

Linux 发展史和 **Linux** 应用

主讲: wisefox

created by openoffice.org

一, *Linux* 发展史

- ◆ Linux ? Unix ? Windows ? FreeBSD ?
- ◆ Linux 是对 **POSIX** 标准的一个实现, 并增加了对 **SYSV** 以及 **BSD** 的扩展.

Linux 发展史——企鹅的诞生

◆ 1990 年，赫尔辛基大学，计算机系硕士 Linus Torvalds

◆ Unix

◆ Minix

◆ 2 个月后完成了基本文件系统

Linux 发展史——企鹅的诞生

◆ 重要的决定

- ◆ 个人开发 集天下英才
- ◆ 源码开放，允许自由分发 / 拷贝
- ◆ 世界各地的 **Minix** 爱好者、骇客，提出改进建议，打补丁，并且为它开发应用程序。

Linux 发展史——企鹅的诞生

◆ Linux 名字的由来

- ◆ ftp 管理员认为这是 Linus 的 Minix，因而就建了一个叫做 Linux 的目录来存放这些文件。
- ◆ 1991 年 8 月下旬，Linux0.01 发布
- ◆ 1991 年 10 月 5 日，Linux0.02 发布
- ◆ 1992 年，Linux0.95 发布

Linux 发展史——企鹅的诞生

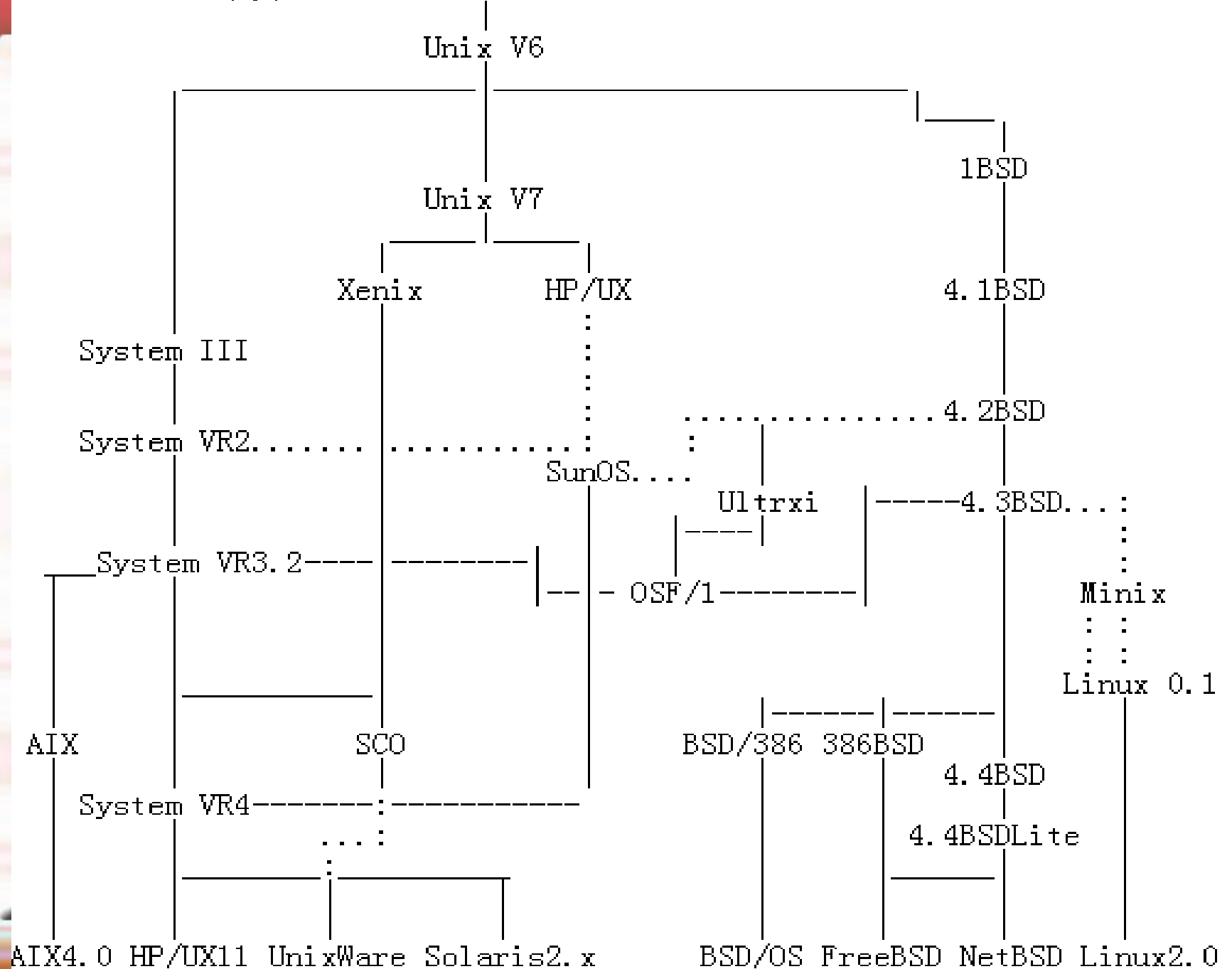
- ◆ Linux 的吉祥物
- ◆ 1994 年，Torvalds 终于推出了 Linux 的正式的 1.0 版本。此时这个操作系统已经变得比较稳定并且拥有很多的高级特性了：支持抢先多任务、对称多处理和完全的 POSIX 兼容性等。
- ◆ Torvalds 在 Linux 的联机文档里这样描述到：“它是一只讨人喜欢的、让人一见就想拥抱的、刚刚饱餐了一顿鲱鱼的企鹅”。

