

**RED HAT
SUMMIT**

**LEARN. NETWORK.
EXPERIENCE OPEN SOURCE.**

June 11-14, 2013
Boston, MA





SELINUX FOR MERE MORTALS

Thomas Cameron, RHCA, RHCSS, RHCDS, RHCVA, RHCX
Chief Architect, Red Hat
06.13.2013

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

- thomas@redhat.com
- thomasdcameron on Twitter
- choirboy on #rhel on Freenode
- <http://people.redhat.com/tcameron>
- <http://excogitat.us>
- thomas.cameron on Google talk

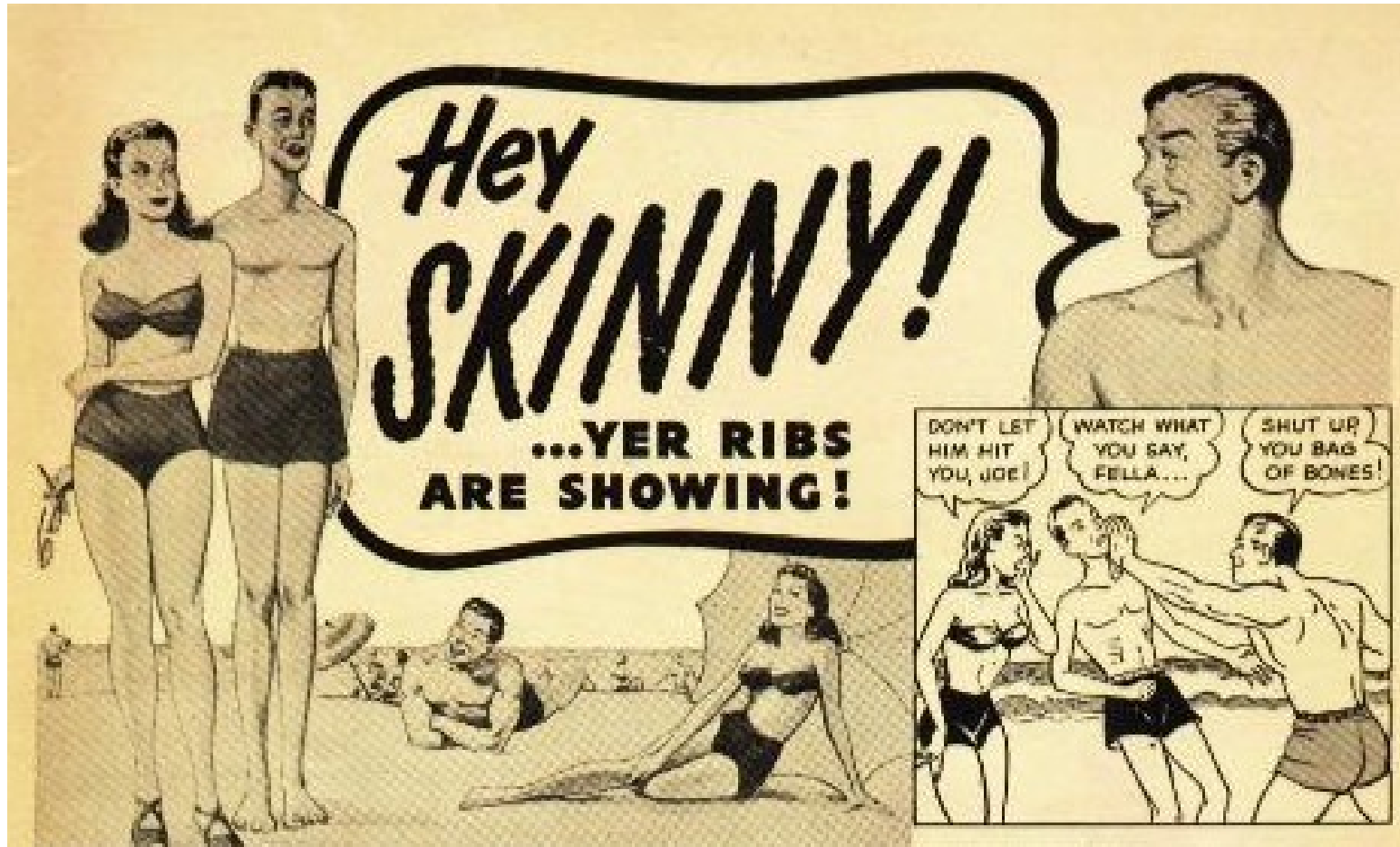
About Us

- 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:
- <http://www.nsa.gov/research/selinux/contrib.shtml>
- 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 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

What Thomas thought SELinux was



If you feel the same way...

If you feel the same way...

- You're in the right place!

What is SELinux?

What is SELinux?

- SELinux is an example of a Mandatory Access Control system for Linux.

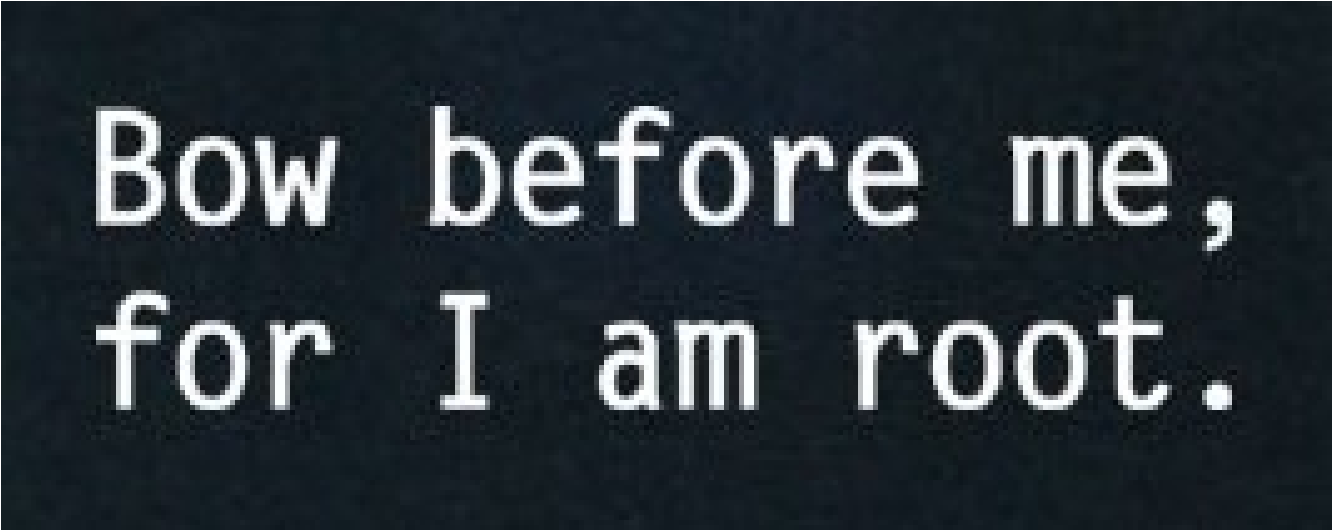
DAC vs. MAC

- 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.



Bow before me,
for I am root.

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

- In Red Hat Enterprise Linux 6, 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@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# 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 - SELinux is fully disabled.  
SELINUX=enforcing  
# SELINUXTYPE= type of policy in use. Possible values are:  
#     targeted - Only targeted network daemons are protected.  
#     strict - Full SELinux protection.  
SELINUXTYPE=targeted  
[root@armitage ~]#
```

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# /usr/sbin/sestatus  
SELinux status:                enabled  
SELinuxfs mount:                /selinux  
Current mode:                   enforcing  
Mode from config file:          enforcing  
Policy version:                 24  
Policy from config file:        targeted  
[root@armitage ~]#
```

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# /usr/sbin/getenforce  
Enforcing  
[root@armitage ~]# █
```

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@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -lZ /usr/sbin/httpd  
-rwxr-xr-x. root root system_u:object_r:httpd_exec_t:s0 /usr/sbin/httpd  
[root@armitage ~]#
```

So How Does SELinux Work?

