

SELinux: Useful Security for Your Security Problems

Chad Sellers
Lead Software Architect,
Technology Solutions Group
Tresys Technology, LLC
August 8, 2007



Outline

- Background on SELinux
- Useful things you can do with SELinux
 - examples of what you can do with SELinux
 - flexibility of SELinux
- New things in SELinux
 - making SELinux easier
 - making SELinux more useful

What is SELinux?

- Enhancement to Linux Kernel and utilities
 - not a stand-alone distribution
 - available as part of major distributions
- Incorporates flexible mandatory access control
 - primarily through type enforcement
 - cooperates with other security enhancements (execshield)
- Provides a rich and flexible MAC policy language
- Maintains binary compatibility for programs
 - unmodified Linux applications can be controlled

Mandatory Access Controls (MAC)

- Access Control in general
 - subjects and objects have security attributes
 - access determined based on policy rules
- Discretionary Access Control
 - users can change security attributes at request
 - allowing programs running on behalf of a user to affect the results of access rules
- Mandatory Access Control
 - users cannot change security attributes at request
 - user programs must work within the constraints of rules
 - MAC rules are controlled by the organization, not the user

Type Enforcement

- A type is an unambiguous identifier
 - created by the policy writer
 - applied to all subjects and objects and for access decisions
- Types group subjects and objects
 - signifies security equivalence
 - everything with the same type has the same access
 - policies have as few or as many types as needed

Type Enforcement

- Access specified between
 - subject type (e.g., process or domain)
 - and object type (e.g., file, dir, socket, etc.)
- Four elements in defining allowed access
 - source type(s) *aka domain(s)*
 - target type(s) *objects to which access allowed*
 - object class(es) *classes to which access applies*
 - permission(s) *type of access allowed*
- SELinux prevents access unless explicitly allowed

Where is SELinux being used

- Distributions
 - Fedora 2-7
 - Red Hat Enterprise Linux 4 and 5
 - Hardened Gentoo
 - Debian Etch
 - others
- Government applications
- Regular users
- Embedded applications

Solving your security problems

- SELinux is useful for solving your security problems
- Examples
 - protecting the system from an application under attack
 - protecting data from disclosure
 - protecting data integrity
 - containing untrusted programs
 - auditing violations

Application under attack

- Bulletin board application
 - frequent exploits
 - need to keep running
 - want to protect the rest of the system from it
- Exploits
 - compromise system
 - use for botnet

Application under attack

- SELinux can
 - limit bulletin board to local interaction with
 - web server
 - database server
 - reduce ability to upload bot
 - prevent bot from accessing the network
- Depending on application architecture
 - cgi-bin - complete isolation
 - mod_php, mod_python, etc.
 - limit to an apache-sized sandbox
- Demo

Protection from disclosure

- Access to customer records
 - web app authenticates user
 - web app grants access to the customer records
- Exploits to access those customer records
 - compromise web app itself
 - not much SELinux can do
 - compromise other portions of the web site
 - compromise something else on the system

Protection from disclosure

- SELinux can
 - limit access to customer data to the web app itself
 - other portions of the same website denied access
 - other processes on the same system denied access
 - support an architecture for protecting data
- Within the limits of SELinux
 - separation is at the process level
 - kernel exploits can bypass SELinux
 - base policy may give privileged domains access
- Demo

