Red Hat Deep Dive Sessions

SELinux: A Key Component in Secure Infrastructures

Shawn D. Wells, RHCE
EMail: swells@redhat.com

Solutions Architect @ Red Hat

Agenda

1) **Why do we need SELinux? What are the principal concepts?**
2) **SELinux Details**
   - **Type Enforcement**
   - **What are the available policies?**
   - **What's a policy actually made of?**
   - **How do I {add, change} a policy?**
   - **What's the associated overhead?**
3) **Usage**
   - **User Perspective**
   - **Admin Perspective**
4) **Scenarios**
   - Fixing the RHT Corporate VPN “update”

Why do we need SELinux?

Linux Access Control Problems

1) Access is based off users' access

Example: Firefox can read SSH keys

```
# ps -x | grep firefox
shawn 21375 1 35 11:38 ? 00:00:01 firefox-bin
```

```
# ls -l id_rsa
  -rw------- 1 shawn shawn 1743 2008-08-10 id_rsa
```

Fundamental Problem: Security properties not specific enough. Kernel can't distinguish applications from users.

Linux Access Control Problems

2) Processes can change security properties

Example: Mail files are readable only by me..... but Thunderbird could make them world readable

Fundamental Problems:

- Standard access control is discretionary
- Includes concept of “resource ownership”
- Processes can escape security policy

Linux Access Control Problems

3) **Only two privilege levels: User & root**

Example: Apache gets hacked, allowing remote access to root. Entire system is compromised.

Fundamental Problems:

- Simplistic security policy
- No way to enforce least-privilege

Linux Access Control Introduction

Linux access control involves the kernel controling

- **Processes** (running programs), which try to access...
  - **Resources** (files, directories, sockets, etc)

For example:

- Apache (process) can read web files
- But **not** the /etc/shadow file (resource)

Traditional methods do not clearly separate the privileges of users and applications acting on the users behalf, increasing the damage that can be caused by application exploits.

So, how should these decisions be made?

Security Architecture

Every subject (i.e process) and object (i.e. data files) are assigned collections of security attributes, called a **security context**

1) Security context of subject & object passed to SELinux

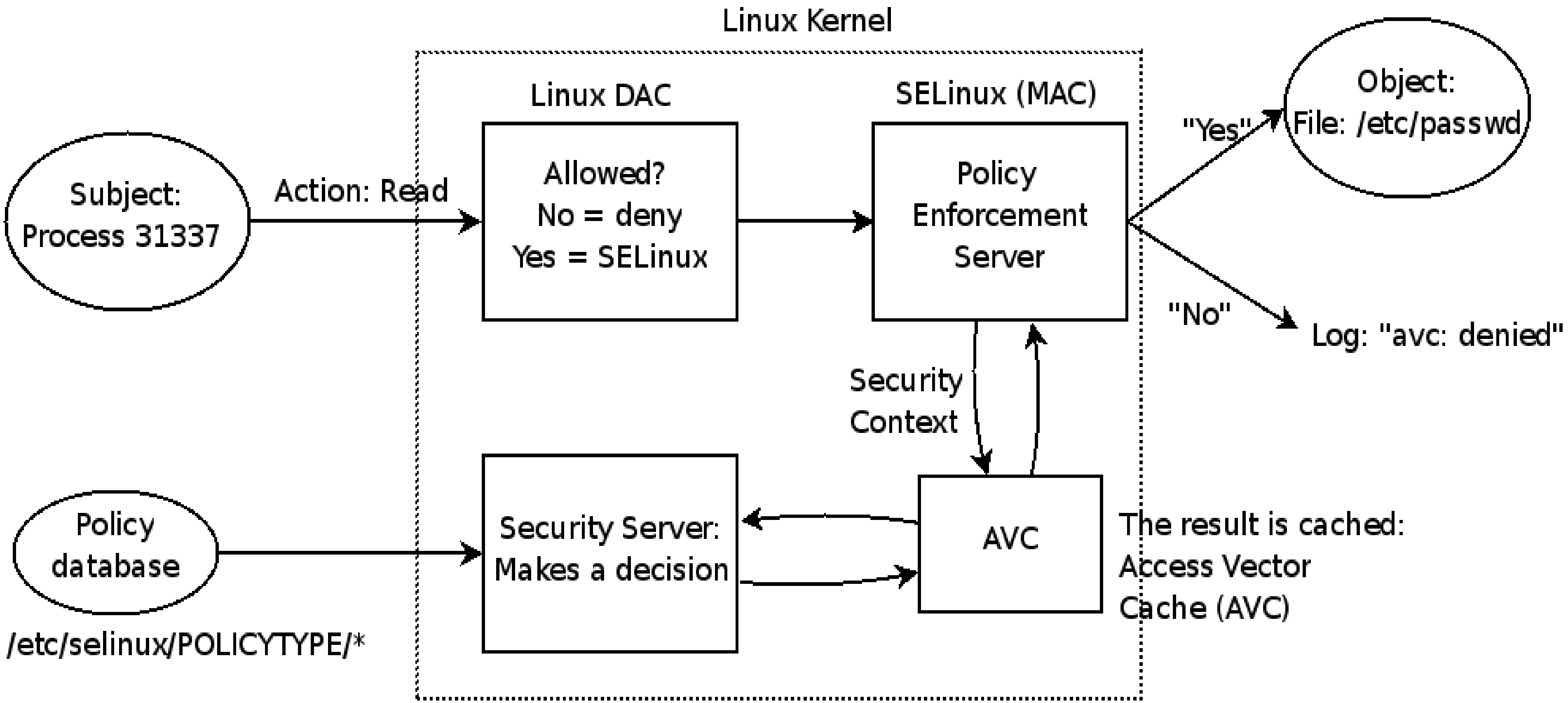
2) Kernel/SELinux check, verify access

2a) Grant access. Record allowance in AVC (Access Vector Cache)

2b) Deny access, log error

Security Architecture

Or in picture view...



Role Based Access Control (RBAC)

“root” really isn't “root”

i.e:

root_u:**WebServerAdmin_r**:SysAdmin_t

root_u:**OracleDBAdmin_r**:SysAdmin_t

SELinux Details

Type Enforcement

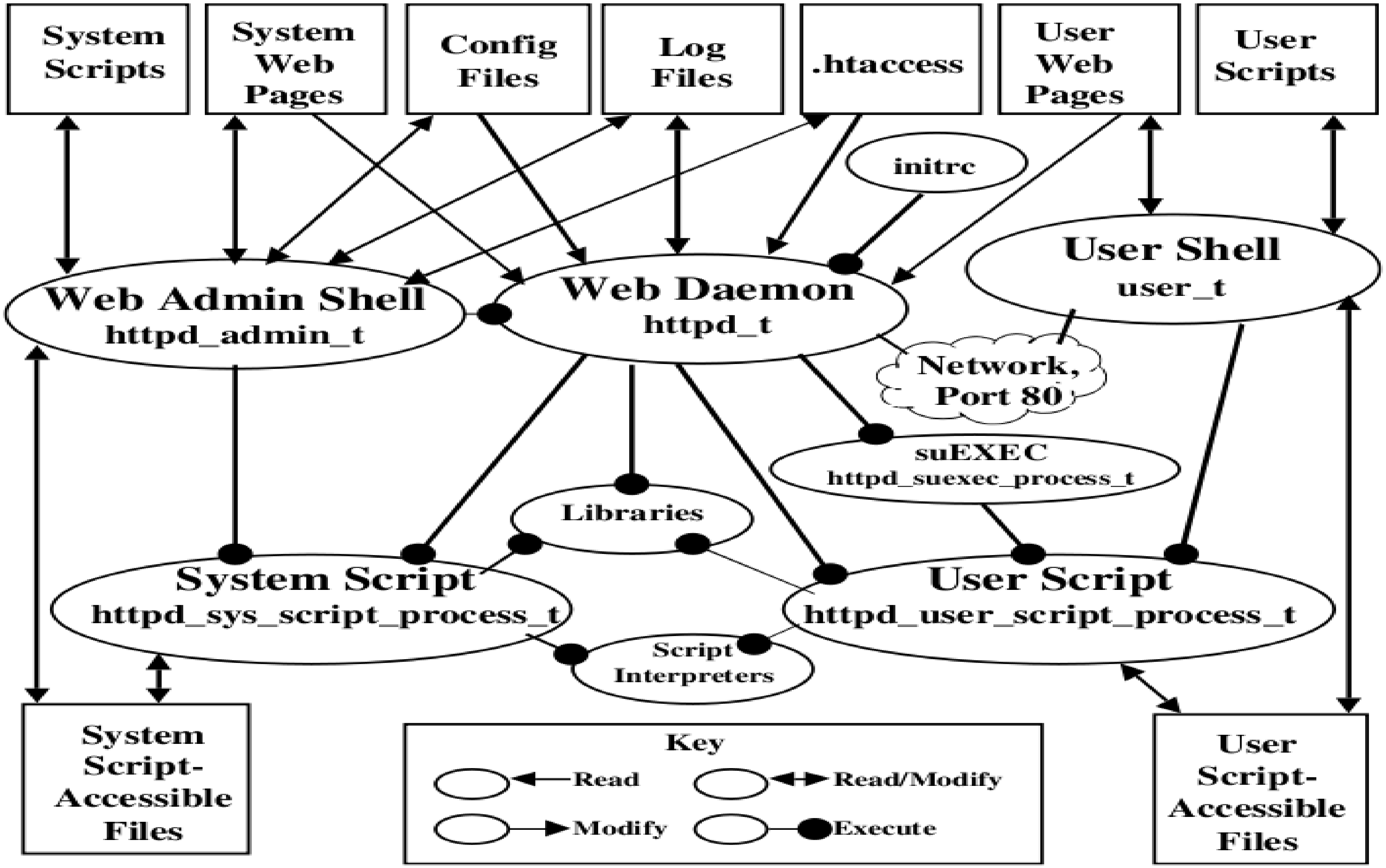
- SELinux implements the MAC model through type enforcement.
- In RHEL5, SELinux also provides RBAC and Bell-LaPadula (MLS), but it uses type enforcement to implement them.
- Type Enforcement involves defining a type for every subject, that is, process, and object on the system.
- Permissions are checked between the source type and the target type for each access.
- Objects include (but are not limited to):
    - Network Sockets
    - Shared Memory Segments
    - Files
    - Processes
    - etc.

