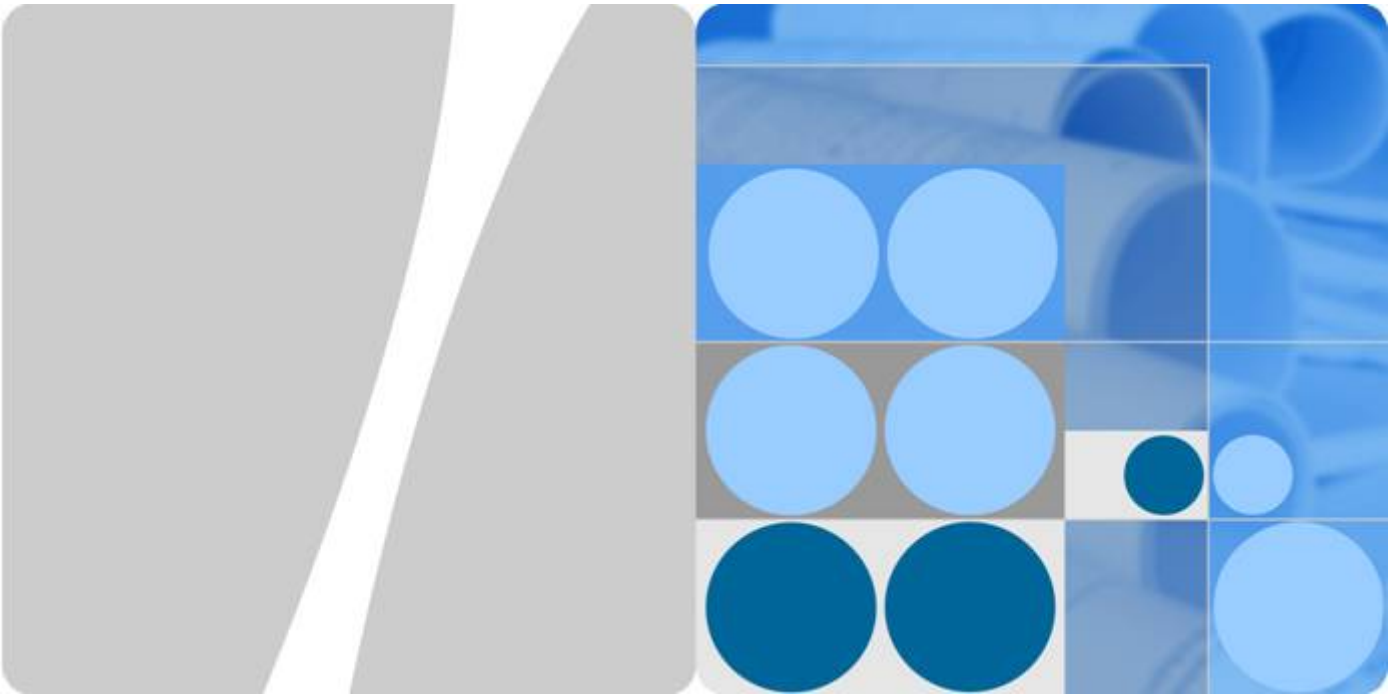


EulerOS V2.0SP2 Repo 部署说明

文档版本　01

发布日期　2019-08-12

华为技术有限公司

HUAWEI

商标声明


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

注意

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

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

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： http://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目 录

1 概述

通过mkrepo工具将EulerOS提供的镜像*EulerOS-V2.0SPx-x86_64-dvd.iso*创建为repo源，并使用nginx进行repo源部署，提供http服务。

说明

EulerOS-V2.0SPx-x86_64-dvd.iso为EulerOS的镜像文件名称，其中SP1简称为EulerOS 2.1，SP2简称EulerOS 2.2。

2 创建/更新本地 repo 源

使用mkrepo和mount挂载，将EulerOS 2.2的镜像EulerOS-V2.0SPx-x86_64-dvd.iso创建为repo源，并能够对repo源进行更新。