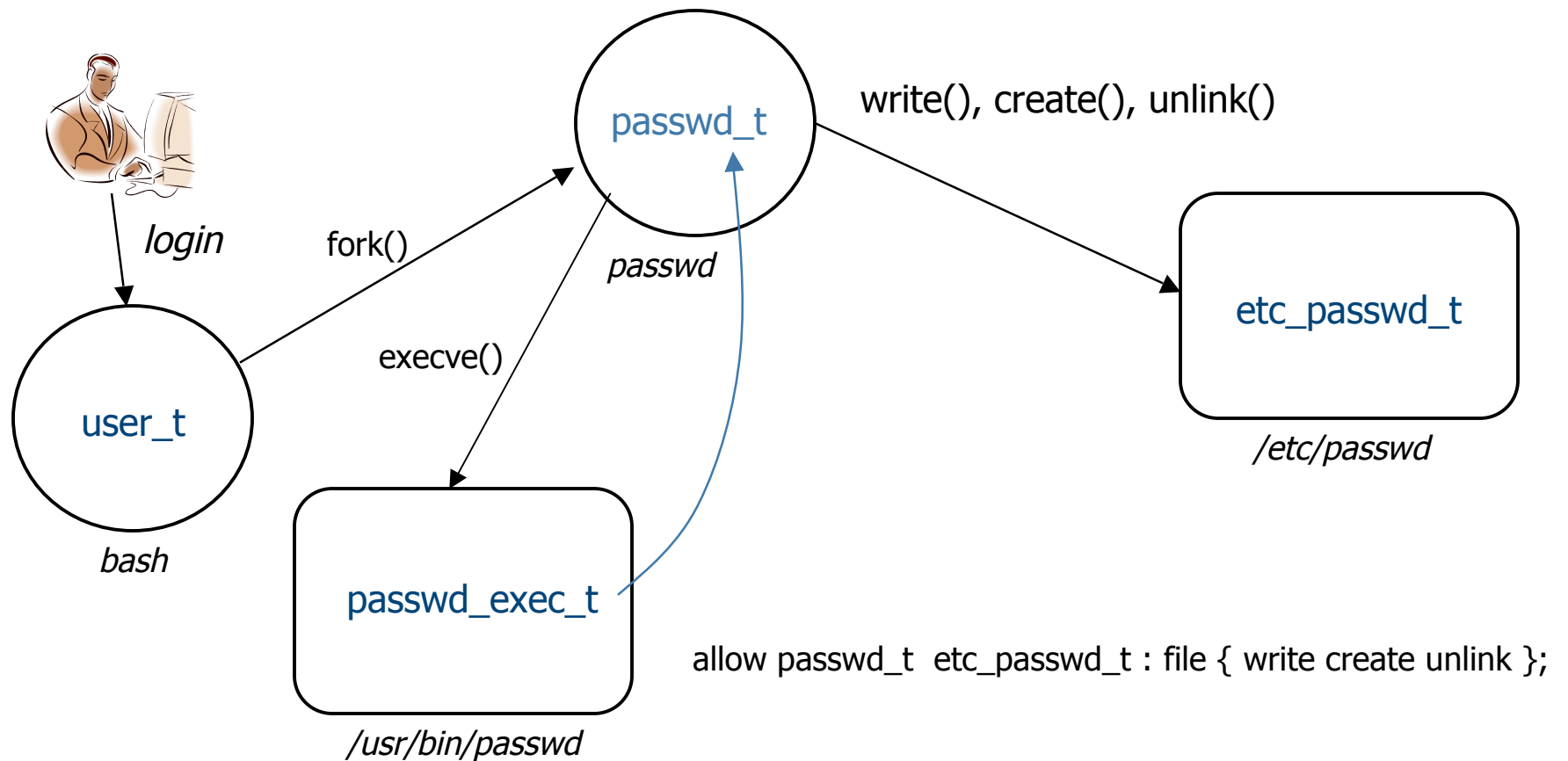
Protecting data integrity

- /etc/passwd
 - pervasive need to read
 - integrity of this file is very important
- Anyone who can modify this file can
 - change their UID to 0
 - lock out another user
 - etc.

Protecting data integrity

- SELinux can
 - limit write access to `/etc/passwd`
 - `/usr/bin/passwd` needs access
 - limit who can run `/usr/bin/passwd`
 - control `passwd`'s power based on who launched it
 - not really useful in this example though
- Still have to trust application to do its job
 - `/usr/bin/passwd` can write to `/etc/passwd`
 - trust it to only change the appropriate user line

Protecting data integrity



Containing untrusted programs

- Great new BitTorrent client
 - great reviews on your favorite forum
 - app written by some guy you've never heard of
- Running it considered dangerous
 - could contain backdoor or other malicious code
 - running it as a regular user insufficient
 - difficult to tell if it's misbehaving

Containing untrusted programs

- SELinux can
 - create a flexible sandbox for the application
 - prevent the application from reading user data
 - log any attempts to escape the sandbox
 - logs all denied accesses
 - can also log allowed actions, if desired
- Demo

SELinux is getting easier

- Improved upstream policy
- Management tools
- Policy development tools
- Debugging tools