SELinux Contexts

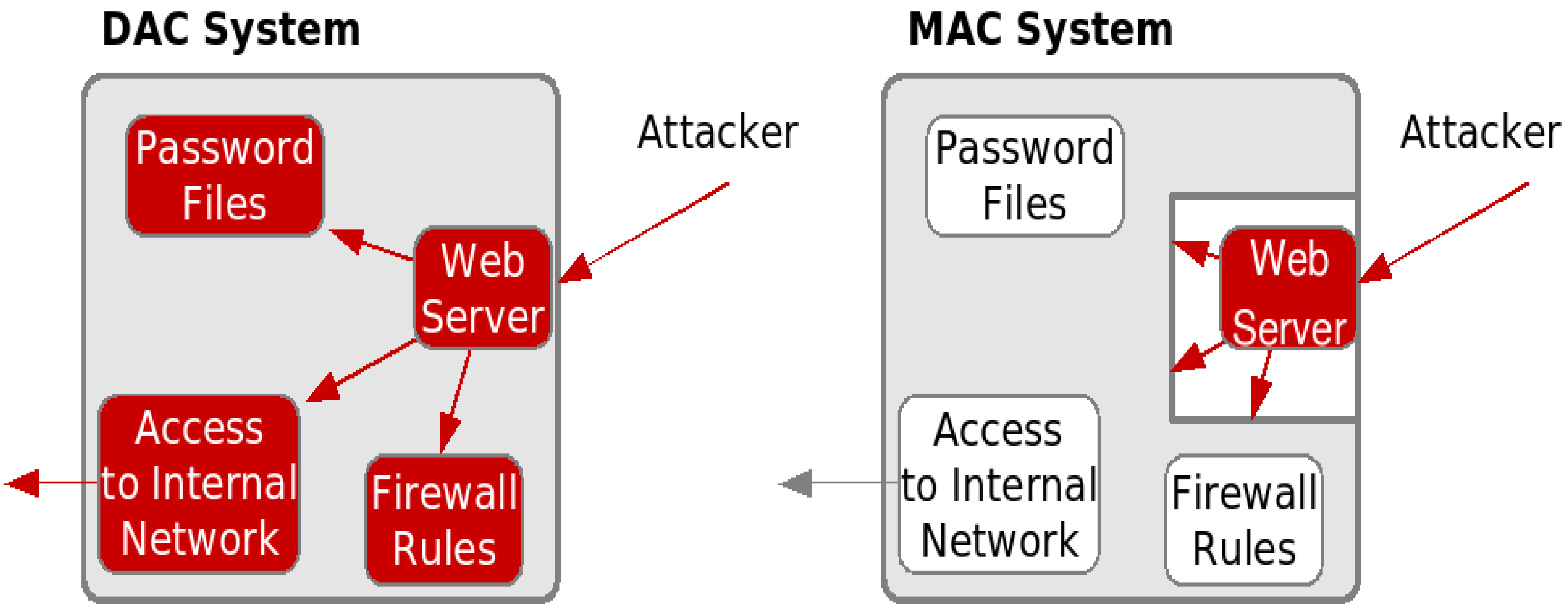
root:object r:sysadm home t:s0:c0

- The above is an SELinux context
- user_t
- role_t
- file_t
- Sensitivity
- category

SELinux Contexts



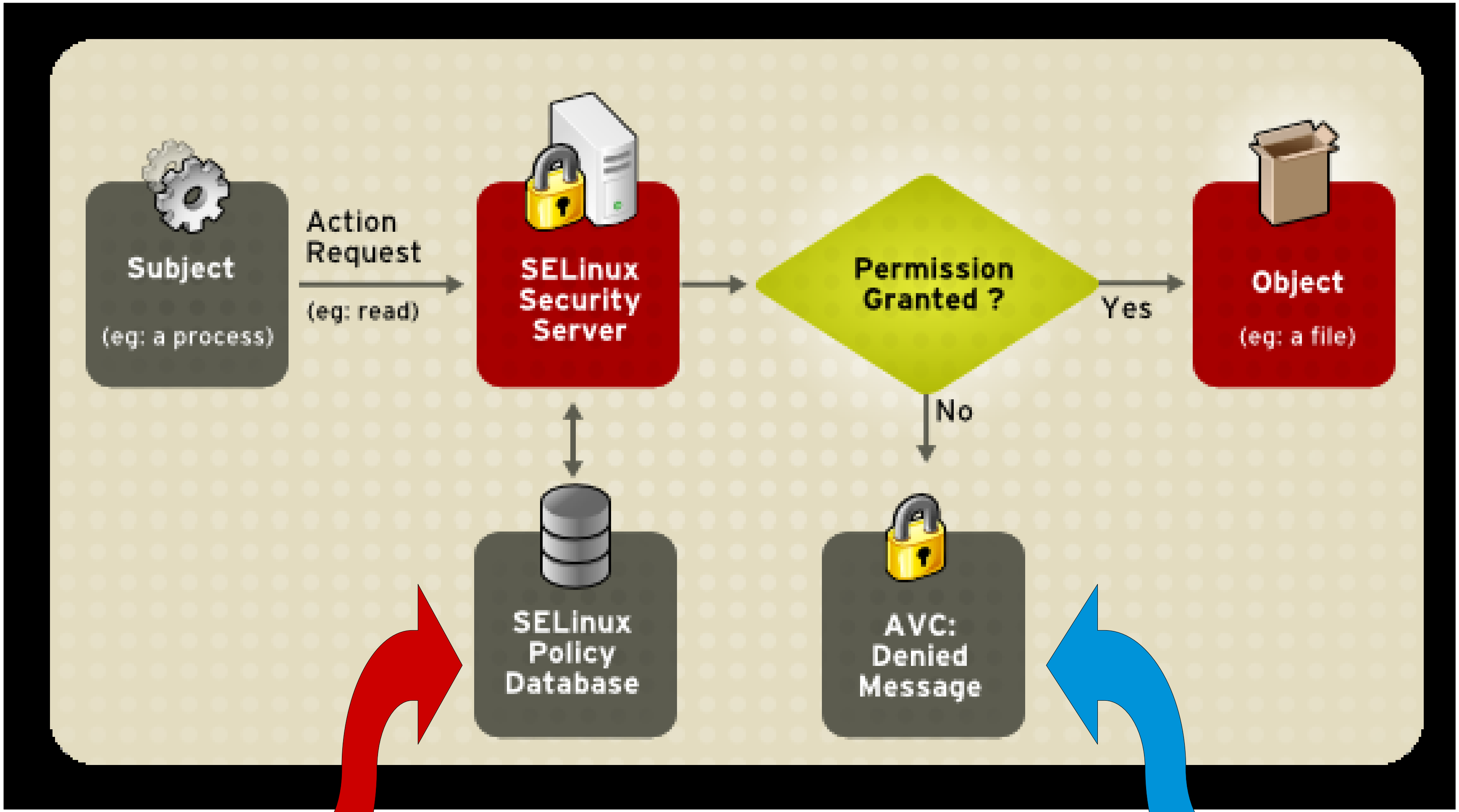
DAC vs MAC



- Application can change attributes
- User privileges
  =
  Process privileges

- Orthogonal to DAC
- Roles, Contexts, Types

How does SELinux Work?



/etc/selinux/{targeted,strict}/policy

/var/log/messages

SELinux Policy

- Policies are matrices of statements which tell SELinux if certain actions are allowed based on the context of the objects attempting those actions.

- There are three SELinux Policy Types

The Three SELinux Policy Types

1) **Targeted Policy**

- *Default policy in RHEL5. Supported by HelpDesk.*
- Targets specific applications to lock down.
- Allows all other applications to run in the unconfined domain (`unconfined_t`)
- Applications running in the unconfined domain run as if SELinux were disabled

The Three SELinux Policy Types

2) **Strict Policy**

- Denies access to everything by default
- Complete protection for all processes on the system
- Requires that policies be written for **all** applications, often requires customization
- Strict is type enforcement with added types for users (e.g. user_t and user_firefox_t).
- Not enabled by Red Hat as default