What is Linux?

- ◆ Linux is a clone of the operating system Unix, written from scratch by Linus Torvalds with assistance from a loosely-knit team of hackers across the Net. It aims towards POSIX and **Single UNIX Specification** compliance.
- ◆ It has all the features you would expect in a modern fully-fledged Unix, including true multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables, proper memory management, and multistack networking including IPv4 and IPv6.

- ◆ Although originally developed first for 32-bit x86-based PCs (386 or higher), today Linux also runs on (at least) the **Compaq Alpha AXP**, Sun **SPARC** and **UltraSPARC**, **Motorola 68000**, **PowerPC**, **PowerPC64**, **ARM**, **Hitachi SuperH**, **IBM S/390**, **MIPS**, **HP PA-RISC**, **Intel IA-64**, **DEC VAX**, **AMD x86-64**, **AXIS CRIS**, and **Renesas M32R** architectures.
- ◆ Linux is easily portable to most general-purpose 32- or 64-bit architectures as long as they have a paged memory management unit (PMMU) and a port of the GNU C compiler (gcc) (part of **The GNU Compiler Collection, GCC**). Linux has also been ported to a number of architectures without a PMMU, although functionality is then obviously somewhat limited. See the **μClinux project** for more info.

贝尔实验室Unix V1 - V5



Linux 发展史——茁壮成长

◆ Linux 与 GNU/GPL → Linux/GNU

- ◆ GNU: GNU's Not Unix.
- ◆ GPL: General Public License.
- ◆ FSF: Free Software Foundation
- ◆ POSIX: Portable Operating System Interface
- ◆ 任何人都可以自由地拷贝、使用、修改（重新发布时必须遵从 **GPL**）和出售 **Linux**（但软件不可收费）。

Linux 发展史——茁壮成长

- ◆ Linux 的优势
 - ◆ “free”
 - ◆ 自由
 - ◆ 开源
 - ◆ 免费
 - ◆ 高度可定制
 - ◆ Unix like
 - ◆ 安全性
 - ◆ 众多的资源
 - ◆ 学习的最佳途径

Linux 发展史——茁壮成长

◆ 支持 Linux 的厂商

◆ IBM

◆ HP

◆ Oracle

◆ Borland

◆ Intel

◆ AMD...

Linux 发展史——茁壮成长

◆ 优秀的开放源码软件

- Bash
- Openssl
- Perl
- Gcc
- Xorg
- Bind
- SendMail

- Apache
- Proftpd
- Samba
- Mozilla
- MySQL
- OpenOffice
- KDE/GNOME

Linux 发展史—— *Distribution*

- ◆ 1，按运营模式划分
 - ◆ 商业模式：Redhat，Suse...
 - ◆ 非商业模式：Debian，Gentoo...
- ◆ 2，按安装包方式划分
 - ◆ RPM：Redhat，Suse...
 - ◆ DEB：Debian，Ubuntu 等
 - ◆ Source：gentoo，LFS

Linux 发展史—— *Distribution*

◆ 主要发行版本概述

◆ Redhat

- ◆ 强大的技术实力
- ◆ 优秀的技术支持
- ◆ 简单易用
- ◆ 良好的软件集成和系统框架
- ◆ 基于 rpm 的包管理系统便于管理，同时也带来了严重的依赖问题。Yum 更新系统很好的解决了依赖问题，但软件库不是特别丰富，且有版本限制。
- ◆ 因为对源文件修改造成有些软件编译困难。

Linux 发展史—— *Distribution*

◆ Mandriva

- ◆ 最漂亮的 Linux ，好看的用户界面
- ◆ 基于 rpm 的包管理系统跟 redhat 有类似的问题。
但 urpm 使用了和 yum 同样的方法在一定程度上解决了该问题，也存在版本间限制。
- ◆ 技术实力欠佳。

Linux 发展史—— *Distribution*

◆ Debian

- ◆ 纯正的 Linux
- ◆ 基于 apt , deb 包管理系统, 拥有庞大的软件仓库, 依赖问题少。
- ◆ 简易的升级方法
- ◆ 良好的开发环境
- ◆ 配置较为复杂

