SECURITY ENHANCED LINUX FOR MERE MORTALS

Or, “Don’t Turn It Off!”

Thomas Cameron, RHCA, RHCSS, RHCDs, RHCVa, RHCX
Digital Transformation Strategist, Red Hat
May 3rd, 2017
Agenda

- About Us
- What is SELinux?
  - Where did it come from?
  - DAC vs. MAC
- So How Does SELinux Work?
  - Labeling and Type Enforcement
- How Do I Deal With Labels?
- Real World Examples
CONTACT INFO
Contact info

- thomas@redhat.com
- @thomasdcameron on Twitter
- https://www.facebook.com/RedHatThomas/
- choirboy on #rhel on Freenode
- http://people.redhat.com/tcameron
ABOUT US
About Us

- Red Hat leads the way in SELinux development. John Dennis, Ulrich Drepper, Steve Grubb, Eric Paris, Roland McGrath, James Morris and Dan Walsh, all Red Hat staffers, acknowledged by the NSA for their contributions to SELinux at:
- Red Hat acknowledged by the NSA as a corporate contributor as well.
WHAT IS SELINUX?
Where did it come from?

- Created by the United States National Security Agency (NSA) as a set of patches to the Linux kernel using Linux Security Modules (LSM)
- Released by the NSA under the GNU General Public License (GPL) in 2000
- Adopted by the upstream Linux kernel in 2003
HOW I FELT ABOUT SELINUX
What Thomas thought SELinux was
If you feel the same way...
If you feel the same way...

- You're in the right place!
DAC VS. MAC
DAC vs. MAC

- Historically, Linux and Unix systems have used discretionary access control.
  - Ownership (user, group, and other) plus permissions.
  - Users have the ability (discretion) to change permissions on their own files. A user can chmod +rwx his or her home directory, and nothing will stop them. Nothing will prevent other users or processes from accessing the contents of his home directory.
DAC vs. MAC

- Historically, Linux and Unix systems have had discretionary access control.
  - The root user is omnipotent.
DAC vs. MAC

- On a mandatory access control system, there is policy which is administratively set and fixed.
- Even if you change the DAC settings on your home directory, if there is a policy in place which prevents another user or process from accessing it, you're generally safe.
DAC vs. MAC

- These policies can be very fine grained. Policies can be set to determine access between:
  - Users
  - Files
  - Directories
  - Memory
  - Sockets
  - tcp/udp ports
  - etc...
POLICY
Policy

- In Red Hat Enterprise Linux, there are two policies you'll generally see.
  - “targeted” - the default policy
    - Only targeted processes (there are hundreds) are protected by SELinux
    - Everything else is unconfined
  - “mls” - multi-level/multi-category security
    - Out of scope for today's presentation
    - Can be very complex
    - Typically used in TLA government organizations
So How Does SELinux Work?

- You can determine what policy your system is set to use by looking at /etc/selinux/config (which is also symlinked to /etc/sysconfig/selinux)
- You can check via /usr/sbin/sestatus
- You can also check via /usr/sbin/getenforce
[root@w541 ~]# cat /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 three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes are protected.
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted

[root@w541 ~]# ls -l /etc/sysconfig/selinux
lrwxrwxrwx. 1 root root 17 Oct 29 2015 /etc/sysconfig/selinux -> ../selinux/config
[root@w541 ~]#
[root@w541 ~]# /usr/sbin/sestatus
SELinux status: enabled
SELinuxfs mount: /sys/fs/selinux
SELinux root directory: /etc/selinux
Loaded policy name: targeted
Current mode: enforcing
Mode from config file: enforcing
Policy MLS status: enabled
Policy deny_unknown status: allowed
Max kernel policy version: 30
[root@w541 ~]#
[root@w541 ~]# /usr/sbin/getenforce
Enforcing
[root@w541 ~]#
HOW DOES SELINUX WORK?
So How Does SELinux Work?

- Two of the important concepts to understand with SELinux are:
  - Labeling
  - Type Enforcement
So How Does SELinux Work?

- **Labeling**
  - Files, processes, ports, etc., are all labeled with an SELinux context.
  - For files and directories, these labels are stored as extended attributes on the filesystem.
  - For processes, ports, etc., the kernel manages these labels.
So How Does SELinux Work?

- Labeling
  - Labels are in the format:
    - user:role:type:level(optional)
  - For the purpose of this presentation, we will not deal with the SELinux user, role or level. These are used in more advanced implementations of SELinux (MLS/MCS).
  - What we really care about for today's presentation is the type (remember, labeling and type enforcement).
So How Does SELinux Work?

- We'll look at a fairly complex service, one which provides access from the network, potentially on several ports, and potentially, access to the whole filesystem.
- The Apache web server is not necessarily insecure, it is just very wide ranging in its access.
So How Does SELinux Work?

- The Apache web server has a binary executable which launches from `/usr/sbin`. When you look at that file's SELinux context, you see its type is `httpd_exec_t:`
[root@w541 ~]# ls -lZ /usr/sbin/httpd
-rwxr-xr-x. 1 root  root  system_u:object_r:httpd_exec_t:s0 536888 Jan 4 00:17 /usr/sbin/httpd
[root@w541 ~]#
So How Does SELinux Work?

- The web server's configuration directory is labeled httpd_config_t:
[root@w541 ~]# ls -dZ /etc/httpd/
/system_u:object_r:httpd_config_t:s0 /etc/httpd/
[root@w541 ~]#
So How Does SELinux Work?

- The web server's logfile directory is labeled httpd_log_t:
[root@w541 ~]# ls -dZ /var/log/httpd/
system_u:object_r:httpd_log_t:s0 /var/log/httpd/
[root@w541 ~]#
So How Does SELinux Work?

- The web server's content directory is labeled httpd_sys_content_t:
[root@w541 ~]# ls -dZ /var/www/html/
/system_u:object_r:httpd_sys_content_t:s0 /var/www/html/
[root@w541 ~]#
So How Does SELinux Work?

- The web server's startup script is labeled httpd_initrc_exec_t:
[root@w541 ~]# ls -lZ /usr/lib/systemd/system/httpd.service
-rw-r--r--. 1 root root system_u:object_r:httpd_unit_file_t:s0 1090 Jan 4 00:12
/usr/lib/systemd/system/httpd.service
[root@w541 ~]#
So How Does SELinux Work?

- As the web server runs, it's process is labeled httpd_t:
[root@w541 ~]# ps axZ | grep [h]ttpd
system_u:system_r:httpd_t:s0  1289  ?  Ss  0:01 /usr/sbin/httpd -DFOR
system_u:system_r:httpd_t:s0  1449  ?  S   0:00 /usr/sbin/httpd -DFOR
system_u:system_r:httpd_t:s0  1451  ?  Sl  0:00 /usr/sbin/httpd -DFOR
system_u:system_r:httpd_t:s0  1452  ?  Sl  0:00 /usr/sbin/httpd -DFOR
system_u:system_r:httpd_t:s0  1454  ?  Sl  0:00 /usr/sbin/httpd -DFOR
system_u:system_r:httpd_t:s0  1457  ?  Sl  0:00 /usr/sbin/httpd -DFOR
system_u:system_r:httpd_t:s0  1459  ?  Sl  0:00 /usr/sbin/httpd -DFOR
[root@w541 ~]#
So How Does SELinux Work?

- If you look at the ports upon which the web server listens, you'll see that even they are labeled.
# redhat # rhsummit

[root@w541 ~]# netstat -tnlpZ | grep [h]ttpd_t
tcp6       0       0     :::80     ::::*            LISTEN
1289/httpd        system_u:system_r:httpd_t:s0
 tcp6       0       0     :::443     ::::*            LISTEN
1289/httpd        system_u:system_r:httpd_t:s0
[root@w541 ~]# semanage port -l | grep [h]ttp
http_cache_port_t tcp 8000, 8118, 8123, 10001-10010
http_cache_port_t udp 3130
http_port_t tcp 80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t tcp 5988
pegasus_https_port_t tcp 5989
[root@w541 ~]#
So How Does SELinux Work?

- Now then... The /etc/shadow file has a type shadow_t:
[root@w541 ~]# ls -lZ /etc/shadow
----------. 1 root root system_u:object_r:shadow_t:s0 1431 Apr 26 12:12 /etc/shadow
[root@w541 ~]#
So How Does SELinux Work?

- Type enforcement
So How Does SELinux Work?