Reference Policy

- A new SELinux policy that
 - reduces the complexity of writing, maintaining, and analyzing policy
 - uses modern software engineering principles
 - is well documented, modular, and configurable
 - provides a single source for all the policy variants
- Together this will make a policy that is...
 - maintainable
 - verifiable
 - usable

Reference Policy

Security Enhanced Linux Reference Policy

- + admin
 - acct
 - alsa
 - amanda
 - anaconda
 - apt
 - backup
 - bootloader
 - certwatch
 - consoletype
 - ddcprobe
 - dmesg
 - dmidecode
 - dpkg
 - firstboot
 - kudzu
 - logrotate
 - logwatch
 - mrtg
 - netutils
 - portage
 - prelink
 - quota
 - readahead
 - rpm
 - su
 - sudo
 - sxid
 - tmpreaper

Layer: admin

Policy modules for administrative functions, such as package management.

Module:	Description:
acct	Berkeley process accounting
alsa	Ainit ALSA configuration tool
amanda	Automated backup program.
anaconda	Policy for the Anaconda installer.
apt	APT advanced package toll.
backup	System backup scripts
bootloader	Policy for the kernel modules, kernel image, and bootloader.
certwatch	Digital Certificate Tracking
consoletype	Determine of the console connected to the controlling terminal.
ddcprobe	ddcprobe retrieves monitor and graphics card information
dmesg	Policy for dmesg.
dmidecode	Decode DMI data for x86/ia64 bioses.
dpkg	Policy for the Debian package manager.