Linux 发展史—— *Distribution*

◆ Gentoo

- ◆ 性能最优
- ◆ 高度定制
- ◆ 软件最新，最为丰富
- ◆ 基于 **emerge** 软件安装系统，可获得最优性能。
- ◆ 安装软件时耗费的时间过长。

Linux 发展史—— *Distribution*

- ◆ TurboLinux

- ◆ 良好的中文支持

- ◆ Redflag

- ◆ 较好的中文支持
- ◆ Windows like 界面
- ◆ 仿 windows 配置方法

- ◆ Suse 、新华、中标等。。。。

二, *Linux* 应用——认识 *Linux*

- ◆ 1, Linux 框架结构
 - ◆ 从系统启动角度介绍系统框架。
 - ◆ Click or type 。
- ◆ 2, Linux 文件系统结构
 - ◆ 按功能概述目录
 - ◆ 按物理结构概述文件系统

2.1 认识 *Linux* —— *Linux* 系统框架

◆ 2.1.1 按启动顺序概述系统框架

◆ 1, Kernel

- ◆ 版本号 2.6.14 “uname -a”
- ◆ 检测硬件
- ◆ 启动各系统功能
- ◆ 挂载根文件系统，调用 `init` 。

◆ 2 , init , rc level

- ◆ 加载驱动, 设置硬件参数 kmodules
- ◆ 检查硬件, 设置系统参数 network
- ◆ 根据 initlevel 批处理运行用户服务。
- ◆ Inittab, init.d, rc0 ~ 6, rcS
- ◆ “S02ku..., K03mo...”

2.1 认识 *Linux* —— *Linux* 系统框架

- ◆ 3 , 服务
 - ◆ Kudzu,Xserver,ftpd,sshd...
 - ◆ “man kudzu”

2.1 认识 *Linux* —— *Linux* 系统框架

- ◆ 4, 图形界面
 - ◆ Xserver
 - ◆ Xwindow
 - ◆ Kde,gnome,Xfce

2.1 认识 *Linux* —— *Linux* 系统框架

- ◆ Click or type

- ◆ 图形界面虽然精美，但不够方便
- ◆ 命令仍然广泛的被使用

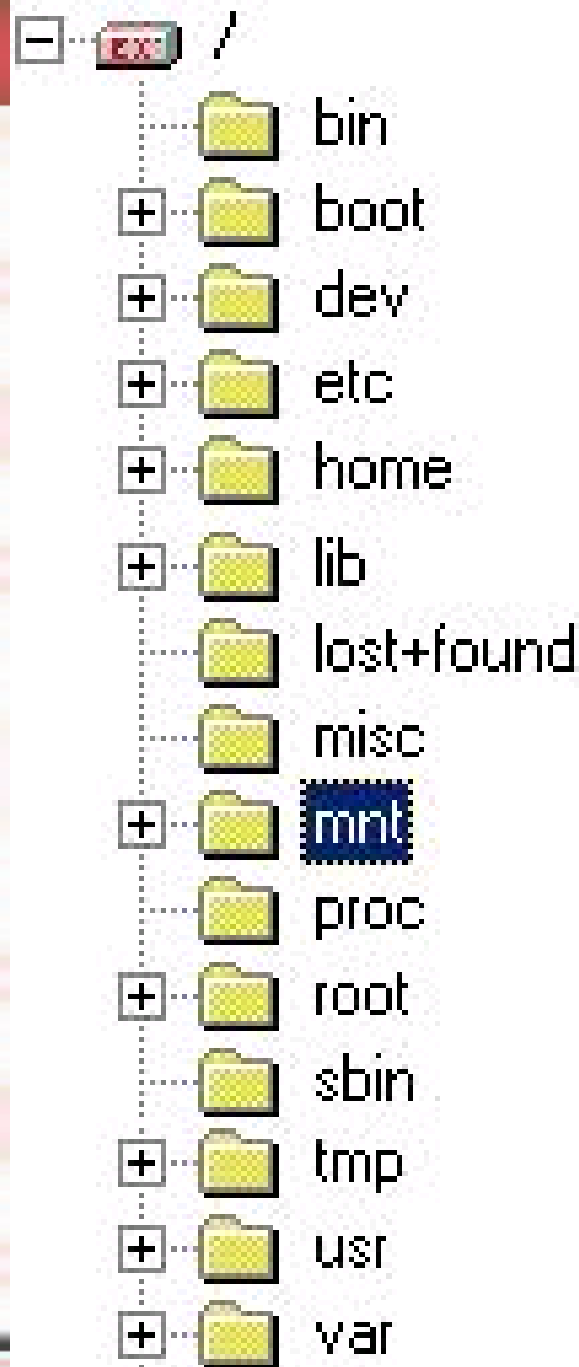
2.1 认识 *Linux* —— *Linux* 系统框架

◆ 2.1.2 Linux 文件系统（按功能划分）

- ◆ /boot : 引导文件目录
- ◆ /root 、 /home : 用户目录
- ◆ /bin 、 /sbin : 系统基本命令、可执行程序
- ◆ /etc : 全局配置文件目录
- ◆ /lib : 系统基本库、内核模块
- ◆ /var : 系统变量目录