The Three SELinux Policy Types

3) **Multi-Level Security (MLS)**

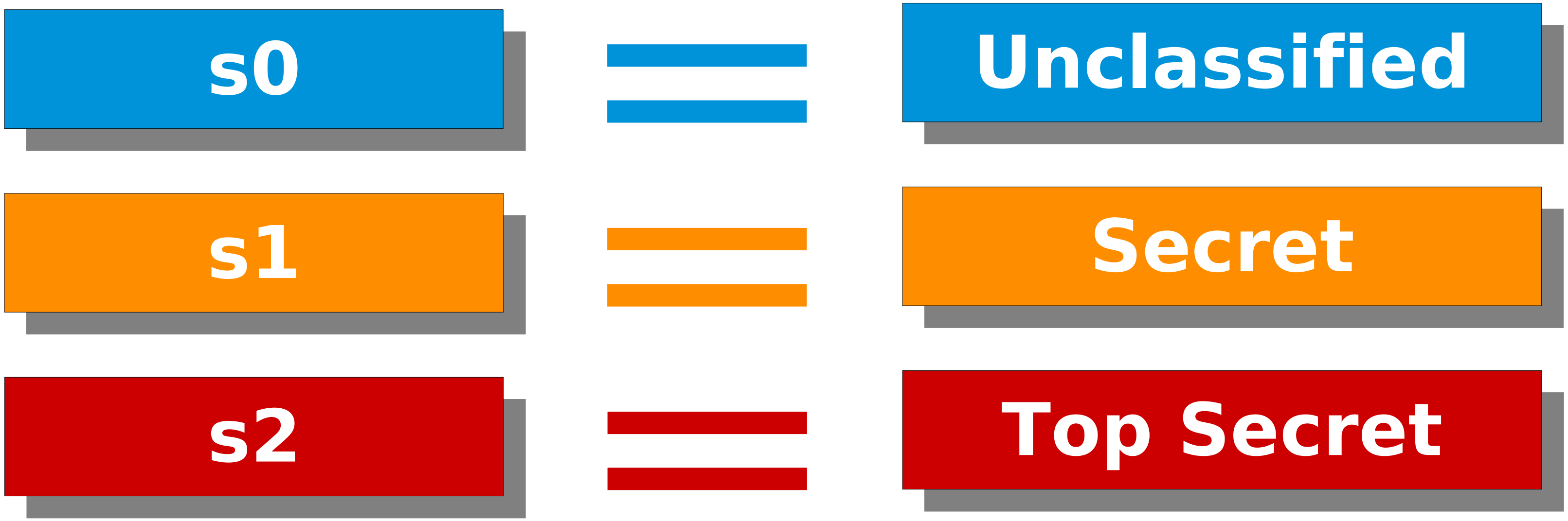
- Focuses on confidentiality (i.e. separation of multiple classifications of data)
- Ability to manage {processes, users} with varying levels of access. (i.e. “*the need to know*”)
- Uses category & sensitivity levels

The Three SELinux Policy Types

3) **Multi-Level Security (MLS)**

(a) **Sensitivity Labels**

- Mostly used by the government – Top Secret, Secret, Unclassified, etc



s0 = Unclassified

s1 = Secret

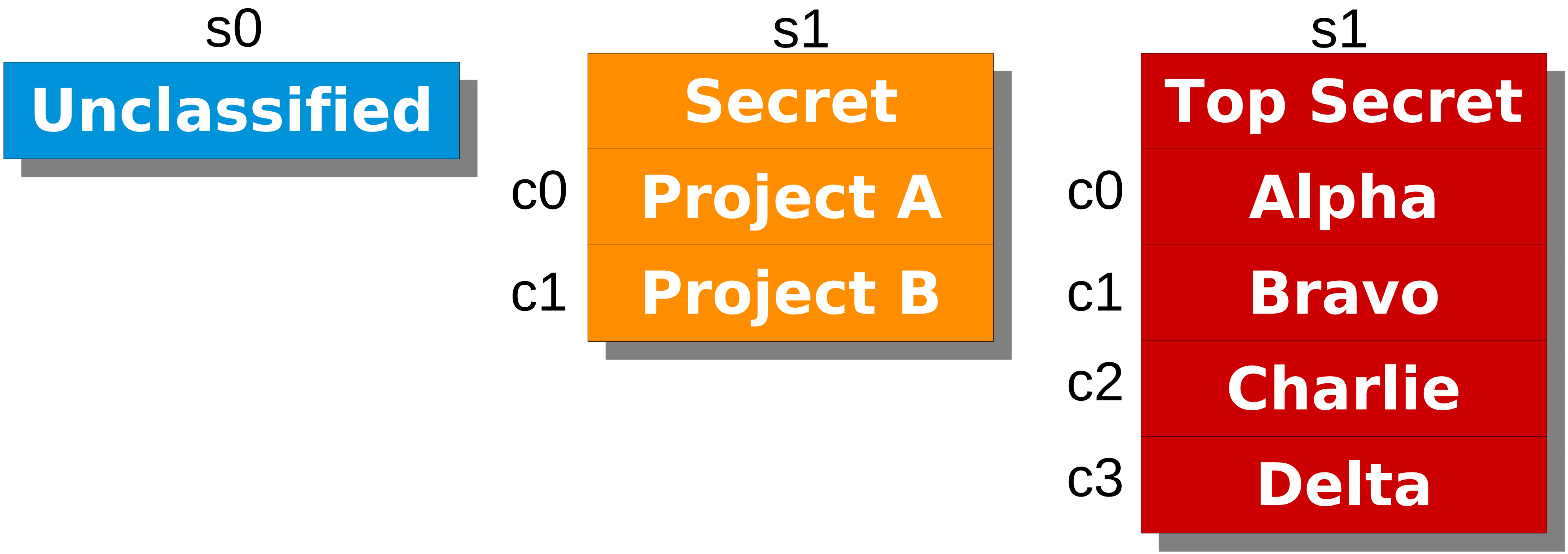
s2 = Top Secret

The Three SELinux Policy Types

3) **Multi-Level Security (MLS)**

(b) **Category Labels**

- Separation of data types, compartments, projects, etc



s0
Unclassified

s1
Secret
c0 Project A
c1 Project B

s1
Top Secret
c0 Alpha
c1 Bravo
c2 Charlie
c3 Delta

The Three SELinux Policy Types

3) **Multi-Level Security (MLS)**

(b) **Polyinstantiation & pam_namespace**

- The pam_namespace PAM module sets up a private namespace for a session with polyinstantiated directories
- A polyinstantiated directory provides a different instance of itself based on user name, or when using SELinux, user name, security context or both

The Three SELinux Policy Types

3) Multi-Level Security (MLS)

(b) Polyinstantiation & pam_namespace

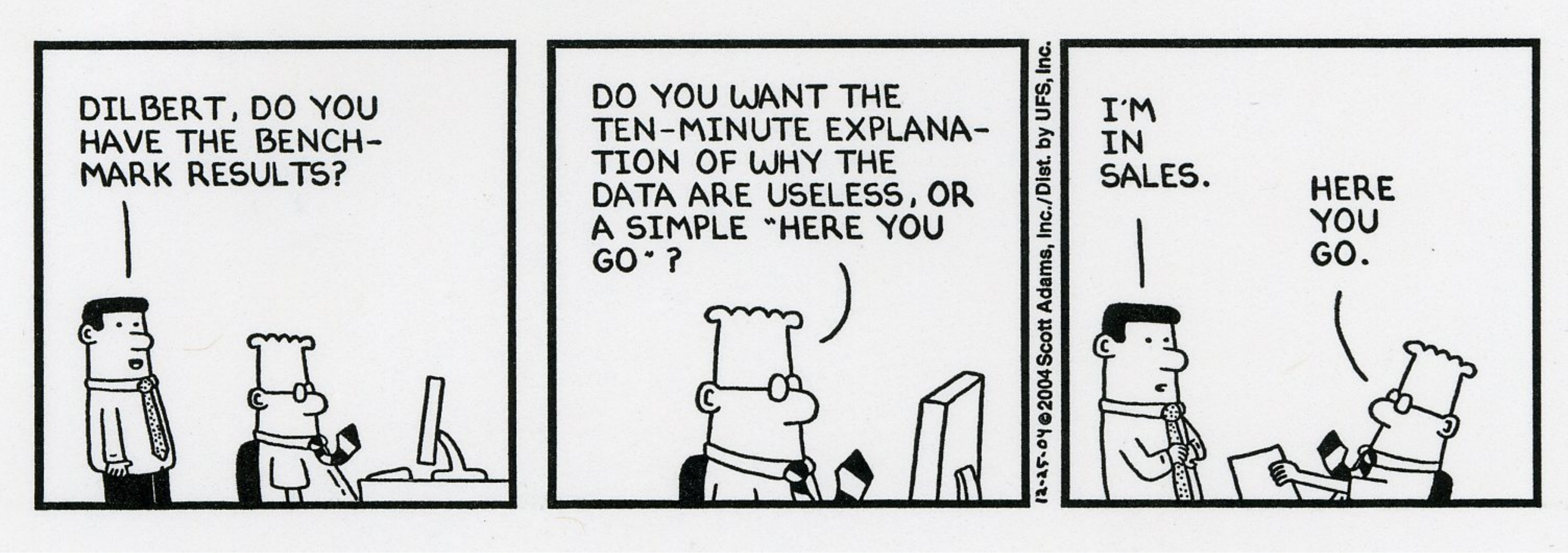
```
# id -Z
staff_u:WebServer_Admin_r:WebServer_Admin_t:s0:c0
# ls -l /data
secret-file-1
secret-file 2

# id -Z
staff_u:WebServer_Admin_r:WebServer_Admin_t:s1:c0
# ls -l /data
secret-file-1
secret-file 2
top-secret-file-1
```