- The web server's configuration directory is labeled `httpd_config_t`:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -dZ /etc/httpd/  
drwxr-xr-x. root root system_u:object_r:httpd_config_t:s0 /etc/httpd/  
[root@armitage ~]#
```

So How Does SELinux Work?

- The web server's logfile directory is labeled `httpd_log_t`:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -dZ /var/log/httpd/  
drwx-----. root root system_u:object_r:httpd_log_t:s0 /var/log/httpd/  
[root@armitage ~]#
```

So How Does SELinux Work?

- The web server's content directory is labeled `httpd_sys_content_t`:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -dZ /var/www/html/  
drwxr-xr-x. root root system_u:object_r:httpd_sys_content_t:s0 /var/www/html/  
[root@armitage ~]#
```

So How Does SELinux Work?

- The web server's startup script is labeled `httpd_initrc_exec_t`:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -Z /etc/rc.d/init.d/httpd  
-rwxr-xr-x. root root system_u:object_r:httpd_initrc_exec_t:s0 /etc/rc.d/init.d/  
httpd  
[root@armitage ~]#
```

So How Does SELinux Work?

- As the web server runs, it's process is labeled httpd_t:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ps axZ | grep [h]ttpd  
unconfined_u:system_r:httpd_t:s0 9448 ? Ss 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9450 ? S 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9451 ? S 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9452 ? S 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9453 ? S 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9454 ? S 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9455 ? S 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9456 ? S 0:00 /usr/sbin/httpd  
unconfined_u:system_r:httpd_t:s0 9457 ? S 0:00 /usr/sbin/httpd  
[root@armitage ~]#
```

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.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# netstat -tnlpZ | grep httpd  
tcp        0      0 :::80                :::*                LISTEN        2135/httpd         unconfined_u:system_r:httpd_t:s0  
tcp        0      0 :::443               :::*                LISTEN        2135/httpd         unconfined_u:system_r:httpd_t:s0  
[root@armitage ~]#
```

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# semanage port -l | grep http  
http_cache_port_t      tcp      3128, 8080, 8118, 8123, 10001-10010  
http_cache_port_t      udp      3130  
http_port_t            tcp      80, 443, 488, 8008, 8009, 8443  
pegasus_http_port_t    tcp      5988  
pegasus_https_port_t   tcp      5989  
[root@armitage ~]#
```

So How Does SELinux Work?

- Now then... The `/etc/shadow` file has a type `shadow_t`:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -Z /etc/shadow  
-----. root root system_u:object_r:shadow_t:s0 /etc/shadow  
[root@armitage ~]#
```

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

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!
- 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!

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`

```
root@armitage:~  
File Edit View Search Terminal Help  
abrt_anon_write --> off  
abrt_handle_event --> off  
allow_console_login --> on  
allow_cvs_read_shadow --> off  
allow_daemons_dump_core --> on  
allow_daemons_use_tcp_wrapper --> off  
allow_daemons_use_tty --> on  
allow_domain_fd_use --> on  
allow_execheap --> off  
allow_execmem --> on  
allow_execmod --> on  
allow_execstack --> on  
allow_ftpd_anon_write --> off  
allow_ftpd_full_access --> off  
allow_ftpd_use_cifs --> off  
allow_ftpd_use_nfs --> off  
allow_gssd_read_tmp --> on  
allow_guest_exec_content --> off  
allow_httpd_anon_write --> off  
allow_httpd_mod_auth_ntlm_winbind --> off  
allow_httpd_mod_auth_pam --> off  
allow_httpd_sys_script_anon_write --> off  
allow_java_execstack --> off  
:
```

```
root@armitage:~  
File Edit View Search Terminal Help  
git_system_use_nfs --> off  
global_ssp --> off  
gpg_agent_env_file --> off  
gpg_web_anon_write --> off  
httpd_builtin_scripting --> on  
httpd_can_check_spam --> off  
httpd_can_network_connect --> off  
httpd_can_network_connect_cobbler --> off  
httpd_can_network_connect_db --> off  
httpd_can_network_memcache --> off  
httpd_can_network_relay --> off  
httpd_can_sendmail --> off  
httpd_dbus_avahi --> on  
httpd_enable_cgi --> on  
httpd_enable_ftp_server --> off  
httpd_enable_homedirs --> off  
httpd_execmem --> off  
httpd_manage_ipa --> off  
httpd_read_user_content --> off  
httpd_setrlimit --> off  
httpd_ssi_exec --> off  
httpd_tmp_exec --> off  
httpd_tty_comm --> on  
:
```

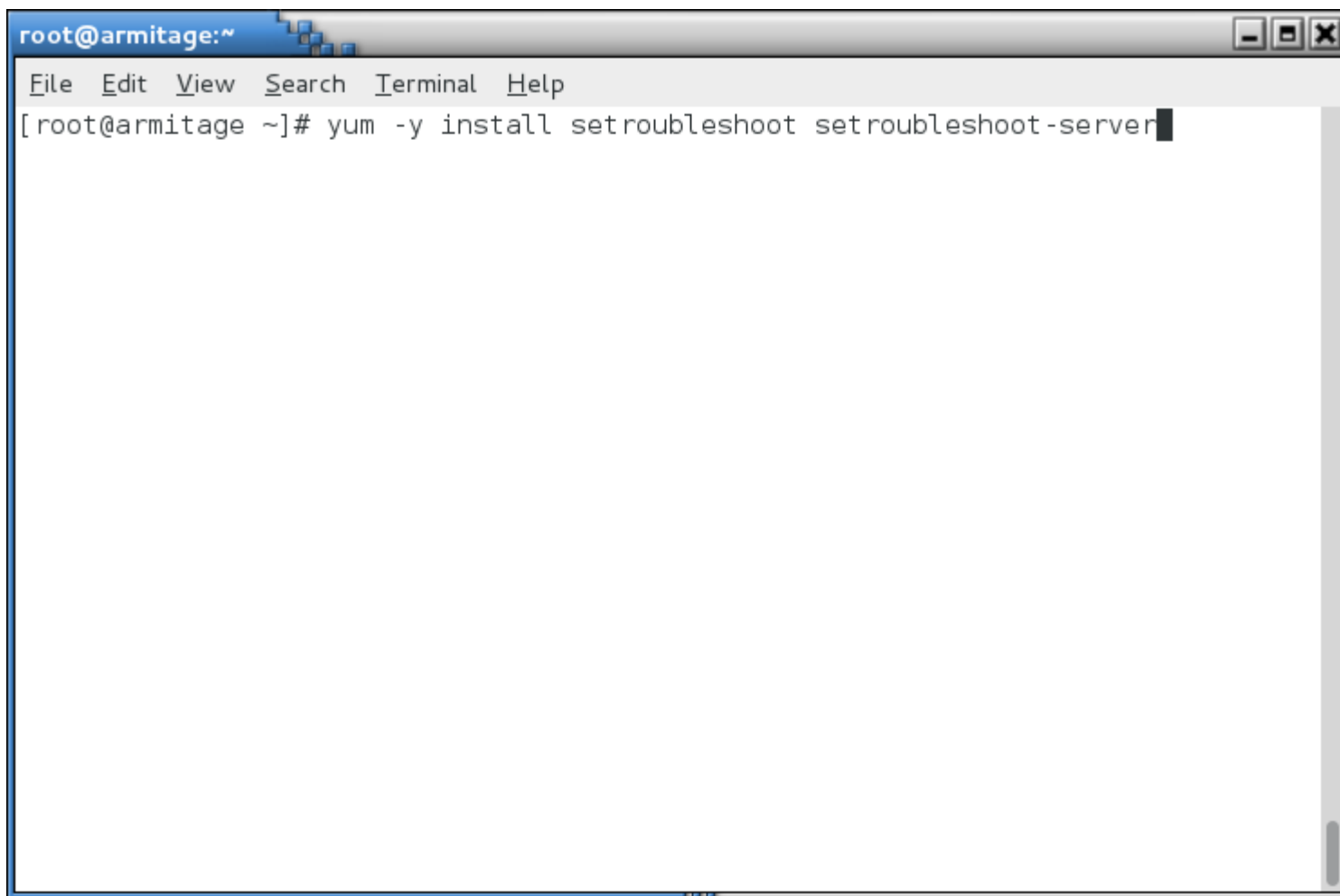
```
root@armitage:~  
File Edit View Search Terminal Help  
secure_mode_policyload --> off  
sepgsql_enable_users_ddl --> on  
sepgsql_unconfined_dbadm --> on  
sge_domain_can_network_connect --> off  
sge_use_nfs --> off  
smartmon_3ware --> off  
spamassassin_can_network --> off  
spamd_enable_home_dirs --> on  
squid_connect_any --> on  
squid_use_tproxy --> off  
ssh_chroot_rw_homedirs --> off  
ssh_sysadm_login --> off  
telepathy_tcp_connect_generic_network_ports --> off  
tftp_anon_write --> off  
tor_bind_all_unreserved_ports --> off  
unconfined_login --> on  
unconfined_mmap_zero_ignore --> off  
unconfined_mozilla_plugin_transition --> off  
use_fusefs_home_dirs --> off  
use_lpd_server --> off  
use_nfs_home_dirs --> on  
use_samba_home_dirs --> off  
user_direct_dri --> on  
:
```

What Are Booleans?