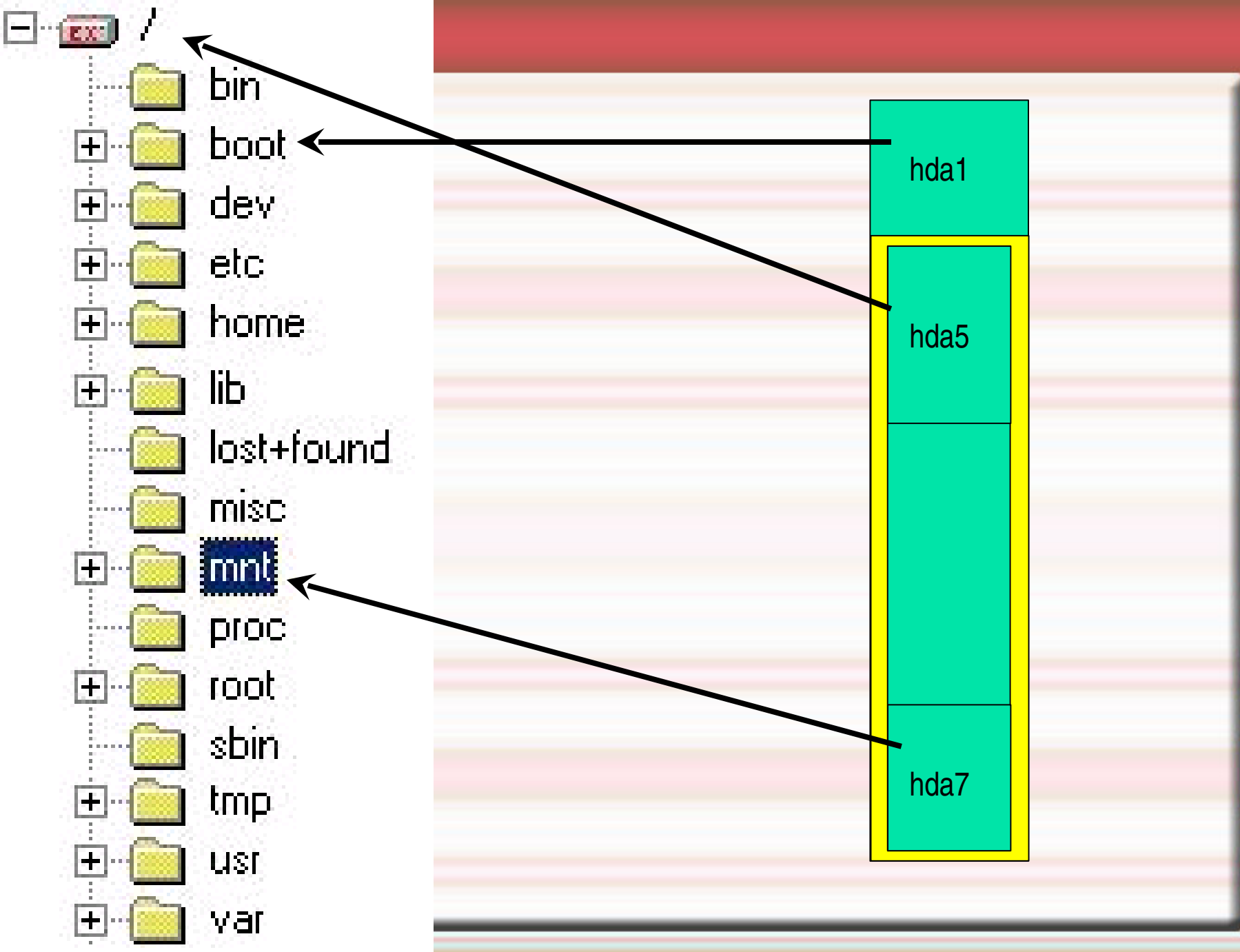
- ◆ /usr : 用户目录
 - ◆ /usr/bin 、 /usr/sbin : 应用程序, 高级命令
 - ◆ /usr/lib : 应用程序库
 - ◆ /usr/local : 自定义软件安装目录
 - ◆ /usr/share : 应用程序共享资源
 - ◆ /usr/X11R6 : Xserver 、 Xclient 目录