The Three SELinux Policy Types

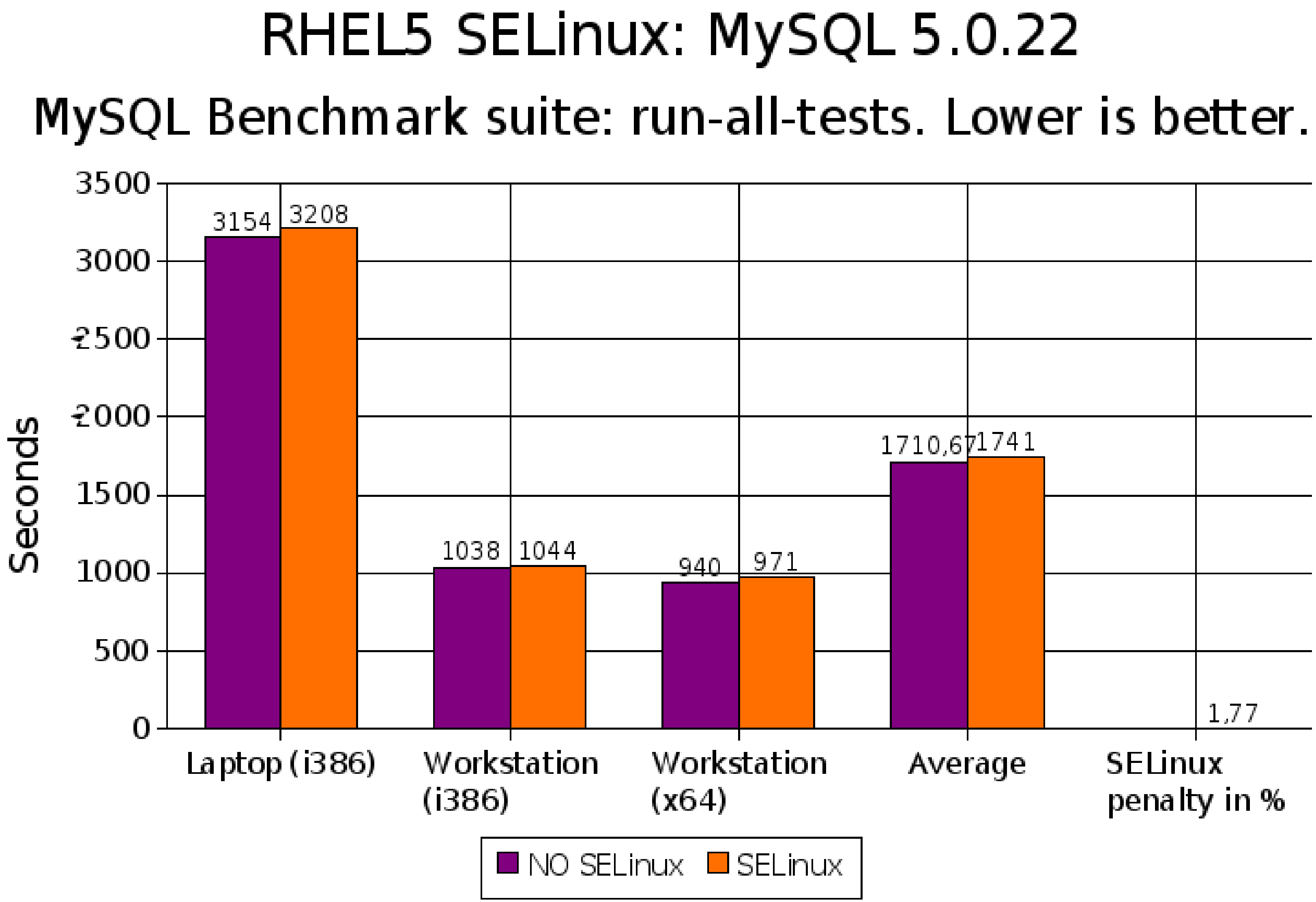
Multi-Level Security (MLS) & Common Criteria

- The Common Criteria (CC) is an international security standard against which systems are evaluated. Many government customers require CC evaluated systems.
- Red Hat Enterprise Linux 5 meets EAL4+ with RBAC/LSPP/CAPP endorcements

What's the Performance Overhead?


DILBERT, DO YOU HAVE THE BENCHMARK RESULTS?
DO YOU WANT THE TEN-MINUTE EXPLANATION OF WHY THE DATA ARE USELESS, OR A SIMPLE "HERE YOU GO"?
I'M IN SALES.
HERE YOU GO.
12-15-04 ©2004 Scott Adams, Inc./Dist. by UFS, Inc.

What's the Performance Overhead?



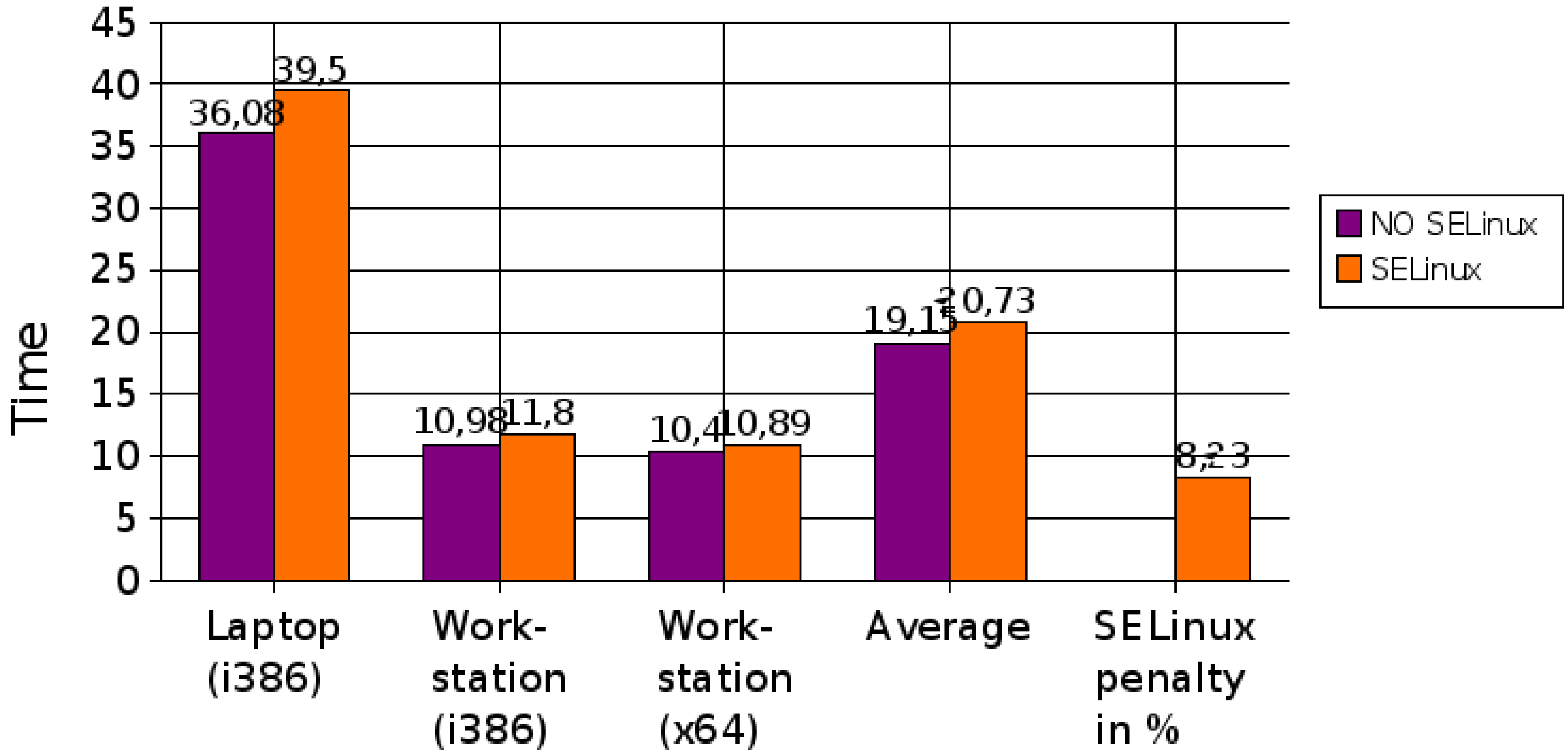
RHEL5 SELinux: MySQL 5.0.22

MySQL Benchmark suite: run-all-tests. Lower is better.

What's the Performance Overhead?

RHEL5 SELinux: Apache 2.2.3 (worker)

11 tests: 100000 requests with 1-255 concurrent connections. Lower is better.



What's the Performance Overhead?

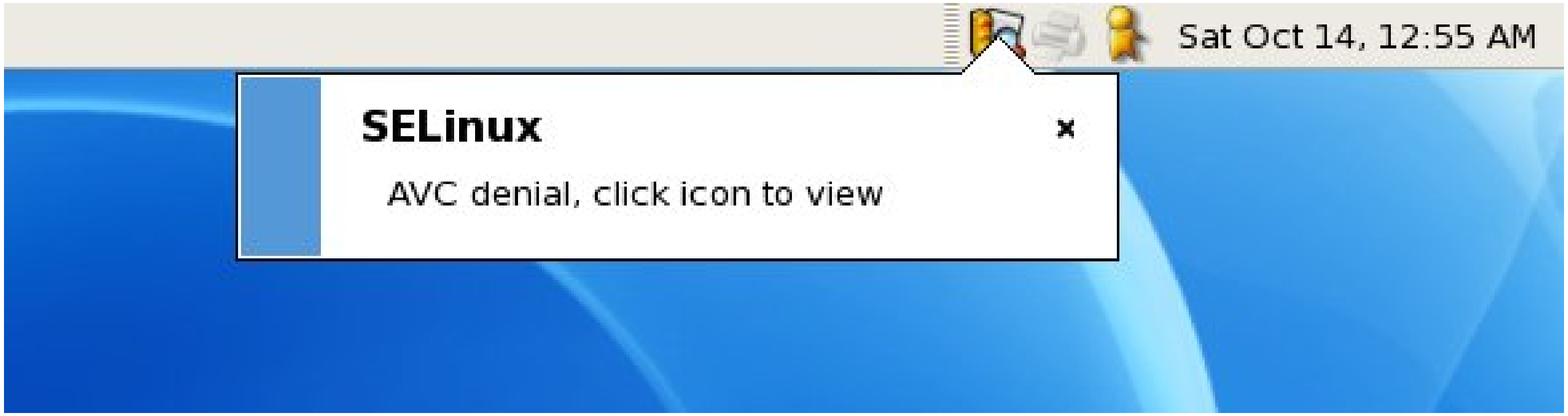
- Not official statistics
- Laptop = 2GHz, 2x 1GB RAM
- Workstation = 2.13GHz, 4x 1GB RAM
- Apache = Lots of threads
- MySQL = Lots of disk I/O