- To set a boolean, run `setsebool [boolean] [0|1]`
- To make it permanent, pass the `-P` argument to `setsebool`

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.



A terminal window titled "root@armitage:~" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal displays the command "[root@armitage ~]# yum -y install setroubleshoot setroubleshoot-server" with a cursor at the end of the line.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# yum -y install setroubleshoot setroubleshoot-server
```

```
root@armitage:~  
File Edit View Search Terminal Help  
libvorbis.x86_64 1:1.2.3-4.el6_2.1  
libwnck.x86_64 0:2.28.0-3.el6  
make.x86_64 1:3.81-20.el6  
notification-daemon.x86_64 0:0.5.0-1.el6  
notify-python.x86_64 0:0.1.1-10.el6  
policycoreutils-python.x86_64 0:2.0.83-19.24.el6  
pulseaudio-libs.x86_64 0:0.9.21-13.el6  
pycairo.x86_64 0:1.8.6-2.1.el6  
pygtk2.x86_64 0:2.16.0-3.el6  
pygtk2-libglade.x86_64 0:2.16.0-3.el6  
python-decorator.noarch 0:3.0.1-3.1.el6  
python-slip.noarch 0:0.2.20-1.el6_2  
python-slip-dbus.noarch 0:0.2.20-1.el6_2  
setools-libs.x86_64 0:3.3.7-4.el6  
setools-libs-python.x86_64 0:3.3.7-4.el6  
setroubleshoot-plugins.noarch 0:3.0.40-1.el6  
sgml-common.noarch 0:0.6.3-32.el6  
sound-theme-freedesktop.noarch 0:0.7-3.el6  
startup-notification.x86_64 0:0.10-2.1.el6  
xcb-util.x86_64 0:0.3.6-1.el6  
xml-common.noarch 0:0.6.3-32.el6  
  
Complete!  
[root@armitage ~]#
```

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# service auditd restart  
Stopping auditd: [ OK ]  
Starting auditd: [ OK ]  
[root@armitage ~]#
```

Real World Examples

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/httpd.conf
 - Restart the web server

```
root@armitage:~  
File Edit View Search Terminal Help  
# 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.  
#  
# See also: http://httpd.apache.org/docs/misc/FAQ.html#forbidden  
#  
<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>  
"/etc/httpd/conf/httpd.conf" 1009L, 34418C written
```

Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
 - Change permissions so the web server can access his home directory.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# chmod o+x /home/fred/  
[root@armitage ~]# ls -ld /home/fred/  
drwx-----x. 2 fred fred 4096 Jun 20 23:17 /home/fred/  
[root@armitage ~]#
```

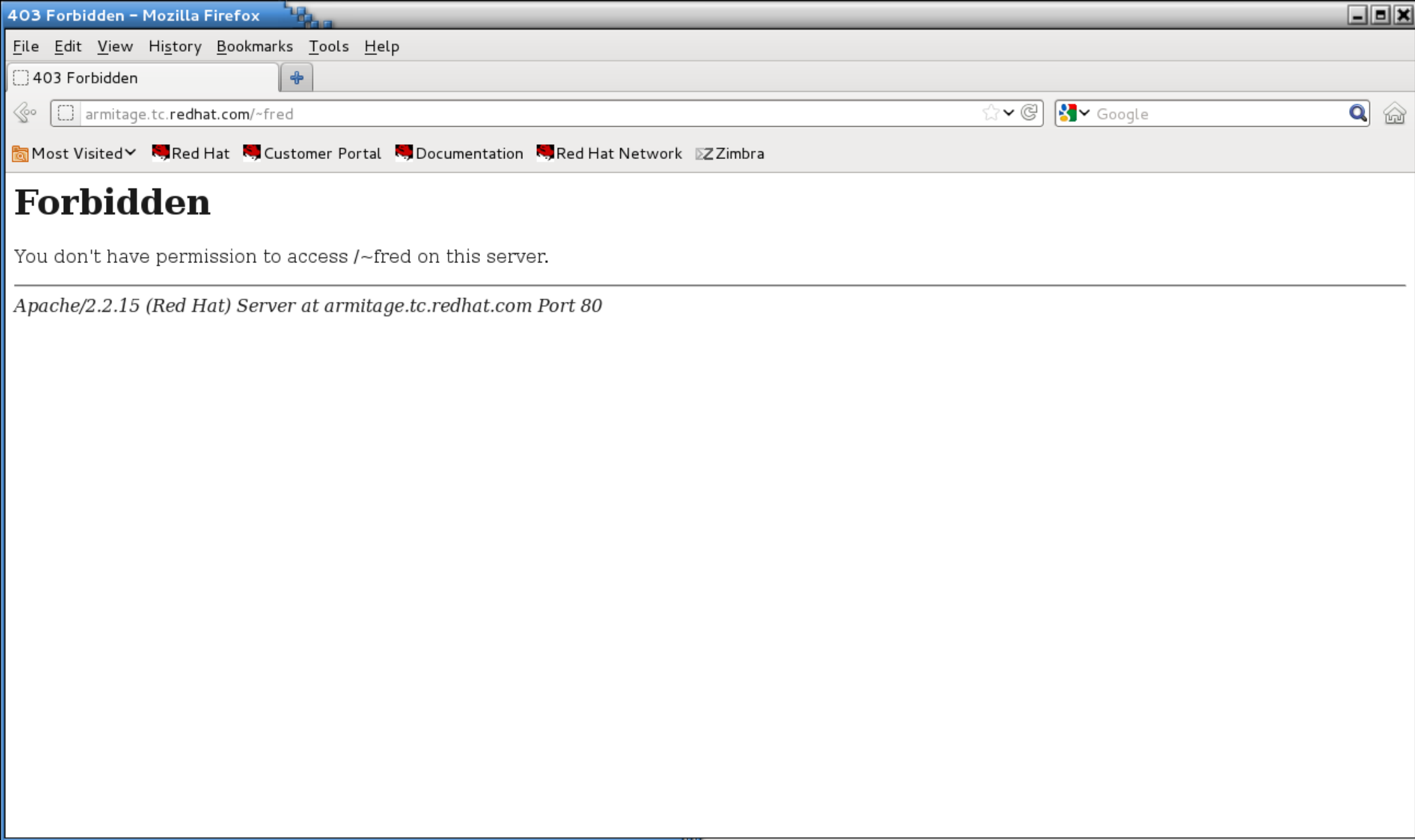
Real World Examples

- 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.

