



Managing Red Hat Enterprise Linux 5

Daniel J Walsh

SELinux Lead Engineer

dwalsh@redhat.com

Karl MacMillan

kmacmill@redhat.com

Principal Software Engineer

Agenda

- 1) Introduction to SELinux Concepts
- 2) Policies and Configuration Files
- 3) Modified OS Commands
- 4) SELinux Utilities
- 5) Understanding Audit Messages
- 6) Managing File Labeling
- 7) Customizing Policy With Booleans
- 8) Managing SELinux Modules
- 9) Managing SELinux Systems
- 10) Configuring Auditing
- 11) Customizing Apache



Introduction to SELinux Concepts

Linux Access Control Introduction

- Linux access control involves the
 - kernel controlling
 - processes (running programs) access to
 - resources (files, directories, sockets, etc.)
- For example:
 - web server processes can read web files
 - but not /etc/shadow
- How are these decisions made?

Standard Linux Access Control

- Processes and files have security properties
 - process: user and group (real and effective)
 - resources: user and group + access bits
 - read, write, and execute for user, group, other
- Kernel has hard-coded policy
- Example:
 - Can firefox read my ssh private key?
 - kmacmill 21375 1 35 11:38 ? 00:00:01 firefox-bin
 - -rw----- 1 kmacmill kmacmill 1743 2006-07-10 id_rsa



Important Concepts

- Security properties: security relevant data
 - associated with processes and resources
 - used to make access control decisions
- Policy: rules for access control decisions
- Kernel enforces access control decisions
 - called reference validation mechanism
 - processes also enforce access control
 - database server, dbus, X, etc.

Standard Linux Security Problems

- **Access is based on users' access**
- Example: Firefox can read ssh keys
 - generally has no reason to read them, but
 - if compromised can – potentially disastrous
- Fundamental problem:
 - security properties not specific enough
 - kernel can't distinguish applications from users

Standard Linux Security Problems

- **Processes can change security properties**
- Example: mail files readable only by me
 - evolution can make them world readable
- Fundamental problem:
 - standard access control is discretionary
 - includes concept of resource ownership
 - processes can escape security policy

Standard Linux Security Problems

- **Only two privilege levels: user and root**
- Example: apache privilege escalation
 - apache bug allows obtaining root shell
 - entire system is compromised
- Fundamental problem:
 - simplistic security policy
 - no way to enforce least-privilege

SELinux Introduction

- SELinux adds additional access control
 - new security properties on processes / resources
 - flexible security policy that can be changed
- Kernel and application based enforcement
- Designed to address security problems
 - mandatory, least-privilege, and fine-grained
 - no all powerful root
- Transparent to most applications

SELinux Access Control

- SELinux has three forms of access control
 - Type Enforcement (TE) - primary mechanism
 - Role-Based Access Control (RBAC)
 - Multi-Level Security (MLS)
- Configurable via policy language
 - central configuration files control all access
 - Several policies available (targeted, strict, mls)
- All access is denied by default

SELinux Security Properties

- Processes and files have a security context
 - `kmacmill:staff_r:firefox_t:s0`
 - `kmacmill:object_r:user_home_t:s0`
 - `user:role:type:level`
- The key field is type
 - used to implement Type Enforcement
- Other fields used for RBAC and MLS
 - more on these later

Exercise: Security Contexts

- Several utilities modified for SELinux
- The “-Z” option usually used to view contexts
- Examples:
 - `ps -aeZ` -> view contexts of processes
 - `ls -Z` -> view contexts of files and directories
- Exercises:
 - What is the security context of `/etc/shadow`?
 - What is the security context of `udev`?

Solving Linux Security Challenges

- Security properties need to identify
 - all relevant security information, e.g.,
 - process is a web server (apache)
 - that was started by init
 - consistent across all process and resources
- Security policy needs to be flexible
 - no assumptions (e.g., no root)
 - capable of enforcing integrity, confidentiality, etc.

