

## 64 位编译环境搭建

### 背景与意义：

编译服务器在编译的时候占用内存巨大，如果采用 32 位的系统只能支持到 3 点几 G 的内存，当用户数量增多时编译显得十分吃力。搭建 64 位的编译环境可以有效的解决 32 位系统内存不足的情况，从而提高了编译的速度。

```
liangjinchao@SVTI:~/b64/PUBLIC/build
top - 22:02:44 up 4:03, 3 users, load average: 9.60, 7.60, 3.51
Tasks: 132 total, 3 running, 129 sleeping, 0 stopped, 0 zombie
Cpu(s): 43.5%us, 6.2%sy, 0.0%ni, 49.5%id, 0.8%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 7706824k total, 7485064k used, 221760k free, 446040k buffers
Swap: 3702780k total, 848k used, 3701932k free, 6393884k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 19820 heyiwen   20   0  9764  4224  504  R   80.4   0.1   0:02.42 bzip2
 19819 heyiwen   20   0 19216  1160  924  S    3.7   0.0   0:00.11 tar
 19367 heyiwen   20   0 10104  2240 1128  S    0.7   0.0   0:00.04 wireshark_cen_c
      3 root       RT    0    0    0    0  S    0.3   0.0   0:00.60 migration/0
    1209 root       20   0    0    0    0  R    0.3   0.0   0:01.66 kondemand/1
 15401 liangjin   20   0 15304  1568  964  R    0.3   0.0   0:00.39 top
      1 root       20   0 19356  1504 1228  S    0.0   0.0   0:00.76 init
      2 root       20   0    0    0    0  S    0.0   0.0   0:00.01 kthreadd
      4 root       20   0    0    0    0  S    0.0   0.0   0:00.10 ksoftirqd/0
      5 root       RT    0    0    0    0  S    0.0   0.0   0:00.00 stopper/0
      6 root       RT    0    0    0    0  S    0.0   0.0   0:00.00 watchdog/0
      7 root       RT    0    0    0    0  S    0.0   0.0   0:00.35 migration/1
      8 root       RT    0    0    0    0  S    0.0   0.0   0:00.00 stopper/1
```

### 资料准备：

确认以下资料是否准备完全：

1. 获得所需的完整的 lib 库，必须要完整，否则缺失文件无法编译成功。
2. 获得以下软件的安装包包含如下几个文件：

ppc-uclibc-cross-v181.tar.bz2

vdfuse.tar.gz

vmware-disklib-5.1.0.tar.gz

x86\_64-uclibc-cross-v181.tar.bz2

3. 获得以下几个库文件（见附件）：

ld-linux.so.2 libc.so.6 libgmp.so.3 libmpfr.so.1 libz.so.1

4. PUBLIC 文件夹及项目文件夹：确保 PUBLIC 文件夹本身是没问题的。

## 环境准备：

1. 安装好 x86\_64 版本的 CentOS7 或 CentOS6。
2. 配置好 IP、网关、DNS，确保可以上网。通过 ping [www.baidu.com](http://www.baidu.com) 正常。
3. 配置好 samba 服务器以便拷贝文件（也可以用 SCP、FTP、SFTP 等）。
4. 检查 CentOS 是否支持 yum 命令，没有则下载 rpm 安装包自行安装。

## 开始搭建：

搭建的步骤最好按顺序来：

1. 解压准备好的安装包到/opt

ppc-uclibc-cross-v181.tar.bz2

vdfuse.tar.gz

vmware-disklib-5.1.0.tar.gz

x86\_64-uclibc-cross-v181.tar.bz2

参考命令：

(1) xxx.tar.gz: tar xzf xxx.tar.gz -C 目的路径

(2) xxx.tar.bz2: tar xjf xxx.tar.bz2 -C 目的路径

解压完成后，

(1) 运行 vdfuse 目录下的 install.sh

(2) 运行 vmware-disklib-5.1.0 中的 vmware-install.pl

2. 拷贝 lib 文件到/var/xxx/下，或者其他路径，路径名决定了编译时-d 后面的路径参数。
3. 将资料准备中的第三步的文件 (\*.so) 拷贝到/lib 目录下。
4. 使用 yum install 命令安装以下软件到最新(主要是 gcc, glibc 相关):