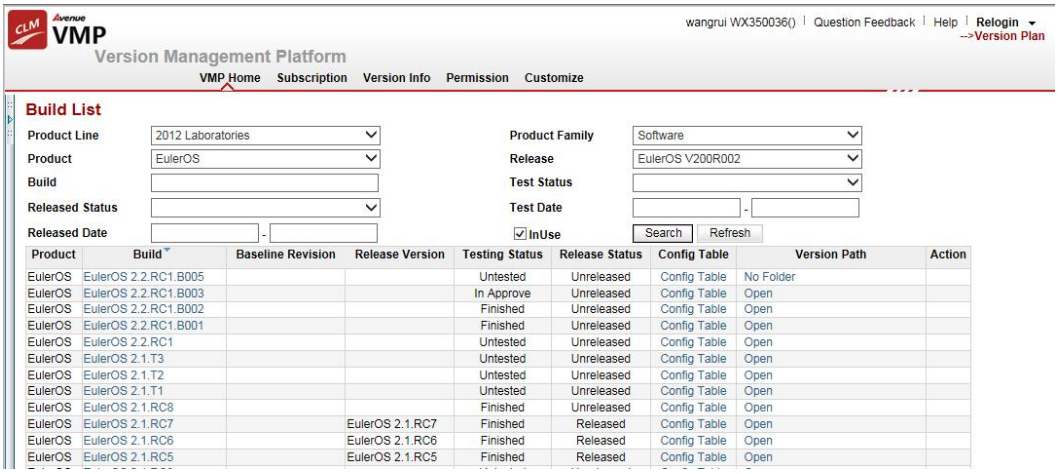
2.1 获取 ISO 镜像

通过VMP上获取EulerOS 2.2版本的ISO。

获取包路径

1. 登录VMP，网址为**http://vmp.huawei.com**。如果没有VMP权限，请向EulerOS的CME申请。
2. 查找EulerOS V200R002下的相应版本，查找您需要的版本即可。如**图 1**所示：

图 2-1 从 VMP 上获取包



2.2 挂载 ISO 创建 repo 源

使用mount命令挂载镜像文件。

示例如下：

```
mount /home/Euler/EulerOS-V2.0SP2-x86_64-dvd.iso  /mnt/
```

挂载好的mnt目录如下：

```
.
├── EFI
├── images
├── isolinux
├── LiveOS
├── Packages
├── repodata
├── ks
├── TRANS.TBL
└── RPM-GPG-KEY-EulerOS
```

其中，Packages为rpm包所在的目录，repodata为repo源元数据所在的目录，RPM-GPG-KEY-EulerOS为EulerOS的签名公钥。

2.3 mkrepo 创建 repo 源

使用repo构建工具mkrepo构建repo源，mkrepo集成rsync、createrepo等命令，mkrepo的使用方式如下：

```
mkrepo
Usage: mkrepo -i isoname -d repodirectory
Options:
        -i, --iso        Specify the iso to make
        -d, --dir        Specify the directory to make repo
        --key-dir        GPG key directory
        -h, --help       Display help information.
Example:
        ./mkrepo -i EulerOS.iso -d /home/Euler/
```

其中，-i/--iso表示ISO所在的路径，-d/--dir表示要构建的repo源目录，--key-dir表示EulerOS提供的GPG公钥存放目录，默认存放在repo的父目录，也可指定。

示例如下：

```
mkrepo -i /home/Euler/EulerOS-V2.0SP2-x86_64-dvd.iso -d /srv/repo/os/2.2/base/x86_64 --key-
dir /srv/repo/
```

构建好的repo目录如下：

```
.
├── os
│   └── 2.2
│       └── base
│           └── x86_64
│               ├── Packages
│               └── repodata
└── RPM-GPG-KEY-EulerOS
```

其中，Packages为rpm包所在的目录，repodata为repo源元数据所在的目录，RPM-GPG-KEY-EulerOS为EulerOS的签名公钥。

2.4 更新 repo 源

更新repo源有两种方式：

- 通过新版本的ISO更新已有的repo源，与创建repo源的方式相同

  ```
  mkrepo -i /home/Euler/EulerOS-V2.0SP2-x86_64-dvd.iso -d /srv/repo/os/2.2/base/x86_64 --key-dir /srv/repo
  ```

  说明

  这里EulerOS-V2.0SP2-x86_64-dvd.iso表示新版本EulerOS的iso文件。

- 在repo源的Packages目录下添加rpm包，然后更新repo源，可通过createrepo命令更新repo源

  ```
  createrepo --update --workers=10 /srv/repo/os/2.2/base/x86_64
  ```

  其中，--update表示更新，--workers表示线程数，可自定义。

对于update升级目录，同样用mkrepo构建repo源，保存到update目录下：

```
mkrepo -i /home/Euler/EulerOS-V2.0SP2-x86_64-dvd.iso -d /srv/repo/os/2.2/update/x86_64
```