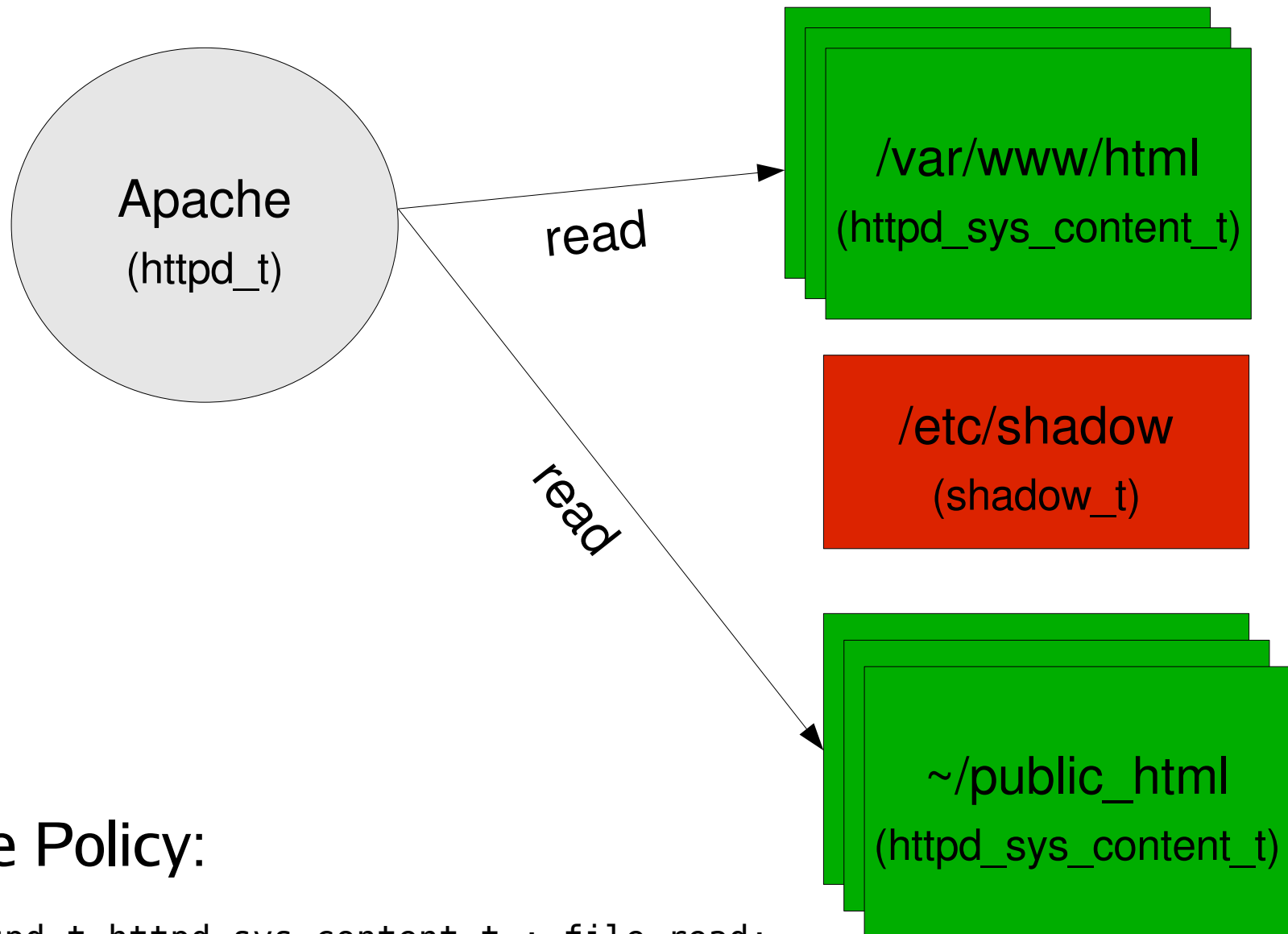
Introduction to Type Enforcement

- Based on a single security property – type
 - applied to processes and resources
 - represents all security relevant information
- Types are assigned to processes and resources
 - Apache processes -> httpd_t
 - /var/www/html/index.html -> httpd_sys_content_t
- Access is allowed between types
 - e.g., httpd_t can read httpd_sys_content_t

Introduction to Object Classes

- Object classes specify the details of access
- Resources divided into classes
 - e.g., file, dir, socket, process
- Each class has permissions
 - e.g., file: read, write, execute, getattr
- Full access in Type Enforcement:
 - allow httpd_t httpd_sys_content_t : file read;

Type Enforcement Overview



Apache Policy:

```
allow httpd_t httpd_sys_content_t : file read;
```

Type Enforcement Concepts

- Access is allowed solely by type
 - many processes and resources have same type
 - simplifies policy by grouping
 - processes with same type have same access
 - same for resources (files)
- Process types called “domains”
 - sometimes applied to resources (e.g., sockets)
- Different resources can have same type

Assigning Initial Types

- Files and directories:
 - configuration file specifies default context
 - called “file contexts”
 - uses path regex: `^/usr/bin/ -> bin_t`
 - Inherited from containing directory at runtime
- Applications can explicitly set context
 - `chcon`: utility to set contexts (think `chown`)
 - `passwd`: maintains context on `/etc/shadow`

Assigning Process Types

■ Process types are:

- (default) inherited from parent process
- set by policy (type transition rule)
- set by application (e.g., login)

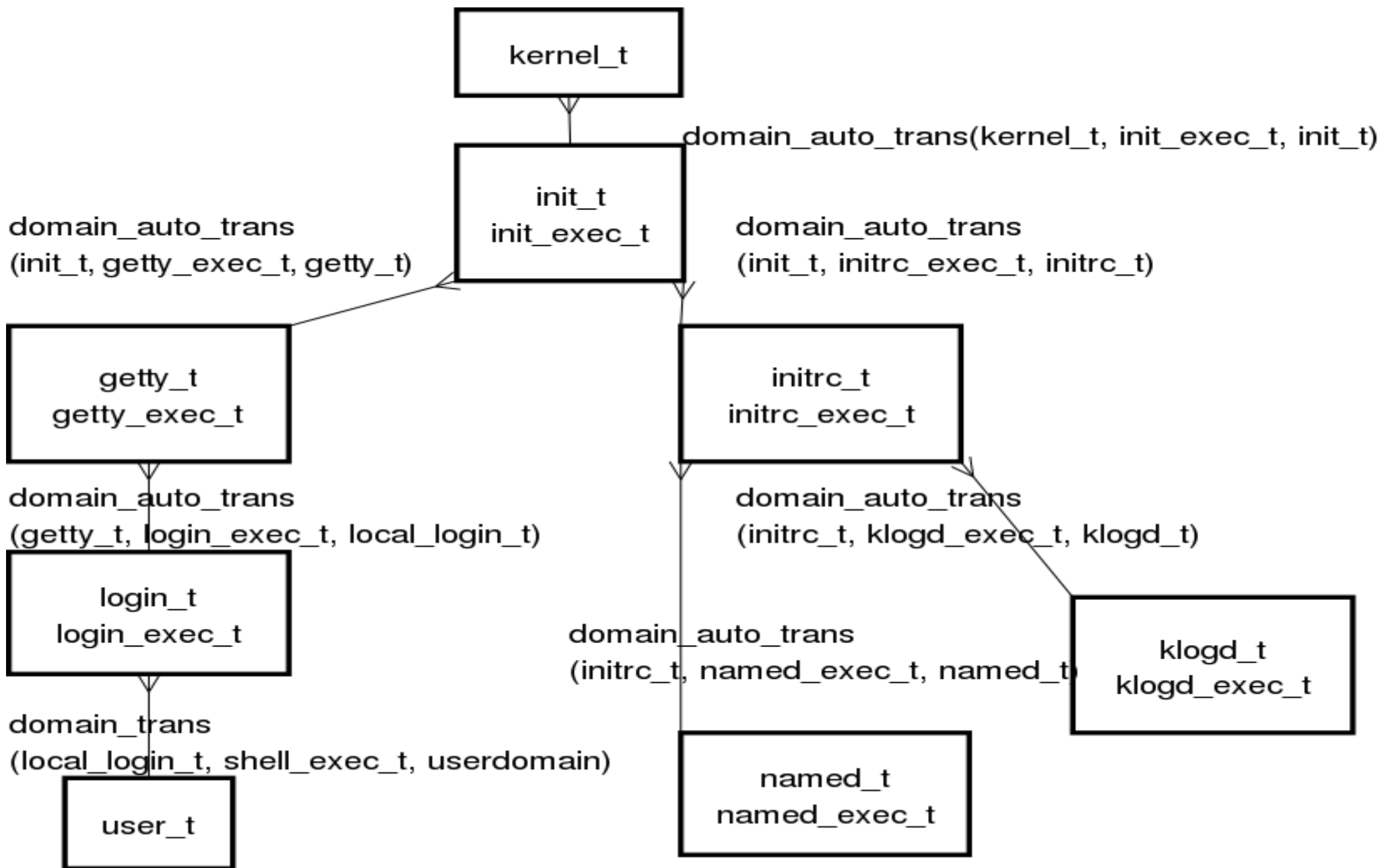
■ Examples:

- bash (user_t) -> ls (user_t)
- init (init_t) -> httpd init script (initrc_t) -> httpd (httpd_t)
- login (login_t) -> bash (user_t)

Type Transition Rules

- Type transition rules set process types using:
 - parent process type and executable file type
 - similar to setuid
- Example: starting name server
 - Rule: `domain_auto_trans(initrc_t, named_exec_t, named_t)`
 - parent process (`initrc_t`)
 - executable file type (`named_exec_t`)
 - result -> `named_t`

Type Transition Rules



Type Transition Notes

- Primary means for setting process type
 - ensures applications run in correct domain
 - does not require application modification
- Must be allowed by policy
 - e.g., apache cannot start processes in `init_t`
 - prevents applications from gaining privilege
- Binds specific executable to domain
 - e.g., only `/usr/bin/passwd` can run in `passwd_t`

User Field Details

- **kmacmill**:user_r:user_mozilla_t:s0
- Not necessarily the same as the Linux user
- Often ends in “_u”: system_u, user_u
- Not currently used in the targeted policy
- Files and directories:
 - user inherited from process
 - system process -> files created with system_u

Role Field Details

- kmacmill:**user_r**:user_mozilla_t:s0
- Used for RBAC
 - role further restricts available type transitions
 - in cooperation with TE (e.g., user_r / user_t)
- Usually ends with “_r”
- Resources have default “object_r” role
- Used in strict and MLS policies
 - user_r, staff_r, secadm_r

MLS Level Field Details

- kmacmill:user_r:user_mozilla_t:s0
- Used for MLS (or MCS)
- Often hidden in targeted and strict (MCS)
- Identifies one level or range
 - single level: s0
 - range: so-s15:c0.c1023
- Usually translated
 - s15:c0.c1023 -> “SystemHigh”

SELinux Security Benefits

- Types capture important security information
 - access is based on user *and* application function
 - transitions capture process call chains
- Processes run with least-privilege
 - only what is allowed for the type
 - e.g., httpd_t can only read web pages
- Privilege escalation tightly controlled
 - a compromise of Apache limited by policy

SELinux Configuration

Strict Policy

- A system where everything is denied by default
 - You must specify allow rules to grant privileges
- SELinux designed to be a strict policy.
 - The policy rules only have allows, no denies
 - Minimal privilege's for every daemon
 - separate user domains for programs like GPG,X, ssh, etc
- Difficult to enforce in general purpose operating system
- Not Supported in RHEL

MLS Policy

- Strict policy with Bell-LaPadula Support
- Supported in RHEL 5 with special license.
- Server only operating system
 - No X-windows support
 - limited package set
- HP/IBM working towards getting EAL4+/LSPP certification

Targeted Policy

System where processes by default are unconfined.

- Only targeted processes are confined
- Unconfined Domains
 - By default user processes run in `unconfined_t`
 - System processes run in `initrc_t`
 - Unconfined processes have the same access they would have without SELinux running
- Daemons with defined policy transition to confined domains
- `httpd` started from `unconfined_t` transitions to `httpd_t` which has limited access.