```
fred@armitage:~/public_html
File Edit View Search Terminal Help
[fred@armitage ~]$ who am i
fred pts/1 2012-06-21 10:07 (armitage.tc.redhat.com)
[fred@armitage ~]$ mkdir public_html
[fred@armitage ~]$ cd public_html/
[fred@armitage public_html]$ echo "this is my home page" > index.html
[fred@armitage 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:



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

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# tail /var/log/httpd/access_log  
172.31.100.4 - - [21/Jun/2012:10:10:14 -0500] "GET / HTTP/1.1" 403 3985 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:13.0) Gecko/20100101 Firefox/13.0"  
172.31.100.4 - - [21/Jun/2012:10:10:14 -0500] "GET /icons/apache_pb2.gif HTTP/1.1" 200 1797 "http://armitage.tc.redhat.com/" "Mozilla/5.0 (X11; Linux x86_64; rv:13.0) Gecko/20100101 Firefox/13.0"  
172.31.100.4 - - [21/Jun/2012:10:10:15 -0500] "GET /favicon.ico HTTP/1.1" 404 298 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:13.0) Gecko/20100101 Firefox/13.0"  
172.31.100.4 - - [21/Jun/2012:10:10:15 -0500] "GET /favicon.ico HTTP/1.1" 404 298 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:13.0) Gecko/20100101 Firefox/13.0"  
172.31.100.4 - - [21/Jun/2012:10:10:22 -0500] "GET /~fred HTTP/1.1" 403 296 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:13.0) Gecko/20100101 Firefox/13.0"  
172.31.100.4 - - [21/Jun/2012:10:12:50 -0500] "GET /~fred HTTP/1.1" 403 296 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:13.0) Gecko/20100101 Firefox/13.0"  
172.31.100.4 - - [21/Jun/2012:10:12:51 -0500] "GET /~fred HTTP/1.1" 403 296 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:13.0) Gecko/20100101 Firefox/13.0"  
[root@armitage ~]#
```

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# tail /var/log/httpd/error_log  
[Thu Jun 21 10:10:03 2012] [notice] Digest: done  
[Thu Jun 21 10:10:03 2012] [warn] mod_wsgi: Compiled for Python/2.6.2.  
[Thu Jun 21 10:10:03 2012] [warn] mod_wsgi: Runtime using Python/2.6.6.  
[Thu Jun 21 10:10:03 2012] [notice] Apache/2.2.15 (Unix) DAV/2 mod_ssl/2.2.15 OpenSSL/1.0.0-fips mod_wsgi/3.2 Python/2.6.6 mod_perl/2.0.4 Perl/v5.10.1 configured -- resuming normal operations  
[Thu Jun 21 10:10:14 2012] [error] [client 172.31.100.4] Directory index forbidden by Options directive: /var/www/html/  
[Thu Jun 21 10:10:15 2012] [error] [client 172.31.100.4] File does not exist: /var/www/html/favicon.ico  
[Thu Jun 21 10:10:15 2012] [error] [client 172.31.100.4] File does not exist: /var/www/html/favicon.ico  
[Thu Jun 21 10:10:22 2012] [error] [client 172.31.100.4] (13)Permission denied: access to /~fred denied  
[Thu Jun 21 10:12:50 2012] [error] [client 172.31.100.4] (13)Permission denied: access to /~fred denied  
[Thu Jun 21 10:12:51 2012] [error] [client 172.31.100.4] (13)Permission denied: access to /~fred denied  
[root@armitage ~]#
```

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 /var/log/messages

```
root@armitage:~  
File Edit View Search Terminal Help  
5167  
Jun 21 09:44:21 armitage audispd: audispd initialized with q_depth=120 and 1 active plugins  
Jun 21 09:44:21 armitage auditd[25165]: Init complete, auditd 2.2 listening for events (startup state enable)  
Jun 21 10:10:24 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from search access on the directory /home/fred. For complete SELinux messages. run sealert -l 9f88e0bb-5f4b-4e3a-96b2-7644917fbfc4  
Jun 21 10:10:24 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from getattr access on the directory /home/fred. For complete SELinux messages. run sealert -l 37acc7d8-e955-4359-8ac5-1d027bfcea72  
Jun 21 10:12:52 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from search access on the directory /home/fred. For complete SELinux messages. run sealert -l 9f88e0bb-5f4b-4e3a-96b2-7644917fbfc4  
Jun 21 10:12:52 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from getattr access on the directory /home/fred. For complete SELinux messages. run sealert -l 37acc7d8-e955-4359-8ac5-1d027bfcea72  
Jun 21 10:12:52 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from search access on the directory /home/fred. For complete SELinux messages. run sealert -l 9f88e0bb-5f4b-4e3a-96b2-7644917fbfc4  
Jun 21 10:12:52 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from getattr access on the directory /home/fred. For complete SELinux messages. run sealert -l 37acc7d8-e955-4359-8ac5-1d027bfcea72  
[root@armitage ~]#
```

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 9f88e0bb-5f4b-4e3a-96b2-7644917fbfc4”
 - It reveals that there are two issues.
 - User content
 - httpd access to home directories

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]#  
[root@armitage ~]# sealert -l 9f88e0bb-5f4b-4e3a-96b2-7644917fbfc4  
SELinux is preventing /usr/sbin/httpd from search access on the directory /home/  
fred.  
  
***** Plugin catchall_boolean (47.5 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 'user_selinux' man page for more details.  
Do  
setsebool -P httpd_read_user_content 1  
  
***** Plugin catchall_boolean (47.5 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' bo  
olean.You can read 'user_selinux' man page for more details.  
Do  
setsebool -P httpd_enable_homedirs 1  
  
***** Plugin catchall (6.38 confidence) suggests *****  
  
If you believe that httpd should be allowed search access on the fred directory
```

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.

```
root@armitage:~  
File Edit View Search Terminal Help  
Do  
setsebool -P httpd_read_user_content 1  
  
***** Plugin catchall_boolean (47.5 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' bo  
olean.You can read 'user_selinux' man page for more details.  
Do  
setsebool -P httpd_enable_homedirs 1  
  
***** Plugin catchall (6.38 confidence) suggests *****  
  
If you believe that httpd should be allowed search access on the fred directory  
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 ~]#
```

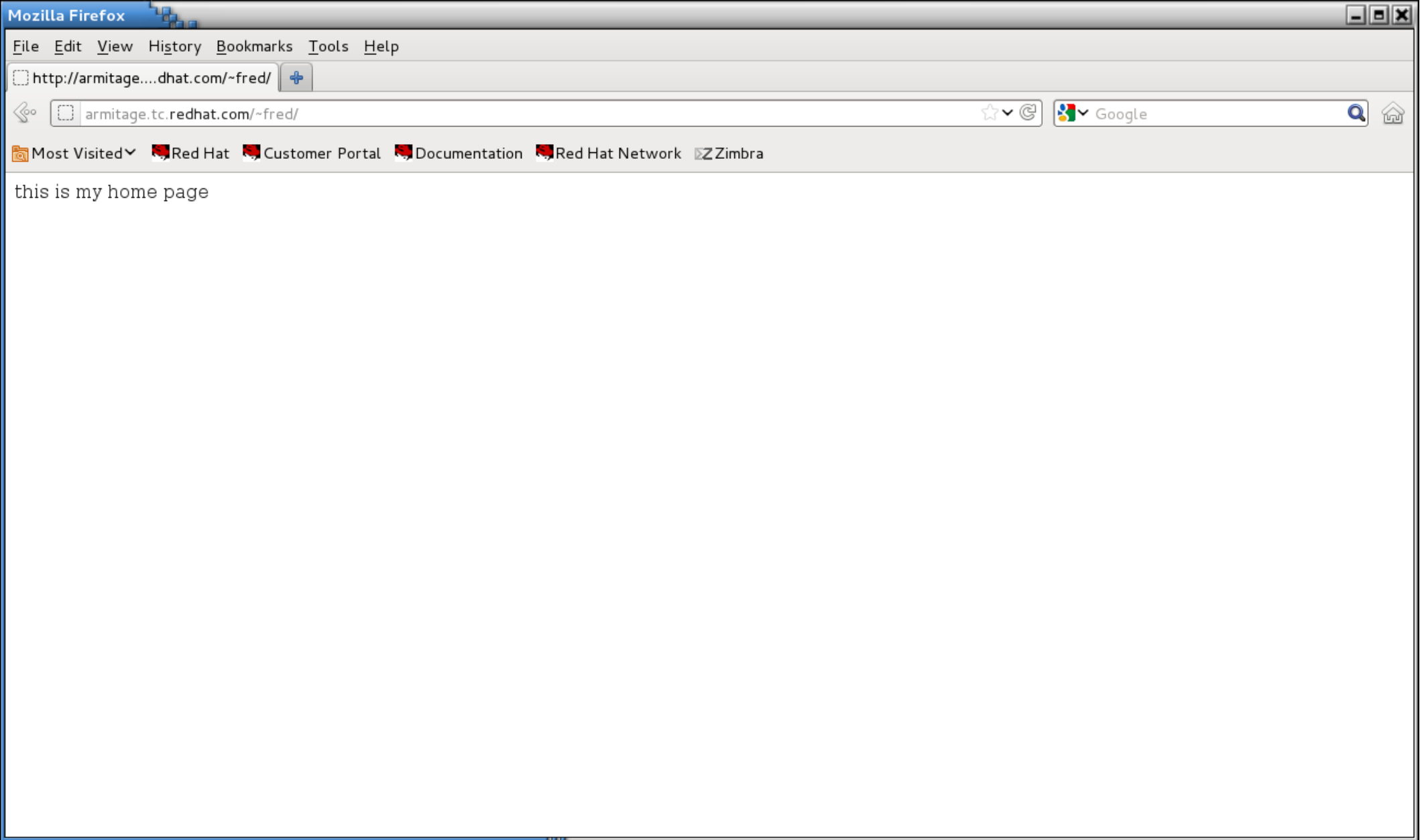
Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
 - Follow the instructions and set the two booleans.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# setsebool -P httpd_read_user_content 1; setsebool -P httpd_enable_homedirs 1
```

