

Atividades da semana 10

Baltazar Alic Borges da Silva – 11711EMT022

01)

Cross-compiling toolchain é um conjunto de ferramentas que permite construir código-fonte em código binário para uma plataforma de destino diferente daquela em que a compilação ocorre para diferentes arquiteturas de CPU, ABIs diferentes, sistema operacional diferente, bibliotecas C diferentes. As três máquinas envolvidas no processo de construção são a máquina de construção, onde a construção ocorre, a máquina host, onde ocorre a execução e a máquina de destino, para a qual os programas geram código.

Autoconf define o conceito de definições do sistema, representadas como tuplas. Por padrão, autoconf é configurado para a máquina atual, uma definição de sistema descreve um sistema: arquitetura de CPU, sistema operacional, fornecedor, ABI, biblioteca C,

As definições de sistema podem ser organizados como: <arch>-<vendor>-<os>-<libc/abi>, formulário completo ou <arch>-<os>-<libc/abi> de forma que:

<arch>, a arquitetura da CPU: arm, mips, powerpc, i386, i686, etc.

<vendor>, string de forma livre, ignorada pelo autoconf

<os>, o sistema operacional.

<libc/abi>, combinação de detalhes sobre a biblioteca C e a ABI em uso

Existem quatro componentes principais em uma cadeia de ferramentas de compilação cruzada do Linux, são eles, binutils, gcc, Cabeçalhos do kernel do Linux, biblioteca C.

Para construir uma biblioteca C para linux, são necessários os cabeçalhos do kernel do Linux: definições de números de chamada do sistema, vários tipos de estrutura e definições. A ABI do kernel para o espaço do usuário deve ser compatível com versões anteriores. Portanto, a versão do kernel usada para os cabeçalhos do kernel deve ser a mesma versão ou anterior à 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.

Várias bibliotecas matemáticas são necessárias para construir o gcc. Eles são compilados para a máquina host, ou seja, não são necessários no destino. O mpfr contém cálculos de ponto flutuante de precisão múltipla, gmp, dependência de mpfr, mpc, para cálculo de números complexos.

O processo de compilação para uma cadeia de ferramentas de compilação cruzada regular do Linux é bastante fácil: primeiro, Construir binutils, Construa as dependências do gcc: mpfr, gmp, mpc, Instale os cabeçalhos do kernel Linux, Construir um gcc de primeiro estágio: sem suporte para uma biblioteca C, suporte apenas para vinculação estática, Construa a biblioteca C usando o primeiro estágio gcc, e por final construa o gcc final com biblioteca C e suporte para vinculação dinâmica.

2)

Construir uma cadeia de ferramentas do zero não é uma tarefa fácil, embora vários projetos muito bons, como “Cross Linux From Scratch”, já existam. Uma abordagem simplificada consiste em usar o crosstool-NG que vem com muitos scripts úteis conduzidos por um front-end. Para instalar o crosstool-NG:

```
$> git clone https://github.com/crosstool-ng/crosstool-ng.git
$> cd crosstool-ng
$> git checkout crosstool-ng-[latest_version] #
# Verifique as dependências do seu sistema de compilação. #
$> ./bootstrap
$> ./configure --enable-local
$> fazer
$> fazer instalação
```

Se as etapas anteriores forem bem-sucedidas, o leitor terá uma instalação funcional do crosstool-NG pronta na máquina de compilação. As etapas a seguir mostram como usar crosstool-NG para construir uma cadeia de ferramentas para o emulador QEMU:

```
$> ./ct-ng distclean
# Lista todas as configurações disponíveis
$> ./ct-ng list-samples
# Escolha o arquivo de configuração correto. Aqui o para o QEMU é escolhido
$> ./ct-ng arm-unknown-linux-gnueabi
# Em "Caminhos e opções diversas" desmarque "Renderizar a cadeia de ferramentas somente leitura"
$> ./ct-ng menuconfig
$> ./ct-ng compilação
```

Se o processo de compilação for bem-sucedido, a cadeia de ferramentas estará localizada em:

```
~/x-tools/arm-unknown-linux-gnueabi/bin
```

Nesta pasta, estão localizadas ferramentas como compilador, depurador, vinculador, todas renomeadas com a identidade da cadeia de ferramentas. Por exemplo, a versão da cadeia de ferramentas de 'ld' será:

```
~/x-tools/arm-unknown-linux-gnueabi/bin/arm-unknown-linux-gnueabi-ld
```

Nesta fase o usuário deve começar a se familiarizar com o novo ambiente. Modificando o PATH variável ambiental também seria benéfica:

```
PATH=${HOME}/x-tools/arm-unknown-linux-gnueabi/bin/:$PATH
```

Embora o procedimento acima seja o que os usuários do QEMU devem seguir, quem possui uma placa física pode passar pelo mesmo procedimento, certificando-se de:

1. Escolha o arquivo de configuração correto.
2. Defina a variável PATH de acordo com sua cadeia de ferramentas.