- Type enforcement
  - It probably makes sense for a process running in the httpd_t context to interact with a file with the httpd_config_t label.
So How Does SELinux Work?

- Type enforcement
  - Do you think it makes sense for a process running with the httpd_t context label to be able to interact with a file with, say, the shadow_t label?
So How Does SELinux Work?

- Type enforcement
  - Type enforcement is the part of the policy that says, for instance, “a process running with the label httpd_t can have read access to a file labeled httpd_config_t”
HOW DO I DEAL WITH LABELS?
How Do I Deal With Labels?

- You've seen me use the -Z argument to several commands to view context. Many commands accept this argument:
  - ls -Z
  - id -Z
  - ps -Z
  - netstat -Z
How Do I Deal With Labels?

- You can actually use the -Z argument to create and modify files and contexts, as well.
  - `cp -Z`
  - `mkdir -Z`
How Do I Deal With Labels?

- You can use SELinux aware tools like chcon or restorecon to change the context of a file (more on this later).
- Contexts are set when files are created, based on their parent directory's context (with a few exceptions).
- RPMs can set contexts as part of installation.
- The login process sets the default context (unconfined in the targeted policy)
How Do I Deal With Labels?

- File transitions (defined by policy)
  - If an application foo_t creates a file in a directory labeled bar_t, policy can require a transition so that file is created with the baz_t label.
  - Example: A process, dhclient, running with the dhclient_t label creates a file, resolv.conf, labeled net_conf_t in a directory, /etc, labeled etc_t. Without that transition, /etc/resolv.conf would have inherited the etc_t label.
How Do I Deal With Labels?

- You've also seen me use the semanage command. It can be used to manage SELinux settings for:
  - login
  - user
  - port
  - interface
  - module
How Do I Deal With Labels?

- You've also seen me use the semanage command. It can be used to manage SELinux settings for:
  - node
  - file context
  - boolean
  - permissive state
  - dontaudit
SELINUX ERRORS
What Does It Mean If I Get An SELinux Error?
What Does It Mean If I Get An SELinux Error?

- If you see an SELinux error, it means that something is wrong!
What Does It Mean If I Get An SELinux Error?

- If you see an SELinux error, it means that something is wrong!
- Turning off SELinux is like turning up the radio really loud when your car is making a strange noise!
What Does It Mean If I Get An SELinux Error?

- It may mean that labeling is wrong
  - Use the tools to fix the labels. We'll talk more about that later.
What Does It Mean If I Get An SELinux Error?

- It may mean that the policy needs to be tweaked.
  - booleans
  - Policy modules
What Does It Mean If I Get An SELinux Error?

- There could be a bug in the policy
  - We need to know about these! Open a ticket (do not file a Bugzilla report - there are no SLAs around BZ).
What Does It Mean If I Get An SELinux Error?

- You have been, or are being, broken into
  - Man the battle stations!
BOOLEANS
What Are Booleans?

- Booleans are just off/on settings for SELinux.
  - From simple stuff like “do we allow the ftp server access to home directories” to more esoteric stuff like “httpd can use mod_auth_ntlm_winbind.”
What Are Booleans?

- To see all the booleans, run `getsebool -a`
abrt_anon_write --> off
abrt_handle_event --> off
abrt_upload_watch_anon_write --> on
antivirus_can_scan_system --> off
antivirus_use_jit --> off
auditadm_exec_content --> on
authlogin_nsswitch_use_ldap --> off
authlogin_radius --> off
authlogin_yubikey --> off
awstats_purge_apache_log_files --> off
boinc_execmem --> on
cdrecord_read_content --> off
cluster_can_network_connect --> off
cluster_manage_all_files --> off
cluster_use_execmem --> off
cobbler_anon_write --> off
cobbler_can_network_connect --> off
cobbler_use_cifs --> off
cobbler_use_nfs --> off
collectd_tcp_network_connect --> off
condor_tcp_network_connect --> off
commcan_can_network --> off
cron_can_relabel --> off

;
TIPS AND TRICKS
Tips and Tricks

- Install setroubleshoot and setroubleshoot-server on machines you'll be developing policy modules on. They drag in a bunch of tools to help diagnose and fix SELinux issues.
- Reboot or restart auditd after you install.
[root@w541 ~]# yum install setroubleshoot setroubleshoot-server
<table>
<thead>
<tr>
<th>Package</th>
<th>Architecture</th>
<th>Version</th>
<th>Source</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>mesa-libglapi</td>
<td>x86_64</td>
<td>10.6.5-3.20150824.el7</td>
<td>rhel-7-server-rpms</td>
<td>39 k</td>
</tr>
<tr>
<td>pango</td>
<td>x86_64</td>
<td>1.36.8-2.el7</td>
<td>rhel-7-server-rpms</td>
<td>287 k</td>
</tr>
<tr>
<td>pixman</td>
<td>x86_64</td>
<td>0.32.6-3.el7</td>
<td>rhel-7-server-rpms</td>
<td>254 k</td>
</tr>
<tr>
<td>policycoreutils-python</td>
<td>x86_64</td>
<td>2.2.5-20.el7</td>
<td>rhel-7-server-rpms</td>
<td>435 k</td>
</tr>
<tr>
<td>pycairo</td>
<td>x86_64</td>
<td>1.8.10-8.el7</td>
<td>rhel-7-server-rpms</td>
<td>157 k</td>
</tr>
<tr>
<td>pygtk2</td>
<td>x86_64</td>
<td>2.24.0-9.el7</td>
<td>rhel-7-server-rpms</td>
<td>914 k</td>
</tr>
<tr>
<td>pygtk2-libglade</td>
<td>x86_64</td>
<td>2.24.0-9.el7</td>
<td>rhel-7-server-rpms</td>
<td>25 k</td>
</tr>
<tr>
<td>python-IPy</td>
<td>noarch</td>
<td>0.75-6.el7</td>
<td>rhel-7-server-rpms</td>
<td>32 k</td>
</tr>
<tr>
<td>rest</td>
<td>x86_64</td>
<td>0.7.92-3.el7</td>
<td>rhel-7-server-rpms</td>
<td>62 k</td>
</tr>
<tr>
<td>satyr</td>
<td>x86_64</td>
<td>0.13-12.el7</td>
<td>rhel-7-server-rpms</td>
<td>508 k</td>
</tr>
<tr>
<td>setools-libs</td>
<td>x86_64</td>
<td>3.3.7-46.el7</td>
<td>rhel-7-server-rpms</td>
<td>485 k</td>
</tr>
<tr>
<td>setroubleshoot-plugins</td>
<td>noarch</td>
<td>3.0.59-2.el7_2</td>
<td>rhel-7-server-rpms</td>
<td>585 k</td>
</tr>
<tr>
<td>systemd-python</td>
<td>x86_64</td>
<td>219-19.el7_2.11</td>
<td>rhel-7-server-rpms</td>
<td>99 k</td>
</tr>
<tr>
<td>xml-common</td>
<td>noarch</td>
<td>0.6.3-39.el7</td>
<td>rhel-7-server-rpms</td>
<td>26 k</td>
</tr>
<tr>
<td>xmlrpc-c</td>
<td>x86_64</td>
<td>1.32.5-1905.svn2451.el7</td>
<td>rhel-7-server-rpms</td>
<td>130 k</td>
</tr>
<tr>
<td>xmlrpc-c-client</td>
<td>x86_64</td>
<td>1.32.5-1905.svn2451.el7</td>
<td>rhel-7-server-rpms</td>
<td>32 k</td>
</tr>
</tbody>
</table>

Transaction Summary

Install 2 Packages (+78 Dependent packages)

Total download size: 30 M
Installed size: 83 M
Is this ok [y/d/N]: y
libxcb.x86_64 0:1.11-4.el7
libxshmfence.x86_64 0:1.2-1.el7
mesa-libEGL.x86_64 0:10.6.5-3.20150824.el7
mesa-libGL.x86_64 0:10.6.5-3.20150824.el7
mesa-libgbm.x86_64 0:10.6.5-3.20150824.el7
mesa-libglapi.x86_64 0:10.6.5-3.20150824.el7
pango.x86_64 0:1.36.8-2.el7
pixman.x86_64 0:0.32.6-3.el7
policycoreutils-python.x86_64 0:2.2.5-20.el7
pycairo.x86_64 0:1.10-8.el7
pygobject2.x86_64 0:2.24.0-9.el7
pygobject2-libglade.x86_64 0:2.24.0-9.el7
python-IPy.noarch 0:0.75-6.el7
rest.x86_64 0:0.7.92-3.el7
satyr.x86_64 0:0.13-12.el7
setools-libs.x86_64 0:3.3.7-46.el7
setroubleshoot-plugins.noarch 0:3.0.59-2.el7_2
systemd-python.x86_64 0:219-19.el7_2.11
xml-common.noarch 0:0.6.3-39.el7
xmlrpc-c.x86_64 0:1.32.5-1905.svn2451.el7
xmlrpc-c-client.x86_64 0:1.32.5-1905.svn2451.el7

