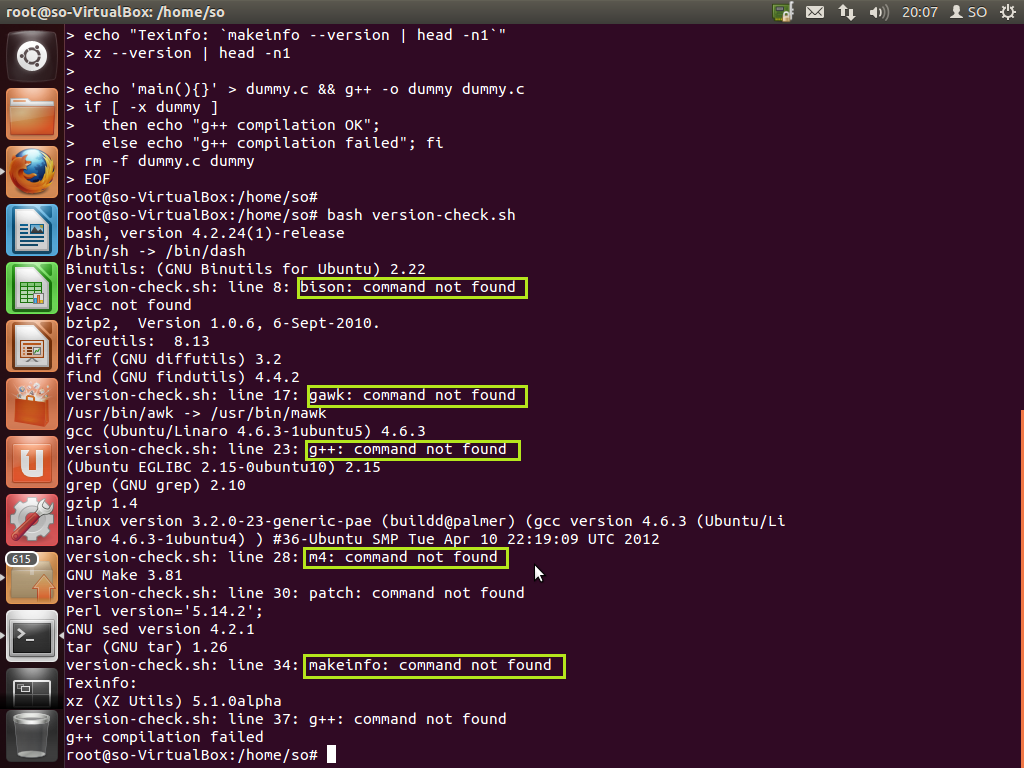
Relatório de trabalho:  
Instalação e configuração do sistema operacional Ubuntu 12.04 em Máquina virtual.  
#feito

Demos uma breve passagem no guia focalinux e no “prefácio”, “introdução” e “preparando uma nova partição” do guia linux from scratch.

Após instalação da máquina virtual foi seguido as instruções do guia linux from scrach para instalação e atualização de pacotes. Após a tentativa, foi indicado pelo terminal a ausência de pacotes como o “bison”, “g++”,”gawk”, “M4” e “makeinfo”.

#anexo.

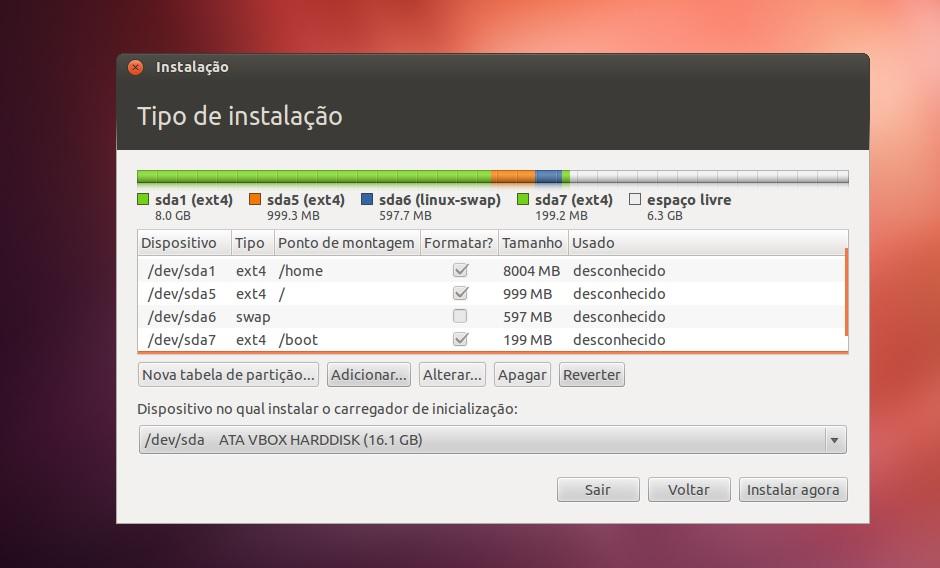


Após uma breve pesquisa encontramos um meio de instalar os patch’s de forma adequada.  
Já em modo root, instalamos os pacotes.   
  
  
apt-get install bison  
apt-get install gawk  
apt-get install textinfo (pacote para instalação do makeinfo)  
apt-get install g++

ao tentar instalar o “m4” apareceu um aviso no terminal como se o pacote já estivesse instalado.

