



Enabling Grids for E-science

Chroot development environments for multiplatform support

Ricardo Mendes
CERN/ADI, GD-ITR

www.eu-egee.org



- **Introduction**
- **Where to get**
- **How to use**

- **Why chroot environments:**
 - Developers require to support an increasing number of platforms, but do not necessarily have easy access to the necessary physical machines.
- **Chroot changes the root directory of a process, thus showing it a different (limited) filesystem**
- **Chroot shares the same resources as the 'hosting' OS**
- **Useful for testing and cross-platform development**
- **Normally if certain rules are followed provides some extra security to the running processes and its children**
- **For this to have any practical use to testing and development we need a image of an OS filesystem to chroot....**

- **Where to get this OS images...**
 - Can be obtained from several places
 - <https://twiki.cern.ch/twiki/bin/view/EGEE/ChrootEnvs>.
 - <https://twiki.cern.ch/twiki/bin/view/EGEE/ChrootEnvsCreation>.
 - *Provides images for SLC4, SLC5, SLC3 and debian and a quick install script. (Maintained by Akos Frohner and Remi Mollon)*
 - <http://cern.ch/vnode>. Provides 32 bit and 64 bit basic images that normally are used in virtual machines but can also be easily used for chroot. *(Maintained by: Ricardo Mendes)*
 - *They contain the groups: Development Libraries, Development Tools, Editors, Java Development, System tools, etc.*
- **As an alternative you can also create this images yourself and customize them to your specific needs with libfsimage.**
 - `cvs -d :pserver:anonymous@isscvcs.cern.ch:/local/repos/xenvirt/ co libfsimage.`

- (1) Download the image from the first link in the previous slide.
- (2) Follow the instructions to configure the environment and execute *'/etc/init.d/chroot-envs start'*
- (3) *ssh root@slc5*
- Configuring the system to use a chrooted image is very easy using the chroot-envs script

- **User account**
 - useradd <username>
- **How to map directories in the hosting system inside the chrooted environment**
 - Mount devices:
 - mount -t <type> <device> <dir>
 - *Example: mount -t proc proc/ proc/*
 - Remount part of the file hierarchy somewhere else:
 - mount –bind olddir newdir
 - *Example: mount –bind /afs <chrootdir>/afs*
- **Graphical Interface**
 - Outside the chroot env and from an account that has access to X server do:
 - xhost + (Access is granted to everyone – Not safe)
 - Xhost +si:localuser:<username> or xhost +local:

- **Example: (The hard way):**
 - OS Host: Fedora Core 8 32bit
 - Chroot Env: SLC-5 32bit
 - Tar zxvf SL-5-i386.tar.gz -C /var/chroot/sl-5-x86/
 - Xhost +local: -> All local users have access to the X Server
 - Mount –bind /dev /var/chroot/sl-5-x86/dev
 - Mount –bind /afs /var/chroot/sl-5-x86/afs
 - Chroot /var/chroot/sl-5-x86/
 - Mount -t devpts dev/pts dev/pts
 - Mount -t proc proc/ proc/
 - Mount -t tmp tmp/ tmp/
 - Other devices may need to be mounted or binded for your specific case
 - Done...
 - When you finished using your chroot env just unmount everything..
- **The easy way – Chroot-envs script**

- **Problems that may appear:**
 - Xlib: connection to “:0.0” refused by server
 - Possible resolution: add user to xhost access control list
 - Reason: get_pty: not enough ptys
 - Possible resolution: devpts not mounted or you reached the maximum pts defined in your system.
 - Check kernel.pty.max and kernel.pty.max if they match increase kernel.pty.max.
 - Some other error:
 - Possible resolution: make sure you mounted all the necessary devices (tmp, sys, dev, proc, devpts , etc)

QUESTIONS?