SELinux Usage

(GUI & console)

End-User Perspective

- sealert Notifications



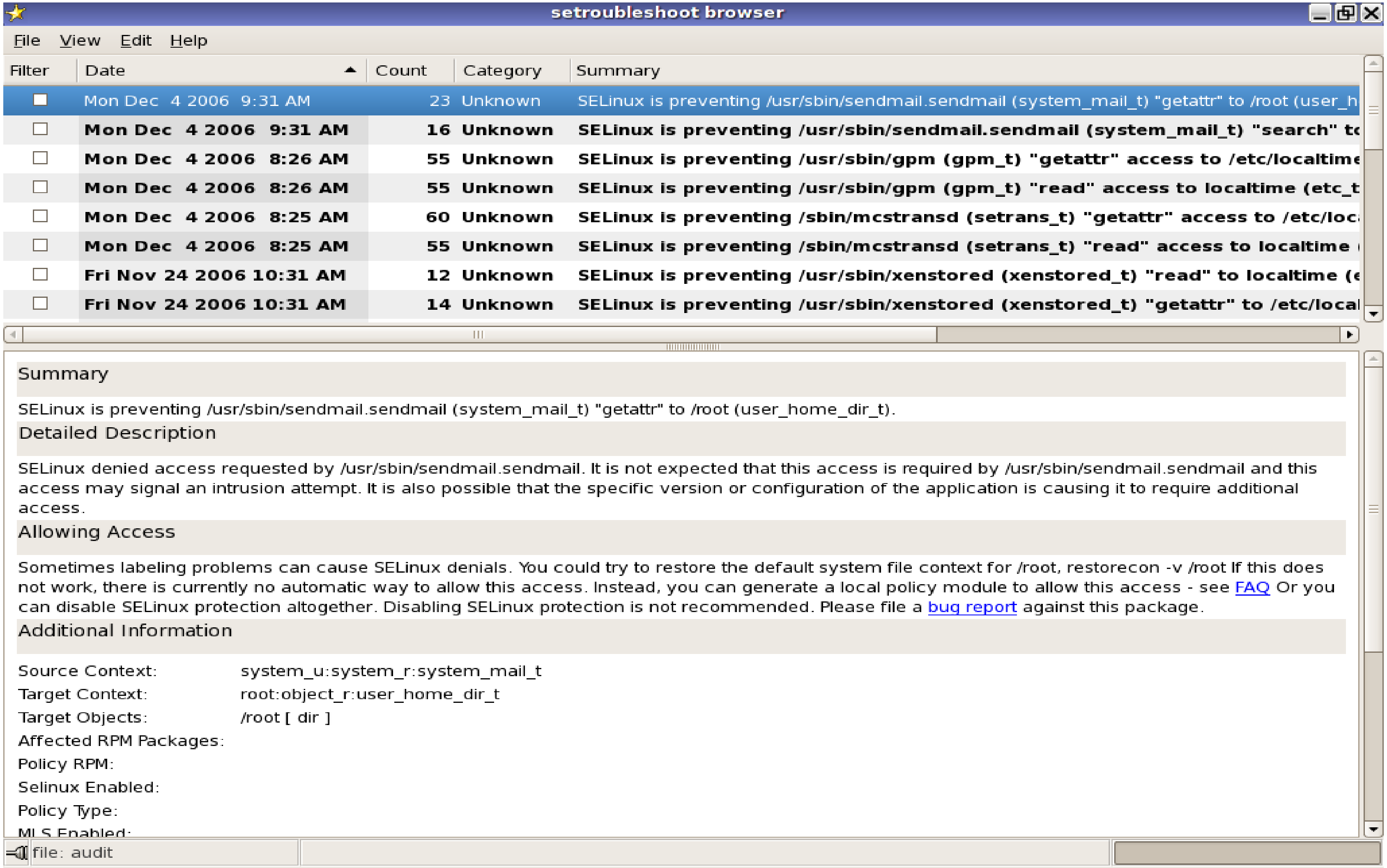
Sat Oct 14, 12:55 AM

SELinux

AVC denial, click icon to view

End-User Perspective

- sealert Browser



Filter	Date	Count	Category	Summary
☐	Mon Dec 4 2006 9:31 AM	23	Unknown	SELinux is preventing /usr/sbin/sendmail.sendmail (system_mail_t) "getattr" to /root (user_h
☐	Mon Dec 4 2006 9:31 AM	16	Unknown	SELinux is preventing /usr/sbin/sendmail.sendmail (system_mail_t) "search" to
☐	Mon Dec 4 2006 8:26 AM	55	Unknown	SELinux is preventing /usr/sbin/gpm (gpm_t) "getattr" access to /etc/localtime
☐	Mon Dec 4 2006 8:26 AM	55	Unknown	SELinux is preventing /usr/sbin/gpm (gpm_t) "read" access to localtime (etc_t
☐	Mon Dec 4 2006 8:25 AM	60	Unknown	SELinux is preventing /sbin/mcstransd (setrans_t) "getattr" access to /etc/loc
☐	Mon Dec 4 2006 8:25 AM	55	Unknown	SELinux is preventing /sbin/mcstransd (setrans_t) "read" access to localtime (
☐	Fri Nov 24 2006 10:31 AM	12	Unknown	SELinux is preventing /usr/sbin/xenstored (xenstored_t) "read" to localtime (e
☐	Fri Nov 24 2006 10:31 AM	14	Unknown	SELinux is preventing /usr/sbin/xenstored (xenstored_t) "getattr" to /etc/local

Summary

SELinux is preventing /usr/sbin/sendmail.sendmail (system_mail_t) "getattr" to /root (user_home_dir_t).

Detailed Description

SELinux denied access requested by /usr/sbin/sendmail.sendmail. It is not expected that this access is required by /usr/sbin/sendmail.sendmail and this access may signal an intrusion attempt. It is also possible that the specific version or configuration of the application is causing it to require additional access.

Allowing Access

Sometimes labeling problems can cause SELinux denials. You could try to restore the default system file context for /root, restorecon -v /root If this does not work, there is currently no automatic way to allow this access. Instead, you can generate a local policy module to allow this access - see FAQ Or you can disable SELinux protection altogether. Disabling SELinux protection is not recommended. Please file a bug report against this package.

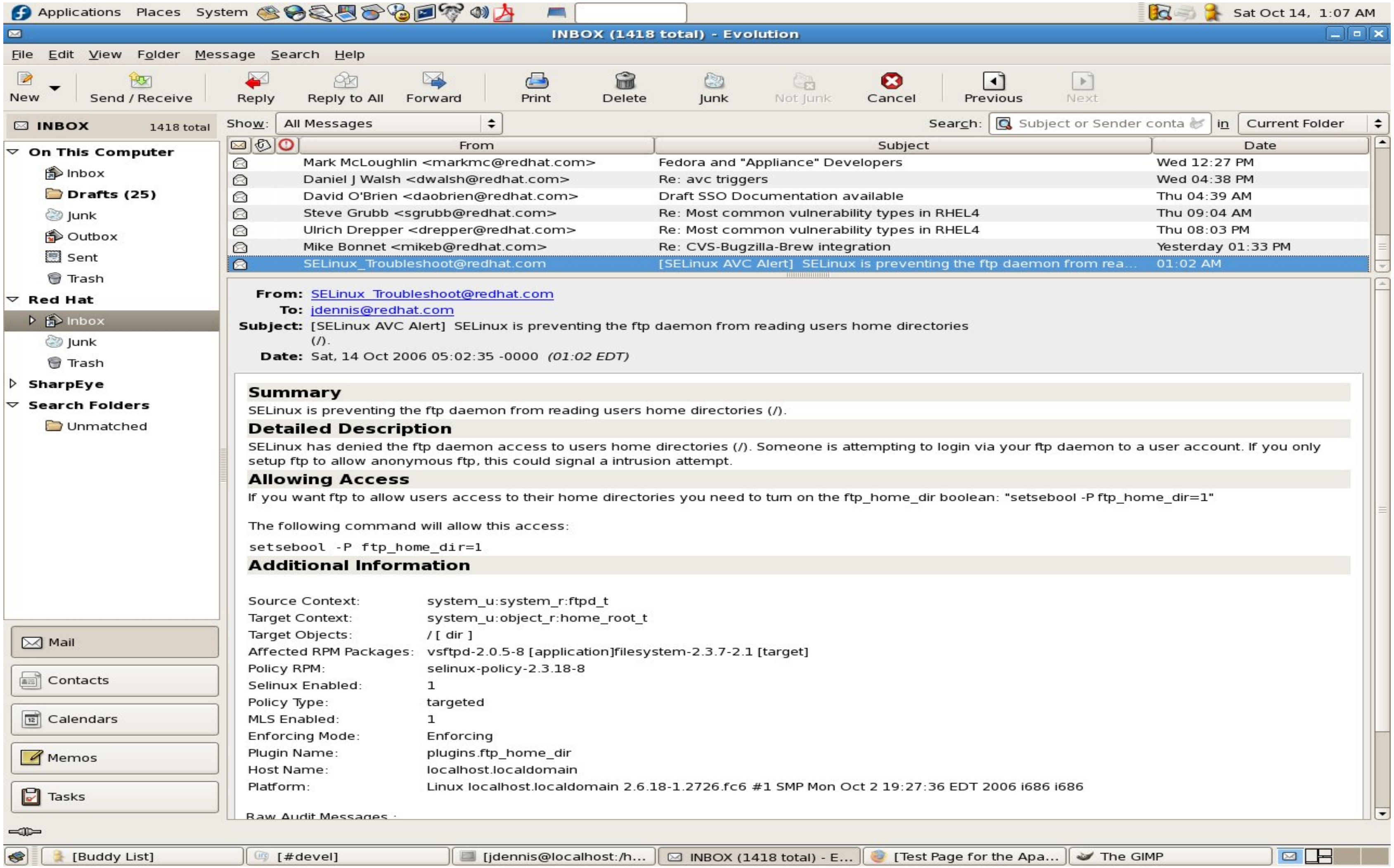
Additional Information

Source Context:	system_u:system_r:system_mail_t
Target Context:	root:object_r:user_home_dir_t
Target Objects:	/root [dir]
Affected RPM Packages:	
Policy RPM:	
Selinux Enabled:	
Policy Type:	
MLS Enabled:	