Targeted Domains

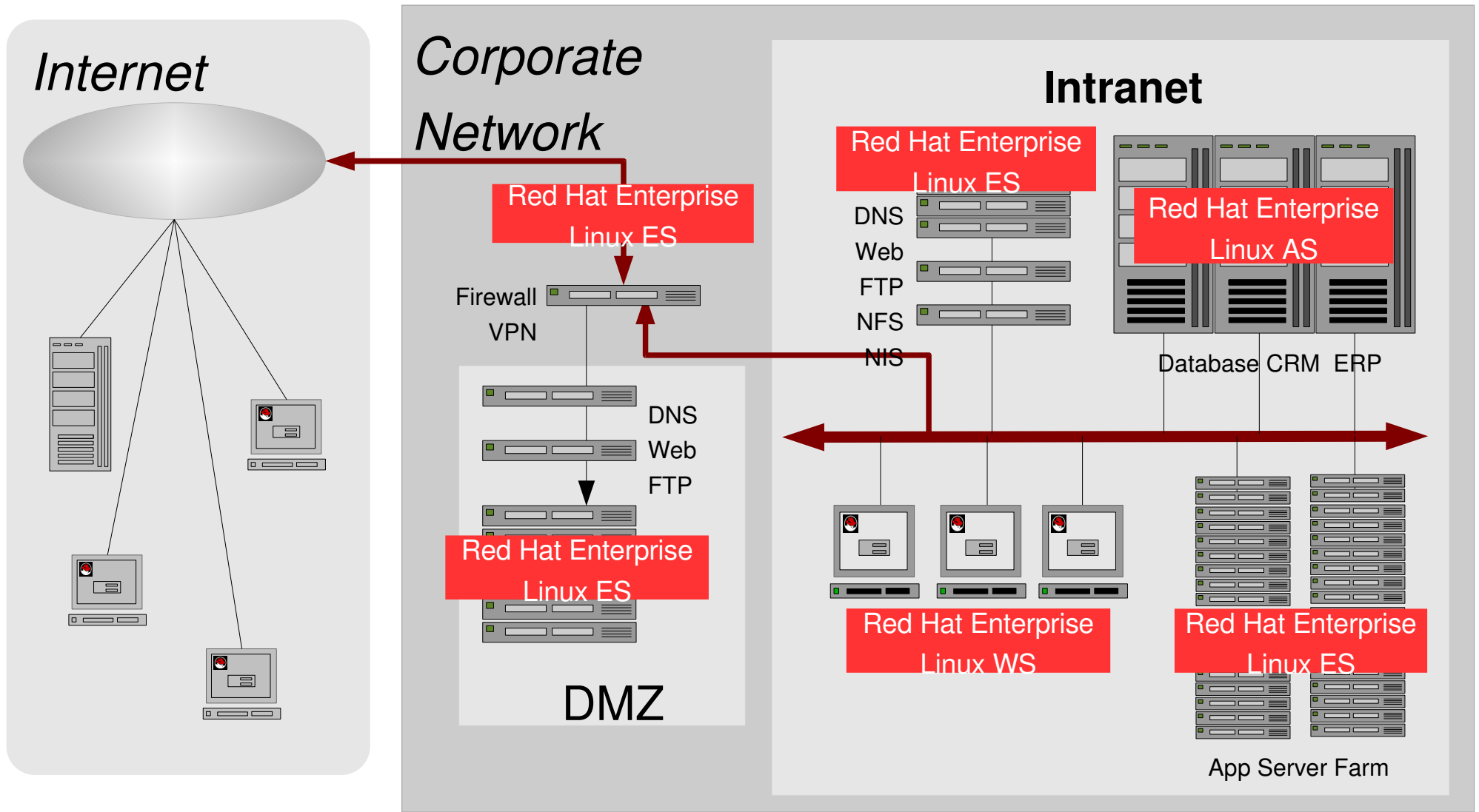
■ In RHEL4

- 15 targets defined
- httpd,squid,pegasus,Mailman,Named, dhcpd,mysqld, nscd,ntpd.
portmap,postgresql,snmpd,syslogd,winbindd

■ In RHEL5

- 200 targets defined
 - Every program shipped by Red Hat and started on boot should have a domain defined
- All system space is confined
- Limited confinement for user space
- 20 unconfined domains

Where should you run SELinux?



Config files

- SELinux stores its config files in `/etc/selinux`

```
ls -l /etc/selinux
-rw-r--r-- 1 root root 515 Jan 18 11:46 config
drwxr-xr-x 7 root root 4096 Jan 23 14:06 strict
drwxr-xr-x 7 root root 4096 Jan 23 14:06 targeted
```

- `/etc/selinux/config` identifies policy and enforcing mode

```
more /etc/selinux/config

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=enforcing

# SELINUXTYPE= can take one of these two values:
#   targeted - Only targeted network daemons are protected.
#   strict - Full SELinux protection.
SELINUXTYPE=targeted
```

Config files

- Directory under policy type follow same format
 - contexts directory contains default contexts files used by SELinux aware applications
 - policy dir contains compiled policy file
 - seusers contains Linux User to SELinux users mapping file
 - setrans.conf contains MLS/MCS translations
 - Modules directory includes current modules used to build policy

```
ls -l /etc/selinux/targeted/  
total 40  
drwxr-xr-x 4 root root 4096 Jan 29 09:00 contexts  
drwxr-xr-x 4 root root 4096 Jan 29 09:00 modules  
drwxr-xr-x 2 root root 4096 Jan 29 09:00 policy  
-rw-r--r-- 1 root root 598 Jan 23 17:24 setrans.conf  
-rw-r--r-- 1 root root 143 Jan 29 09:00 seusers
```

Config files

- /etc/selinux/targeted/contexts/files/
 - file_contexts
 - file_contexts.local
 - file_contexts.homedir
 - homedir_template

Kernel Boot Parameters

- Kernel parameters override `/etc/selinux/config` settings
- `selinux=0`
 - Boots the kernel with SELinux turned off
 - All files will no longer get created with file context.
 - Will require a relabel if the machine gets booted again with selinux turned on.
- `enforcing=0`
 - Boots the kernel in permissive mode
 - File labeling continues
 - May NOT give the same error messages as in enforcing mode.

Target Policy Man Pages

- Target man pages explain custom features of the policy booleans and file context

```
httpd_selinux(8)  httpd SELinux Policy documentation  httpd_selinux(8)
```

NAME

httpd_selinux - Security Enhanced Linux Policy for the httpd daemon

DESCRIPTION

Security-Enhanced Linux secures the httpd server via flexible mandatory access control.

FILE_CONTEXTS

SELinux requires files to have an extended attribute to define the file type. Policy governs the access daemons have to these files. SELinux httpd policy is very flexible allowing users to setup their web services in as secure a method as possible.

The following file contexts types are defined for httpd:

```
httpd_sys_content_t
```

- Set files with httpd_sys_content_t for content which is available from all httpd scripts and the daemon.

```
httpd_sys_script_exec_t
```

- Set cgi scripts with httpd_sys_script_exec_t to allow them to

Exercises

- Read through a couple of SELinux Policy man pages
- Which policy is the system currently running?
- Reboot in permissive mode
 - Do you see additional AVC messages?