Complete!
[root@w541 ~]#
[root@w541 ~]# systemctl restart auditd.service
Failed to restart auditd.service: Operation refused, unit auditd.service may be requested by dependency only.
[root@w541 ~]#
[root@w541 ~]# service auditd restart
Stopping logging: [ OK ]
Redirecting start to /bin/systemctl start auditd.service
[root@w541 ~]#
auditd

- This is not a bug. See https://bugzilla.redhat.com/show_bug.cgi?id=1026648 for details.
Real World Examples

- A user, fred, wants to have his own web page in /home/fred/public_html on a web server.
  - You enable UserDir in /etc/httpd/conf.d/userdir.conf
  - Restart the web server
The path to the end user account 'public_html' directory must be accessible to the webserver userid. This usually means that ~userid must have permissions of 711, ~userid/public_html must have permissions of 755, and documents contained therein must be world-readable. Otherwise, the client will only receive a "403 Forbidden" message.

<IfModule mod_userdir.c>

  # UserDir is disabled by default since it can confirm the presence of a username on the system (depending on home directory permissions).
  # UserDir disabled

  # To enable requests to /~user/ to serve the user's public_html directory, remove the "UserDir disabled" line above, and uncomment the following line instead:
  # UserDir public_html
</IfModule>
[root@w541 ~]# systemctl restart httpd
Real World Examples

- A user, fred, wants to start having his own web page in /home/fred/public_html
  - Change permissions so the web server can access his home directory.
[root@w541 ~]# chmod o+x /home/fred/
[root@w541 ~]# ls -ld /home/fred/
  drwx------. 3 fred fred 4096 Jun 26 22:58 /home/fred/
[root@w541 ~]#
Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
  - Fred logs in, creates his public_html directory and an index.html file.
tcameron@w541 ~]$ ssh fred@192.168.124.1
Warning: Permanently added '192.168.124.1' (ECDSA) to the list of known hosts.
fred@192.168.124.1's password:
[fred@w541 ~]$ mkdir public_html
[fred@w541 ~]$ cd public_html/
[fred@w541 public_html]$ echo "this is my home page" > index.html
[fred@w541 public_html]$
Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
  - We fire up the web browser, and:
Forbidden

You don't have permission to access /fred/ on this server.
Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
  - So now we check the usual suspects.
    - /var/log/httpd/access_log
    - /var/log/httpd/error_log
# tail -2 /var/log/httpd/access_log
192.168.124.1 - - [26/Jul/2016:03:02:27 -0700] "GET /~fred HTTP/1.1" 301 235 "-" "Mozilla/5.0 (X11; Fedora; Linux x86_64; rv:47.0) Gecko/20100101 Firefox/47.0"
192.168.124.1 - - [26/Jul/2016:03:02:27 -0700] "GET /~fred/ HTTP/1.1" 403 215 "-" "Mozilla/5.0 (X11; Fedora; Linux x86_64; rv:47.0) Gecko/20100101 Firefox/47.0"
[root@w541 ~]# tail -2 /var/log/httpd/error_log
[Sun Jun 26 23:03:27.658743 2016] [core:error] [pid 17755] (13)Permission denied:
[client 192.168.124.1:44820] AH00035: access to /~fred/index.html denied (file
system path '/home/fred/public_html/index.html') because search permissions are
missing on a component of the path
[Sun Jun 26 23:03:27.658823 2016] [negotiation:error] [pid 17755] (13)Permission
denied: [client 192.168.124.1:44820] AH00686: cannot read directory for multi:
/home/fred/public_html/
[root@w541 ~]#
Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
  - We already knew that!
Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
  - So now we look at journalctl
[root@w541 ~]# journalctl -b -1
**** Plugin catchall_boolean (24.7 confidence) suggests  ***************

If you want to allow httpd to read home directories
Then you must tell SELinux about this by enabling the 'httpd_enable_homedirs' boolean.
You can read 'None' man page for more details.

Do
setsebool -P httpd_enable_homedirs 1

**** Plugin catchall_boolean (24.7 confidence) suggests  ***************

If you want to allow httpd to read user content
Then you must tell SELinux about this by enabling the 'httpd_read_user_content' boolean.
You can read 'None' man page for more details.

Do
setsebool -P httpd_read_user_content 1

**** Plugin catchall_boolean (24.7 confidence) suggests  ***************

If you want to allow HTTPS/HTTPD handling of all content files.
Then you must tell SELinux about this by enabling the 'httpd_unified' boolean.
You can read 'None' man page for more details.

Do
setsebool -P httpd_unified 1

**** Plugin public_content (24.7 confidence) suggests  ***************

If you want to treat index.html as public content
Then you need to change the label on index.html to public_content_t or public_content_rw_t.

Do
# semanage fcontext -a -t public_content_t '/home/fred/public_html/index.html'
# semanage fcontext -a -t public_content_rw_t '/home/fred/public_html/index.html'

**** Plugin catchall (3.53 confidence) suggests  ***************

If you believe that httpd should be allowed getattr access on the index.html file by default.
Then you should report this as a bug.

**** Plugin catchall (3.53 confidence) suggests  ***************

If you believe that httpd should be allowed getattr access on the index.html file by default.
Then you should report this as a bug.
Real World Examples

• A user, fred, wants to start have his own web page in /home/fred/public_html
  - AH-HAH! Follow the instructions and run “sealert -l c36627c9-7b99-44c9-a78c-a9a737ff119b”
  - It reveals that there are several potential issues/solutions.
    • httpd access to home directories
    • httpd access to user content
    • httpd unified access to all content
    • relabel the content as public
SELinux is preventing /usr/sbin/httpd from getattr access on the file /home/fred/public_html/index.html.

***** Plugin catchall_boolean (24.7 confidence) suggests ***************

If you want to allow httpd to read home directories
Then you must tell SELinux about this by enabling the 'httpd_enable_homedirs' boolean.
You can read 'None' man page for more details.
Do
setsebool -P httpd_enable_homedirs 1

***** Plugin catchall_boolean (24.7 confidence) suggests ***************

If you want to allow httpd to read user content
Then you must tell SELinux about this by enabling the 'httpd_read_user_content' boolean.
You can read 'None' man page for more details.
Do
setsebool -P httpd_read_user_content 1

***** Plugin catchall_boolean (24.7 confidence) suggests ***************

If you want to unify HTTPD handling of all content files.
Then you must tell SELinux about this by enabling the 'httpd_unified' boolean.
You can read 'None' man page for more details.
Do
setsebool -P httpd_unified 1

***** Plugin public_content (24.7 confidence) suggests ***************

If you want to treat index.html as public content
Then you need to change the label on index.html to public_content_t or public_content_rw_t.
Do
# semanage fcontext -a -t public_content_t '/home/fred/public_html/index.html'
# restorecon -v '/home/fred/public_html/index.html'

***** Plugin catchall (3.53 confidence) suggests ***************

If you believe that httpd should be allowed getattr access on the index.html file by default.
Then you should report this as a bug.
You can generate a local policy module to allow this access.
Do
allow this access for now by executing:
# ausearch -c '/usr/sbin/httpd' --raw | audit2allow -M my-usersbinhttpd
# semodule -X 360 -i my-usersbinhttpd.pp

Additional Information:
Source Context    system_u:system_r:httpd_t:s0
Target Context   unconfined_u:object_r:httpd_user_content_t:s0
Real World Examples

• A user, fred, wants to start have his own web page in /home/fred/public_html
  – It also says we can create a policy module to allow this, but in this case, setting a boolean is easier and makes more sense.
***** Plugin public_content (24.7 confidence) suggests ********************

If you want to treat index.html as public content
Then you need to change the label on index.html to public_content_t or public_content_rw_t.
Do
# semanage fcontext -a -t public_content_t '/home/fred/public_html/index.html'
# restorecon -v '/home/fred/public_html/index.html'