Real World Examples

- A user, fred, wants to start have his own web page in /home/fred/public_html
 - And... Voila!



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/`

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# cat /etc/selinux/targeted/modules/active/booleans.local  
# This file is auto-generated by libsemanage  
# Do not edit directly.  
  
httpd_read_user_content=1  
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:~  
File Edit View Search Terminal Help  
[root@armitage ~]# touch marker  
[root@armitage ~]# setsebool -P httpd_read_user_content=1; setsebool -P httpd_en  
able_homedirs=1  
[root@armitage ~]# find /etc/selinux/ -newer marker █
```

```
root@armitage:~  
File Edit View Search Terminal Help  
/etc/selinux/targeted  
/etc/selinux/targeted/modules  
/etc/selinux/targeted/modules/active  
/etc/selinux/targeted/modules/active/file_contexts.homedirs  
/etc/selinux/targeted/modules/active/file_contexts  
/etc/selinux/targeted/modules/active/base.pp  
/etc/selinux/targeted/modules/active/modules  
/etc/selinux/targeted/modules/active/modules/firewallgui.pp  
/etc/selinux/targeted/modules/active/modules/ulogd.pp  
/etc/selinux/targeted/modules/active/modules/howl.pp  
/etc/selinux/targeted/modules/active/modules/shutdown.pp  
/etc/selinux/targeted/modules/active/modules/smartmon.pp  
/etc/selinux/targeted/modules/active/modules/ncftool.pp  
/etc/selinux/targeted/modules/active/modules/webalizer.pp  
/etc/selinux/targeted/modules/active/modules/canna.pp  
/etc/selinux/targeted/modules/active/modules/qmail.pp  
/etc/selinux/targeted/modules/active/modules/portreserve.pp  
/etc/selinux/targeted/modules/active/modules/w3c.pp  
/etc/selinux/targeted/modules/active/modules/comsat.pp  
/etc/selinux/targeted/modules/active/modules/xguest.pp  
/etc/selinux/targeted/modules/active/modules/dictd.pp  
/etc/selinux/targeted/modules/active/modules/jabber.pp  
/etc/selinux/targeted/modules/active/modules/nagios.pp  
:
```

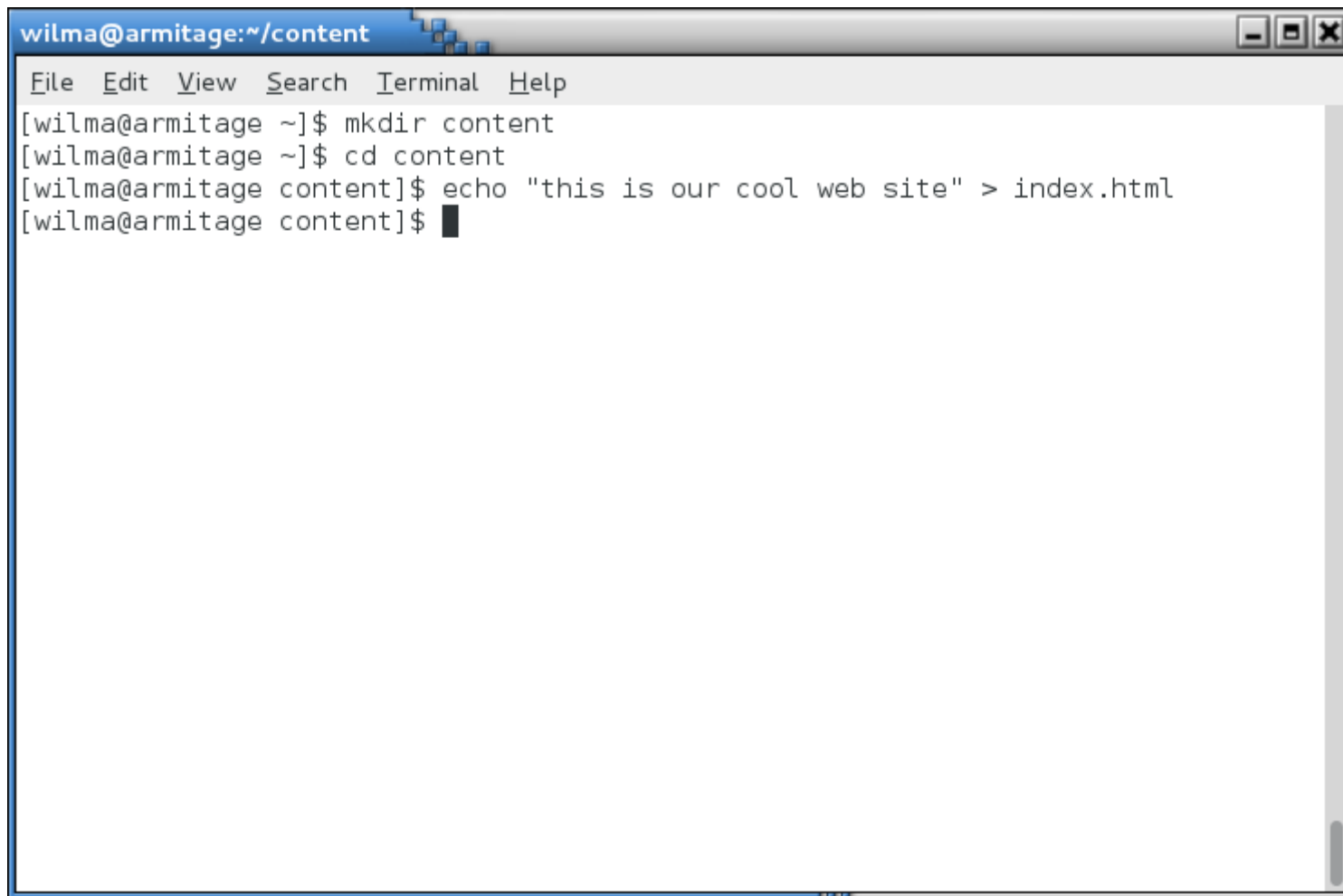
```
root@armitage:~  
File Edit View Search Terminal Help  
/etc/selinux/targeted/modules/active/modules/quantum.pp  
/etc/selinux/targeted/modules/active/modules/ntp.pp  
/etc/selinux/targeted/modules/active/modules/afs.pp  
/etc/selinux/targeted/modules/active/modules/fail2ban.pp  
/etc/selinux/targeted/modules/active/modules/amanda.pp  
/etc/selinux/targeted/modules/active/modules/fetchmail.pp  
/etc/selinux/targeted/modules/active/policy.kern  
/etc/selinux/targeted/modules/active/commit_num  
/etc/selinux/targeted/modules/active/users_extra  
/etc/selinux/targeted/modules/active/seusers  
/etc/selinux/targeted/modules/active/seusers.final  
/etc/selinux/targeted/modules/active/booleans.local  
/etc/selinux/targeted/modules/active/netfilter_contexts  
/etc/selinux/targeted/modules/active/homedir_template  
/etc/selinux/targeted/modules/active/file_contexts.template  
/etc/selinux/targeted/seusers  
/etc/selinux/targeted/contexts  
/etc/selinux/targeted/contexts/files  
/etc/selinux/targeted/contexts/files/file_contexts.homedirs  
/etc/selinux/targeted/contexts/files/file_contexts  
/etc/selinux/targeted/contexts/netfilter_contexts  
/etc/selinux/targeted/policy  
/etc/selinux/targeted/policy/policy.24  
(END)
```

Real World Examples

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

Real World Examples

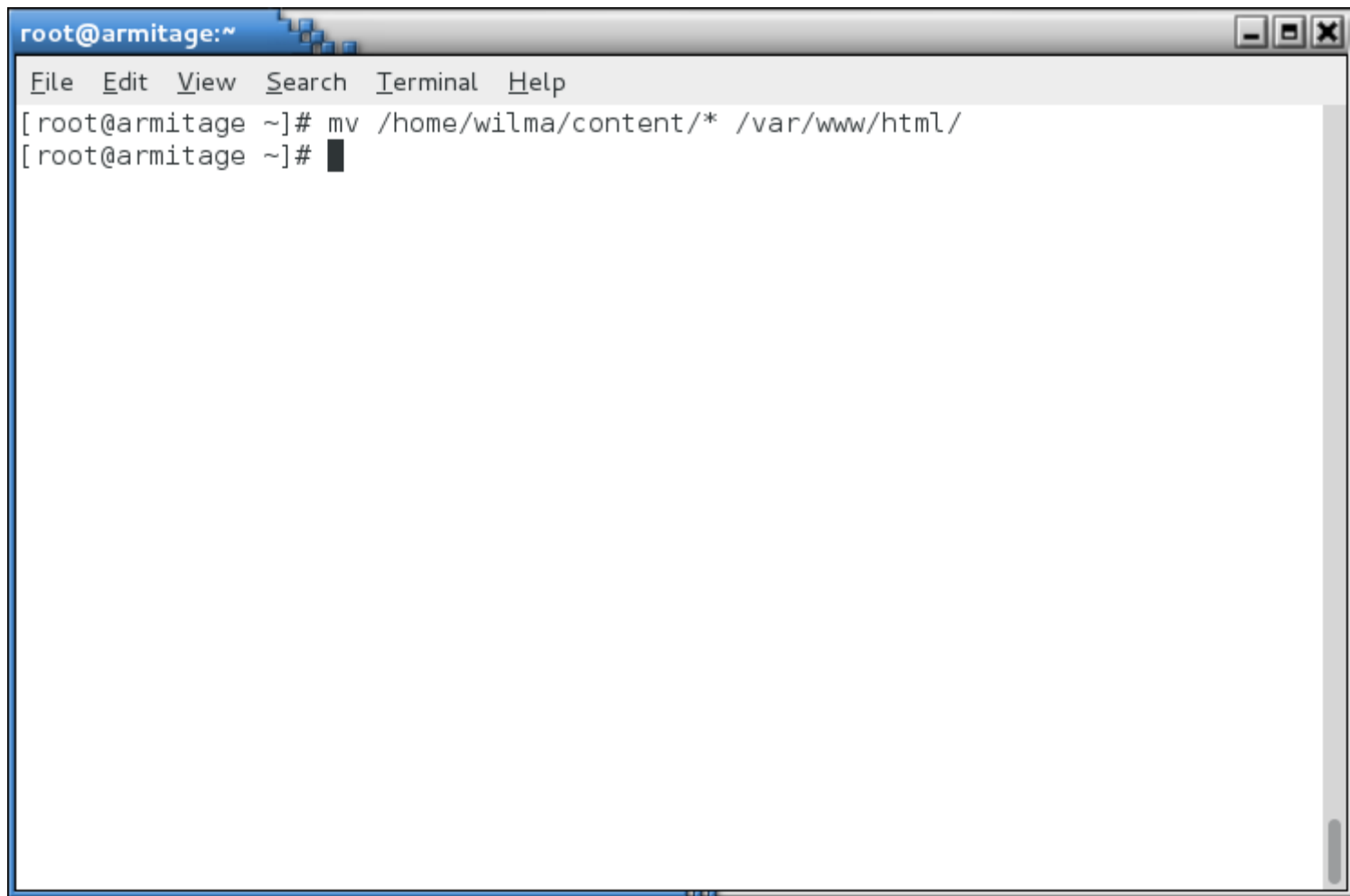
- A user, Wilma, is a web content author. She has created content in her home directory and asked that you move it to the web site.

A terminal window titled 'wilma@armitage:~/content' with standard window controls. The terminal shows a sequence of commands: 'mkdir content', 'cd content', and 'echo "this is our cool web site" > index.html'. The prompt returns to the user after each command. A menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help' is visible at the top of the terminal area.