Modified Operating System Commands

Modified Utilities

- **“Z”** is your friend
 - ls -Z
 - id -Z
 - ps auxZ
 - lsof -Z
 - netstat -Z
 - find / -context=

Modified Utilities

- cp
 - Adopts destination directory or files security context
 - -a problems
- mv
 - Maintains Sources Destination Security context
- install
 - Sets default security context based on system defaults

Modified Programs

- Login Programs - PAM
 - sshd, login, xdm
- Password utilities
 - passwd, useradd, groupadd
- rpm

Backup and disc management

- tar, zip
 - Both now have extended attribute support
- rsync
 - -X, -xattrs
- star
 - `star -xattr -H=exustar -c -f output.tar [files]`
- amanda
- `tar xv | restorecon -f -`; still might be best option

Exercises: Modified Linux Utilities

- What security context is on `/etc/resolv.conf`?
- Explore other security context in `/etc`
- What is the context is the apache process running with?
- What is your security context?
- Create a file in `/tmp` and mv it to etc
 - What is the security context on the file?
 - Is this a problem?
- Create a new account on your machine
 - What is the security context on `/etc/passwd`? `/etc/shadow`?
 - Why do you suppose they are different?



SELinux Utilities

SELinux Utilities

- libselinux rpm
- libselinux is the default SELinux library used by SELinux aware applications
- libselinux utilities
 - getenforce – Tell whether machine is in enforcing/permissive/disabled
 - setenforce 1/0 – Sets the machine in enforcing/permissive
 - selinuxenabled – Used by scripts to tell whether SELinux enabled.
 - matchpathcon – Tells you the default context of file/directory
 - avcstat - Display SELinux AVC statistics
- libselinux-python
 - Python bindings to libselinux

SELinux Utilities - Policycoreutils

- genhomedircon, fixfiles, restorecon, restorecond, setfiles, chcon, chcat
- audit2allow, audit2why (See [Understanding SELinux log messages](#))
- secon - See an SELinux context, from a file, program or user input.
- semodule, semodule_deps, semodule_expand, semodule_link, semodule_package (See [Managing an SELinux Policy Modules](#))
- load_policy – load a new SELinux policy into the kernel
- run_init – Run a init script in the proper SELinux context (mls, strict)
- semanage, system-config-selinux - (See [Managing an SELinux system](#))
- sestatus – SELinux status tool
- setsebool, getsebool - (See [Customizing the policy with booleans](#))
- newrole – Run a shell with a new SELinux role/level (mls, Strict)

Exercises: SELinux Utilities

- Is your machine in enforcing mode?
 - Turn on permissive mode
 - What AVC message was generated?
 - Return machine to enforcing mode.
- What is the SELinux status of your machine?
- Use `sestatus` to check the file context on `/etc/shadow`
- Create the file `/etc/apache`
 - Change its context type to `httpd_exec_t`
 - How would you get this application to run as `httpd_t`?
- Correct the context of all the files in `etc`

Understanding Audit Messages

Understanding SELinux log messages

- AVC Access Vector Cache
 - messages in /var/log/messages or /var/log/audit/audit.log

```
type=AVC msg=audit(1140184056.443:78): avc: denied { use } for pid=2185  
comm="mingetty" name="ptmx" dev=tmpfs ino=699 scontext=system_u:system_r:getty_t:s0  
tcontext=system_u:system_r:kernel_t:s0 tclass=fd
```

```
type=AVC msg=audit(1166017682.366:876): avc: denied { getattr } for pid=23768  
comm="httpd" name="index.html" dev=dm-0 ino=7996439  
scontext=user_u:system_r:httpd_t:s0 tcontext=user_u:object_r:user_home_t:s0 tclass=file
```

Understanding SELinux log messages

- AVC Messages can get created for a variety of reasons.
 - A mislabeled file
 - A process running under the wrong context
 - A bug in policy.
 - Basically an application goes down a code path that was never tested by the policy writer and gets an unexpected AVC.
 - An intruder

Understanding SELinux log messages

■ audit2allow

- Tool that generates policy allow rules from logs of denied operations
- `audit2allow -i /var/log/audit/audit.log`
 - `allow httpd_t user_home_t:file getattr;`

■ audit2why

- Translates SELinux audit messages into a description of why the access was denied
- Not very helpful to novice users, used by policy developers

Analyzing SELinux AVC Messages

- AVC Messages referring to files labeled *:file_t
 - Major Labeling problem, all files probably require labels
 - SELinux kernel labels files with no security context file_t
 - File was created when running with selinux=0 or a new disk.
 - It is safest to relabel the system - touch /.autorelabel; reboot
 - On a new disk you can restorecon -R -v /MOUNTPOINT
- AVC Messages containing default_t
 - Probably a labeling problem
 - If not in / you probably need to relabel
 - If in / and you want confined domains to have access. You need to relabel the file/directory using chcon

