libpcap库的交叉编译

作者：滑国青

历史：2023.07.06 v0.1

[前言 1](#_Toc139547734)

[这是5ev的petalinux在source settings.sh以后的编译器 1](#_Toc139547735)

[这是10.240.13.120上缺省安装的编译器，7ev使用的 2](#_Toc139547736)

[Libpcap下载 3](#_Toc139547737)

[交叉编译 3](#_Toc139547738)

[使用libpcap的几个问题 4](#_Toc139547739)

[如何使用poll来获取包 4](#_Toc139547740)

[如何性能优化libpcap抓包 4](#_Toc139547741)

[结束 4](#_Toc139547742)

# 前言

目前发现，使用gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu是可以交叉编译libpcap的。而用aarch64-none-elf-gcc是不行的，可能这个不是linux下用的编译器，而是 FreeRTOS版本。

## 这是5ev的petalinux在source settings.sh以后的编译器

a23286@libpcap-1.6.2$which aarch64-none-elf-gcc

/home/a23286/alinx/petalinux/tools/xsct/gnu/aarch64/lin/aarch64-none/bin/aarch64-none-elf-gcc

a23286@libpcap-1.6.2$aarch64-none-elf-gcc -v

Using built-in specs.

COLLECT\_GCC=/home/a23286/alinx/petalinux/tools/xsct/gnu/aarch64/lin/aarch64-none/bin/../x86\_64-oesdk-linux/usr/bin/aarch64-xilinx-elf/aarch64-xilinx-elf-gcc.real

COLLECT\_LTO\_WRAPPER=/home/a23286/alinx/petalinux/tools/xsct/gnu/aarch64/lin/aarch64-none/x86\_64-oesdk-linux/usr/bin/aarch64-xilinx-elf/../../libexec/aarch64-xilinx-elf/gcc/aarch64-xilinx-elf/9.2.0/lto-wrapper

Target: aarch64-xilinx-elf

Configured with: ../../../../../../work-shared/gcc-9.2.0-r0/gcc-9.2.0/configure --build=x86\_64-linux --host=x86\_64-oesdk-linux --target=aarch64-xilinx-elf --prefix=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr --exec\_prefix=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr --bindir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/bin/aarch64-xilinx-elf --sbindir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/bin/aarch64-xilinx-elf --libexecdir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/libexec/aarch64-xilinx-elf --datadir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/share --sysconfdir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/etc --sharedstatedir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/com --localstatedir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/var --libdir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/lib/aarch64-xilinx-elf --includedir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/include --oldincludedir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/include --infodir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/share/info --mandir=/usr/local/oecore-x86\_64/sysroots/x86\_64-oesdk-linux/usr/share/man --disable-silent-rules --disable-dependency-tracking --with-libtool-sysroot=/scratch/mhatle/toolchain/baremetal-toolchains/20200422-094955/build/aarch64-tc-x86\_64/work/x86\_64-nativesdk-oesdk-linux/gcc-cross-canadian-aarch64/9.2.0-r0/recipe-sysroot --with-gnu-ld --enable-shared --enable-languages=c,c++ --enable-threads=posix --enable-multilib --enable-c99 --enable-long-long --enable-libstdcxx-pch --program-prefix=aarch64-xilinx-elf- --without-local-prefix --enable-lto --disable-libssp --enable-libitm --disable-bootstrap --disable-libmudflap --with-system-zlib --enable-linker-build-id --with-ppl=no --with-cloog=no --enable-checking=release --enable-cheaders=c\_global --without-isl --with-gxx-include-dir=/not/exist/usr/include/c++/9.2.0 --with-build-time-tools=/scratch/mhatle/toolchain/baremetal-toolchains/20200422-094955/build/aarch64-tc-x86\_64/work/x86\_64-nativesdk-oesdk-linux/gcc-cross-canadian-aarch64/9.2.0-r0/recipe-sysroot-native/usr/aarch64-xilinx-elf/bin --with-build-sysroot=/scratch/mhatle/toolchain/baremetal-toolchains/20200422-094955/build/aarch64-tc-x86\_64/work/x86\_64-nativesdk-oesdk-linux/gcc-cross-canadian-aarch64/9.2.0-r0/recipe-sysroot --enable-poison-system-directories --enable-nls --with-glibc-version=2.28 --enable-initfini-array --without-headers --with-newlib --disable-libstdcxx-pch --with-newlib --disable-threads --enable-plugins --with-gnu-as --disable-libitm --disable-multiarch --with-arch=armv8-a --enable-multilib --with-sysroot=/not/exist

Thread model: single

gcc version 9.2.0 (GCC)

## 这是10.240.13.120上缺省安装的编译器，7ev使用的

a23286@~$echo $CC

/opt/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gnu-gcc

a23286@libpcap-1.6.2$/opt/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gnu-gcc -v

Using built-in specs.