3 远端存储 repo 源

如果开通了华为对象存储服务（OBS），可将构建好的repo源通过s3fs-fuse工具上传，今后可直接挂载桶使用repo源。

3.1 s3fs-fuse 工具

s3fs是一款用于Amazon S3在Linux环境下的命令行工具，并为商业和私人免费使用，只需支付对应的存储服务的费用。它的主要功能是能够把桶（Bucket）挂载到本地目录，进行文件、文件夹的拷入、拷出和删除等操作，即完成对桶内文件、文件夹的上传、下载和删除操作。

s3fs-fuse已经集成到EulerOS-V2.0SP2-x86_64-dvd.iso中，若需要安装可通过配置repo，直接yum install进行安装：

```
yum install -y s3fs-fuse
```

3.2 s3fs-fuse 使用说明

使用之前，需要事先在华为企业云服务官方网站（**www.hwclouds.com**）中注册华为云服务账号，并开通华为对象存储服务（OBS），开通方法请参考**《华为对象存储服务（OBS）用户指南》**。

说明

这里以华为企业云为例进行说明，其他云平台的获取方式请参考云平台的说明。

1. 将使用凭证保存到/etc/passwd-uds中：

```
echo 'EDYOURYCMXYYOQJBET5U:AEWNLQVKYHHFW2PX3FUVBRCVR00CNCWO57IIDU4T' > /etc/passwd-uds
```

说明

EDYOURYCMXYYOQJBET5U:AEWNLQVKYHHFW2PX3FUVBRCVR00CNCWO57IIDU4T为使用凭证示例，请用户根据实际情况输入对应的使用凭证。

2. 创建目录/srv/repo，挂载euleros-repo桶：

```
mkdir repo
s3fs euleros-repo /srv/repo -o host=http://10.107.193.157:5080,passwd_file=/etc/passwd-uds,use_path_request_style,uid=0,gid=0,nomultipart
```

说明

10.107.193.157为内部测试使用的服务器，用户需要根据实际环境的IP地址进行配置。

3. 卸载euleros-repo桶：

```
umount /srv/repo
```

3.3 同步 repo 源

1. 在本地/home/Euler/目录下构建repo源：

```
mkrepo -i /home/Euler/EulerOS-V2.0SP2-x86_64-dvd.iso -d /home/Euler/os/2.2/base/x86_64
```

2. 将本地构建的repo同步到挂载的euleros-repo桶中：

```
rsync -a --progress /home/Euler/os /srv/repo
```

说明

同步速度比较慢，请耐心等待。

3. repo源同步完成之后，可直接挂载euler-repo桶当作repo源使用。

3.4 参考文档

华为对象存储服务在这里将不详细介绍，可参考如下文档：

- **华为对象存储服务-第三方客户端用户指南（s3fs）**
- **对象存储服务-客户端操作指南**
- **华为对象存储服务-帮助中心**
- **OBS客户端**

4 部署远端 repo 源

安装操作系统EulerOS2.2（EulerOS-V2.0SP2-x86_64-dvd.iso），在EulerOS2.2上通过nginx部署repo源。

4.1 nginx 安装与配置

1. nginx已经集成到EulerOS-V2.0SP2-x86_64-dvd.iso中，若需要安装可通过配置repo，直接yum install进行安装：

```
yum install -y nginx
```

2. 安装nginx之后，配置/etc/nginx/nginx.conf

```
user  root;
worker_processes  auto;                          # 建议设置为core-1
error_log  /var/log/nginx/error.log  warn;       # log存放位置
pid        /var/run/nginx.pid;

events {
    worker_connections  1024;
}

http {
    include       /etc/nginx/mime.types;
    default_type  application/octet-stream;

    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;
    sendfile        on;
    keepalive_timeout  65;

    server {
        listen       80;
        server_name  localhost;                  # 服务器名（url）
        client_max_body_size 4G;
        root         /srv/repo;                  # 服务默认目录

        location / {
            autoindex            on;             # 开启访问目录下层文件
            autoindex_exact_size on;
```



```
            autoindex_localtime  on;
        }

    }

}
```

4.2 启动 nginx 服务

1. 通过systemd启动nginx服务：

```
systemctl enable nginx
systemctl start nginx
```

2. nginx是否启动成功可通过下面命令查看：

```
systemctl status nginx
```

   - 图 1表示nginx服务启动成功

图 4-1 nginx 服务启动成功

```
[root@localhost ~]# systemctl status nginx
●  nginx.service - SYSV: Nginx is an HTTP(S) server, HTTP(S) reverse proxy and IMAP/POP3 p
roxy server
   Loaded: loaded (/etc/rc.d/init.d/nginx)
   Active: active (running) since Wed 2016-12-21 05:20:31 EST; 2 days ago
     Docs: man:systemd-sysv-generator(8)
 Main PID: 1965 (nginx)
   CGroup: /system.slice/system-hostos.slice/nginx.service
           ├─1965 nginx: master process /usr/sbin/nginx -c /etc/nginx/nginx.conf
           └─1967 nginx: worker process

Dec 21 05:20:30 localhost.localdomain systemd[1]: Starting SYSV: Nginx is an HTTP(S) s....
Dec 21 05:20:31 localhost.localdomain nginx[1446]: Starting nginx: [  OK  ]
Dec 21 05:20:31 localhost.localdomain systemd[1]: Started SYSV: Nginx is an HTTP(S) se....
Hint: Some lines were ellipsized, use -l to show in full.
```

