Final exam is Thursday, May 13 @ noon

- No and midterm in this class

(Cheat sheet next Wednesday)

- Lab 4 is up, due on [date]

Presentation due Thursday, April 13

(Final problem set due, Thursday, April 13)

- In-class review of papers today

- Quiz today

Announcements

Security - Linux Security (part 2)
Then problems (a bit obsolete or less secure)

/etc/hosts.allow + /etc/hosts.deny

list of access control in any service, treats hosts at

before allowing any TCP connection

TCP wrappers

- (0-day vulnerabilities)
- Patch management
- Limit applications

Avoid System hardening!
- Logging

- User/Password Management

- Symantec, McAfee

- Many commercial anti-virus programs: McAfee

- Some free ware, eg. Clam AV

- More worms than viruses historically

- Anti-virus Software

- Trouble a few powerful.

- Firewalls - perhaps.
Loggrings

- Generally run by ships
- Better choice is Skype - NC
- Sonorous + descriptions
- Rules engine is much more fragile

"supports loggrings via TC (which can be encrypted)"

- Loggrings via TC
Nessus - Security Scanners

Smart - Powerful (Red) IDS

Shadow of the Passwords
to trace the critical issues and changes in the database. What functions

Edocuments as systems hardening philosophy

Other tools worth mentioning...
- eCryptphen
- Loggin
- Chroot jail

- Minimal process running as root
  - Modularity: Postfix versus Sendmail

Application Security
(Adds complexity, but increases security)

So well "map" this directory to look

Like root, / as that the domain

Can't see anything else.

The system can access anything else in the

Then the domain should not hang

directory, eg. /etc/ftpdb. html

It's a process is only working to one

Chroot jail
less commonly used.

No "root takes all" issue; root is
change the system.
other accounts that can actually
System administration is done with
security policy only.
In MAC settings, root is used to
MAC is separate; consider root.

As we've said, Linux is based on DAC.
Mandatory Access Controls in Linux
Subject and objects

but adds what on tap.

- Doesn't change basic "PC in Linux"

- NSA's implementation of MAC for Linux

SELinux
and objects to be grouped.

- 2. Permission is denied if a permitted object is not expressly permitted subjects, permissions

Rule: 1. That which is not expressly

```
Dir
|-- parent
|   |-- get attr
|   |-- read
|   `-- search

Each category of object has a set
```
(called Type Enforcement)

Kerberos + SFTinux

#

add to global

gdm

Scand box

"Is user, role, domain"

Security context which is a context separate from Linux, needs a security context for user, subject, object.
Different "sandbox"

a process or the for a

Sometimes you'll need to create

Transition decisions

To exit, etc.

Can you read/write, etc.

Access decisions

Two types of decisions for analysis:
Other models in SELinux are possible.

- RBAC can be added.
- Multilevel security (Bell-LaPadula Model) can be added.
- Mandatory label enforcement via file system labelings.
Wicked Learning Curve

- System - Conroy - SecurityLevel
  - In Red Hat and Fedora, use:
  - http://www.freshy.com
  - Also GUI-based admin tools:

Also in /etc/security/sectrules:

These SE Linux policies are actually
Application as Covered
Root is still root, unless the
No RBAC or multi-level security

only works for a subset of

applications

primary goal is to restrict

More limited: (Ubuntu)
Rarely available for Linux

Much easier for administrators

Novell AppArmor