```
wilma@armitage:~/content
File Edit View Search Terminal Help
[wilma@armitage ~]$ mkdir content
[wilma@armitage ~]$ cd content
[wilma@armitage content]$ echo "this is our cool web site" > index.html
[wilma@armitage content]$
```

Real World Examples

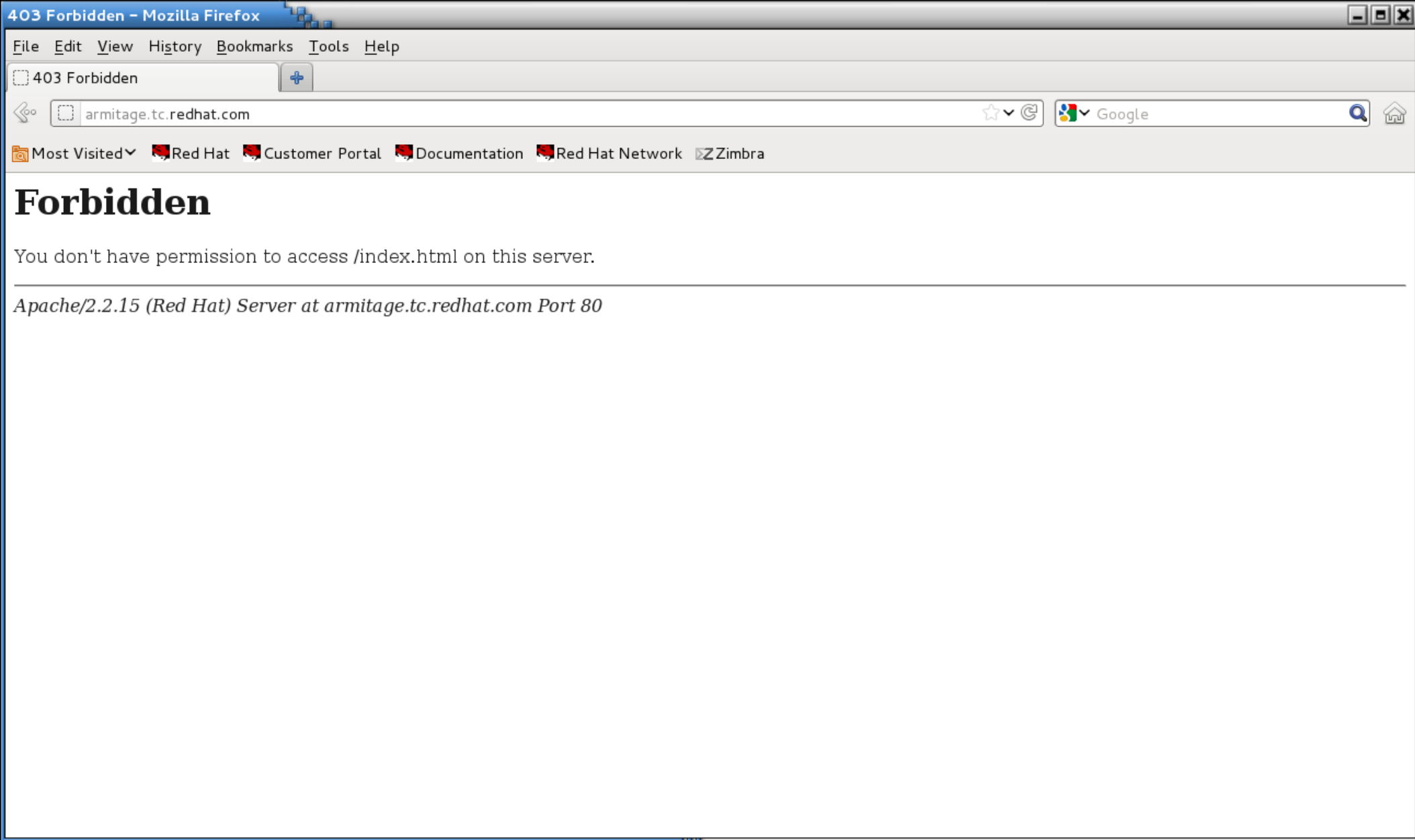
- So, you move it over.

A terminal window titled 'root@armitage:~' with standard window controls. It features a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal content shows a root user at the 'armitage' host in the '~' directory. The command 'mv /home/wilma/content/* /var/www/html/' has been entered and executed. A second prompt is visible with a black cursor.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# mv /home/wilma/content/* /var/www/html/  
[root@armitage ~]# █
```

Real World Examples

- And when you go to test...