Prosseguindo com o manual, chegando na parte de particionamento, verificamos que a máquina virtual que, foi criada de modo que se expandia à nossa necessidade, não nos dava suporte para poder criar uma nova partição já que não tinhamos como ampliar seu tamanho apenas tentando criar uma nova partição. A solução encontrada por hora foi criar uma nova máquina virtual com um espaço fixo de 15 GB (quase o dobro que o recomendado no virtual box) para que continuássemos nosso trabalho.

Com uma nova máquina virtual e mais espaço livre, foram alocados : 8 GB para o home, 1 GB para a raiz, 600 MB para o swap, 200 mb para o boot e foi deixado livre para a continuação do projeto 6,3 GB.

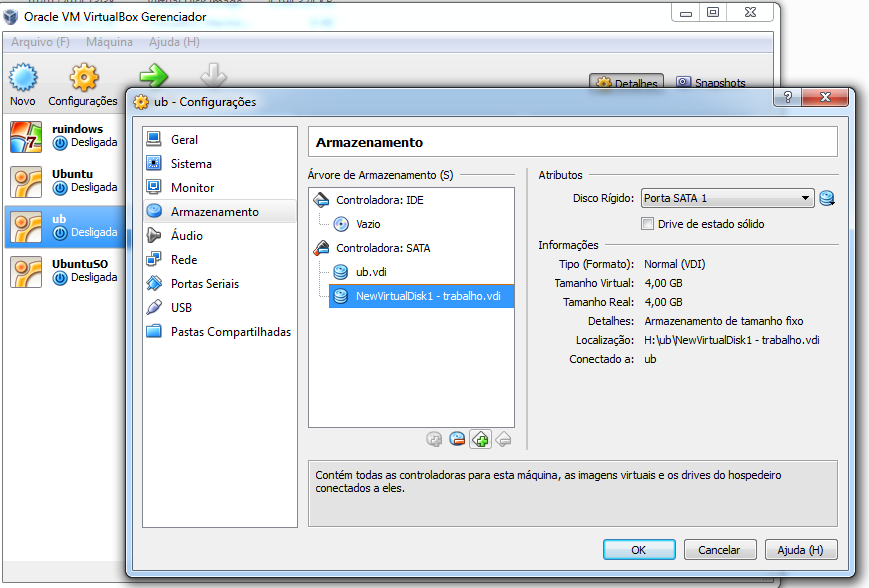


porém ao tentar instalar, foi detectado que existia memoria insuficiente em certas partições , então , mais uma vez, iremos tentar instalar outra vez , apos pesquisar o tamanho minimo nessesario para cada partição.

apos algumas falhas ao tentar redimensionar a maquina virtual , resolvemos utilizar o tamanho minimo recomendado (no momento da instalação),mesmo lendo alguns artigos onde se dizia possível instalá-lo em menos de 5gb, porém nenhum deles ensinava como , ou simplesmente recomendava algum outro SO (Linux) menor.

--------------------------------------------------------------------------------------------------------------------------

Problema parcialmente resolvido: Foi dedicado á máquina virtual uma nova partição de 4 GB afim de continuar os passos do guia.



(depois de tentativas a máquina virtual bugou - refazendo mas creio que não vai dar tempo de particionar ).

--------------------------------------------------------------------------------------------------------------------------

14/02/14  
  
Nova máquina virtual criada após um bug na transferencia de uma máquina virtual para outra.   
Após verificação e instalação dos pacotes(e compilação), foi adicionado uma nova partição, como segue no Linux from scratch, e em seguida o “/dev/sdb”, assim identificado na tabela do ubuntu, foi particionado em dois : principal e SWAP.

--------------------------------------------------------------------------------------------------------------------------

14/02/14

passo 3  
Criação e montagem de pastas de arquivos para a partição nova “/dev/sdb1” criada a partir da extensão adicionada.

passo 4

Download dos arquivos para compilação para o diretório “mnt/lfs”.