   - 若nginx服务启动失败，查看错误信息：

```
systemctl status nginx.service --full
```

图 4-2 nginx 服务启动失败

```
[root@localhost ~]# systemctl status nginx.service --full
●  nginx.service - SYSV: Nginx is an HTTP(S) server, HTTP(S) reverse proxy and IMAP/POP3 p
roxy server
   Loaded: loaded (/etc/rc.d/init.d/nginx)
   Active: failed (Result: exit-code) since Thu 2016-12-08 06:13:45 EST; 3min 8s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 24340 ExecStart=/etc/rc.d/init.d/nginx start (code=exited, status=1/FAILURE)

Dec 08 06:13:45 localhost.localdomain systemd[1]: Starting SYSV: Nginx is an HTTP(S) serve
r, HTTP(S) reverse proxy and IMAP/POP3 proxy server...
Dec 08 06:13:45 localhost.localdomain nginx[24340]: Starting nginx: nginx: [emerg] mkdir()
 "/var/spool/nginx/tmp/client_body" failed (13: Permission denied)
Dec 08 06:13:45 localhost.localdomain nginx[24340]: [FAILED]
Dec 08 06:13:45 localhost.localdomain systemd[1]: nginx.service: control process exited, c
ode=exited status=1
Dec 08 06:13:45 localhost.localdomain systemd[1]: Failed to start SYSV: Nginx is an HTTP(S
) server, HTTP(S) reverse proxy and IMAP/POP3 proxy server.
Dec 08 06:13:45 localhost.localdomain systemd[1]: Unit nginx.service entered failed state.
Dec 08 06:13:45 localhost.localdomain systemd[1]: nginx.service failed.
```

如图 2所示nginx服务创建失败，是由于目录/var/spool/nginx/tmp/client_body创建失败，手动进行创建，类似的问题也这样处理：


```
mkdir -p /var/spool/nginx/tmp/client_body
mkdir -p /var/spool/nginx/tmp/proxy
mkdir -p /var/spool/nginx/tmp/fastcgi
mkdir -p /usr/share/nginx/uwsgi_temp
mkdir -p /usr/share/nginx/scgi_temp
```

4.3 repo 源部署

1. 创建nginx配置文件/etc/nginx/nginx.conf中指定的目录/srv/repo：
   ```
   mkdir -p /srv/repo
   ```
2. SELinux设置为宽容模式：
   ```
   setenforce permissive
   ```
   说明

   repo server重启后，需要重新设置。
3. 设置防火墙规则，开启nginx设置的端口（此处为80端口），通过firewall设置端口开启：
   ```
   firewall-cmd --add-port=80/tcp --permanent
   firewall-cmd --reload
   ```
   查询80端口是否开启成功，输出为yes则表示80端口开启成功：
   ```
   firewall-cmd --query-port=80/tcp
   ```
   也可通过iptables来设置80端口开启：
   ```
   iptables -I INPUT -p tcp --dport 80 -j ACCEPT
   ```
4. nginx服务设置好之后，即可通过ip直接访问网页，如图 1：

