


openssh交叉编译与问题

 zynq 同时被 2 个专栏收录 ▾

交叉编译 openssh

使用版本
编译

- zlib-1.2.11 编译
- openssl-1.0.2u 编译
- openssh-8.3p1 编译

sshd 移植
编译错误问题

早期交叉编译过 sshd，使用的是动态编译，需要添加动态库，去年重做文件系统时又重新编译了下，最近比较闲，写个记录文章，这回用的静态编译，网上的教程没发现静态编译的（应该是静态编译出来整体大小过大，能达到34.4M左右，对于小型的系统用不上），本文记录下使用 **静态编译** 的方法，基本就是配置文件的不同。静态编译时，会出现各种问题，这些是在动态编译时不会出现的，详见文章末尾。发现了个轻量级的 sshd 软件，叫做 **dropbear**，这个编译起来不仅简单，而且静态编译出来还比较小（共4个文件，才3.52M，dropbear 才1.03M），有兴趣的看下篇文章。

使用版本

交叉编译总共需要用到3个文件，本文使用如下3个版本：

- openssh-8.3p1.tar.gz**
- openssl-1.0.2u.tar.gz** [openssl-1.0.2u 下载地址](#)
- zlib-1.2.11.tar.gz**

编译

文件不建议放在虚拟机共享文件夹下，可能会出现错误，本文放在共享文件夹下是把坑踩过了，后面做具体说明。

zlib -1.2.11 编译

解压缩包，进入 zlib-1.2.11 目录，对其进行编译前的配置：

```
1 | prefix=/mnt/hgfs/openssh/zlib_install/ CC=arm-linux-gnueabi-gcc CFLAGS="-static -fPIC" ./configure
```

然后 **make** 即可编译完成，而后 **make install** 将其安装在上面配置的 **prefix** 文件夹。

openssl -1.0.2u 编译

解压缩包，进入 openssl-1.0.2u 目录（**注：openssl-1.0.2u.tar.gz 解压缩不可放在共享文件夹下，否则会报错，安装在共享文件夹不影响后续编译，但是会出现无法创建软链接，后面具体说明**），对其进行编译前的配置：

```
1 | ./Configure --prefix=/mnt/hgfs/openssh/openssl_install --cross-compile-prefix=arm-linux-gnueabi- no-asm no-shared linux-armv7
```

然后 **make** 即可编译完成，而后 **make install** 将其安装在上面配置的 **prefix** 文件夹。

openssh -8.3p1 编译

解压缩包，进入 openssh-8.3p1 目录，对其进行编译前的配置：

```
1 | ./configure --host=arm-linux-gnueabi --with-zlib=/mnt/hgfs/openssh/zlib_install --with-ssl-dir=/mnt/hgfs/openssh/openssl_install
```

其中 **--with-zlib** 与 **--with-ssl-dir** 分别是上面的 **zlib-1.2.11** 与 **openssl-1.0.2u** 编译好的安装目录。
配置完成后，使用 **make** 即可编译完成。

上述主要使用了静态编译，若要动态编译，配置时，zlib-1.2.11 与 openssl-1.0.2u 删除 **LDFLAGS** 参数，openssl-1.0.2u 配置的 no-shared 修改成 **shared** 即可。

sshd 移植

新建一个文件夹 **local**，将 **openssh-8.3p1** 目录下生成的 **scp sftp ssh sshd ssh-add ssh-agent ssh-keygen ssh-keyscan** 文件复制到 **local** 文件夹下的 **bin** 目录，将 **sftp-server ssh-keysign** 2 个文件复制到 **local** 文件夹下的 **libexec** 目录，将 **moduli ssh_config sshd_config** 3 个文件复制到 **local** 文件夹下的 **etc** 目录。

以下进行嵌入式文件系统的操作：

1. 将整理好的 **local** 文件夹放到嵌入式文件系统的 **usr** 目录下；
2. 使用 **ssh-keygen** 生成公钥和私钥并修改权限（有其中一个就可以使用 sshd），生成的文件放置的目录可自行定义，可在 **sshd_config** 文件中修改

```
1 | ssh-keygen -t rsa -f ssh_host_rsa_key -N ""
2 | ssh-keygen -t ecdsa -f ssh_host_ecdsa_key -N ""
3 | ssh-keygen -t ed25519 -f ssh_host_ed25519_key -N ""
4 | chmod 600 ssh_host_*
```