compat-gcc-34.x86\_64 3.4.6-19.el6 @base

compat-gcc-34-c++.x86\_64 3.4.6-19.el6 @base

compat-gcc-34-g77.x86_64	3.4.6-19.el6	@base
gcc.x86_64	4.4.7-11.el6	@base
gcc-c++.x86_64	4.4.7-11.el6	@base
libgcc.i686	4.4.7-11.el6	@base
libgcc.x86_64	4.4.7-11.el6	@anaconda-CentOS-201410241409.x86_64/6.6
compat-glibc.x86_64	1:2.5-46.2	@base
compat-glibc-headers.x86_64	1:2.5-46.2	@base
glibc.i686	2.12-1.149.el6_6.9	@updates
glibc.x86_64	2.12-1.149.el6_6.9	@updates
glibc-common.x86_64	2.12-1.149.el6_6.9	@updates
glibc-devel.i686	2.12-1.149.el6_6.9	@updates
glibc-devel.x86_64	2.12-1.149.el6_6.9	@updates
glibc-headers.x86_64	2.12-1.149.el6_6.9	@updates
glibc-static.i686	2.12-1.149.el6_6.9	@updates
glibc-static.x86_64	2.12-1.149.el6_6.9	@updates
glibc-utils.x86_64	2.12-1.149.el6_6.9	@updates
compat-libf2c-34.i686	3.4.6-19.el6	@base
compat-libf2c-34.x86_64	3.4.6-19.el6	@base

5. 将 PUBLIC 文件夹拷贝到工作环境中，cd PUBLIC/build 运行编译命令，第一次编译时只放 PUBLIC，并且加-d 选项。

命令示例：./simware7.sh -e 64sim7cen.ipe -d /var/svti

## 遇到问题及解决方法：

情况一：

```
.../opt/x86_64-uclibc-cross-v181/bin/x86_64-unknown-linux-uclibc-gcc: error  
while loading shared libraries: libc.so.6: cannot open shared object file: No such  
file or directory  
/opt/x86_64-uclibc-cross-v181/bin/x86_64-unknown-linux-uclibc-gcc: error while  
loading shared libraries: libc.so.6: cannot open shared object file: No such file or  
directory
```

解决方法：缺少对应的 `lib*.so` 文件，将其拷贝到 `/lib` 下，如果提示覆盖，输入 `n`。

情况二：

```
make: /opt/x86_64-uclibc-cross-v181/bin/x86_64-unknown-linux-uclibc-gcc:  
Command not found  
make[1]: /opt/x86_64-uclibc-cross-v181/bin/x86_64-unknown-linux-uclibc-gcc:  
Command not found  
make[1]: /opt/x86_64-uclibc-cross-v181/bin/x86_64-unknown-linux-uclibc-gcc:
```

解决方法：重新安装 `gcc` 相关软件包到最新版本。

情况三：

```
In file included from /usr/include/features.h:399:0,
                 from /usr/include/stdio.h:27,
                                     from
/home/compile/Compile/PUBLIC/kernel/scripts/mod/mk_elfconfig.c:1:
/usr/include/gnu/stubs.h:7:27: fatal error: gnu/stubs-32.h: No such file or
directory
# include <gnu/stubs-32.h>
```

解决方法：缺乏兼容 32 位的软件包。将搭建环境步骤中的第 4 步完整做一遍，不要遗漏了。

情况四：

[illegible]

解决方法：lib 库不完整，如....\lib\target\x86\_64\x86\_64\_debug\_cen\_wlan.7z 等，如果没有则找公司经理或相关人员获取。

情况五：

[illegible]

解决方法：之前的编译出错导致。清空编译目录，重新拷贝 PUBLIC 文件夹。

情况六：

```
/bin/sh: /opt/x86_64-uclibc-cross-v181/bin/x86_64-unknown-linux-uclibc-gcc:
/lib/ld-linux.so.2: bad ELF interpreter: No such file or directory
```

解决办法: \*.so 库文件不对或不兼容。可以从曾经编译成功的机器上拷贝下来。