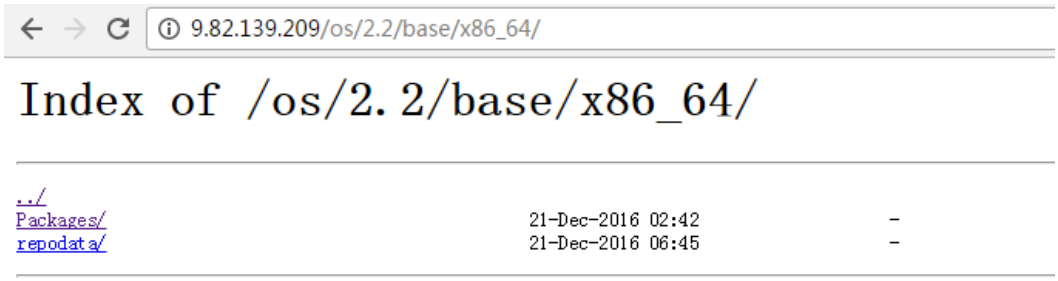
   图 4-3 nginx 部署成功

5. 通过下面几种方式将repo源放入到/srv/repo下：
   - 在/srv/repo下直接创建repo源
     ```
     mkrepo -i /home/Euler/EulerOS-V2.0SP2-x86_64-dvd.iso -d /srv/repo/os/2.2/base/x86_84 --key-dir /srv/repo
     ```
     EulerOS-V2.0SP2-x86_64-dvd.iso存放在/home/Euler目录下。
   - 在/srv/repo下创建repo源的软链接
     ```
     ln -s /home/Euler/os /srv/repo/os
     ```
     /home/Euler/os为已经创建好的repo源，/srv/repo/os将指向/home/Euler/os。
   - 通过s3fs-fuse工具挂载远端存储的repo源桶
     ```
     s3fs euleros-repo /srv/repo/ -o host=http://10.107.193.157:5080,passwd_file=/etc/passwd-uds,use_path_request_style,uid=0,gid=0,nomultipart
     ```
     其中，“euleros-repo”为远端存储的repo源桶，“10.107.193.157”为内部测试IP。

如图 2所示，repo源部署成功：

图 4-4 repo 部署成功



5 使用 repo 源

repo可配置为yum源，yum（全称为 Yellow dog Updater, Modified）是一个Shell前端软件包管理器。基于RPM包管理，能够从指定的服务器自动下载RPM包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软体包，无须繁琐地一次次下载和安装。

5.1 repo 配置为 yum 源

构建好的repo可以配置为yum源使用，在/etc/yum.repos.d/目录下创建***.repo的配置文件（必须以.repo为扩展名），分为本地和http服务器配置yum源两种方式：

- 配置本地yum源

  在/etc/yum.repos.d目录下创建EulerOS-2.2-Base.repo文件，使用构建的本地repo作为yum源，Euler0S-2.2-Base.repo的内容如下：

  ```
  [base]
  name=base
  baseurl=file:///srv/repo/os/2.2/base/x86_64
  enabled=1
  gpgcheck=1
  gpgkey=file:///srv/repo/RPM-GPG-KEY-EulerOS
  ```

  说明

  gpgcheck可设置为1或0，1表示进行gpg（GNU Private Guard）校验，0表示不进行gpg校验，gpgcheck可以确定rpm 包的来源是有效和安全的。

  gpgkey为签名公钥的存放路径。

- 配置http服务器yum源

  在/etc/yum.repos.d目录下创建EulerOS-2.2-Base.repo文件，使用http服务端的repo作为yum源，Euler0S-2.2-Base.repo的内容如下：

  ```
  [base]
  name=base
  baseurl=http://192.168.1.2/os/2.2/base/x86_64
  enabled=1
  gpgcheck=1
  gpgkey=http://192.168.1.2/RPM-GPG-KEY-EulerOS
  ```


说明

“192.168.1.2”为示例地址，请用户根据实际情况进行配置。

5.2 repo 优先级

当有多个repo源时，可通过在.repo文件的priority参数设置repo的优先级。其中，1为最高优先级，99为最低优先级，如给EulerOS-2.2-Base.repo配置优先级为2：

```
[base]
name=base
baseurl=http://192.168.1.2/os/2.2/base/x86_64
enabled=1
priority=2
gpgcheck=1
gpgkey=http://192.168.1.2/RPM-GPG-KEY-EulerOS
```

说明

gpgcheck可设置为1或0，1表示进行gpg（GNU Private Guard）校验，0表示不进行gpg校验，gpgcheck可以确定rpm 包的来源是有效和安全的。

gpgkey为签名公钥的存放路径。

5.3 yum 相关命令

yum命令在安装升级时能够自动解析包的依赖关系，一般的使用方式如下：

```
yum <command> <packages name>
```

常用的命令如下：

- 安装
  ```
  yum install <packages name>
  ```
- 升级
  ```
  yum update <packages name>
  ```
- 回退
  ```
  yum downgrade <packages name>
  ```
- 检查更新
  ```
  yum check-update
  ```
- 卸载
  ```
  yum remove <packages name>
  ```
- 查询
  ```
  yum search <packages name>
  ```
- 本地安装
  ```
  yum localinstall <absolute path to package name>
  ```
- 查看历史记录
  ```
  yum history
  ```
- 清除缓存目录
  ```
  yum clean all
  ```
- 更新缓存
  ```
  yum makecache
  ```