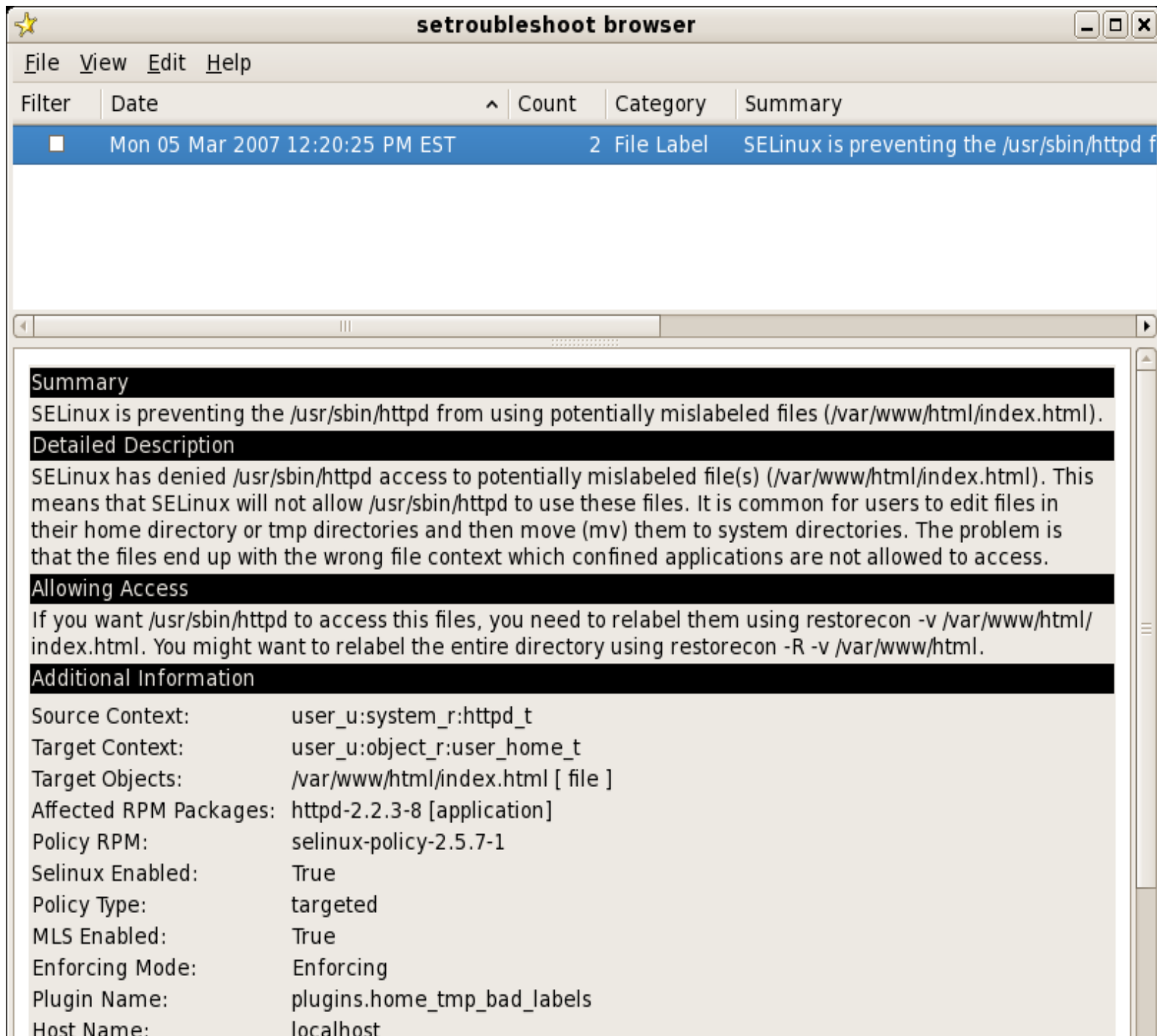
Analyzing SELinux AVC Messages

- Many similar messages about the same file
 - This usually indicates a labeling problem
 - For example:
 - Create /home/dwalsh/resolv.conf
 - mv /home/dwalsh/resolv.conf /etc
 - ls -lZ /etc/resolv.conf
 - Confined domains will report errors accessing user_home_t
 - restorecon /etc/resolv.conf

SELinux Troubleshoot Tool

■ setroubleshoot

- Service listens to audit daemon for AVC messages
- Then processes plugin database for known issues
 - /usr/share/setroubleshoot/plugins
- Displays knowledge base of how to handle avc message
- sealert command can launch browser or analyze log files
- Configure /etc/setroubleshoot/setroubleshoot.cfg to send mail



The screenshot shows the 'setroubleshoot browser' window. At the top, there is a menu bar with 'File', 'View', 'Edit', and 'Help'. Below the menu is a table with columns: Filter, Date, Count, Category, and Summary. A single entry is highlighted in blue, showing a date of 'Mon 05 Mar 2007 12:20:25 PM EST', a count of '2', a category of 'File Label', and a summary of 'SELinux is preventing the /usr/sbin/httpd f'. Below the table is a scrollable area containing the following text:

Summary
SELinux is preventing the `/usr/sbin/httpd` from using potentially mislabeled files (`/var/www/html/index.html`).

Detailed Description
SELinux has denied `/usr/sbin/httpd` access to potentially mislabeled file(s) (`/var/www/html/index.html`). This means that SELinux will not allow `/usr/sbin/httpd` to use these files. It is common for users to edit files in their home directory or tmp directories and then move (`mv`) them to system directories. The problem is that the files end up with the wrong file context which confined applications are not allowed to access.

Allowing Access
If you want `/usr/sbin/httpd` to access this files, you need to relabel them using `restorecon -v /var/www/html/index.html`. You might want to relabel the entire directory using `restorecon -R -v /var/www/html`.

Additional Information

Source Context:	user_u:system_r:httpd_t
Target Context:	user_u:object_r:user_home_t
Target Objects:	<code>/var/www/html/index.html</code> [file]
Affected RPM Packages:	httpd-2.2.3-8 [application]
Policy RPM:	selinux-policy-2.5.7-1
Selinux Enabled:	True
Policy Type:	targeted
MLS Enabled:	True
Enforcing Mode:	Enforcing
Plugin Name:	plugins.home_tmp_bad_labels
Host Name:	localhost

Missing AVC messages

Sometimes applications fail with no AVC messages

- Setting setenforce 0 and the application works???
- dontaudit rules
 - Expected AVCs that cause apps to take different code paths.
 - Sometimes cover up Real errors
- RHEL 4
 - Install selinux-policy-targeted-sources
 - make -C /etc/selinux/targeted/src/policy enableaudit load
- RHEL 5
 - semodule -b /usr/share/selinux/targeted/enableaudit.pp
 - semodule -b /usr/share/selinux/targeted/base.pp

Exercises: SELinux Utilities

- Lets create some avc messages?
 - touch /var/www/html/index.html
 - chcon -t user_home_t /var/www/html/index.html
 - service httpd start
 - firefox localhost
 - what happens?
 - Check the log files for AVC messages.
- What audit rules could you add to solve these AVC?
- Why did policy refuse this access?

