

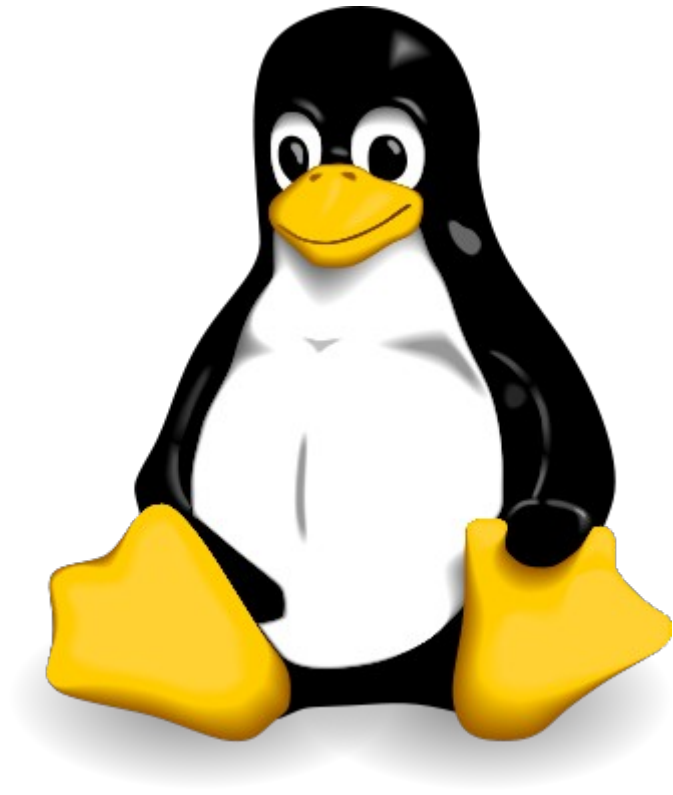
# Interact with Linux

Deep [in]security  
knowledge

# GNU/Linux?

- Ever heard linux?
- Ever heard GNU?
- So, Please explain GNU/Linux?

Please, Met Tux!



# Security

- Why do we need security?
- Securing what?
- How to do that?



# Physical Security

Secure the environment



IS IT SAFE?



# {Physical Security

- Room Security (lock)
- CPU security (lock case, no cdrom... else)\_
- Bios Security

Disallow booting from floppy/cdrom/usb drive and network

# NO Ctrl+Alt+Del

- Pressing Ctrl Alt-Delete will shutdown the system
- Prevent machine from being rebooted
- Edit /etc/inittab and comment out the following:

```
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

# BootLoader

- Hacking grub/lilo
- Bypass , using “single rw  
init=/bin/bash”
- Boot using another Rescue CD
  - Take out passwd&shadow file
  - Add entry
  - Remove it

# Securing BootLoader

- Password for grub
  - GRUB configuration files is `/boot/grub/menu.lst`
  - Add `timeout=00` – do not show menu
  - Generate md5 password by running:  
`Grub-md5-crypt`
  - Add `password –md5 <md5 password>`

# User/Account Security

with your credentials



# Password

- Most IMPORTANT – often neglected
- Set the right values in `/etc/login.def`
  - Change `PASS_MIN_LEN` 5
  - To `PASS_MIN_LEN` 8
  - Change `PASS_MAX_DAY` 99999
  - To `PASS_MAX_DAY` 63



# Root Account

- Root is GOD in unix machine
- Never login as root on your server
- Set login time out for root account

Set TMOUT to the time in seconds

– edit /etc/profile and set:

```
TMOUT = 7200
```

# File & Resources Security restriction



# File Security

- Chmod
- Chown
- Chgrp
- Chattr

# Fork bomb

- `:(){ :|& };:`
- `/etc/security/limits.conf` – important to set limits, to prevent denial of service attacks
- Add/Change the lines in `limits.conf` to read:
  - \* `hard core 0 # prohibit core files`
  - \* `hard rrs 5000 # memory usage 5M`
  - \* `hard nproc 20 # number of process`
- Edit `/etc/pam.d/login` and add  
`session required /lib/security/pam_limits.so`

# Network Security

safe your way home



# Network Security

- Firewall
- ACL
- Network security tools
- IPS/IDS

# Ports?>

- Close all unneeded applications
  - “netstat -anp” or lsof to see what’s open
  - Ntsysv, linuxconf to shut down
- Update-rc.d list apps from /etc/init.d

# Protocol

- Avoid using plaintext protocol  
telnet, ftp, http
- Sniffer will Do the best  
Wireshark (formerly ethereal), ettercap,  
tcpdump, e.t.c
- Use (relatively) Secure Protocol  
Ssh, sftp, https



# Firewall

- Packet filtering
  - Deployed on routers to allow only authorized network traffic to the extent possible
- Application proxies
  - An application program that runs on a firewall system between two networks
  - Application proxies make more complex filtering and access control decision
- Dynamic packet filtering
  - Stateful inspection filtering allows both complex combinations of payload and context filtering decision

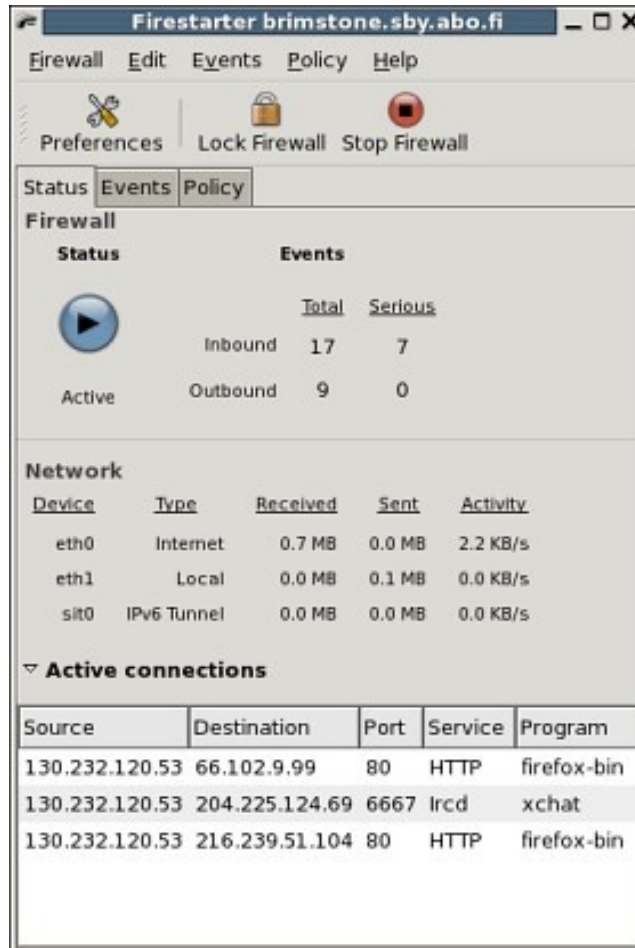
# Iptables?

- A sample rule to drop all incoming traffic from a specific IP