- ◆ **/proc** : 系统信息目录
- ◆ **/dev** : 设备目录
- ◆ **/mnt** : 挂载点



◆ 2.1.2 Linux 文件系统（按挂载点划分）

- ◆ 块设备：软硬盘， loop 设备…
- ◆ 设备名： fd ， hda1 ， sda1
- ◆ 挂载点：任意目录



Linux 应用——熟悉 *Linux*

◆ 2.2.1 桌面应用

◆ 日常应用软件及其对照

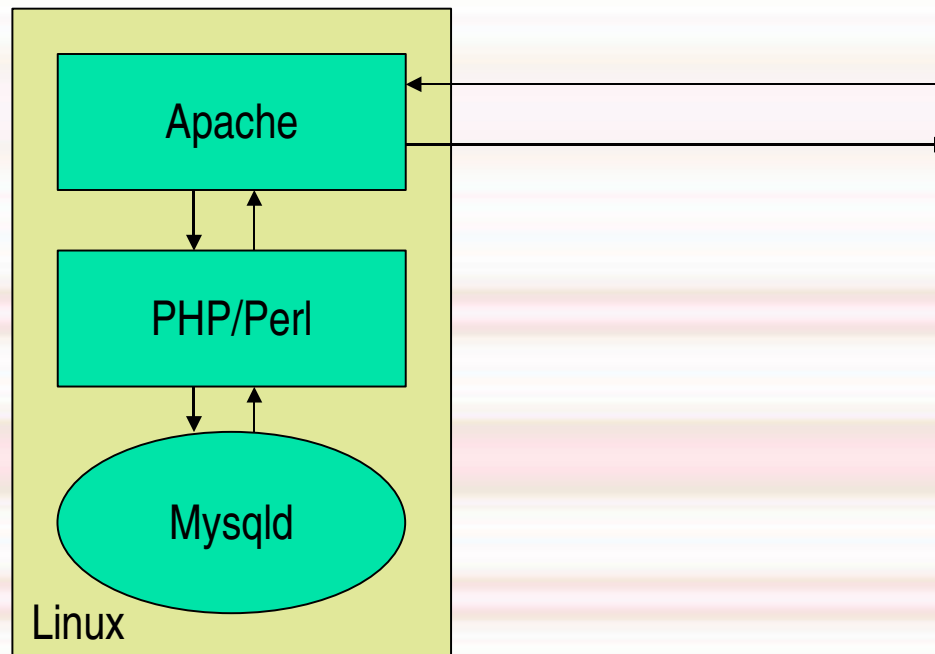
VC studio	Anjuta , eclipse	Gcc , gdb 后台
Realone	Realone , gxine	中文字幕问题
Winamp	Xmms	Midi 问题
Adobe reader	Adobe reader	
Flashget	D4x	有点难用
IE	Firefox , opera	非标准 javascript
leapftp	gftp	

紫光输入法	Fcitx	
Cterm	Qterm	
翻译软件	Stardict	
Office	金山 wps ， openoffice2	
Tencent QQ	openQ ， lumaqq	
UltraEdit	Vim ， Emacs	
绘图	gimp	
Outlook	Evolution,kmail	
Q3 ， d3 ， wolf ， heros3	Q3 ， d3 ， wolf ， heros3	