Managing File Labeling

Managing file labeling

- Changing a files context
- chcon
 - Fundamental utility used to change a files context
 - `chcon -R -t httpd_sys_script_rw_t /var/www/myapp/data`
 - `chcon -t httpd_sys_script_t /var/www/cgi-bin/myapp`
 - Modeled after `chmod` command
 - `-t` type qualifier
 - `customizable_types`
 - `/etc/selinux/targeted/contexts/customizable_types`

Managing file labeling

- restorecon
 - Used to set a file back to the system defaults
- setfiles
 - Used to initialize a system. Used at the File system level
 - Requires you to specify file_context file
- fixfiles
 - Script that wraps setfiles/restorecon with several useful features
 - Use rpm to list files within specified packages to restore file contexts
 - restorecon changes between previous file context and new one
- touch /.autorelabel; reboot

Managing file labeling

- Genhomedircon
 - Used to generate file_contexts.homedir
 - Sometimes has problems with homedir locations.
- /etc/selinux/targeted/contexts/files/file_context.local
- system-config-securitylevel

Exercises: Managing file labeling

- Modify `/etc/resolv.conf`
 - `cp /etc/resolv.conf to /tmp`
 - Make some changes to the search string
 - `mv` it back the `/etc`
 - Fix its security context

- Homedirs
 - Create an a new directory `/export/homes`
 - Add a user account to that directory
 - Add a user account to the `/var` directory
 - Run `genhomedircon` – What happens?
 - Fix the context on these directories

Customizing Policy With Booleans

Customizing policy with booleans

Booleans are if/then/else statements in policy

- Configure policy without editing policy
- `getsebool`
 - `getsebool -a`
- `setsebool`
 - `setsebool -P allow=[1|0]`
- `system-config-selinux` (`system-config-securitylevel-RHEL4`)
- Turn on/off sections of policy
 - `setsebool -P allow_nfs_home_dirs 1`
 - `/etc/selinux/targeted/booleans`



File Help

Select:

- Status
- Boolean**
- File Labeling
- User Mapping
- SELinux User Translation
- Network Port
- Policy Module

- ▶ Databases
- ▶ FTP
- ▶ Games
- ▼ HTTPD Service
 - Allow Apache to use mod_auth_pam.
 - Allow HTTPD cgi support
 - Allow httpd daemon to write files in directories labeled public_content_rw_t
 - Allow HTTPD scripts and modules to connect to the network.
 - Allow HTTPD scripts and modules to network connect to databases.
 - Allow httpd scripts to write files in directories labeled public_content_rw_t
 - Allow httpd to act as a relay.
 - Allow HTTPD to read home directories
 - Allow HTTPD to run as a ftp server
 - Allow HTTPD to run SSI executables in the same domain as system CGI scripts.
 - Allow HTTPD to support built-in scripting
 - Disable SELinux protection for httpd daemon
 - Disable SELinux protection for http suexec



Configuring Policy

Apache Example

- System administrator has multiple choices of policy
 - Booleans
 - httpd_disable_trans, httpd_enable_cgi httpd_enable_homedirs
httpd_tty_comm, httpd_unified
- <http://fedora.redhat.com/docs/selinux-apache-fc3/>
- man httpd_selinux

Exercises: Managing Booleans

- List all booleans on your machine
- Check the contents of `/etc/selinux/targeted/booleans`
- Temporarily change a booleans state
- Did the `/etc/selinux/targeted/booleans` file change?
- If you have time try the previous exercises after you turn on the `httpd_tty_comm` boolean

Managing SELinux Modules

SELinux Modules

■ Modular Policy

- In RHEL 5 /Fedora Core 5 and later, the concept of Policy Modules was introduced

■ The semodule command

- Copies the policy package (pp) files to `/etc/selinux/targeted/modules/active/modules`
- Compiles all installed pp files into new policy file `/etc/selinux/targeted/policy/policy.21`
- Creates new `file_context` file and `file_context.homedirs`
- Loads new policy

SELinux Policy Modules

- semodule command
 - `semodule -l` ; List all modules currently loaded
 - `semodule -b /usr/share/selinux/targeted/enableaudit.pp`
 - `semodule -b /usr/share/selinux/targeted/base.pp`
 - `semodule -i myapache.pp`
 - `semodule -r myapache`



SELinux Management Tool



File Help

Select:

- Status
- Boolean
- File Labeling
- User Mapping
- SELinux User
- Translation
- Network Port
- Policy Module**



Add



Remove



Enable Audit



Disable Audit

Module Name ▾ | Version

amavis 1.1.0

audio_entropy 1.0.0

calamaris 1.1.0

ccs 1.0.1

cdrecord 1.1.0

certwatch 1.0

cipe 1.1.0

clamav 1.2.0

consolekit 1.0.0

daemontools 1.1.0

dcc 1.1.0

dhcp 1.0

ethereal 1.1.1

evolution 1.1.1

Generating policy modules

- Policy modules consists of three files.
 - Type Enforcement File (te)
 - Contains the allow rules and interface calls associated with the confined domain
 - File Context File (fc)
 - Contains all of the labeling file context for the policy module.
 - Interface File (if)
 - Contains all interfaces used by other domains to interact with this confined domain.
 - DOMAIN_domtrans, DOMAIN_read_config

Creating Policy modules with audit2allow

- Making small customizations to policy
- In RHEL4
 - You needed to install selinux-policy-sources to modify policy
 - `cd /etc/selinux/targeted/src/policy`
 - `grep http_t /var/log/messages | audit2allow >> domain/misc/local.te`
 - `make install`
- In RHEL5
 - `grep http_t /var/log/audit/audit.log | audit2allow -M mypolicy`
 - This command will generate a te file and compile it into a pp file.
 - `semodule -i mypolicy.pp`

Building policy modules

- Install selinux-policy-devel
 - Includes interfaces for all installed policy modules
 - /usr/share/selinux/devel
 - policygentool – helper app to begin construction of te, if, fc file
 - include/... directory has interfaces
 - kernel, services, system, apps, admin
 - Makefile (used to compile policy modules).

