ESTR 3102

Gentoo Installation and Kernel Compilation

Helen Chan
SHB 118
hwchan@cse.cuhk.edu.hk
Office Hours: Fri 10am-12pm, or by appointment

Thanks to Dr. Q. Huang and Dr. T.Y. Wong for their slides :(
Outline

• Pre-installation

• Installation
  1. Boot from Gentoo installation CD
  2. Prepare the disk partitions
  3. Prepare the base system

• Kernel Compilation
  1. Get kernel source and configure kernel options
  2. Compile kernel
  3. Set up the new system

• Reference
Outline

• Pre-installation
  1. Boot from Gentoo installation CD
  2. Prepare the disk partitions
  3. Prepare the base system

• Kernel Compilation
  1. Get kernel source and configure kernel options
  2. Compile kernel
  3. Set up the new system

• Reference
Pre-installation

• Tools to host a virtual machine (VM)
  ▫ e.g. VirtualBox, VMware Player

• Gentoo installation image (iso)

• Enable VT-x for Intel CPU
  ▫ An option available in BIOS (if CPU supports VT-x)
    ▪ Disabled by default (as far as I know ... )
  ▫ Required for running 64-bit VMs
Pre-installation

1. Create a new VM
   ▫ For VMware, choose “I will install the operating system later” at the first step
   ▫ Size of Hard Disk: no less than 12GB

2. Edit VM setting
   ▫ Select the Gentoo CD image for CD-drive
Outline

• Pre-installation

• Installation
  1. Boot from Gentoo installation CD
  2. Prepare the disk partitions
  3. Prepare the base system

• Kernel Compilation
  1. Get kernel source and configure kernel options
  2. Compile kernel
  3. Set up the new system

• Reference
Step 1. Boot from Gentoo CD

• Boot VM from CD
  ▫ Press “Enter” when the VM boots up
  ▫ Press “Enter” to use default keyboard setting
Step 1. Boot from Gentoo CD

• Boot VM from CD
  □ After pressing “Enter” twice,
Step 2. Prepare Disk Partitions

• Show hard disk information

```
livecd ~ # fdisk -l /dev/sda

Disk /dev/sda: 12 GiB, 12884901888 bytes, 25165824 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
livecd ~ #
```

• Disk partition plan

<table>
<thead>
<tr>
<th>Partition</th>
<th>File System</th>
<th>Size</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sda1</td>
<td>(boot loader)</td>
<td>2MB</td>
<td>BIOS boot</td>
</tr>
<tr>
<td>/dev/sda2</td>
<td>ext2</td>
<td>128MB</td>
<td>Gentoo boot</td>
</tr>
<tr>
<td>/dev/sda3</td>
<td>(swap)</td>
<td>1024MB</td>
<td>Swap partition</td>
</tr>
<tr>
<td>/dev/sd4</td>
<td>ext4</td>
<td>All the rest</td>
<td>Root</td>
</tr>
</tbody>
</table>
Step 2. Prepare Disk Partitions

1. Create the partitions (first two partitions)

```
livecd # fdisk /dev/sda

Welcome to fdisk (util-linux 2.25.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x350651ef.

Command (m for help): n
Partition type
  p  primary (0 primary, 3 extended, 4 free)
  e  extended (contains logical partitions)
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-5165823, default 2048):
Last sector, +sectors or +size(K,M,G,T,P) (2048-5165823, default 5165823):
Created a new partition 1 of type 'Linux' and of size 2 MiB.

Command (m for help): n
Partition type
  p  primary (1 primary, 0 extended, 3 free)
  e  extended (contains logical partitions)
Select (default p): p
Partition number (2-4, default 2):
First sector (6144-25165823, default 6144):
Last sector, +sectors or +size(K,M,G,T,P) (6144-25165823, default 25165823):
+128M
Created a new partition 2 of type 'Linux' and of size 128 MiB.
```
Step 2. Prepare Disk Partitions