--------------------------------------------------------------------------------------------------------------------------

19/02/14

Passo 4.3

Criação do usuario LFS, criação da senha , criação e modificação de permissão das pastas source e tools para o LFS (usuario)  
  
passo 4.4

Preparando projeto , criando um novo usuario bash , criando uma instancia para o novo shell e instalando arquivos temporarios do “tools”.

//os passos entre os 2 eram conceitos , avisos, etc ...

Passo 5.4

Extração dos pacotes utilizando binutils 2.24 e utilizando o comando ‘xjfv’ extraimos tudo : o "x" significa , extrair os dados do arquivo;o "j" , significa extrair o arquivo bzip2. O "v" significa ,mostrar a atividade em “tempo real” e o "f" significa, utilize o arquivo que estou especificando.

Descompactação ok , preparando psicologicamente para copilar.

--------------------------------------------------------------------------------------------------------------------------

22/03/2014

(lembrar de falar que montamos um diretório somente para o LFS dentro da pasta “mnt”.)

- lembrar que as alterações feitas tem q ser feitas não em root, mas no novo usuário criado (lfs)  
para a compilação dos dados como se fossem já para o noso sistema operacional e o usuário seria o “root” daquele sistema.

Apos a desconpactação dos arquivos , alteramos o diretorio criado.   
Começa a compilação dos pcotes baixados. Logo sem seguida fizems a montagem de uma nova pasta , a “Binutils-build” para compilação da Binutils.

Com a pastas Binutils -build agora está montadas. – passo 5.4 pela metade.   
Pastas montadas. Em seguida , comando “make install “ e a pasta “tools” dentro de “mnt/lsf”.  
Após feito tudo isso, por questão de aconomia de espaço, as pastas “Binutils-2.24” e “Binutils -build” são deletadas já que os arquivos já foram executados.  
  
fim do passo 5.4 – complação da Binutils

--------------------------------------------------------------------------------------------------------------------------

Passo 5.5 – compilação do GCC  
  
Descompactação dos pacotes dentro do pacote GCC.   
**tar -Jxf ../mpfr-3.1.2.tar.xz**

**mv -v mpfr-3.1.2 mpfr**

**tar -Jxf ../gmp-5.1.3.tar.xz**

**mv -v gmp-5.1.3 gmp**

**tar -zxf ../mpc-1.0.2.tar.gz**

**mv -v mpc-1.0.2 mpc**

--------------------------------------------------------------------------------------------------------------------------

23/03/2014

Utilizando o commando “tar xvJf “ extraimos o linux – 3.13.2, após isso , utilizamos o comando

**make headers\_check**

**make INSTALL\_HDR\_PATH=dest headers\_install**

**cp -rv dest/include/\* /tools/include**

para testar e compilar o pacote , entao deletamos a pasta conforme é dito no livro e continuamos com o proximo pacote.

repetindo a operação de extração para glibc – 2.19 para extrair, é feita checagem para poder compilar , como é recomendado fazer a compilação fora da pasta source criamos uma pasta, preparamos para compilar igual como ja foi feito anteriormente para outro pacote utilizando o comando :

**../glibc-2.19/configure \**

**--prefix=/tools \**

**--host=$LFS\_TGT \**

**--build=$(../glibc-2.19/scripts/config.guess) \**

**--disable-profile \**

**--enable-kernel=2.6.32 \**

**--with-headers=/tools/include \**

**libc\_cv\_forced\_unwind=yes \**

**libc\_cv\_ctors\_header=yes \**

**libc\_cv\_c\_cleanup=yes**

e entao compilamos , e instalamos e excluimos a pasta conforme o livro seguindo para o proximo pacote.

Mesmos passos foram seguidos para Libstdc++ - 4.8.2

# 5.9. Binutils-2.24 - Passo 2

usando comando tar xjvf extraimos e fizemos o resto dos procedimentos ja citados em outros pacotes para extrair , compilar e instalar . peparando ,entretanto , o binutils antes de compilar com o comando :

**CC=$LFS\_TGT-gcc \**

**AR=$LFS\_TGT-ar \**

**RANLIB=$LFS\_TGT-ranlib \**

**../binutils-2.24/configure \**

**--prefix=/tools \**

**--disable-nls \**

**--with-lib-path=/tools/lib \**

**--with-sysroot**

Apos todo o processo , o livro pede para preparar os links para fase de re-ajuste(que ele diz ser o proximo capitulo)  
  
utilizando :

**make -C ld clean**

**make -C ld LIB\_PATH=/usr/lib:/lib**

**cp -v ld/ld-new /tools/bin**

ele basicamente diz para o “make” limpar a pasta indicada e refazer o caminho apontando para um diretorio final (e nao temporario).

# 5.10. GCC-4.8.2 - Pass 2