Exercises: Managing Policy Modules.

- List all modules on your machine
- Remove the pcscd policy module.
- What is the label of the running process?
- Why?
- `service pcscd stop`
- `restorecon -R -v /usr/sbin/pcscd /var/run`
- Advanced Topic:
 - using `/usr/share/selinux/devel/policygentool` try to generate pcscd policy



Managing SELinux Systems

Managing SELinux systems

- In RHEL5/Fedora Core 5 and beyond a new semanage framework was added
- In RHEL4 often required custom policy
 - allowing apache to listen on port 81
 - required policy sources and tools
- In RHEL5
 - `semanage port -a -t http_port_t -P tcp 81`

Semanage Commands

- SELinux Users
 - semanage user -l
 - semanage user -a guest_u
- Linux User to SELinux user mapping
 - semanage login -a -s guest_u dwalsh
- File Context
 - semanage fcontext -a -t
httpd_bugzilla_script_exec_t
/usr/share/bugzilla/cgi(/.*)?



File Help

Select:

- Status
- Boolean
- File Labeling
- User Mapping
- SELinux User
- Translation
- Network Port**
- Policy Module



Add



Properties



Delete



Group View

SELinux Port Type	Protocol	Port
gopher_port_t	udp	70
hi_reserved_port_t	tcp	600-1023
hi_reserved_port_t	udp	600-1023
howl_port_t	tcp	5335
howl_port_t	udp	5353
hplip_port_t	tcp	1782, 2207, 2208, 50000, 50002, 8292, 9100, 9101, 9
http_cache_port_t	tcp	3128, 8080, 8118
http_cache_port_t	udp	3130
http_port_t	tcp	80, 443, 488, 8008, 8009, 8443
i18n_input_port_t	tcp	9010
imaze_port_t	tcp	5323
imaze_port_t	udp	5323
inetd_child_port_t	tcp	7, 9, 13, 19, 37, 512, 543, 544, 891, 892, 2105, 5666
inetd_child_port_t	udp	7, 9, 13, 19, 37, 512, 543, 544, 891, 892, 2105, 5666

Select:

- Status
- Boolean
- File Labeling
- User Mapping
- SELinux User
- Translation
- Network Port**
- Policy Module



Group View

SELinux Port Type	Protocol	SELinux Type	MLS/MCS Level	Port Number
http_cache_port_t	tcp	http_port_t	s0	81
http_cache_port_t	tcp	http_port_t	s0	8008
http_cache_port_t	tcp	http_port_t	s0	8009
http_cache_port_t	tcp	http_port_t	s0	8443
http_port_t	tcp	http_port_t	s0	9010
http_port_t	tcp	http_port_t	s0	5323
http_port_t	tcp	http_port_t	s0	37
http_port_t	tcp	http_port_t	s0	513

Add Network Port [X]

Port Number:

Protocol:

SELinux Type:

MLS/MCS Level:

Exercises: Managing SELinux

- List all SELinux Users
- Add an SELinux user to your machine
- Create a directory /opt/www/html
- Label it so that apache can read it.
- Make a permanent change to the file context so that relabeling the file system will not change this.

Configuring Audit

Auditing

- Audit system receives SELinux Events
 - No auditd running
 - AVC in /var/log/messages and dmesg
 - auditd running
 - AVCs in /var/log/audit/audit.log
- audit=1 Command required for full auditing

Auditing CAPP/EAL4+

- CAPP – Controlled Access Protection Profile
 - DAC Profile
 - Security features selection
- eal4+. - E Assurance Level
 - Level of testing and documentation
- `cp /usr/share/doc/audit-1.0.12/capp.rules /etc/audit.rules`

auditctl

- Utility to control the kernel's audit system
 - -e [0|1] Disable, Enable audit
 - SEE Steve Grubb Audit BOF...

aureport

- Generate summary reports of audit logs
 - -a Report about AVC messages
 - -i interpret numeric fields for human consumption
 - -ts “Time Start” -te “Time End”
 - aureport -a -ts 1:00:00
 - Generate a avc report since 1 AM
 - --success/--failed (Both if you select neither.)
 - --summary (Totals of events)

ausearch

- Search Audit Daemon Logs
 - -m avc
 - -ts
 - -x executable
 - ausearch -m avc -ts 1:00:00 -x named

Exercises: Audit

- Use aureport on audit messages on the system
- Search for apache avc messages
- Turn off auditing? Where to the AVC message end up?



Customizing Apache

Customizing Apache Policy

- httpd - most complex daemon in RHEL 4
- Most complex and configurable of any of the SELinux policies.
 - Confine compromised Apache web server from damaging the rest of the system
 - Finer grained goals
 - Preventing a compromised wiki CGI script from corrupting a blog installation owned by the same person

Exercises: Apache

- Is httpd running under a confined domain?
 - Stop httpd
 - Start apache directly `/usr/sbin/httpd`
 - Which context is it running under?
 - Why?
- Kill httpd and start it within a confined domain
- Setup an apache web site which supports cgi-scripts where the data is located in `/src/www/data` directory
- Setup apache to use users home directories and place a html file there
- Advanced: Add a cgi script that needs to write to a particular directory, turn off `httpd_unified` and make the script work in enforcing mode.

Q/A

■ More Information Red Hat Enterprise Linux Resource

- <http://www.redhat.com/software/rhel/>

■ SELinux Resources

- <http://www.nsa.gov/selinux>
- <http://fedora.redhat.com/projects/selinux/>
- <http://fedoraproject.org/wiki/SELinux>

■ Mailing Lists

- selinux@tycho.nsa.gov - NSA List
- fedora-selinux-list@redhat.com - Fedora SELinux List