1. Create the partitions (last two partitions)

```
Command (m for help): n
Partition type
  p  primary (2 primary, 0 extended, 2 free)
  e  extended (container for logical partitions)
Select (default p): p
Partition number (1, 4, default 3):
First sector (268288-25165823, default 268288):
Last sector, +sectors or +size{k,M,G,T,P} (268288-25165823, default 25165823): +1024M
Created a new partition 3 of type 'Linux' and of size 1 GiB.
Command (m for help): n
Partition type
  p  primary (3 primary, 0 extended, 1 free)
  e  extended (container for logical partitions)
Select (default e): p
Selected partition 4
First sector (2365440-25165823, default 2365440):
Last sector, +sectors or +size{k,M,G,T,P} (2365440-25165823, default 25165823): +1024M
Created a new partition 4 of type 'Linux' and of size 10.9 GiB.
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

w (Confirm and write the partition table to disk !!)
Step 2. Prepare Disk Partitions

2. Mark the partitions

```
livecd ~ # fdisk /dev/sda
Welcome to fdisk (util-linux 2.25.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): t
Partition number (1-4, default 4): 1
Hex code (type L to list all codes): ef
Changed type of partition '1' to 'EFI (FAT-12/16/32)'.

Command (m for help): a
Partition number (1-4, default 4): 2
The bootable flag on partition '2' is enabled now.

Command (m for help): t
Partition number (1-4, default 4): 3
Hex code (type L to list all codes): 82
Changed type of partition 'Linux' to 'Linux swap / Solaris'.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```
Step 2. Prepare Disk Partitions

3. Outcome

livecd ~ # fdisk -l /dev/sda

Disk /dev/sda: 12 GiB, 12884901888 bytes, 25165824 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x350651ef

Device Boot Start End Sectors Size Id Type
/dev/sda1 * 2048 6143 4096 2M ef EFI (FAT-12/16/32)
/dev/sda2 6144 268287 262144 128M 83 Linux
/dev/sda3 268288 2365439 2097152 1G 82 Linux swap / Solaris
/dev/sda4 2365440 25165823 22800384 10.9G 83 Linux

(it is alright to be different)
Step 2. Prepare Disk Partitions

4. Create file systems
   - `# mkfs.ext2 /dev/sda2`
   - `# mkfs.ext4 /dev/sda4`
   - `# mkswap /dev/sda3`

5. Enable swap
   - `# swapon /dev/sda3`

6. Mount file systems
   - `# mount /dev/sda4 /mnt/gentoo`
   - `# mkdir /mnt/gentoo/boot`
   - `# mount /dev/sda2 /mnt/gentoo/boot`
Step 3. Prepare the Base System

1. Get the stage tarball
   - `# cd /mnt/gentoo`
   - `# wget -O stage3.tar.bz2 goo.gl/YxTj8P`
   - `# tar xjpf stage3.tar.bz2`
Step 3. Prepare the Base System

2. Configure compile options
   - `# nano /mnt/gentoo/etc/portage/make.conf`
   - Navigate using the “arrow keys”
   - Find “CFLAGS=-O2 -pipe”, and change it to `CFLAGS="-march=native -O2 -pipe"`
   - Save and exit by pressing “Ctrl+x”, “y” and “Enter”

3. Select mirrors
   - `# mirrorselect -i -o >> /mnt/gentoo/etc/portage/make.conf`
     - Select “Hong Kong: aditsu.net”
   - `# mirrorselect -i -r -o >> /mnt/gentoo/etc/portage/make.conf`
     - Select “Japan” or “Taiwan”
Step 3. Prepare the Base System

4. Set up DNS
   - # cp -L /etc/resolv.conf /mnt/gentoo/etc/

5. Mount the (special) file systems
   - # mount -t proc proc /mnt/gentoo/proc
   - # mount --rbind /sys /mnt/gentoo/sys
   - # mount --rbind /dev /mnt/gentoo/dev

6. Enter the new environment
   - # chroot /mnt/gentoo /bin/bash
   - # source /etc/profile
   - # export PS1="(chroot) $PS1"

(chroot) livedc #
Step 3. Prepare the Base System

7. Install portage snapshot
   - `# wget -O portage.tar.bz2 goo.gl/dDMGVk`
   - `# tar xjf portage.tar.bz2 -C /usr`
Outline