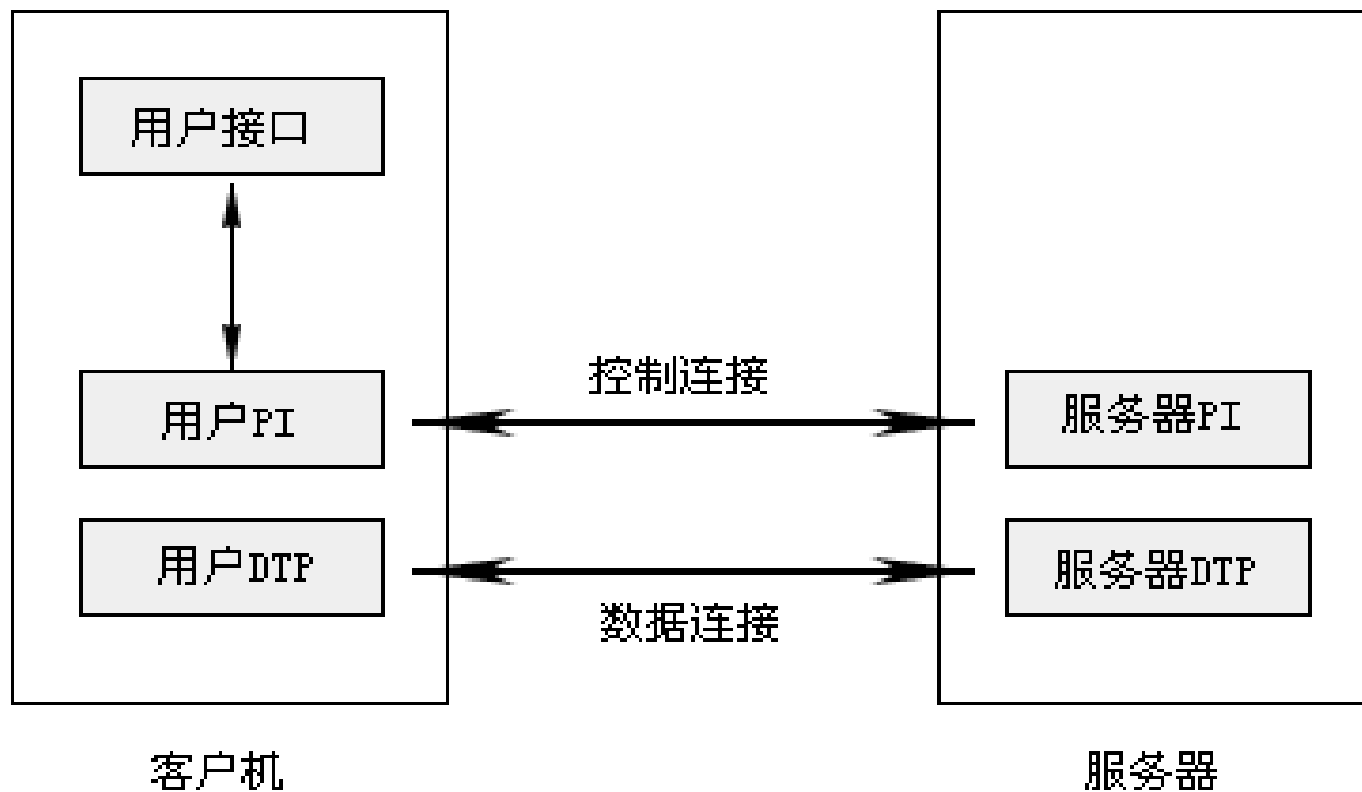
2.2.2 *Linux* 服务器应用

◆ 1 , LAMP

◆ Linux+Apache+Mysql+PHP/Perl

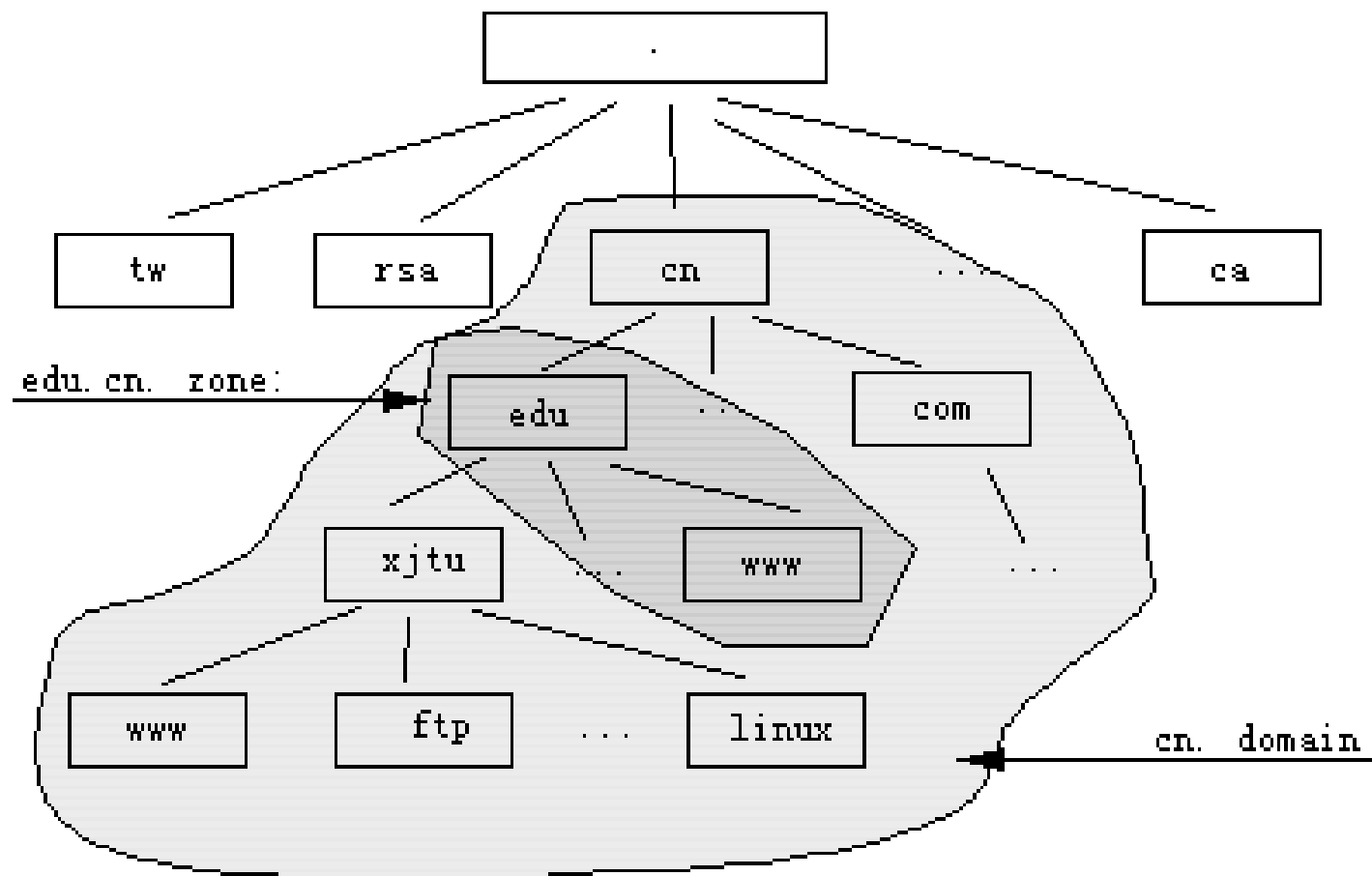


2,proftpd



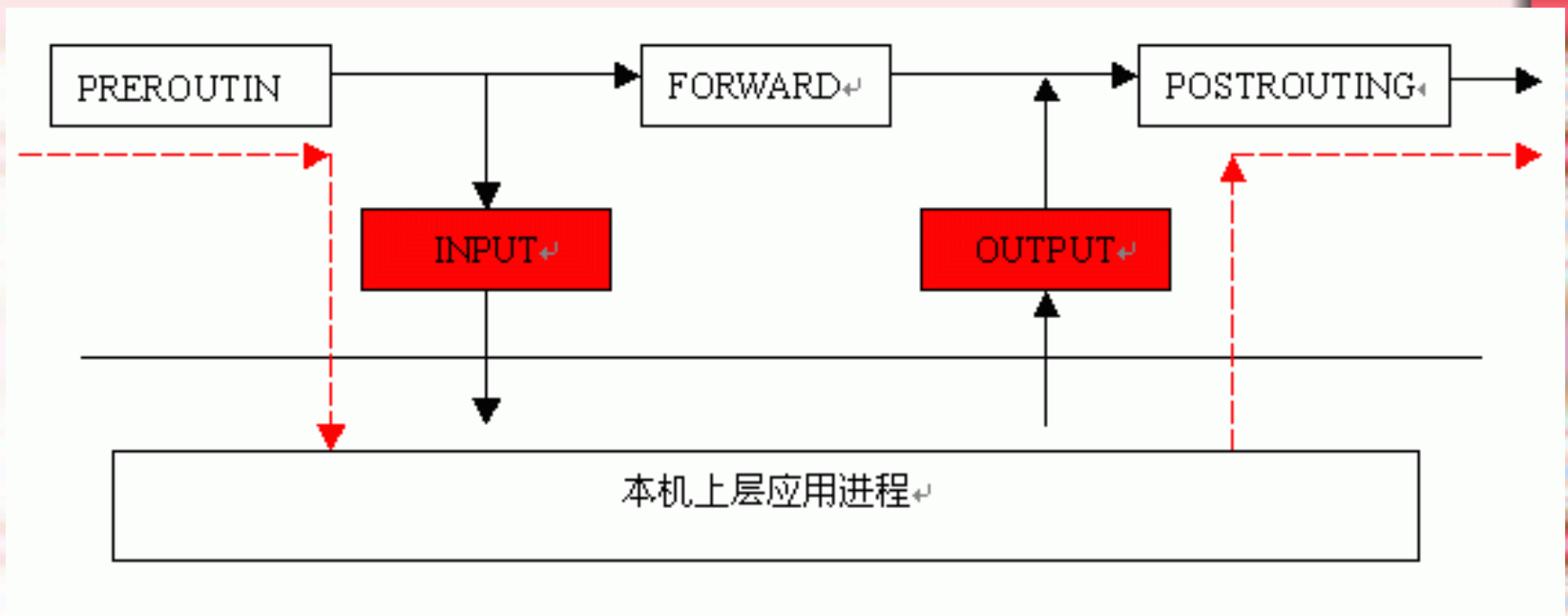
主动连接和被动连接

3,DNS



nslookup

4, *iptables* 应用举例



4, *iptables* 应用举例

◆ (1) 防火墙

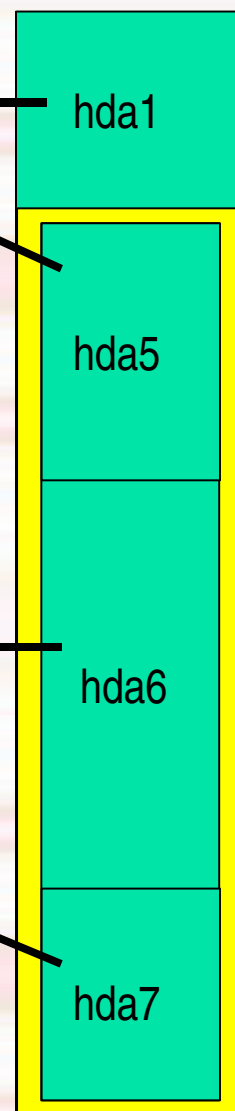
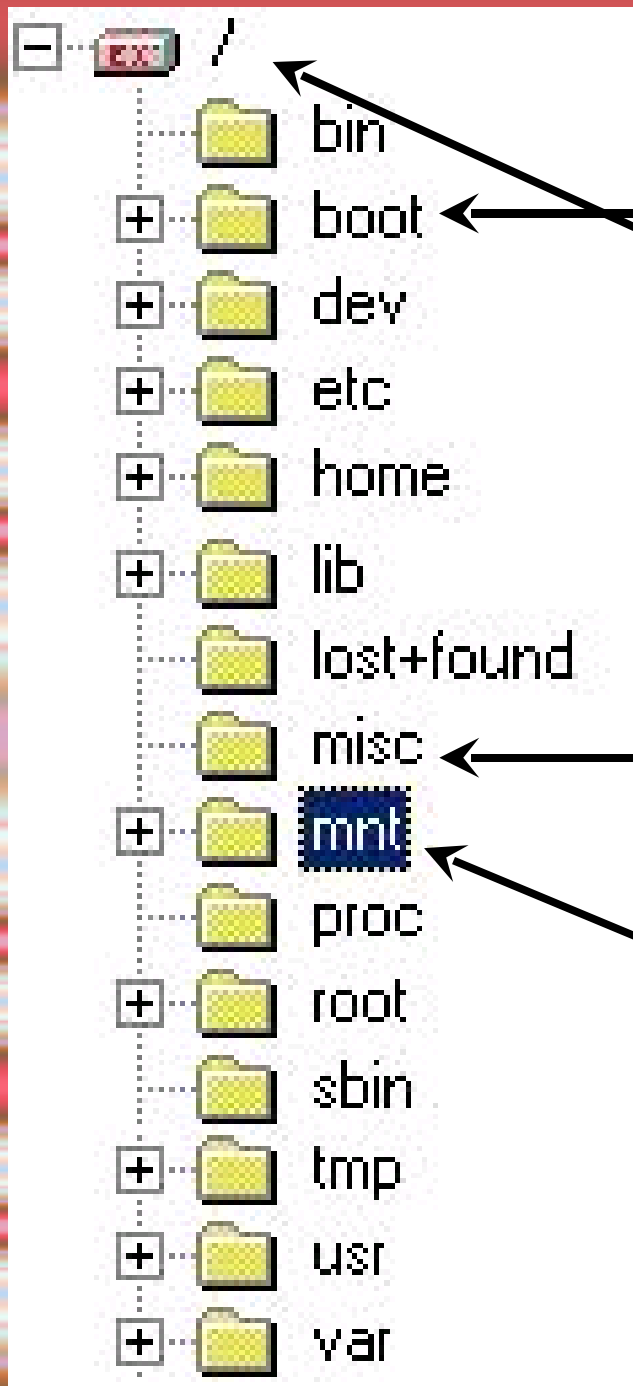
◆ `iptables -A INPUT -p tcp --syn \`
`--destination-port 22 -j DROP`

◆ (2) NAT 转发

◆ `iptables -t nat -A POSTROUTING -o eth0 \`
`-s 192.168.0.0/24 -j MASQUERADE`