Real World Examples

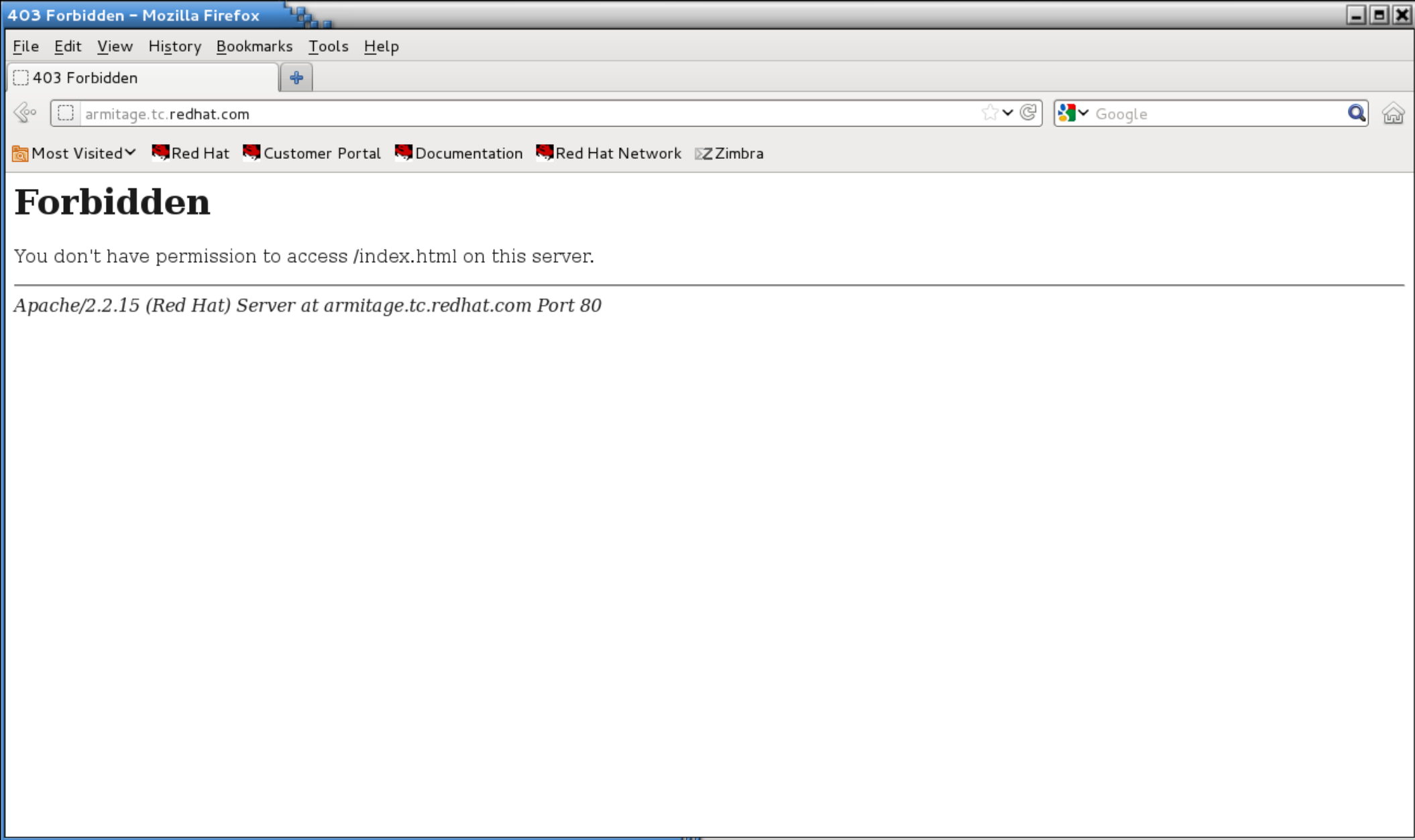
- Ah, it's the wrong owner, right?

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -l /var/www/html/  
total 4  
-rw-rw-r--. 1 wilma wilma 26 Jun 21 10:41 index.html  
[root@armitage ~]#
```

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# chown root:root /var/www/html/index.html  
[root@armitage ~]# ls -l /var/www/html/  
total 4  
-rw-rw-r--. 1 root root 26 Jun 21 10:41 index.html  
[root@armitage ~]#
```

Real World Examples

- But when you test...



Real World Examples

- Checking `/var/log/messages` again tells you to run `sealert`.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# tail /var/log/messages  
Jun 21 10:29:04 armitage setsebool: The httpd_enable_homedirs policy boolean was  
changed to 1 by root  
Jun 21 10:39:58 armitage dbus: avc: received policyload notice (seqno=4)  
Jun 21 10:39:58 armitage dbus: [system] Reloaded configuration  
Jun 21 10:39:58 armitage setsebool: The httpd_read_user_content policy boolean w  
as changed to 0 by root  
Jun 21 10:40:24 armitage dbus: avc: received policyload notice (seqno=5)  
Jun 21 10:40:24 armitage dbus: [system] Reloaded configuration  
Jun 21 10:40:25 armitage setsebool: The httpd_enable_homedirs policy boolean was  
changed to 0 by root  
Jun 21 10:43:11 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd f  
rom read access on the file index.html. For complete SELinux messages. run seale  
rt -l 0feb4ad8-bfa5-4d27-ab6d-9f061ef1f162  
Jun 21 10:45:57 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd f  
rom read access on the file index.html. For complete SELinux messages. run seale  
rt -l 0feb4ad8-bfa5-4d27-ab6d-9f061ef1f162  
Jun 21 10:45:57 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd f  
rom read access on the file index.html. For complete SELinux messages. run seale  
rt -l 0feb4ad8-bfa5-4d27-ab6d-9f061ef1f162  
[root@armitage ~]#
```

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.

```
root@armitage:~  
File Edit View Search Terminal Help  
rt -l 0feb4ad8-bfa5-4d27-ab6d-9f061ef1f162  
[root@armitage ~]# sealert -l 0feb4ad8-bfa5-4d27-ab6d-9f061ef1f162  
SELinux is preventing /usr/sbin/httpd from read access on the file index.html.  
  