***** Plugin catchall (3.53 confidence) suggests ********************

If you believe that httpd should be allowed getattr access on the index.html file by default.
Then you should report this as a bug.
You can generate a local policy module to allow this access.
Do
# ausearch -c '/usr/sbin/httpd' --raw | audit2allow -M my-usrsbinhttpd
# semodule -X 300 -i my-usrsbinhttpd.pp

Additional Information:
Source Context: system_u:system_r:httpd_t:s0
Target Context: unconfined_u:object_r:httpd_user_content_t:s0
Target Objects: /home/fred/public_html/index.html [ file ]
Source: /usr/sbin/httpd
Source Path: /usr/sbin/httpd
Port: -u unknow
Host: w541.tc.redhat.com
Source RPM Packages: httpd-2.4.18-1.fc23.x86_64
Target RPM Packages: selinux-policy-3.13.1-150.15.fc23.noarch
Selinux Enabled: True
Policy Type: targeted
Enforcing Mode: Enforcing
Host Name: w541.tc.redhat.com
Platform: Linux w541.tc.redhat.com 4.5.7-200.fc23.x86_64 #1
SMP Wed Jun 8 17:41:50 UTC 2016 x86_64 x86_64
Alert Count: 6
First Seen: 2016-06-26 23:03:27 PDT
Last Seen: 2016-06-26 23:04:31 PDT
Local ID: c36627c9-7b99-14c9-a78c-a9e737f119b

Raw Audit Messages:
type=AVC msg=audit(1467018471.39:424): avc: denied { getattr } for pid=1489 comm="/usr/sbin/httpd" path="/home/fred/public_html/index.html" dev="dm-0" ino=19803817 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:httpd_user_content_t:s0 tclass=file permissive=0

Hash: /usr/sbin/httpd;httpd_t;httpd_user_content_t;file,getattr
[root@w541 ~]#
Real World Examples

• A user, fred, wants to start have his own web page in /home/fred/public_html
  – Follow the instructions and set the boolean to allow httpd access to home directories.
[root@w541 ~]# setsebool httpd_enable_homedirs 1 -P
[root@w541 ~]#
Real World Examples

• A user, fred, wants to start have his own web page in /home/fred/public_html
  – And... Voila!
This is my home page
Real World Examples

- And people say this SELinux thing is too hard! Pffft!
HOW CAN I SEE WHAT BOOLEANS HAVE BEEN SET?
How Can I See What Booleans Have Been Set?

- Look at the booleans.local file under /etc/selinux/targeted/modules/active/
File Edit View Terminal Tabs Help
[root@armitage ~]# cat /etc/selinux/targeted/modules/active/booleans.local
# This file is auto-generated by libsemanage
# Do not edit directly.

httpd_enable_homedirs=1
[root@armitage ~]#
How Can I See What Booleans Have Been Set?

- Note that when you use setsebool -P (and other commands we'll cover later), the entire /etc/selinux/targeted directory is regenerated. That file doesn't actually do anything - it just tells you what's been set. Believe it when it says “Do not edit directly” - it won't do anything.
[root@armitage ~]# touch marker
[root@armitage ~]# setsebool -P httpd_enable_homedirs=1
[root@armitage ~]# find /etc/selinux/ -newer marker
Terminal - root@armitage:

File   Edit   View   Terminal   Tabs   Help

[root@armitage ~]# setsebool -P httpd_enable_homedirs=1
[root@armitage ~]# find /etc/selinux/ -newer marker
/etc/selinux/targeted
/etc/selinux/targeted/contexts
/etc/selinux/targeted/contexts/files
/etc/selinux/targeted/contexts/files/file_contexts
/etc/selinux/targeted/contexts/files/file_contexts.homedirs
/etc/selinux/targeted/contexts/files/file_contexts.bin
/etc/selinux/targeted/contexts/files/file_contexts.local.bin
/etc/selinux/targeted/contexts/files/file_contexts.homedirs.bin
/etc/selinux/targeted/contexts/filefilter_contexts
/etc/selinux/targeted/modules
/etc/selinux/targeted/modules/active
/etc/selinux/targeted/modules/active/base.pp
/etc/selinux/targeted/modules/active/commit_num
/etc/selinux/targeted/modules/active/file_contexts
/etc/selinux/targeted/modules/active/file_contexts.homedirs
/etc/selinux/targeted/modules/active/file_contexts.template
/etc/selinux/targeted/modules/active/homedir_template
/etc/selinux/targeted/modules/active/modules
/etc/selinux/targeted/modules/active/modules/bcfg2.pp
/etc/selinux/targeted/modules/active/modules/colord.pp
/etc/selinux/targeted/modules/active/modules/cipe.pp
/etc/selinux/targeted/modules/active/modules/dcc.pp
REAL WORLD EXAMPLES
Real World Examples

- This next example assumes an unmodified SELinux environment, so ignore the changes from the last example.
Real World Examples

- A user, Wilma, is a web content author. She has created content in her home directory and asked that you move it to the web site.
mkdir content
cd content/
echo "this is our cool web site" > index.html
Real World Examples

- So, you move it over.
[root@w541 ~]# mv /home/wilma/content/* /var/www/html/
[root@w541 ~]#
Real World Examples

- And when you go to test...
Forbidden

You don't have permission to access /index.html on this server.
Real World Examples

• Ah, it's the wrong owner, right?
[root@w541 ~]# ls -l /var/www/html/index.html
-rw-rw-r--. 1 wilma wilma 26 Jun 27 00:23 /var/www/html/index.html
[root@w541 ~]#
[root@w541 ~]# chown root:root /var/www/html/index.html
[root@w541 ~]# ls -l /var/www/html/index.html
[root@w541 ~]#
Real World Examples

- But when you test...
Forbidden

You don't have permission to access /index.html on this server.
Real World Examples

● Checking journalctl again tells you to run sealert.
Jun 27 00:35:26 wsl1 tc.redhat.com kernel: wlp300: authenticated
Jun 27 00:35:26 wsl1 tc.redhat.com kernel: wlp300: associate with 02:ja:ff:00:4e (try 1/3)
Jun 27 00:35:26 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: scanning -> authenticating
Jun 27 00:35:26 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): RX AssocResp from 02:ja:ff:00:4e (capab=0x411 status=0 audit)
Jun 27 00:35:26 wsl1 tc.redhat.com kernel: wlp300: authenticated
Jun 27 00:35:26 wsl1 tc.redhat.com kernel: wlp300: associate with 02:ja:ff:00:4e (try 1/3)
Jun 27 00:35:26 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: authenticating -> associating
Jun 27 00:35:26 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: associating -> associated
Jun 27 00:35:26 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: associated -> 4-way handshake
Jun 27 00:35:26 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: 4-way handshake -> completed
Jun 27 00:36:59 wsl1 tc.redhat.com audit[1491]: AVC avc: denied ( read ) for pid=1469 core="/usr/sbin/httpd" name="index.html" dev="in-
 Jun 27 00:37:02 wsl1 tc.redhat.com dmesg[2071]: [system] Activating service name="org.fedoraproject.anaconda" (using servicehelper)
Jun 27 00:37:02 wsl1 tc.redhat.com dmesg[2071]: [system] Successfully activated service "org.fedoraproject.anaconda"
Jun 27 00:37:02 wsl1 tc.redhat.com dmesg[2071]: Deleting alert c36572c9-7069-44cb-87be-7e973f7f116b, it is allowed in current policy
Jun 27 00:37:02 wsl1 tc.redhat.com dmesg[2071]: SELinux is preventing /usr/sbin/httpd from read access on the file index.html. For complete SELinux messages, run server -t c8b82e33
Jun 27 00:37:02 wsl1 tc.redhat.com python(36728): SELinux is preventing /usr/sbin/httpd from read access on the file index.html.