```
iptables - I INPUT - i eth0 - s 192.168.0.2 - j DROP
```

- iptables - is the command
- -I INPUT – insert into INPUT chain
- -i eth0 – input interface
- -s 192.168.0.2 – source IP address
- -j DROP - target

# Firestarter?



<http://www.fs-security.com> - A Modern Linux Firewall

# Network Security tools

- Nmap
- Nessus
- Thc amap

# IDS/IPS

- Snort – Network intrusion detection system
  - Performs real-time traffic analysis and packet logging on IP networks
  - It can perform protocol analysis, content searching/matching and can be used to detect a variety of attacks and probes, such as buffer overflows, stealth port scans, CGI attacks, SMB probes, and OS fingerprinting
  - Snort uses a flexible rules language to describe traffic that it should collect or pass
  - For implementation: [www.snort.org/docs/](http://www.snort.org/docs/)

# Portsentry

- portsentry – protects against portscan
  - runs as a daemon on the protected host, it listens to TCP/UDP ports and will block scanning hosts from connecting to server
  - For implementation:  
<http://sourceforge.net/projects/sentrytools/>

# Auditing & Logging

Watch for a foot print



# Anti Logging

- UNSET HISTFILE
- History -c
- Rm -rf .bash\_history

# Logging

- Lastlog
- Last
- History



# Shell Logging

- bash shell stores up to 500 old commands in the `~/.bash_history` file
- Every user will have this file `.bash_history`
- Reducing the number of old commands the `.bash_history` file can hold will protect against storing passwords typed on the command line
- Set `HISTFILESIZE` and `HISTSIZE` lines in the `/etc/profile` to:  
    `HISTFILESIZE = 20`  
    `HISTSIZE = 20`

# Auditing

- Bastille
- Chkrootkit
- Rkhunter
- CIS manually

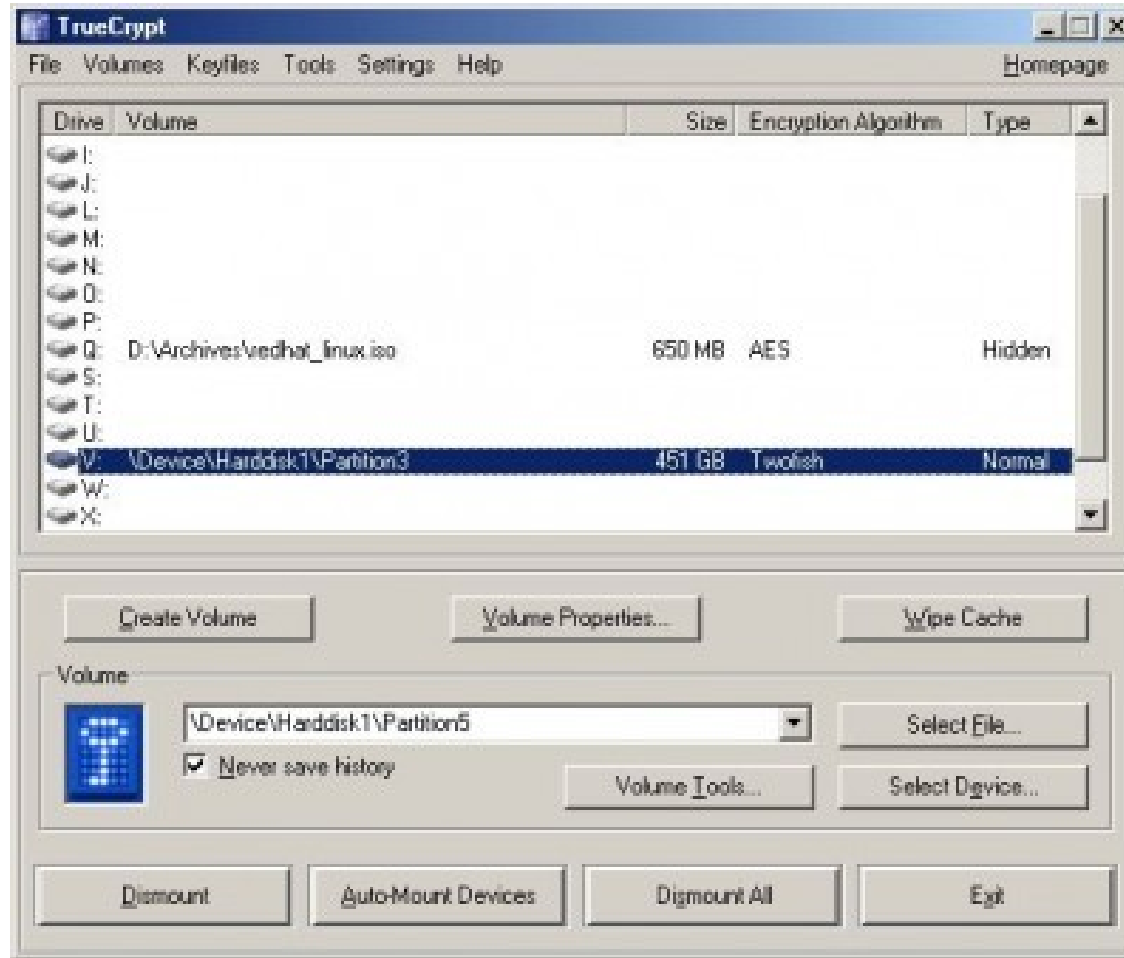


# Encryption & Backup

Your data are so  
expensive



# Encryption



<http://www.truecrypt.org>; Free open-source disk encryption software for Windows Vista/XP, Mac OS and Linux

# Backup

- Hardware failure like disk breaking
- accidentally deleting wrong file
- computer being stolen
- ?

# Kernel Security

Tune your linux



# Kernel tunable parameters

- Parameters can be set in `/etc/sysctl.conf`
- Prevent system from responding to ping
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.icmp.echo.ignore.all = 1`
  - restart the network by typing  
`/etc/init.d/network restart`
- Refuse responding to broadcast request
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.icmp.echo.ignore.broadcasts = 1`

continued...

# Kernel tunable parameters cont.

- Disable IP source routing
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.conf.all.accept_source_route = 0`
  - restart the network by typing  
`/etc/init.d/network restart`
- Enable TCP SYN Cookie Protection
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.tcp_syncookies = 1`
  - restart the network by typing  
`/etc/init.d/network restart`

continued...

# Kernel tunable parameters cont.

- Disable ICMP redirect acceptance
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.conf.all.accept_redirects = 0`
  - restart the network by typing  
`/etc/init.d/network restart`
- Enable always-defragging protection
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.ip_always_defrag = 1`
  - restart the network by typing  
`/etc/init.d/network restart`

continued...

# Kernel tunable parameters cont.

- Enable bad error message protection
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.icmp_ignore_bogus_error_responses = 1`
  - restart the network by typing  
`/etc/init.d/network restart`
- Enable IP spoofing protection
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.conf.all.rp_filter = 1`
  - restart the network by typing  
`/etc/init.d/network restart`

continued...

# Kernel tunable parameters cont.

- Log spoofed, source routed and redirected packets
  - edit `/etc/sysctl.conf` and add  
`net.ipv4.conf.all.log_martians = 1`
  - restart the network by typing  
`/etc/init.d/network restart`

# Advanced

- Chrooted
- Virtualized
- Kernel security;  
PIE+SSP/SELinux/grsec/PAX
- Hardened profile

# Reference

- “Securing A Host Machine”; -Raj Nagendra,William Zereneh
- “Basic Linux/System Security”; **Bill Stearns**
- **Linux Manual**
- “Linux Security Howto”; Kevin fenzi , Dave wreski