• Pre-installation

• Installation
  1. Boot from Gentoo installation CD
  2. Prepare the disk partitions
  3. Prepare the base system

• Kernel Compilation
  1. Get kernel source and configure kernel options
  2. Compile kernel
  3. Set up the new system

• Reference
Step 1. Get Src. and Config. Kernel

1. Get kernel source code
   - # emerge gentoo-sources

2. Configure kernel options
   - # cd /usr/src/linux
   - EITHER manually configure the options
     - # make menuconfig
     - (press “Enter” to jump to menu, press “Space” to change an option)
     - Configure based on Reference
   - OR download the configure file
     - For VMware,  # wget -O .config goo.gl/DYqCpy
     - For VirtualBox,  # wget -O .config goo.gl/NoHIHN
Step 2. Compile Kernel

1. Compile and install
   - # make
   - # make install
   - # make modules_install
Step 3. Set up the New System

1. Set up grub bootloader
   - `# emerge sys-boot/grub`
   - `# grub2-install /dev/sda`
   - `# grub2-mkconfig -o /boot/grub/grub.cfg`

2. Set up root password
   - `# passwd`
Step 3. Set up the New System

3. Set up /etc/fstab

- `# nano /etc/fstab`

<table>
<thead>
<tr>
<th></th>
<th>&lt;mountpoint&gt;</th>
<th>&lt;type&gt;</th>
<th>&lt;opts&gt;</th>
<th>&lt;dump/pass&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sda2</td>
<td>/boot</td>
<td>ext2</td>
<td>defaults,noatime</td>
<td>0 2</td>
</tr>
<tr>
<td>/dev/sda4</td>
<td>/</td>
<td>ext4</td>
<td>noatime</td>
<td>0 1</td>
</tr>
<tr>
<td>/dev/sda3</td>
<td>none</td>
<td>swap</td>
<td>sw</td>
<td>0 0</td>
</tr>
<tr>
<td>/dev/cdrom</td>
<td>/mnt/cdrom</td>
<td>auto</td>
<td>noauto,ro</td>
<td>0 0</td>
</tr>
<tr>
<td>/dev/fd0</td>
<td>/mnt/floppy</td>
<td>auto</td>
<td>noauto</td>
<td>0 0</td>
</tr>
</tbody>
</table>

- Press “Ctrl + x”, “y” and “Enter” to save and exit
Step 3. Set up the New System

4. Set up the network
   - `# cd /etc/init.d/
   - For VirtualBox,
     - `# ln -s net.lo net.enp0s3`
     - `# rc-update add net.enp0s3`
   - For VMware,
     - `# ln -s net.lo net.eno16777736`
     - `# rc-update add net.eno16777736`
   - Reboot!
     - `# reboot`
   - Login as “root”
Outline

• Pre-installation

• Installation
  1. Boot from Gentoo installation CD
  2. Prepare the disk partitions
  3. Prepare the base system

• Kernel Compilation
  1. Get kernel source and configure kernel options
  2. Compile kernel
  3. Set up the new system

• Reference
Reference

• Gentoo Handbook

• Gentoo HK Mirror
  ▫ http://gentoo.aditsu.net:8000/
  ▫ “release” : CD images and stage tarballs
  ▫ “snapshots”: portage tarballs

• Kernel Options
  ▫ [VMware] https://forums.gentoo.org/viewtopic-t-961502.html
Appendix

• If you power off the machine before compiling the kernel, do the following steps before resuming the compilation

  ▪ **Installation:** Enable swap and mount file systems:
    Step 2.5-2.6
  ▪ **Installation:** Set up DNS, mount (special) file systems and enter the new environment:
    Step 3.4-3.6
Appendix

• For VMware,

  ▪ To get more time at boot to enter BIOS, add the following option to the end of “.vmx” file after poweroff
    - bios.bootDelay = "5000"
  ▪ This option tells the player to delay for 5 seconds before booting from the default device.
  ▪ This will be useful when hard disk is bootable but you want to boot from other devices, e.g. CD, removable drives