**** Plugin catchell Boolean (89.3 confidence) suggests ********************

If you want to allow httpd to read user content
Then you must tell SELinux about this by enabling the 'httpd_read_user_content' boolean.
Do
setsebool -P httpd_read_user_content 1

**** Plugin catchell (1.6 confidence) suggests ****************************

If you believe that httpd should be allowed read access on the index.html file by default,
Then you should report this as a bug.
You can generate a local policy module to allow this access.
Do
allow this access for now by executing:
# semanage -X -i -v httpd_read_user_content
# seflush -e
# semanage module -r http
# sed -i -e '/^AllowUsers /i httpd_read_user_content' /etc/selinux/targeted/contexts/rhselinux/httpd.pp

Jun 27 00:37:05 wsl1 tc.redhat.com NetworkManager[1279]: <warn> Connection disconnected (reason -4)
Jun 27 00:37:05 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: disconnected -> completed
Jun 27 00:37:05 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: disconnected -> scanning
Jun 27 00:37:08 wsl1 tc.redhat.com kernel: wlp300: authenticate with 02:ja:ff:00:4e
Jun 27 00:37:08 wsl1 tc.redhat.com kernel: wlp300: associate with 02:ja:ff:00:4e (try 1/3)
Jun 27 00:37:08 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: scanning -> authenticating
Jun 27 00:37:08 wsl1 tc.redhat.com kernel: wlp300: authenticated
Jun 27 00:37:08 wsl1 tc.redhat.com kernel: wlp300: associate with 02:ja:ff:00:4e (try 1/3)
Jun 27 00:37:08 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: authenticating -> associating
Jun 27 00:37:08 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: associating -> associated
Jun 27 00:37:08 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: associated -> 4-way handshake
Jun 27 00:37:08 wsl1 tc.redhat.com NetworkManager[1279]: <info> (wlp300): supplicant interface state: 4-way handshake -> completed

Then 5760-5175156: 99:3
Real World Examples

- But this time, sealert is still talking about user content and home directories... We're dealing with content in the system web content directory, /var/www/html.
** Plugin catchall_boolean (89.3 confidence) suggests ********************

If you want to allow httpd to read user content
Then you must tell SELinux about this by enabling the 'httpd_read_user_content' boolean.
You can read 'None' man page for more details.
Do
setsebool -P httpd_read_user_content 1

** Plugin catchall (11.6 confidence) suggests ********************

If you believe that http should be allowed read access on the index.html file by default.
Then you should report this as a bug.
You can generate a local policy module to allow this access.
Do
allow this access for now by executing:
# ausearch -c /usr/sbin/httpd --raw | audit2allow -M my-usrsbinhttpd
# semodule -X 300 -i my-usrsbinhttpd.pp

Additional Information:
Source Context  system_u:system_r:httpd_t:s0
Target Context  unconfined_u:object_r:user_home_t:s0
Target Objects  index.html [ file ]
Source  /usr/sbin/httpd
Source Path  /usr/sbin/httpd
Port  <Unknown>
Host  w541.tc.redhat.com
Source RPM Packages  httpd-2.4.18-1.fc23.x86_64
Target RPM Packages  httpd-2.4.18-1.fc23.x86_64
Policy RPM  selinux-policy-3.13.1-150.15.fc23.noarch
Policy Enabled  True
Policy Type  targeted
Enforcing Mode  Enforcing
Host Name  w541.tc.redhat.com
Platform  Linux w541.tc.redhat.com 4.5.7-200.fc23.x86_64 #1
SMP Wed Jun 8 17:41:58 UTC 2016 x86_64 x86_64
Alert Count  1
First Seen  2016-06-27 00:36:59 PDT
Last Seen  2016-06-27 00:36:59 PDT
Local ID  c0882e13-8637-4564-8916-8288b28bf145c

Raw Audit Messages
type=AVC msg.nr=auditd(1467013019,225:592): avc: denied { read } for pid=1491 conn="/usr/sbin/httpd" name="index.html" dev="dm-0" ino=19865083 scontext=sysm_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:user_home_t:s0 tclass=file permissive=0

#redhat #rhsummit
Real World Examples

- A quick `ls -Z` reveals the issue.
[root@w541 ~]# ls -lZ /var/www/html/index.html
-rw-rw-r--. 1 root root unconfined_u:object_r:user_home_t:s0 26 Jun 27 00:23 /var/www/html/index.html
[root@w541 ~]#
Real World Examples

- We moved instead of copied, so the file kept its original context.
- To change the context, we can run one of a couple of commands.
Real World Examples

- First we need to figure out what the label should be. Look at a known good file label.
[root@w541 ~]# ls -ldZ /var/www/html/
drwxr-xr-x. 4 root root system_u:object_r:httpd_sys_content_t:s0 4096 Jun 27 00:27 /var/www/html/
[root@w541 ~]#
Real World Examples

- Use that information as arguments for the chcon (change context) command
- The long form is:
[root@w541 ~]# ls -ldZ /var/www/html/
drw-r-xr-x. 4 root root system_u:object_r:httpd_sys_content_t:s0 4096 Jun 27 00:27 /var/www/html/
[root@w541 ~]# chcon -u system_u -r object_r -t httpd_sys_content_t /var/www/html/index.html
Real World Examples

- Remember that the targeted policy doesn't use the SELinux user or role. The short form is:
[root@w541 ~]# chcon -t httpd_system_content_t /var/www/html/index.html
Real World Examples

- I'm lazy. If I just want to reference a known good context, the shortest form is:
[root@w541 ~]# chcon --reference /var/www/html/ /var/www/html/index.html
[root@w541 ~]#
Real World Examples

- If you just want to restore a directory and all its files to the default context, the easiest to remember is restorecon:
This is our cool web site
Where Are These Contexts Stored?

- `restorecon` uses information from `/etc/selinux/targeted/contexts/files/file_contexts` (and other files in that directory) to determine what a file or directory's context should be.
- There are over 4000 entries in this file. Don't modify this file directly, your changes will be lost!
etc/selinux/targeted/contexts/files/file_contexts
Terminal - root@w541:~

/var/www/html/[\^/]*/cgi-bin/(.*).* system_u:object_r:httpd_sys_script_exec_t:s0
/usr/acroread/(.*).*?intellinux/nppdf\.so -- system_u:object_r:textrile_shlib_t:s0
/usr/acroread/(.*).*?lib/[\^/]*\.(\.[^/]+)?* -- system_u:object_r:textrile_shlib_t:s0
/usr/lib/gems/.*/*ApplicationPoolServerExecutable -- system_u:object_r:passenger_exec_t:s0
/usr/bin/preupg.* -- system_u:object_r:preupgrade_exec_t:s0
/var/run/bacula.* -- system_u:object_r:bacula_var_run_t:s0
/usr/lib/ipsec/.* -- system_u:object_r:bin_t:s0
/usr/bin/pingus.* -- system_u:object_r:bin_t:s0
/etc/ppp/ip-up\.* -- system_u:object_r:bin_t:s0
/etc/cp/cp-up\.* -- system_u:object_r:bin_t:s0
/usr/sbin/cipact.* -- system_u:object_r:cipact_exec_t:s0
/var/log/vsftpd.* -- system_u:object_r:xferlog_t:s0
/usr/lib/gnutls/.* -- system_u:object_r:gpg_exec_t:s0
/var/run/charger.* -- system_u:object_r:ipsec_var_run_t:s0
/dev/shm/llcmap.* -- system_u:object_r:llcmap_tmpfs_t:s0
/var/log/mcelog.* -- system_u:object_r:mcelog_log_t:s0
/usr/sbin/rmmmod.* -- system_u:object_r:insmod_exec_t:s0
/var/run/fstatd.* -- system_u:object_r:mon_statd_var_run_t:s0
/usr/bin/umount.* -- system_u:object_r:mount_exec_t:s0

REAL WORLD EXAMPLES
Real World Examples

- Someone tells you to create a web directory somewhere non-standard - /foo/bar - for a virtual web site.
Real World Examples

- You create the directory:
[root@w541 ~]# mkdir -p /foo/bar
[root@w541 ~]# ls -l /foo/
total 4
drwxr-xr-x. 2 root root 4096 Jun 27 01:53 bar
[root@w541 ~]#
Real World Examples

- You define the virtual web site in httpd.conf:
<VirtualHost *:80>
    ServerAdmin webmaster@dummy-host.example.com
    DocumentRoot /foo/bar
    DirectoryIndex index.html
    ServerName dummy-host.example.com
    ErrorLog logs/dummy-host.example.com-error_log
    CustomLog logs/dummy-host.example.com-access_log common
</Directory>
</VirtualHost>

<VirtualHost *:80>
    ServerAdmin webmaster@w541.tc.redhat.com
    DocumentRoot /var/www/html
    DirectoryIndex index.html
    ServerName w541.tc.redhat.com
    ErrorLog logs/w541.tc.redhat.com-error_log
    CustomLog logs/w541.tc.redhat.com-access_log common
</VirtualHost>

"/etc/httpd/conf.d/virtnhost.conf" 23L, 609C written
Real World Examples

- You create an index.html file:
[root@w541 ~]# echo "this is the dummy-host.example.com web page" > /foo/bar/index.html
[root@w541 ~]# cat /foo/bar/index.html
this is the dummy-host.example.com web page
[root@w541 ~]#
Real World Examples

- Restart the web server:
[root@w541 ~]# systemctl restart httpd.service
Real World Examples

- When you test the page...
This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

For information on Red Hat Enterprise Linux, please visit the Red Hat, Inc. website. The documentation for Red Hat Enterprise Linux is available on the Red Hat, Inc. website.