5, chroot 应用举例

- ◆ (1) 服务器安全
 - ◆ 尽量减少服务器端损失
- ◆ (2) 测试开发
 - ◆ 目标测试系统



chroot /misc

仅限于当前 shell

Linux 应用——程序开发

◆ 2.3.1 认识 Gcc

- ◆ GCC, the GNU Compiler Collection, includes front ends for C, C++, Objective-C, Fortran, Java, and Ada, as well as libraries for these languages (libstdc++, libgccj,...).
- ◆ 目前版本
 - ◆ Gcc3.4 —— 3.4.4
 - ◆ Gcc3.3 —— 3.3.6
 - ◆ Gcc4.0 —— 4.0.2

认识 *Gcc* —— *gcc* 基本语法

- ◆ 基本编译

- ◆ `#gcc -o test test.c`

- ◆ 调用共享库

- ◆ `#gcc -o test -lpthread -lm test.c`

- ◆ `#gcc -o test -L/path/to/ -lyourlib test.c`

- ◆ `#gcc -o test -I/path/to/include test.c`

认识 *Gcc* —— *gcc* 基本参数

- ◆ 优化参数、调试参数
 - ◆ `#gcc -g -O2 -o test test.c`
- ◆ 平台参数
 - ◆ `#gcc -O3 -mtune=pentium4 -o test test.c`
- ◆ CFLAGS , CC
 - ◆ `#CFLAGS='-O3 -mtune=pentium4' CC -o test test.c`

认识 *Gcc* —— *g++*

- ◆ G++ 用于编译 C++ 程序
- ◆ CXX 、 CXXFLAGS

认识 **Gcc** —— 软件安装

- ◆ 0, # 获得源码, 解压
- ◆ 1, # ./configure --prefix=/usr/local/test
- ◆ 2, # make
- ◆ 3, # make install

2.3.2 集成开发环境

- ◆ Anjuta
- ◆ Eclipse + cdt