



## Linux recovery 部署说明

文档历史:

版本	作者	注释
1.0	张延明	新建

## 目录

一、 产品板端.....	3
1. 资料说明.....	3
2. 编译 recovery.img.....	5
3. 编译 x-loader.....	6
二、服务器端.....	7
三、QA.....	8

# 一、产品板端

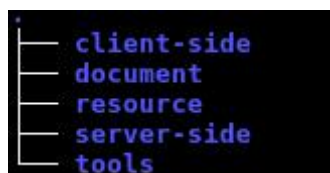
## 1. 资料说明

首先解压 linux-recovery.tar.xz

```
mkdir linux-recovery
```

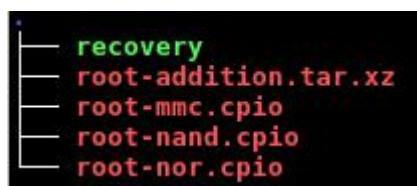
```
tar -xvf linux-recovery.tar.xz -C linux-recovery
```

压缩包包含以下内容：



```
├── client-side
├── document
├── resource
├── server-side
└── tools
```

- client-side:包含了 recovery 应用和文件系统，如下图：



```
├── recovery
├── root-addition.tar.xz
├── root-mmc.cpio
├── root-nand.cpio
└── root-nor.cpio
```

用户需要将 recovery 应用放到文件系统的/sbin/目录下，以 nand 为例：

先解开 root-nand.cpio

```
mkdir root-nand
```

```
cd root-nand
```

```
sudo cpio -idm < ../root-nand.cpio
```

```
cp -a ../recovery/sbin/
```

重新打包

```
find ./ -print | cpio -H newc -ov > ../root-nand.cpio
```

root-nand.cpio 仅仅包含基本的 busybox、libc.so 和 ld.so 以及一些图片资源。

recovery 应用所依赖的其他资源在运行时通过挂载 rootfs，从 rootfs 拷贝过来。这样可以减小 recovery.img 分区。rootfs-addition.tar.xz 包含了 recovery 应用运行所依赖的库和其他资源。**如果用户担心 rootfs 分区不能正常挂载，那么就需要将 root-addition.tar.xz 里的内容按照已有的目录结构拷贝到 root-nand.cpio 中。这样 recovery 分区会变大，但是只要 bootloader 分区没有被破坏，系统任何一个分区都可以升级/修复，因为 recovery.img 完全在内存里运行！**

recovery 应用的配置文件 recovery.conf 放在/etc 下，如下图：



可配置的内容如下：

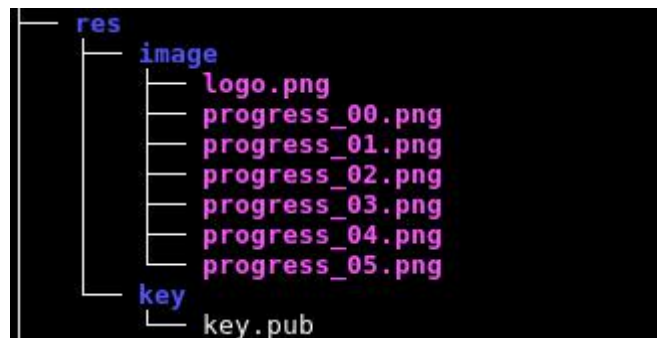
```

Version="2.0";

Application:
{
    Server:
    {
        ip="194.169.2.59";
        url="http://194.169.2.59:8008/data/recovery";
    };
};
  
```

其中，Version 字段配置 recovery 版本，不可修改，ip 字段配置 OTA 服务器的 IP 地址，url 字段配置升级包在服务器上的路径。注意，配置文件名和路径一定不能更改！