If you are the website administrator:

You may now add content to the directory /var/www/html/. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file /etc/httpd/conf.d/99منذوم.com.conf.

You are free to use this image on your site if the Apache HTTP Server is the source of the content.
Real World Examples

- What logfile should we check?
Real World Examples

- journalctl
Real World Examples

- Note that at the end it tells you to restorecon!
Real World Examples

- What directory should we look at to get the correct context label?
[root@w541 ~]# ls -ldZ /var/www/html/
drwxr-xr-x. 4 root root system_u:object_r:httpd_sys_content_t:s0 4096 Jun 27 02:27 /var/www/html/
[root@w541 ~]#
Real World Examples

- We actually want all of the files under /foo to have the right context, so we'll use a regular expression (you can get the syntax from /etc/selinux/targeted/contexts/files/file_contexts):
[root@w541 ~]# ls -ldZ /var/www/html/
drw-r-xr-x. 4 root root system_u:object_r:httpd_sys_content_t:s0 4096 Jun 27 02:27 /var/www/html/
[root@w541 ~]# semanage fcontext -a -t httpd_sys_content_t "/foo(/.*)?"
Real World Examples

• Or, if you're like me (lazy), you can use the `-e` (equals) argument to `semanage fcontext`:
[root@w541 ~]# semanage fcontext -a -e /var/www/html /foo
Real World Examples

- Now run restorecon against the directory:
[root@w541 ~]# semanage fcontext -a -e /var/www/html /foo
[root@w541 ~]# restorecon -vR /foo
restorecon reset /foo context unconfined_u:object_r:default_t:s0->unconfined_u:object_r:httpd_sys_content_t:s0
restorecon reset /foo/bar context unconfined_u:object_r:default_t:s0->unconfined_u:object_r:httpd_sys_content_t:s0
restorecon reset /foo/bar/index.html context unconfined_u:object_r:default_t:s0->unconfined_u:object_r:httpd_sys_content_t:s0
[root@w541 ~]#
Real World Examples

• Test the site:
this is the dummy-host.example.com web page
POLICY MODULES
Creating Policy Modules

- In the case that a boolean or labeling does not fix your issue, you might have to create a policy module.
Creating Policy Modules

- In this example, I want to install squirrelmail on a RHEL 6 mail server. Other than using journalctl instead of /var/log/messages, the process is the same.
SquirrelMail - Login - Mozilla Firefox

SquirrelMail version 1.4.22
By the SquirrelMail Project Team

SquirrelMail Login

Name: fred
Password: ********

Login
SquirrelMail version 1.4.22
By the SquirrelMail Project Team

ERROR
Error connecting to IMAP server: localhost.
13 : Permission denied
Go to the login page
type=AVC msg=audit(1340321054.097:32692): avc: denied { name_connect } for pid=3593 comm="httpd" dest=143 scontext=unconfined_u:system_r:httpd_t:s0 tcontext=sys_user:r:pop_port_t:s0 tclass=tcp_socket
  type=SYSCALL msg=audit(1340321054.097:32692): arch=x86_64 syscall=42 success=no exit=-13 a0=13 a1=7f0939a05bb0 a2=1c a3=ff00 items=0 ppid=3590 pid=3593 auid=0
  uid=48 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=1 comm="httpd" exe="/usr/sbin/httpd" subj=unconfined_u:system_u:httpd_t:s0 key=(null)

type=AVC msg=audit(1340321054.098:32693): avc: denied { name_connect } for pid=3593 comm="httpd" dest=143 scontext=unconfined_u:system_r:httpd_t:s0 tcontext=sys_user:r:pop_port_t:s0 tclass=tcp_socket
  type=SYSCALL msg=audit(1340321054.098:32693): arch=x86_64 syscall=42 success=no exit=-13 a0=13 a1=7f0939a05250 a2=10 a3=7f093691814c items=0 ppid=3590 pid=3593 auid=0
  uid=48 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(none) ses=1 comm="httpd" exe="/usr/sbin/httpd" subj=unconfined_u:system_u:httpd_t:s0 key=(null)
usr/share/setroubleshoot/plugins/catchall_boolean.py", line 76, in check_for_man
    man_page = name.split("_")[0] + ".selinux"#012AttributeError: 'tuple' object has no attribute 'split'
Jun 21 18:23:31 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from name_connect access on the tcp_socket. For complete SELinux messages. run s
ealert -t f64ca3e4-4fe2-4998-85eb-de402ba79db2
Jun 21 18:23:31 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from name_connect access on the tcp_socket. For complete SELinux messages. run s
ealert -t f64ca3e4-4fe2-4998-85eb-de402ba79db2
Jun 21 18:24:15 armitage setroubleshoot: [avc.ERROR] Plugin Exception catchall_boolean #012Traceback (most recent call last):#012 File "/usr/lib64/python2.6/site-packages/setroubleshoot/analyze.py", line 191, in analyze_avc#012 report =
    plugin.analyze(avc)#012 File "/usr/share/setroubleshoot/plugins/catchall_boolean.py", line 90, in analyze#012 man_page = name.split("_")[0] + ".selinux"#012AttributeError: 'tuple' object has no attribute 'split'
Jun 21 18:24:15 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from name_connect access on the tcp_socket. For complete SELinux messages. run s
ealert -t f64ca3e4-4fe2-4998-85eb-de402ba79db2
Jun 21 18:24:15 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from name_connect access on the tcp_socket. For complete SELinux messages. run s
ealert -t f64ca3e4-4fe2-4998-85eb-de402ba79db2
[root@armitage ~]#
Creating Policy Modules

- Now that I know there is an SELinux issue, I set SELinux enforcement to “permissive” and then run the application through all its paces. In this case, sending and receiving mail.
- This will log denials but not act on them. If you don't do this, you'll fix one, trigger a second, fix the second, trigger a third, etc. It's easier to run the app in permissive mode and catch all of them.
[root@armitage ~]# setenforce 0
[root@armitage ~]#
To: barney@armitage.tc.redhat.com
Cc: 
Bcc: 
Subject: test from SquirrelMail
Priority: Normal
Receipt: □ On Read □ On Delivery

this is a test of sending e-mail!

Send
SquirrelMail Login

Name: barney
Password: ******

Login
From: fred@armitage.tc.redhat.com
Date: 6:07 pm
Subject: test from SquirrelMail
root@armitage:~# sealert -l f64ca3e4-4fe2-4998-85eb-de402ba79db2
Gtk-Message: Failed to load module "pk-gtk-module": libpk-gtk-module.so: cannot
open shared object file: No such file or directory
SELinux is preventing /usr/sbin/httpd from name_connect access on the tcp_socket.

***** Plugin catchall (100. confidence) suggests **************************
If you believe that httpd should be allowed name_connect access on the tcp_socket by default.
Then you should report this as a bug.
You can generate a local policy module to allow this access.
Do allow this access for now by executing:
# grep httpd /var/log/audit/audit.log | audit2allow -M mypol
# semodule -i mypol.pp

[root@armitage ~]#
[root@armitage ~]# grep httpd /var/log/audit/audit.log | audit2allow -M squirrel

                           IMPORTANT                           
To make this policy package active, execute:
semodule -i squirrellocal.pp

[root@armitage ~]#
Note

- Actually, this error could be fixed by setting a boolean. I am just creating a policy module so you can see it being done.
File Edit View Search Terminal Help
[root@armitage ~]# cat squirrellocal.te

module squirrellocal 1.0;

require {
    type httpd_t;
    type smtp_port_t;
    type pop_port_t;
    class tcp_socket name_connect;
}

#------------- httpd_t -------------
#!!! This avc can be allowed using one of the these booleans:
#    httpd_can_sendmail, allow_ypbind, httpd_can_network_connect

allow httpd_t pop_port_t:tcp_socket name_connect;
#!!! This avc can be allowed using one of the these booleans:
#    httpd_can_sendmail, allow_ypbind, httpd_can_network_connect

allow httpd_t smtp_port_t:tcp_socket name_connect;
[root@armitage ~]#
[root@armitage ~]# smodule -i squirrellocal.pp
[root@armitage ~]# setenforce 1
[root@armitage ~]#
SquirrelMail Login