http://oss.tresys.com/docs/refpolicy/api/services_apache.html

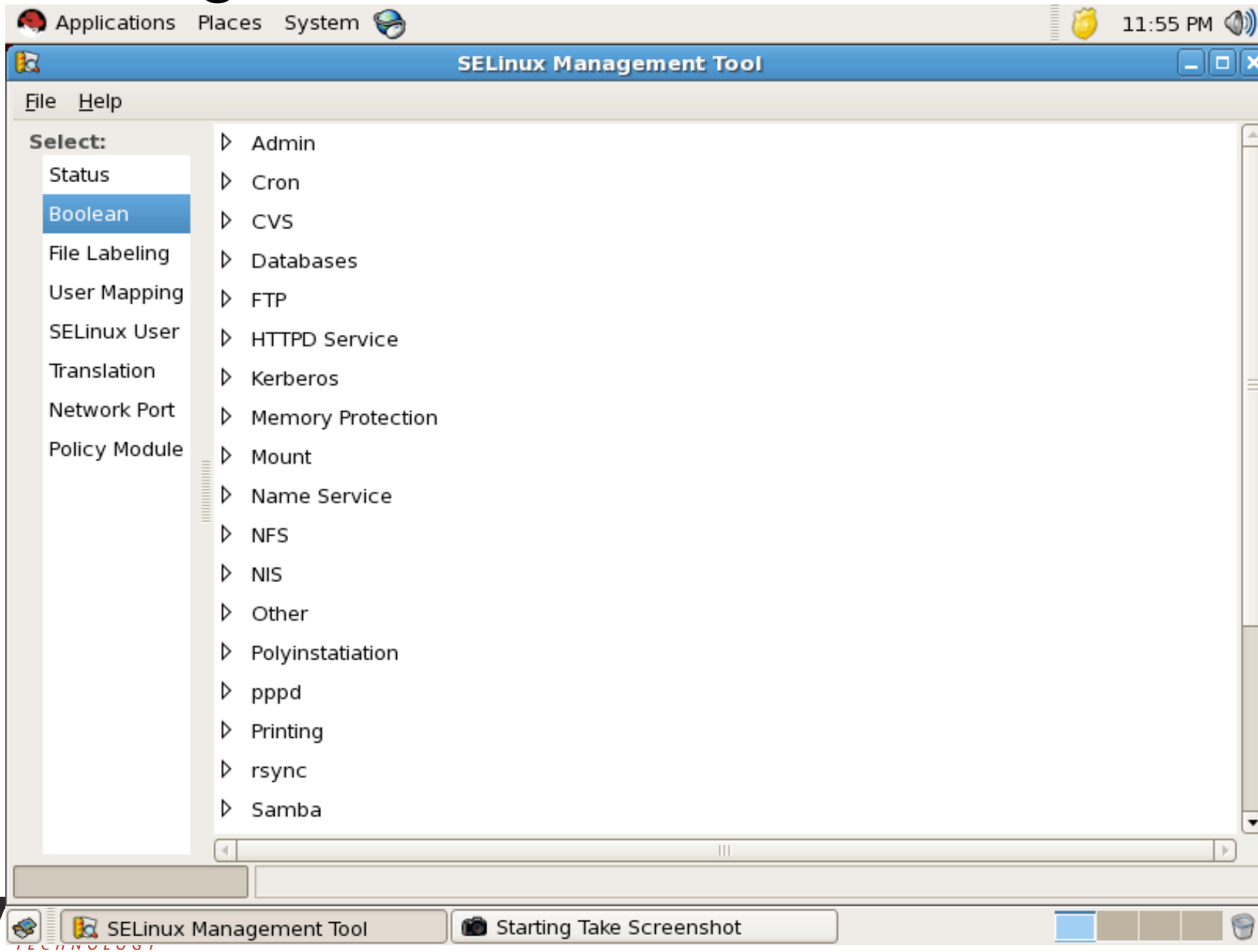
Management tools

- libsemanage
 - library for policy management
 - provides ability to make changes to the policy
- Tools built on libsemanage
 - semodule (policy modules)
 - semanage
 - GUIs (system-config-selinux)
- Commercial applications
 - Tresys Brickwall

Management tools

- Policy modules
 - present in current distributions
 - breaks policy into loadable modules
 - allows for local customization while still tracking vendor base policy
 - third-party module support
 - need only install modules for your applications

Management tools



Management tools

The screenshot displays the Tresys Brickwall Professional interface. The window title is "Tresys Brickwall Professional". The menu bar includes "File", "Edit", "Options", and "Help". Below the menu bar are icons for "Apply", "Revert", "Settings", and "Definitions".

The left sidebar shows a list of services, with "timesheet" selected under the "Custom" category. Other services listed include apache, cups, cyrus, dhcpd, dovecot, ftp, inn, jabber, kadmind, krb5kdc, ldap, mailman, mta, mysqld, named, nmbd, nscd, ntpd, openvpn, pegasus, portmap, postfix, postgresql, radius, sendmail, smbd, snmpd, squid, sshd, syslogd, winbind, ypbind, and Custom.

The main content area is titled "timesheet - Hour Tracking and Billing". It is divided into several sections:

- Network Interfaces**: A section for configuring network interfaces.
- Hosts**: A section for configuring hosts. It includes a table with columns "Name", "Addresses", and "RAW Networking".

Name	Addresses	RAW Networking
Database Server	10.1.4.45 / 255.255.255.255	<input type="checkbox"/>
Employee Subnet	10.100.0.0 / 255.255.0.0	<input checked="" type="checkbox"/>
- Remote Ports**: A section for configuring remote ports.
- Service Ports**: A section for configuring service ports. It includes a table with columns "Name" and "Ports".

Name	Ports
Timesheet Service	6262:tcp
- Security Options**: A section for configuring security options.
- Additional File Access**: A section for configuring additional file access. It includes a table with columns "Name", "Paths", "R", "W", "X", "C", and "D".

Name	Paths	R	W	X	C	D
Console Devices	/dev/console (character file)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
pts Devices	/dev/pts (directory)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
tty Devices	/dev/tty (character file)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Configuration Files	/etc/cups/client.conf (regular file)...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Localization	/var/spool/postfix/etc/localtime (regular file)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Random Devices	/var/named/chroot/dev/random (character fil	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
urandom Device	/dev/urandom (character file)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

At the bottom of the interface, there is a status bar that reads "Brickwall: Active, Enforcing".



SLIDE - policy development

- IDE for Reference Policy
 - utilizes generated xml to make editing easier
 - provides wizards for common tasks
 - syntax highlighting
- Full policy or individual module
- Remote testing and debugging
 - through SLIDE Remote