/res 存放公钥、Logo 以及进度条图片，图片只支持 PNG 格式，这些文件名和路径一定不能修改。如果用户更换进度条图片，文件名必须是 progress\_XX.png，XX 必须是连续的数字。



res/image 可以存放名为 font.png 的字体图片，recovery 只支持 ASCII 码显示。字体图片如下：

```

!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
  
```

图片高度必须为 96x2，高度为 96 个像素，上面一行为正常字体，下面一行为加粗字体。如果没有提供该文件则使用内置的 10x18 的字体。

- document: 包含说明文档
- resource: 包含了预置的一些数字证书和相应的私钥
- server-side: 包含了服务器端的资源，详细请见第二章服务器端
- tools: 包含了主机端的工具，如下图：



dumpkey.jar 用于从数字证书提取公钥，用法如下：

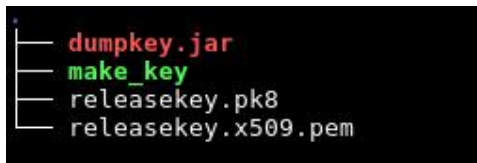
```
java -jar dumpkey.jar ../resource/security/testkey.x509.pem > key.pub
```

将生成的 key.pub 预置到 root-nand.cpio 中 res/key 目录下。

make\_key 用于生成用户自定义的数字证书和相应的私钥，用法如下：

```
make_key releasekey '/C=CN/ST=Guangdong/L=Shenzhen/O=Company/OU=Department/CN=Your
Name/emailAddress=YourE-mailAddress'
```

命令执行完后会在当前目录下生成数字证书和相应的私钥，如下图：



## 2. 编译 recovery.img

在配置好 root-nand.cpio 后下面需要编译 recovery.img

首先需要一份内核配置，该配置可以和正常启动的配置一样，也可以根据具体情况裁减一些，可以减小 kernel 镜像大小。recovery 内核配置中需要增加以下配置：

```
CONFIG_BLK_DEV_INITRD=y
CONFIG_INITRAMFS_SOURCE=root-nand.cpio 文件路径
CONFIG_RD_XZ=y
CONFIG_INITRAMFS_COMPRESSION_XZ=y
```

配置如下所示：

General setup --->

```
[*] Initial RAM filesystem and RAM disk (initramfs/initrd) support
    (../linux-recovery/client-side/root-nand.cpio) Initramfs source file(s)
[]   Support initial ramdisks compressed using gzip
[]   Support initial ramdisks compressed using bzip2
[]   Support initial ramdisks compressed using LZMA
[*]  Support initial ramdisks compressed using XZ
[]   Support initial ramdisks compressed using LZO
Built-in initramfs compression mode (XZ) --->
```

在添加好 recovery 配置后，开始编译：

以 halley2\_V20 为例：

```
make halley2_sfcnand_recovery_defconfig
```

```
make recovery
```

将生成的 recovery 烧录到 recovery 分区。

### 3. 编译 x-loader

recovery.img 生成后还需要 bootloader 支持引导 recovery.img。

下面编译 bootloader：

```
cd x-loader
```

```
vim boards/BOARD_NAME/board.mk
```

修改变量 CONFIG\_RECOVERY := y

```
make
```

将生成的 bootloader 烧录到 bootloader 分区。

## 二、服务器端

## 三、QA

Q: 如何更换的数字证书?

A: 参考 1.1 章节 `make_key` 和 `dumpkey.jar` 的用法

Q: 如何更换 logo 和升级进度条以及字体?

A: 参考 1.1 章节关于资源的描述

Q: 如何编译 recovery 镜像?

A: 参考 1.1 章节关于编译的描述

Q: 如何配置 OTA 服务器地址?

A: 参考 1.1 章节关于 `recovery.conf` 的描述

Q: 如何使用 SD 卡或者 U 盘进行本地升级?

A: 将升级包放在 SD 卡或 U 盘中 `recovery-update` 目录下, 插入 SD 卡或 U 盘, 进入 `recovery` 模式, 执行 `recovery` 应用