3. 修改之前复制的 **sshd_config**，修改如下，其他默认就好。

```
1 | HostKey /usr/local/etc/ssh_host_rsa_key
2 | HostKey /usr/local/etc/ssh_host_ecdsa_key
3 | HostKey /usr/local/etc/ssh_host_ed25519_key
4 | PermitRootLogin yes
5 | Subsystem sftp /usr/local/libexec/sftp-server
```

4. 在 **/etc/passwd** 文件末尾添加

```
1 | sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/usr/sbin/nologin
```

5. 使用命令 **passwd** 配置 **root** 用户密码，若 **/etc** 文件夹中没有 **shadow** 文件，密码会直接存在 **/etc/passwd** 文件中。
6. 在嵌入式系统中添加 **/var/empty** 文件夹，其所有者需要是 **root** 用户，若不是，需使用命令修改。

```
1 | chown -R root:root /var/empty
```

完成上述步骤，可使用 **/usr/local/bin/sshd** 命令运行。可使用软件 **putty** 或者 **winscp** 软件登录试验。

编译错误问题

1. 将 **openssl-1.0.2u** 解压到共享目录，配置时会出现操作不支持，如下：

```
1 | verify_extra_test.c => dummytest.c
2 | ln: failed to create symbolic link 'clienthellotest.c': Operation not supported
3 | clienthellotest.c => dummytest.c
4 | ln: failed to create symbolic link 'ssl2confctest.c': Operation not supported
5 | ssl2confctest.c => dummytest.c
6 | ln: failed to create symbolic link 'dtl1stest.c': Operation not supported
7 | dtl1stest.c => dummytest.c
8 | ln: failed to create symbolic link 'bad_dtls_test.c': Operation not supported
9 | bad_dtls_test.c => dummytest.c
10 | ln: failed to create symbolic link 'fatalerrtest.c': Operation not supported
11 | fatalerrtest.c => dummytest.c
```

解决办法：将 **openssl-1.0.2u** 解压到**非共享目录**

2. 在配置 **openssh-8.3p1** 时，卡死在 **checking whether OpenSSL's PRNG is internally seeded...**
可能是编译器和 **openssl** 的新版本不兼容，用的 **T1** 的编译器不会卡死，使用 **Xilinx 2017.4** 的编译器出现卡死。

解决办法：使用版本 openssl-1.0.2u 可正常配置，版本 openssl-1.1.0l 以上的包括最新的试了好几个都会卡死，但使用 openssl-1.1.0l 动态编译正常。

3. 在配置 openssh-8.3p1 时，出现错误 `configure: error: *** working libcrypto not found, check config.log`，这个问题是在使用 openssl-1.1.0l 出现的，在 openssl-1.0.2u 版本中没有这个问题。
打开 config.log 文件，发现有以下问题（截取部分）：

```
1 | /mnt/hgfs/share/openssh/openssl_install/lib/libcrypto.a(threads_pthread.o): In function `CRYPTO_THREAD_get_local':
2 | threads_pthread.c:(.text+0xf0): undefined reference to `pthread_getspecific'
3 | /mnt/hgfs/share/openssh/openssl_install/lib/libcrypto.a(threads_pthread.o): In function `CRYPTO_THREAD_set_local':
4 | threads_pthread.c:(.text+0xfc): undefined reference to `pthread_setspecific'
5 | /mnt/hgfs/share/openssh/openssl_install/lib/libcrypto.a(threads_pthread.o): In function `CRYPTO_THREAD_cleanup_local':
6 | threads_pthread.c:(.text+0x114): undefined reference to `pthread_key_delete'
7 | /mnt/hgfs/share/openssh/openssl_install/lib/libcrypto.a(threads_pthread.o): In function `CRYPTO_THREAD_get_current_id':
8 | threads_pthread.c:(.text+0x124): undefined reference to `pthread_self'
```

解决办法：这个问题应该是 openssl 使用了 pthread 的库，而 openssh-8.3p1 使用了 openssl 的库，所以在配置 openssh-8.3p1 时，在参数 **LDFLAGS** 加上 **-pthread** 就可以了。

4. 在运行 sshd 时出现 `Privilege separation user sshd does not exist`

解决办法：在 `/etc/passwd` 文件中添加：

```
1 | sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/usr/sbin/nologin
```

5. 使用软件登录时，用户名和密码都正确，一直提示 `Access denied`。

解决办法：出现这个问题，主要是在第4个问题上，在 `/etc/passwd` 文件中添加内容时，不小心引入了Windows的换行符 `\r\n`，将换行符修改成 linux 的 `\n` 结尾就可以了。