file: audit

System Administrator Perspective

- **sealert + EMail Notifications**



From	Subject	Date
Mark McLoughlin <markmc@redhat.com>	Fedora and "Appliance" Developers	Wed 12:27 PM
Daniel J Walsh <dwalsh@redhat.com>	Re: avc triggers	Wed 04:38 PM
David O'Brien <daobrien@redhat.com>	Draft SSO Documentation available	Thu 04:39 AM
Steve Grubb <sgrubb@redhat.com>	Re: Most common vulnerability types in RHEL4	Thu 09:04 AM
Ulrich Drepper <drepper@redhat.com>	Re: Most common vulnerability types in RHEL4	Thu 08:03 PM
Mike Bonnet <mikeb@redhat.com>	Re: CVS-Bugzilla-Brew integration	Yesterday 01:33 PM
SELinux_Troubleshoot@redhat.com	[SELinux AVC Alert] SELinux is preventing the ftp daemon from rea...	01:02 AM

From: SELinux_Troubleshoot@redhat.com
To: jdennis@redhat.com
Subject: [SELinux AVC Alert] SELinux is preventing the ftp daemon from reading users home directories (/).
Date: Sat, 14 Oct 2006 05:02:35 -0000 *(01:02 EDT)*

Summary

SELinux is preventing the ftp daemon from reading users home directories (/).

Detailed Description

SELinux has denied the ftp daemon access to users home directories (/). Someone is attempting to login via your ftp daemon to a user account. If you only setup ftp to allow anonymous ftp, this could signal a intrusion attempt.

Allowing Access

If you want ftp to allow users access to their home directories you need to turn on the ftp_home_dir boolean: "setsebool -P ftp_home_dir=1"

The following command will allow this access:

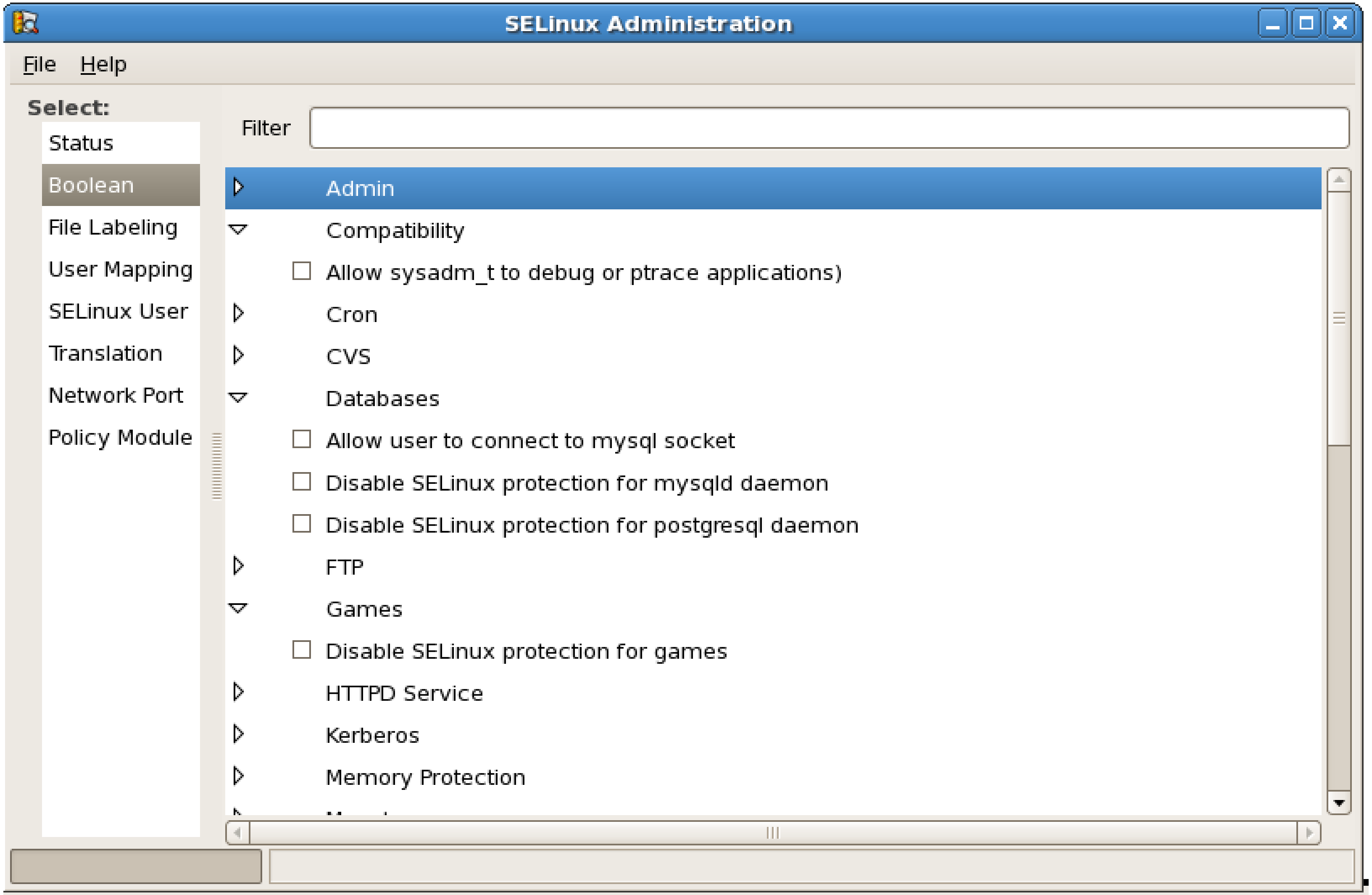
```
setsebool -P ftp_home_dir=1
```

Additional Information

```
Source Context:          system_u:system_r:ftpd_t
Target Context:          system_u:object_r:home_root_t
Target Objects:          / [ dir ]
Affected RPM Packages:   vsftpd-2.0.5-8 [application]filesystem-2.3.7-2.1 [target]
Policy RPM:              selinux-policy-2.3.18-8
Selinux Enabled:         1
Policy Type:             targeted
MLS Enabled:             1
Enforcing Mode:          Enforcing
Plugin Name:             plugins.ftp_home_dir
Host Name:               localhost.localdomain
Platform:                Linux localhost.localdomain 2.6.18-1.2726.fc6 #1 SMP Mon Oct 2 19:27:36 EDT 2006 i686 i686

Raw Audit Messages :
```