Name: fred
Password: ********

Login
THIS FOLDER IS EMPTY
ENABLING SELINUX
Enabling SELinux

- To enable SELinux on a system, edit /etc/selinux/config and set SELINUX=permissive
- Do not set it to enforcing, as it will more than likely hang at boot time.
# 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=permissive
#SELINUXTYPE= can take one of three two values:
# targeted - Targeted processes are protected,
# minimum - Modification of targeted policy. Only selected processes are protected.
# mls - Multi Level Security protection.
SELINUXTYPE=targeted

"/etc/selinux/config" 14L, 547C written
Enabling SELinux

- Then create a file in the root of the filesystem called .autorelabel
[root@gluster-dev-aus-001 ~]# vi /etc/selinux/config
[root@gluster-dev-aus-001 ~]# touch /.autorelabel
[root@gluster-dev-aus-001 ~]#
Enabling SELinux

- Reboot, and the system will relabel the filesystem.
Enabling SELinux

- You can also run fixfiles relabel.
  - Don't do it in runlevel 5 - it deletes everything in /tmp and your X font server will get real cranky about that.
- Reboot after it's done.
Files in the /tmp directory may be labeled incorrectly. This command can remove all files in /tmp. If you choose to remove files from /tmp, a reboot will be required after completion.

Do you wish to clean out the /tmp directory (Y/N)?

Cleaning out /tmp

Warning: Skipping the following R/W filesystems:

Processing the following R/W filesystems:

Relabeling /boot /dev/adm /dev/audio /dev/vg_music /dev/vg_root /dev/vg_smp /dev/vg_smp /dev/vg_swap /dev/vg_smp

Cleaning up labels on /tmp
Enabling SELinux

- After everything is relabeled, then set it to enforcing in /etc/selinux/config and reboot or run setenforce 1.
GRAPHICAL TOOLS
Graphical Tools

- This stuff is so easy, even a Windows admin can do it!
  - Install xorg-x11-xauth, a font (I like bitmap-fixed-fonts, or you can do yum groupinstall fonts), and policycoreutils-gui. and you can ssh -Y into the box and run system-config-selinux
[root@gluster-dev-aus-001 ~]# yum -y install xorg-x11-xauth policycoreutils-gui bitmap-fixed-fonts
<table>
<thead>
<tr>
<th>Package</th>
<th>Architecture</th>
<th>Version</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>psmisc</td>
<td>x86_64</td>
<td>22.20-9.el7</td>
<td>140 K</td>
</tr>
<tr>
<td>pulseaudio-libs</td>
<td>x86_64</td>
<td>6.0-7.el7</td>
<td>576 K</td>
</tr>
<tr>
<td>pycairo</td>
<td>x86_64</td>
<td>1.8.10-8.el7</td>
<td>157 K</td>
</tr>
<tr>
<td>pygtk2</td>
<td>x86_64</td>
<td>2.24.0-9.el7</td>
<td>914 K</td>
</tr>
<tr>
<td>pygtk2-libglade</td>
<td>x86_64</td>
<td>2.24.0-9.el7</td>
<td>25 K</td>
</tr>
<tr>
<td>pyorbit</td>
<td>x86_64</td>
<td>2.24.0-15.el7</td>
<td>51 K</td>
</tr>
<tr>
<td>python-IPy</td>
<td>noarch</td>
<td>0.75-6.el7</td>
<td>32 K</td>
</tr>
<tr>
<td>rest</td>
<td>x86_64</td>
<td>0.7.92-3.el7</td>
<td>62 K</td>
</tr>
<tr>
<td>selinux-policy-devel</td>
<td>noarch</td>
<td>3.13.1-60.el7_2.7</td>
<td>33M</td>
</tr>
<tr>
<td>setools-libs</td>
<td>x86_64</td>
<td>3.3.7-46.el7</td>
<td>485 K</td>
</tr>
<tr>
<td>sound-theme-freedesktop</td>
<td>noarch</td>
<td>0.8-3.el7</td>
<td>377 K</td>
</tr>
<tr>
<td>startup-notification</td>
<td>x86_64</td>
<td>0.12-8.el7</td>
<td>39 K</td>
</tr>
<tr>
<td>udisks2</td>
<td>x86_64</td>
<td>2.1.2-6.el7</td>
<td>312 K</td>
</tr>
<tr>
<td>usermode-gtk</td>
<td>x86_64</td>
<td>1.111-5.el7</td>
<td>110 K</td>
</tr>
<tr>
<td>xcb-util</td>
<td>x86_64</td>
<td>0.4.0-2.el7</td>
<td>16 K</td>
</tr>
<tr>
<td>xml-common</td>
<td>noarch</td>
<td>0.6.3-39.el7</td>
<td>26 K</td>
</tr>
</tbody>
</table>

Transaction Summary
=======================
Install 3 Packages (+120 Dependent packages)

Total download size: 51 M
Installed size: 172 M
Is this ok [y/d/N]: y
mesa-libglapi.x86_64 0:10.6.5-3.20150824.el7
pango.x86_64 0:1.36.8-2.el7
pixman.x86_64 0:0.32.6-3.el7
policycoreutils-devel.x86_64 0:2.2.5-20.el7
policycoreutils-python.x86_64 0:2.2.5-20.el7
psmisc.x86_64 0:22.20-9.el7
pulseaudio-libs.x86_64 0:6.0-7.el7
pycairo.x86_64 0:1.18.10-8.el7
pygtk2.x86_64 0:2.24.0-9.el7
pygtk2-libglade.x86_64 0:2.24.0-9.el7
pyorbit.x86_64 0:2.24.0-15.el7
python-IPy.noarch 0:0.75-6.el7
rest.x86_64 0:0.7.92-3.el7
selinux-policy-devel.noarch 0:3.13.1-60.el7_2.7
setools-libs.x86_64 0:3.3.7-46.el7
sound-theme-freedesktop.noarch 0:0.8-3.el7
startup-notification.x86_64 0:0.12-8.el7
udisks2.x86_64 0:2.1.2-6.el7
usermode-gtk.x86_64 0:1.111-5.el7
xcb-util.x86_64 0:0.4.0-2.el7
xml-common.noarch 0:0.6.3-39.el7

Complete!
[root@gluster-dev-aus-001 ~]#
File Edit View Terminal Tabs Help
[tcameron@w541 ~]$ ssh -Y root@gluster-dev-aus-001
Warning: Permanently added 'gluster-dev-aus-001,192.168.124.125' (ECDSA) to the list of known hosts.
root@gluster-dev-aus-001's password:
Last login: Mon Jun 27 18:07:04 2016 from w541.tc.redhat.com
This is a private system. Trespassers will be violated!
/usr/bin/xauth:  file /root/.Xauthority does not exist
[root@gluster-dev-aus-001 ~]#  

Terminal - root@gluster-dev-aus-001:

Last login: Mon Jun 27 18:07:04 2016 from w541.tc.redhat.com
This is a private system. Trespassers will be violated!
/usr/bin/xauth: file /root/.Xauthority does not exist
[root@gluster-dev-aus-001 ~]# system-config-selinux
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
/usr/share/system-config-selinux/system-config-selinux.py:77: Warning: g_object_get_valist: object class 'GnomeProgram' has no property named 'default-icon'
    xml = gtk.glade.XML("/usr/share/system-config-selinux/system-config-selinux.glade", domain=PROGNAME)
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
GConf Error: Client failed to connect to the D-BUS daemon:
/bin/dbus-launch terminated abnormally without any error message
GConf Error: Client failed to connect to the D-BUS daemon:
**SELinux Administration**

