

José Divino Ferreira Júnior - 11621EMT010

## Anatomy of cross-compilation toolchains

### O que é uma cadeia de ferramentas de compilação cruzada?

Um conjunto de ferramentas que permite construir código fonte em código binário para um destino de plataforma diferente daquela em que a compilação ocorre.

Podendo conter:

- Diferentes arquiteturas de processadores;
- Diferentes ABI;
- Sistemas operacionais diferentes;
- Bibliotecas em C diferentes;

Para a construção temos três máquinas envolvidas no processo de construção.

- máquina de construção, onde a construção ocorre
- máquina host, onde a execução ocorre
- máquina de destino, para a qual os programas geram código

Uma definição de sistema descreve um sistema: arquitetura da CPU, sistema operacional, fornecedor, ABI, biblioteca C

Formas diferentes:

`<arch>-<vendor>-<os>-<libc/abi>`, full form

`<arch>-<os>-<libc/abi>`

### Componentes:

- a arquitetura da CPU: arm, mips, powerpc, i386, i686, etc.
- (principalmente) string de forma livre, ignorada pelo autoconf
- o sistema operacional. Ou nenhum ou linux para o propósito desta palestra.
- combinação de detalhes sobre a biblioteca C e a ABI em uso
- arm-foo-none-eabi, cadeia de ferramentas bare-metal visando a arquitetura ARM, do fornecedor foo
- arm-unknown-linux-gnueabi, cadeia de ferramentas Linux visando a arquitetura ARM, usando a EABI ABI e a biblioteca glibc C, de um fornecedor desconhecido
- armeb-linux-uclibcgnueabi, cadeia de ferramentas Linux visando a arquitetura big-endian ARM, usando a EABI ABI e a biblioteca uClibc C
- mips-img-linux-gnu, conjunto de ferramentas Linux visando a arquitetura MIPS, usando a biblioteca glibc C, fornecida pela Imagination Technologies.

Dois valores principais para:

- nenhum para cadeias de ferramentas bare-metal
- Usado para desenvolvimento sem um sistema operacional
- A biblioteca C usada geralmente é newlib
- Fornece serviços de biblioteca C que não requerem um sistema operacional
- Permite fornecer chamadas básicas de sistema para alvos de hardware específicos
- Pode ser usado para construir carregadores de inicialização ou o kernel Linux, não pode construir código de espaço de usuário Linux para cadeias de ferramentas Linux
- Usado para desenvolvimento com um sistema operacional Linux
- Escolha de bibliotecas C específicas do Linux: glibc, uclibc, musl
- Suporta chamadas de sistema Linux
- Pode ser usado para construir código de espaço de usuário do Linux, mas também código bare-metal, como carregadores de inicialização ou o próprio kernel

## Componentes

Existem quatro componentes principais em uma cadeia de ferramentas de compilação cruzada do Linux

1. binutils
2. gcc
3. Cabeçalhos do kernel do Linux
4. Biblioteca C

Além disso, algumas dependências são necessárias para construir o próprio gcc.

Principais ferramentas

- ld, o vinculador. Vincula vários arquivos de objeto em uma biblioteca compartilhada, um executável ou outro arquivo de objeto. Como o montador. Recebe código assembly específico da arquitetura em forma de texto e produz um arquivo de objeto correspondente com código binário.
- Ferramentas de depuração/análise e outras ferramentas
- addr2line, ar, c++filt, gold, gprof, nm, objcopy, objdump, ranlib, readelf, size, strings, strip (Precisa ser configurado)

Para construir uma biblioteca C, são necessários os cabeçalhos do kernel do Linux: definições de números de chamada do sistema, vários tipos de estrutura e definições.

- Qual versão dos cabeçalhos do kernel deve ser usada em uma cadeia de ferramentas?
- A ABI do kernel para o espaço do usuário é compatível com versões anteriores.
- Portanto, a versão do kernel usada para os cabeçalhos do kernel deve ser a mesma versão ou mais antiga que a versão do kernel em execução no sistema de destino.
- Caso contrário, a biblioteca C pode usar chamadas de sistema que não são fornecidas pelo kernel.