System Administrator Perspective

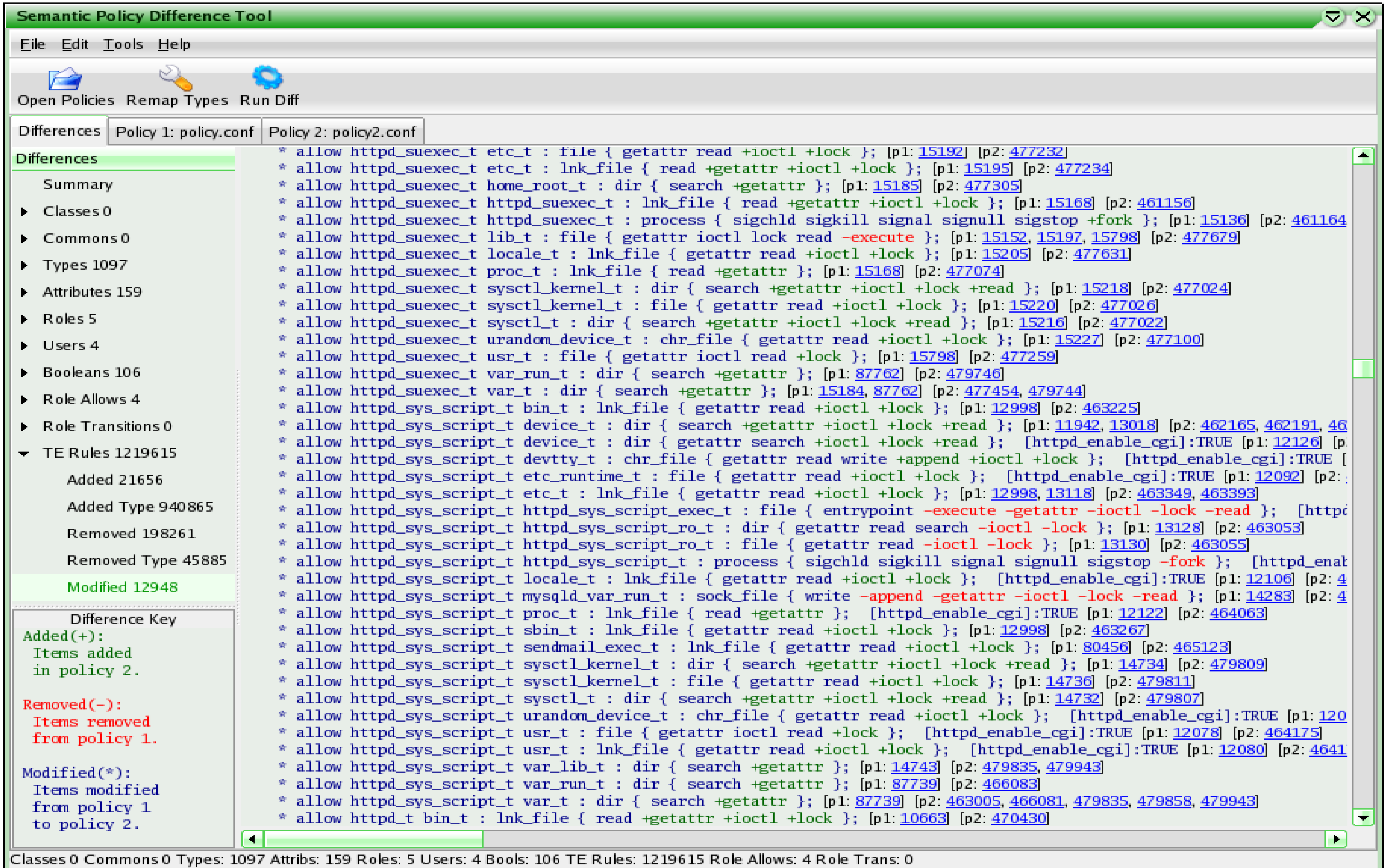
- **system-config-selinux**



- ▷ Admin
- ▽ Compatibility
  - □ Allow sysadm_t to debug or ptrace applications)
- ▷ Cron
- ▷ CVS
- ▽ Databases
  - □ Allow user to connect to mysql socket
  - □ Disable SELinux protection for mysqld daemon
  - □ Disable SELinux protection for postgresql daemon
- ▷ FTP
- ▽ Games
  - □ Disable SELinux protection for games
- ▷ HTTPD Service
- ▷ Kerberos
- ▷ Memory Protection

System Administrator Perspective

- **sediffx**



```
Differences
    Summary
  ▸ Classes 0
  ▸ Commons 0
  ▸ Types 1097
  ▸ Attributes 159
  ▸ Roles 5
  ▸ Users 4
  ▸ Booleans 106
  ▸ Role Allows 4
  ▸ Role Transitions 0
  ▾ TE Rules 1219615
        Added 21656
        Added Type 940865
        Removed 198261
        Removed Type 45885
        Modified 12948
```

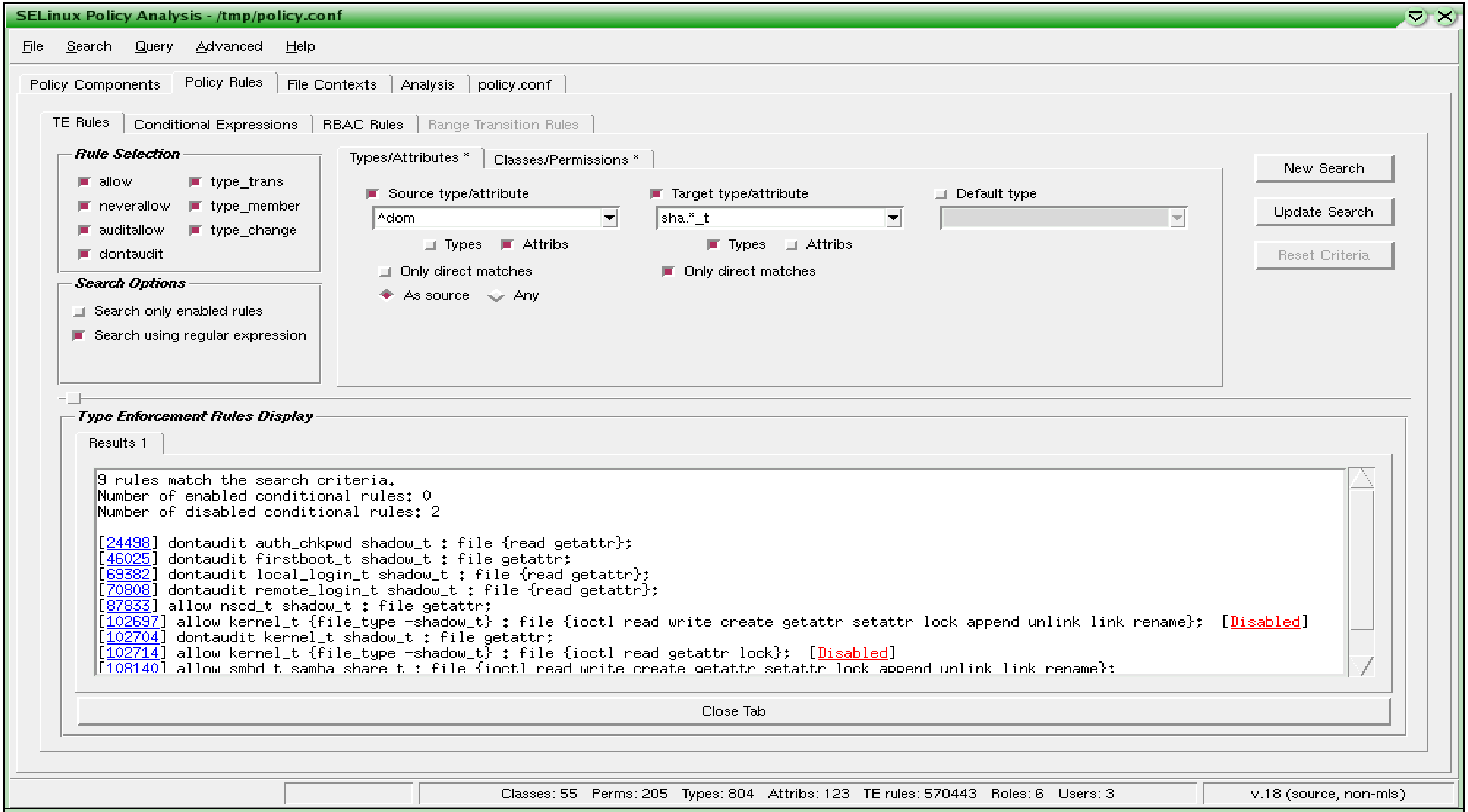
```
Difference Key
Added(+):
  Items added
  in policy 2.

Removed(-):
  Items removed
  from policy 1.

Modified(*):
  Items modified
  from policy 1
  to policy 2.
```

```
* allow httpd_suexec_t etc_t : file { getattr read +ioctl +lock }; [p1: 15192] [p2: 477232]
* allow httpd_suexec_t etc_t : lnk_file { read +getattr +ioctl +lock }; [p1: 15195] [p2: 477234]
* allow httpd_suexec_t home_root_t : dir { search +getattr }; [p1: 15185] [p2: 477305]
* allow httpd_suexec_t httpd_suexec_t : lnk_file { read +getattr +ioctl +lock }; [p1: 15168] [p2: 461156]
* allow httpd_suexec_t httpd_suexec_t : process { sigchld sigkill signal signull sigstop +fork }; [p1: 15136] [p2: 461164
* allow httpd_suexec_t lib_t : file { getattr ioctl lock read -execute }; [p1: 15152, 15197, 15798] [p2: 477679]
* allow httpd_suexec_t locale_t : lnk_file { getattr read +ioctl +lock }; [p1: 15205] [p2: 477631]
* allow httpd_suexec_t proc_t : lnk_file { read +getattr }; [p1: 15168] [p2: 477074]
* allow httpd_suexec_t sysctl_kernel_t : dir { search +getattr +ioctl +lock +read }; [p1: 15218] [p2: 477024]
* allow httpd_suexec_t sysctl_kernel_t : file { getattr read +ioctl +lock }; [p1: 15220] [p2: 477026]
* allow httpd_suexec_t sysctl_t : dir { search +getattr +ioctl +lock +read }; [p1: 15216] [p2: 477022]
* allow httpd_suexec_t urandom_device_t : chr_file { getattr read +ioctl +lock }; [p1: 15227] [p2: 477100]
* allow httpd_suexec_t usr_t : file { getattr ioctl read +lock }; [p1: 15798] [p2: 477259]
* allow httpd_suexec_t var_run_t : dir { search +getattr }; [p1: 87762] [p2: 479746]
* allow httpd_suexec_t var_t : dir { search +getattr }; [p1: 15184, 87762] [p2: 477454, 479744]
* allow httpd_sys_script_t bin_t : lnk_file { getattr read +ioctl +lock }; [p1: 12998] [p2: 463225]
* allow httpd_sys_script_t device_t : dir { search +getattr +ioctl +lock +read }; [p1: 11942, 13018] [p2: 462165, 462191, 46
* allow httpd_sys_script_t device_t : dir { getattr search +ioctl +lock +read };  [httpd_enable_cgi]:TRUE [p1: 12126] [p.
* allow httpd_sys_script_t devtty_t : chr_file { getattr read write +append +ioctl +lock };  [httpd_enable_cgi]:TRUE [
* allow httpd_sys_script_t etc_runtime_t : file { getattr read +ioctl +lock };  [httpd_enable_cgi]:TRUE [p1: 12092] [p2:
* allow httpd_sys_script_t etc_t : lnk_file { getattr read +ioctl +lock }; [p1: 12998, 13118] [p2: 463349, 463393]
* allow httpd_sys_script_t httpd_sys_script_exec_t : file { entrypoint -execute -getattr -ioctl -lock -read };  [httpd
* allow httpd_sys_script_t httpd_sys_script_ro_t : dir { getattr read search -ioctl -lock }; [p1: 13128] [p2: 463053]
* allow httpd_sys_script_t httpd_sys_script_ro_t : file { getattr read -ioctl -lock }; [p1: 13130] [p2: 463055]
* allow httpd_sys_script_t httpd_sys_script_t : process { sigchld sigkill signal signull sigstop -fork };  [httpd_enab
* allow httpd_sys_script_t locale_t : lnk_file { getattr read +ioctl +lock };  [httpd_enable_cgi]:TRUE [p1: 12106] [p2: 4
* allow httpd_sys_script_t mysqld_var_run_t : sock_file { write -append -getattr -ioctl -lock -read }; [p1: 14283] [p2: 4
* allow httpd_sys_script_t proc_t : lnk_file { read +getattr };  [httpd_enable_cgi]:TRUE [p1: 12122] [p2: 464063]
* allow httpd_sys_script_t sbin_t : lnk_file { getattr read +ioctl +lock }; [p1: 12998] [p2: 463267]
* allow httpd_sys_script_t sendmail_exec_t : lnk_file { getattr read +ioctl +lock }; [p1: 80456] [p2: 465123]
* allow httpd_sys_script_t sysctl_kernel_t : dir { search +getattr +ioctl +lock +read }; [p1: 14734] [p2: 479809]
* allow httpd_sys_script_t sysctl_kernel_t : file { getattr read +ioctl +lock }; [p1: 14736] [p2: 479811]
* allow httpd_sys_script_t sysctl_t : dir { search +getattr +ioctl +lock +read }; [p1: 14732] [p2: 479807]
* allow httpd_sys_script_t urandom_device_t : chr_file { getattr read +ioctl +lock };  [httpd_enable_cgi]:TRUE [p1: 120
* allow httpd_sys_script_t usr_t : file { getattr ioctl read +lock };  [httpd_enable_cgi]:TRUE [p1: 12078] [p2: 464175]
* allow httpd_sys_script_t usr_t : lnk_file { getattr read +ioctl +lock };  [httpd_enable_cgi]:TRUE [p1: 12080] [p2: 4641
* allow httpd_sys_script_t var_lib_t : dir { search +getattr }; [p1: 14743] [p2: 479835, 479943]
* allow httpd_sys_script_t var_run_t : dir { search +getattr }; [p1: 87739] [p2: 466083]
* allow httpd_sys_script_t var_t : dir { search +getattr }; [p1: 87739] [p2: 463005, 466081, 479835, 479858, 479943]
* allow httpd_t bin_t : lnk_file { read +getattr +ioctl +lock }; [p1: 10663] [p2: 470430]
```

```
Classes 0 Commons 0 Types: 1097 Attribs: 159 Roles: 5 Users: 4 Bools: 106 TE Rules: 1219615 Role Allows: 4 Role Trans: 0
```