SLIDE - policy development

The screenshot displays the SLIDE Eclipse SDK interface. The main editor shows the SELinux policy for the 'arpwatch' process. The policy includes two interfaces: 'arpwatch_search_data' and 'arpwatch_manage_data_files'. The 'arpwatch_search_data' interface allows access to search directories, while 'arpwatch_manage_data_files' allows read and write access to temporary files. The 'Interfaces' view on the right shows a tree structure of interfaces, with 'auth_append_login_records' selected. The bottom panel shows the declaration for 'auth_append_login_records'.

```
9## </param>
10#
11interface('arpwatch_search_data',
12    gen_require('
13        type arpwatch_data_t;
14    ')
15
16    allow $1 arpwatch_data_t:dir search;
17')
18
19#####
20## <summary>
21## Create arpwatch data files.
22## </summary>
23## <param name="domain">
24## Domain allowed access.
25## </param>
26#
27interface('arpwatch_manage_data_files',
28    gen_require('
29        type arpwatch_data_t;
30    ')
31
32    allow $1 arpwatch_data_t:dir rw_dir_perms;
33    allow $1 arpwatch_data_t:file create_file_perms;
34')
35
36#####
37## <summary>
38## Read and write arpwatch temporary files.
39## </summary>
40## <param name="domain">
41## Domain allowed access.
42## </param>
43## </interface>
44## </interface>
45## </interface>
46## </interface>
47## </interface>
48## </interface>
49## </interface>
50## </interface>
51## </interface>
52## </interface>
53## </interface>
54## </interface>
55## </interface>
56## </interface>
57## </interface>
58## </interface>
59## </interface>
60## </interface>
61## </interface>
62## </interface>
63## </interface>
64## </interface>
65## </interface>
66## </interface>
67## </interface>
68## </interface>
69## </interface>
70## </interface>
71## </interface>
72## </interface>
73## </interface>
74## </interface>
75## </interface>
76## </interface>
77## </interface>
78## </interface>
79## </interface>
80## </interface>
81## </interface>
82## </interface>
83## </interface>
84## </interface>
85## </interface>
86## </interface>
87## </interface>
88## </interface>
89## </interface>
90## </interface>
91## </interface>
92## </interface>
93## </interface>
94## </interface>
95## </interface>
96## </interface>
97## </interface>
98## </interface>
99## </interface>
100## </interface>
```

```
## Summary: Append to login records (wtmp).
## Parameter domain
## no description available
interface('auth_append_login_records',
    gen_require('
        type wtmp_t;
    ')

    allow $1 wtmp_t:file { getattr append };
)
```

Debugging tools

- setroubleshoot
 - monitor logs for denials
 - attempt to infer a reason and suggest a resolution
- SLIDE Remote
 - mentioned previously
 - debugging during development
- More in the pipeline

Debugging tools

The screenshot shows the 'setroubleshoot browser' window. The top bar includes 'Applications', 'Places', 'System', and a clock showing '12:24 AM'. The window title is 'setroubleshoot browser'. The menu bar contains 'File', 'View', 'Edit', and 'Help'. Below the menu is a table with columns: Filter, Date, Count, Category, and Summary. Three entries are visible, all dated 'Tue 29 May 2007 01:00:08 AM EDT' and categorized as 'File Label'. The second entry is selected, and its details are shown in the main pane.

Filter	Date	Count	Category	Summary
<input type="checkbox"/>	Tue 29 May 2007 01:00:08 AM EDT	1	File Label	SELinux is preventing the bash from using
<input checked="" type="checkbox"/>	Tue 29 May 2007 01:00:08 AM EDT	1	File Label	SELinux is preventing the bash from using
<input type="checkbox"/>	Tue 29 May 2007 01:00:08 AM EDT	1	File Label	SELinux is preventing the /sbin/consoletyp

Summary
SELinux is preventing the bash from using potentially mislabeled files (/home/csellers/.bash_profile).

Detailed Description
SELinux has denied bash access to potentially mislabeled file(s) (/home/csellers/.bash_profile). This means that SELinux will not allow bash 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 bash to access this files, you need to relabel them using `restorecon -v /home/csellers/.bash_profile`. You might want to relabel the entire directory using `restorecon -R -v /home/csellers`.

Additional Information

Source Context: user_u:system_r:httpd_t
Target Context: user_u:object_r:user_home_t
Target Objects: /home/csellers/.bash_profile [file]
Affected RPM Packages:

Audit Listener

Conclusion

- SELinux can probably address your problems
 - flexible enough to address many situations
 - must understand limitations of SELinux and security architectures to fully utilize
 - not just for government high-security needs
- SELinux is getting easier
 - solid foundation
 - tools developing rapidly to increase usability

Links

- <http://www.tresys.com>
- <http://www.usefulsecurity.com>
- <http://securityblog.org>
- csellers@tresys.com