COLLECT\_GCC=/opt/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gnu-gcc

COLLECT\_LTO\_WRAPPER=/opt/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/bin/../libexec/gcc/aarch64-none-linux-gnu/10.3.1/lto-wrapper

Target: aarch64-none-linux-gnu

Configured with: /data/jenkins/workspace/GNU-toolchain/arm-10/src/gcc/configure --target=aarch64-none-linux-gnu --prefix= --with-sysroot=/aarch64-none-linux-gnu/libc --with-build-sysroot=/data/jenkins/workspace/GNU-toolchain/arm-10/build-aarch64-none-linux-gnu/install//aarch64-none-linux-gnu/libc --with-bugurl=https://bugs.linaro.org/ --enable-gnu-indirect-function --enable-shared --disable-libssp --disable-libmudflap --enable-checking=release --enable-languages=c,c++,fortran --with-gmp=/data/jenkins/workspace/GNU-toolchain/arm-10/build-aarch64-none-linux-gnu/host-tools --with-mpfr=/data/jenkins/workspace/GNU-toolchain/arm-10/build-aarch64-none-linux-gnu/host-tools --with-mpc=/data/jenkins/workspace/GNU-toolchain/arm-10/build-aarch64-none-linux-gnu/host-tools --with-isl=/data/jenkins/workspace/GNU-toolchain/arm-10/build-aarch64-none-linux-gnu/host-tools --enable-fix-cortex-a53-843419 --with-pkgversion='GNU Toolchain for the A-profile Architecture 10.3-2021.07 (arm-10.29)'

Thread model: posix

Supported LTO compression algorithms: zlib

gcc version 10.3.1 20210621 (GNU Toolchain for the A-profile Architecture 10.3-2021.07 (arm-10.29))

# Libpcap下载

<http://www.tcpdump.org/release/libpcap-1.6.2.tar.gz>

# 交叉编译

请参考：

https://adasgitlab.autel.com/DP-Embed/skyfend\_embeded\_team\_mgr.git

产品开发\产品-TracerLinux\资料集合\pcap使用\ (31条消息) Linux网络编程：libpcap 移植及使用\_libpcap下载\_錦鈊銀的博客-CSDN博客

关键点：

CC=/opt/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gnu-gcc

CXX=/opt/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gnu-g++

export PATH=$PATH: /opt/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/bin

./configure --host=aarch64-none-elf --with-pcap=linux --prefix=$PWD/\_install

make

make install

# 使用libpcap的几个问题

## 如何使用poll来获取包

我们的tracer产品，会使用libev作为多路IO框架，我们希望libpcap提供socket fd,然后我们通过libev的多路IO来收包，而不是使用libpcap的pcap\_next直接收包。

多路IO机制，可以让我们的remoteID线程即能接收消息总线上msg报文，也可以同时接收抓包的报文。

答：libpcap提供了此机制，参见源码里的测试程序：

libpcap-1.6.2/tests/selpolltest.c

使用此函数，即可获取fd:

pcap\_get\_selectable\_fd()

pcap\_setnonblock() 改为非阻塞方式

## 如何性能优化libpcap抓包

最佳的方法是能达到直接通过mmap映射到用户空间后，获取报文本身。

此文章对性能优化作了较好的说明：

参见资料：

https://adasgitlab.autel.com/DP-Embed/skyfend\_embeded\_team\_mgr.git

产品开发\产品-TracerLinux\资料集合\pcap使用\ linux libpcap的性能问题，请大家注意绕行。 - \_备忘录 - 博客园

## 如何抓wifi包并解析

Wifi包头有radiotap头。

<https://github.com/niym/wicap>

# 结束