System Administrator Perspective

- apol



Rule Selection

☒ allow ☒ type_trans
☒ neverallow ☒ type_member
☒ auditallow ☒ type_change
☒ dontaudit

Search Options

□ Search only enabled rules
☒ Search using regular expression

☒ Source type/attribute: ^dom
□ Types ☒ Attribs
□ Only direct matches
● As source ○ Any

☒ Target type/attribute: sha.*_t
☒ Types □ Attribs
☒ Only direct matches

□ Default type

```
9 rules match the search criteria.
Number of enabled conditional rules: 0
Number of disabled conditional rules: 2

[24498] dontaudit auth_chkpwd shadow_t : file {read getattr};
[46025] dontaudit firstboot_t shadow_t : file getattr;
[69382] dontaudit local_login_t shadow_t : file {read getattr};
[70808] dontaudit remote_login_t shadow_t : file {read getattr};
[87833] allow nscd_t shadow_t : file getattr;
[102697] allow kernel_t {file_type -shadow_t} : file {ioctl read write create getattr setattr lock append unlink link rename};  [Disabled]
[102704] dontaudit kernel_t shadow_t : file getattr;
[102714] allow kernel_t {file_type -shadow_t} : file {ioctl read getattr lock};  [Disabled]
[108140] allow smbd_t samba_share_t : file {ioctl read write create getattr setattr lock append unlink link rename};
```

Classes: 55 Perms: 205 Types: 804 Attribs: 123 TE rules: 570443 Roles: 6 Users: 3

v.18 (source, non-mls)

System Administrator Perspective

- **semanage**

  Configure elements of SELinux policy without modification/recompilation of policy sources

  . . . . aka on the fly

Example: Dynamically Allowing Apache to listen on port 1234

```
# semanage port -a -t httpd_port_t -p tcp 1234
```

System Administrator Perspective

- **semanage** (more examples)

Example: Allow shawn to join “webadmin_u” group

```
# semanage login -a -s webadmin_u shawn
```

Example: Relabel files for access by Apache

```
# semanage fcontext -a -t \
  httpd_sys_content_t "/data/webpages(/.*)?"
```

System Administrator Perspective

- **semanage** (most important example)

You don't need to disable SELinux to fix a single error!

```
type=SYSCALL msg=audit(1204719775.306:738): arch=40000003 syscall=54
success=no exit=-19 a0=4 a1=8933 a2=bfcec1bc a3=bfcec1bc items=0
ppid=3900 pid=5003 auid=501 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0
sgid=0 fsgid=0 tty=(none) comm="ip" exe="/sbin/ip"
subj=user_u:system_r:ifconfig_t:s0 key=(null)
```

The Fix:

```
# semanage permissive -a ifconfig_t
```

System Administrator Perspective

- **audit2allow**

  Allows generation of SELinux policy rules from logs of denied operations

Example: Fix all the errors on the system (completely not a good idea on a real system)

```
# cat /var/log/audit/audit.log | audit2allow -M FixAll
Generating type enforcment file: FixAll.te
Compiling policy: checkmodule -M -m -o FixAll.mod FixAll.te
Building package: semodule_package -o FixAll.pp -m FixAll.mod

# semodule -i FixAll.pp
```

Scenarios

Scenario: Fixing the RHT corporate VPN “update”

- Red Hat has a Corporate Standard Build (CSB) for desktop environments
- Red Hat pushes updates to said CSB
- I “tweak” my configuration files
- When RHT pushed a CSB update, it broke my VPN settings

Scenario: Fixing the RHT corporate VPN “update”

/var/log/messages:

```
type=SYSCALL msg=audit(1204719775.306:738): arch=40000003 syscall=54
success=no exit=-19 a0=4 a1=8933 a2=bfcec1bc a3=bfcec1bc items=0
ppid=3900 pid=5003 auid=501 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0
sgid=0 fsgid=0 tty=(none) comm="ip" exe="/sbin/ip"
subj=user_u:system_r:ifconfig_t:s0 key=(null)
```

Now what?

Scenario: Fixing the RHT corporate VPN “update”

```
type=SYSCALL msg=audit(1204719775.306:738): arch=40000003 syscall=54
success=no exit=-19 a0=4 a1=8933 a2=bfcec1bc a3=bfcec1bc items=0
ppid=3900 pid=5003 auid=501 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0
sgid=0 fsgid=0 tty=(none) comm="ip" exe="/sbin/ip"
subj=user_u:system_r:ifconfig_t:s0 key=(null)
```

What I Know:

1) AVC Event ID 738
2) syscall=54 (I'd have to google this)
3) root (or an application on its behalf) was running /sbin/ip
4) context = user_u:system_r:ifconfig_t:s0

Scenario: Fixing the RHT corporate VPN “update”

```
type=SYSCALL msg=audit(1204719775.306:738): arch=40000003 syscall=54
success=no exit=-19 a0=4 a1=8933 a2=bfcec1bc a3=bfcec1bc items=0
ppid=3900 pid=5003 auid=501 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0
sgid=0 fsgid=0 tty=(none) comm="ip" exe="/sbin/ip"
subj=user_u:system_r:ifconfig_t:s0 key=(null)
```

My Options:

1) Create a SELinux Policy Module

```
# ausearch -x “/sbin/ip” | audit2allow -M MyVPNFix
```

Scenario: Fixing the RHT corporate VPN “update”

```
type=SYSCALL msg=audit(1204719775.306:738): arch=40000003 syscall=54
success=no exit=-19 a0=4 a1=8933 a2=bfcec1bc a3=bfcec1bc items=0
ppid=3900 pid=5003 auid=501 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0
sgid=0 fsgid=0 tty=(none) comm="ip" exe="/sbin/ip"
subj=user_u:system_r:ifconfig_t:s0 key=(null)
```

My Options:

1) Create a SELinux Policy Module

```
# ausearch -x “/sbin/ip” | audit2allow -M MyVPNFix
# semodule -i MyVPNFix.pp
```

Scenario: Fixing the RHT corporate VPN “update”

```
type=SYSCALL msg=audit(1204719775.306:738): arch=40000003 syscall=54
success=no exit=-19 a0=4 a1=8933 a2=bfcec1bc a3=bfcec1bc items=0
ppid=3900 pid=5003 auid=501 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0
sgid=0 fsgid=0 tty=(none) comm="ip" exe="/sbin/ip"
subj=user_u:system_r:ifconfig_t:s0 key=(null)
```

My Options:

2) Disable enforcement of ifconfig_t (**there is no need to turn SELinux completely off!**)

```
# semanage permissive -a ifconfig_t
```

Questions