<table>
<thead>
<tr>
<th>Select:</th>
<th>System Default Enforcing Mode</th>
<th>Current Enforcing Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Permissive</td>
<td>Permissive</td>
</tr>
<tr>
<td>Boolean</td>
<td></td>
<td></td>
</tr>
<tr>
<td>File Labeling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User Mapping</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SELinux User</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network Port</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Policy Module</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Process Domain</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- System Default Policy Type: targeted
- Relabel on next reboot.
<table>
<thead>
<tr>
<th>Active</th>
<th>Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>abrt</td>
<td>Determine whether abrt-handle-upload shall be permitted</td>
</tr>
<tr>
<td></td>
<td>abrt</td>
<td>Determine whether ABRT can run in threads</td>
</tr>
<tr>
<td></td>
<td>abrt</td>
<td>Allow ABRT to modify public files</td>
</tr>
<tr>
<td></td>
<td>antivirus</td>
<td>Determine whether can antivirus programs to read non</td>
</tr>
<tr>
<td></td>
<td>antivirus</td>
<td>Allow antivirus programs to read non</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Allow httpd to access cifs file syst</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Allow Apache to modify public files</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Don't audit Apache to search dirs.</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Allow Apache to query NS records</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Allow httpd cgi support</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Allow Apache to communicate with sss</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Allow httpd to run gpg</td>
</tr>
<tr>
<td></td>
<td>apache</td>
<td>Allow HTTPD scripts and modules to communicate with</td>
</tr>
<tr>
<td>File</td>
<td>Selinux Type</td>
<td>File Type</td>
</tr>
<tr>
<td>------</td>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>/</td>
<td>root_t:s0</td>
<td>direct</td>
</tr>
<tr>
<td>./*</td>
<td>default_t:s0</td>
<td>all</td>
</tr>
<tr>
<td>/[^/]+</td>
<td>etc_runtime_t:s0</td>
<td>regu</td>
</tr>
<tr>
<td>/afs</td>
<td>mnt_t:s0</td>
<td>direct</td>
</tr>
<tr>
<td>/a?[quota.(user</td>
<td>group)</td>
<td>quota_db_t:s0</td>
</tr>
<tr>
<td>/.autofsck</td>
<td>etc_runtime_t:s0</td>
<td>regu</td>
</tr>
<tr>
<td>/.autorelabel</td>
<td>etc_runtime_t:s0</td>
<td>regu</td>
</tr>
<tr>
<td>/bacula(/.*)?</td>
<td>bacula_store_t:s0</td>
<td>all</td>
</tr>
<tr>
<td>/bin</td>
<td>bin_t:s0</td>
<td>all</td>
</tr>
<tr>
<td>/bin/.*</td>
<td>bin_t:s0</td>
<td>all</td>
</tr>
<tr>
<td>/bin/alsaunmute</td>
<td>alsa_exec_t:s0</td>
<td>regu</td>
</tr>
<tr>
<td>/bin/bash</td>
<td>shell_exec_t:s0</td>
<td>regu</td>
</tr>
<tr>
<td>/bin/bash2</td>
<td>shell_exec_t:s0</td>
<td>regu</td>
</tr>
<tr>
<td>/bin/dash</td>
<td>shell_exec_t:s0</td>
<td>regu</td>
</tr>
</tbody>
</table>
### SELinux Administration

**Select:**
- Status
- Boolean
- File Labeling
- User Mapping
- SELinux User
- Network Port
- Policy Module
- Process Domain

**Add**
- Properties
- Delete
- Customized

**Filter**
- `foo`

<table>
<thead>
<tr>
<th>File Specification</th>
<th>Selinux File Type</th>
<th>File Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/foo(/.*)?</code></td>
<td><code>httpd_sys_content_t:s0</code></td>
<td>all files</td>
</tr>
<tr>
<td><code>/usr/share/foomatic/db/oldprinterids</code></td>
<td><code>cupsd_rw etc_t:s0</code></td>
<td>regular files</td>
</tr>
<tr>
<td><code>/var/cache/foomatic(/.*)?</code></td>
<td><code>cupsd_rw etc_t:s0</code></td>
<td>all files</td>
</tr>
</tbody>
</table>
### SELinux Port Table

<table>
<thead>
<tr>
<th>SELinux Port Type</th>
<th>Protocol</th>
<th>MLS/MCS Level</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>afs3_callback_port_t</td>
<td>udp</td>
<td>s0</td>
<td>7001</td>
</tr>
<tr>
<td>afs3_callback_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>7001</td>
</tr>
<tr>
<td>afs_bos_port_t</td>
<td>udp</td>
<td>s0</td>
<td>7007</td>
</tr>
<tr>
<td>afs_fs_port_t</td>
<td>udp</td>
<td>s0</td>
<td>7000</td>
</tr>
<tr>
<td>afs_fs_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>2040</td>
</tr>
<tr>
<td>afs_fs_port_t</td>
<td>udp</td>
<td>s0</td>
<td>7005</td>
</tr>
<tr>
<td>afs_ka_port_t</td>
<td>udp</td>
<td>s0</td>
<td>7004</td>
</tr>
<tr>
<td>afs_pl_port_t</td>
<td>udp</td>
<td>s0</td>
<td>7002</td>
</tr>
<tr>
<td>afs_vl_port_t</td>
<td>udp</td>
<td>s0</td>
<td>7003</td>
</tr>
<tr>
<td>agentx_port_t</td>
<td>udp</td>
<td>s0</td>
<td>705</td>
</tr>
<tr>
<td>agentx_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>705</td>
</tr>
<tr>
<td>amanda_port_t</td>
<td>udp</td>
<td>s0</td>
<td>10080-10082</td>
</tr>
</tbody>
</table>
### SELinux Administration

#### Select:
- Status
- Boolean
- File Labeling
- User Mapping
- SELinux User
- Network Port
- Process Domain

#### SELinux Port Type

<table>
<thead>
<tr>
<th>SELinux Port Type</th>
<th>Protocol</th>
<th>MLS/MCS Level</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>http_cache_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>8118</td>
</tr>
<tr>
<td>http_cache_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>8080</td>
</tr>
<tr>
<td>http_cache_port_t</td>
<td>udp</td>
<td>s0</td>
<td>3130</td>
</tr>
<tr>
<td>http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>488</td>
</tr>
<tr>
<td>http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>443</td>
</tr>
<tr>
<td>http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>81</td>
</tr>
<tr>
<td><strong>http_port_t</strong></td>
<td><strong>tcp</strong></td>
<td><strong>s0</strong></td>
<td><strong>80</strong></td>
</tr>
<tr>
<td>http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>8008</td>
</tr>
<tr>
<td>http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>8009</td>
</tr>
<tr>
<td>http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>8443</td>
</tr>
<tr>
<td>http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>9000</td>
</tr>
<tr>
<td>pegasus_http_port_t</td>
<td>tcp</td>
<td>s0</td>
<td>5988</td>
</tr>
</tbody>
</table>
### Policy Module

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>abrt</td>
<td>1.2.0</td>
</tr>
<tr>
<td>accountsd</td>
<td>1.0.0</td>
</tr>
<tr>
<td>ada</td>
<td>1.4.0</td>
</tr>
<tr>
<td>afs</td>
<td>1.5.3</td>
</tr>
<tr>
<td>micu</td>
<td>1.0.0</td>
</tr>
<tr>
<td>aide</td>
<td>1.5.0</td>
</tr>
<tr>
<td>aiseexec</td>
<td>1.0.0</td>
</tr>
<tr>
<td>amanda</td>
<td>1.12.0</td>
</tr>
<tr>
<td>amavisi</td>
<td>1.10.3</td>
</tr>
<tr>
<td>antu</td>
<td>1.2.0</td>
</tr>
<tr>
<td>apache</td>
<td>2.1.2</td>
</tr>
<tr>
<td>apcupsd</td>
<td>1.6.1</td>
</tr>
<tr>
<td>arpswatch</td>
<td>1.8.1</td>
</tr>
<tr>
<td>asterisk</td>
<td>1.7.1</td>
</tr>
</tbody>
</table>
CONCLUSION
Hopefully you feel like this, now!
Final Thoughts

- Don't turn it off!
- SELinux can really save you in the event of a breach.
- It's much easier to use SELinux today than it was just a few years ago
- NSA grade security is available at no extra cost - use it!
Thank You!

- If you liked today's presentation, please rate it!
More Information

- Fedora Project SELinux Docs: http://fedoraproject.org/wiki/SELinux
- fedora-selinux-list (mailing list):
  - https://www.redhat.com/mailman/listinfo
More Information

- http://access.redhat.com has several videos about SELinux. Dave Egts and Dan Walsh have covered topics from confining users to sandboxing.
- Dan Walsh's blog:
  - http://danwalsh.livejournal.com/
THANK YOU

plus.google.com/+RedHat
linkedin.com/company/red-hat
facebook.com/redhatinc
twitter.com/RedHatNews
youtube.com/user/RedHatVideos
LEARN. NETWORK. EXPERIENCE OPEN SOURCE.