***** Plugin catchall_boolean (47.5 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 'user_selinux' man page for more details.  
Do  
setsebool -P httpd_read_user_content 1  
  
***** Plugin catchall_boolean (47.5 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' bo  
olean.You can read 'user_selinux' man page for more details.  
Do  
setsebool -P httpd_enable_homedirs 1  
  
***** Plugin catchall (6.38 confidence) suggests *****  
  
If you believe that httpd should be allowed read access on the index.html file b  
y default.
```

Real World Examples

- A quick `ls -Z` reveals the issue.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -Z /var/www/html/  
-rw-rw-r--. root root unconfined_u:object_r:user_home_t:s0 index.html  
[root@armitage ~]#
```

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@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -lZ /var/www/  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_script_exec_t:s0 cgi-bin  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 error  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 html  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 icons  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 manual  
drwxr-xr-x. webalizer root system_u:object_r:httpd_sys_content_t:s0 usage  
[root@armitage ~]#
```

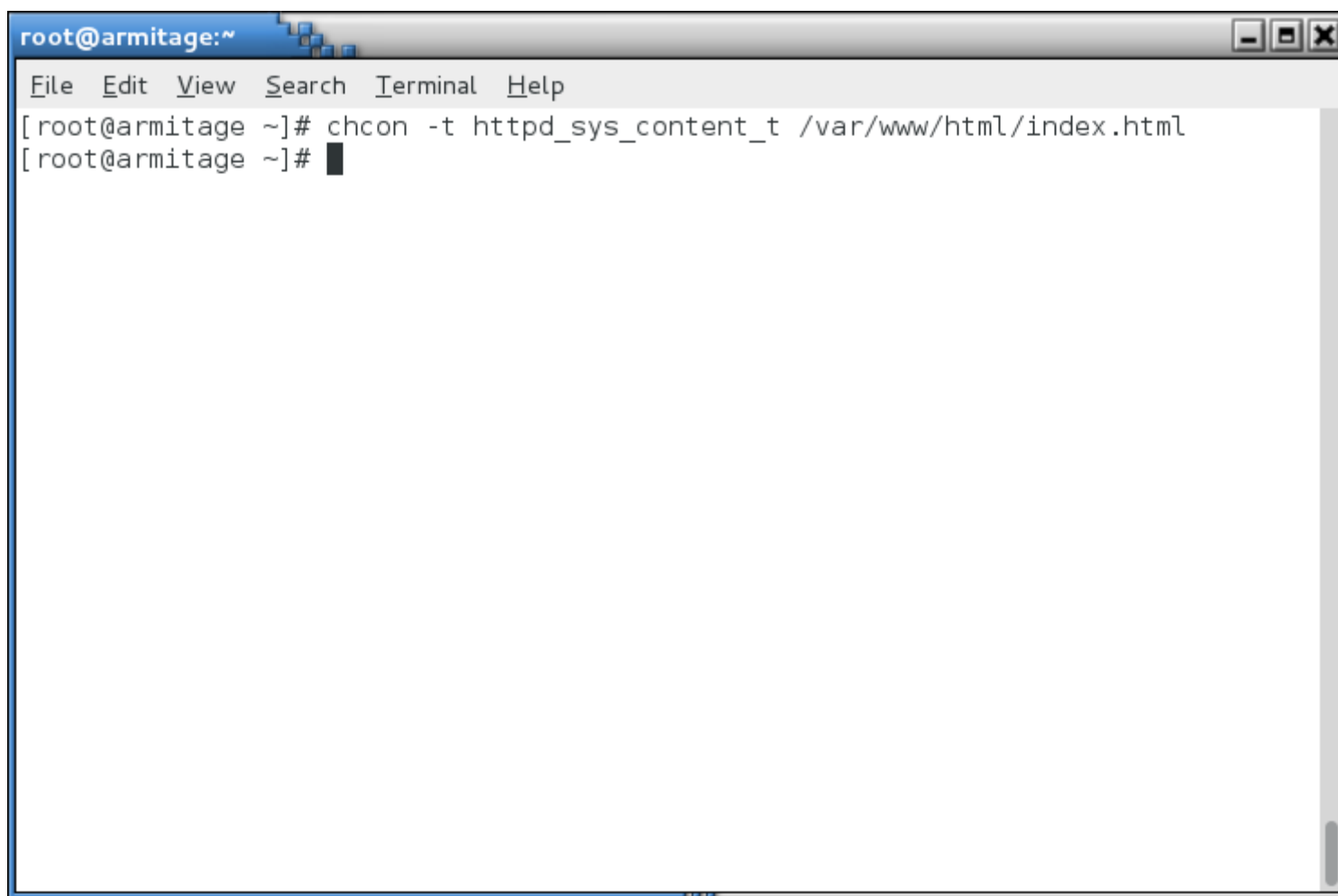
Real World Examples

- Use that information as arguments for the chcon (change context) command
- The long form is:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# chcon -u system_u -r object_r -t httpd_sys_content_t /var/www  
/html/index.html
```

Real World Examples

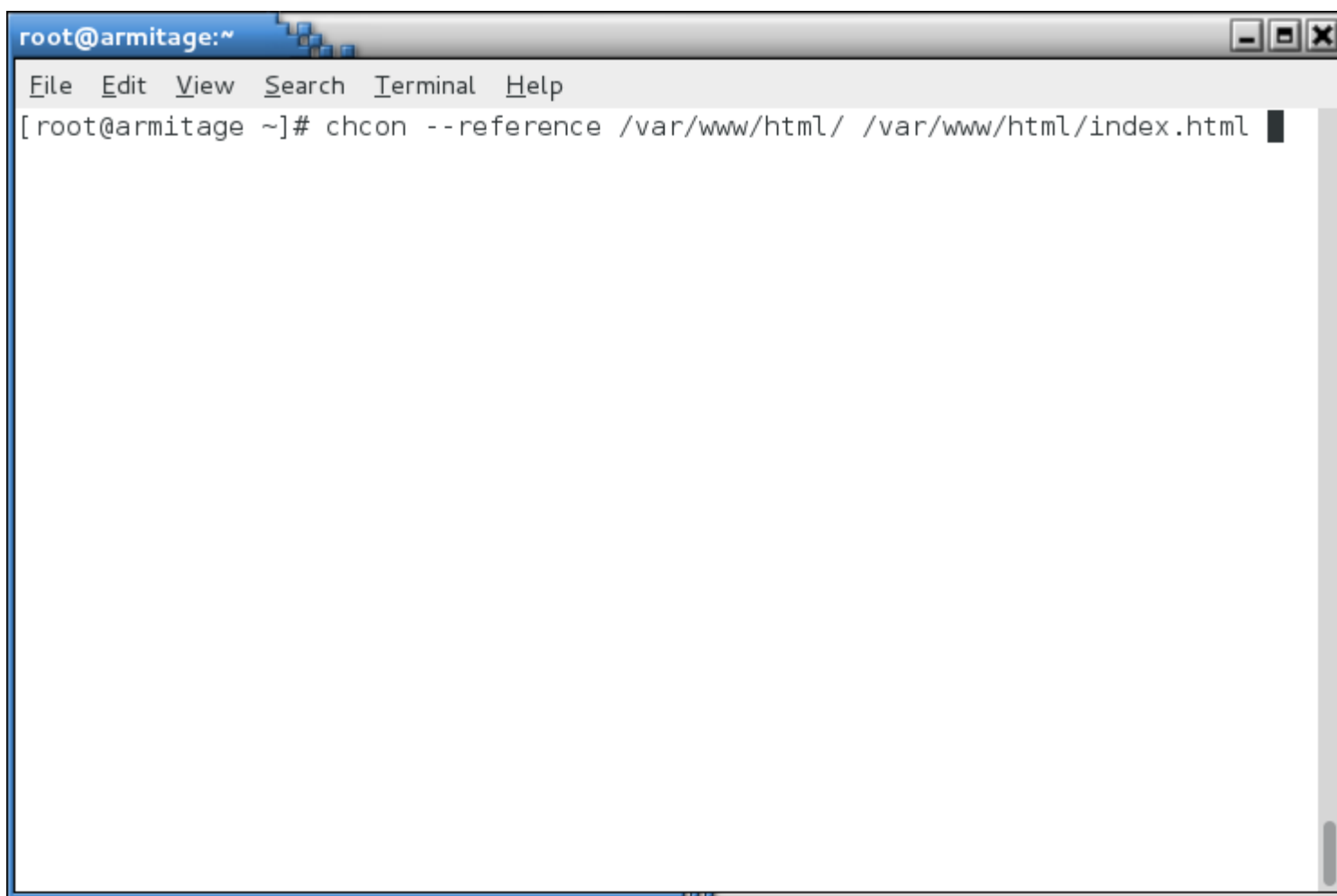
- Remember that the targeted policy doesn't use the SELinux user or role. The short form is:

A terminal window titled 'root@armitage:~' with standard window controls. It features a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The command history shows '[root@armitage ~]# chcon -t httpd_sys_content_t /var/www/html/index.html' followed by a new prompt '[root@armitage ~]#'.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# chcon -t httpd_sys_content_t /var/www/html/index.html  
[root@armitage ~]#
```

Real World Examples

- I'm lazy. If I just want to reference a known good context, the shortest form is:

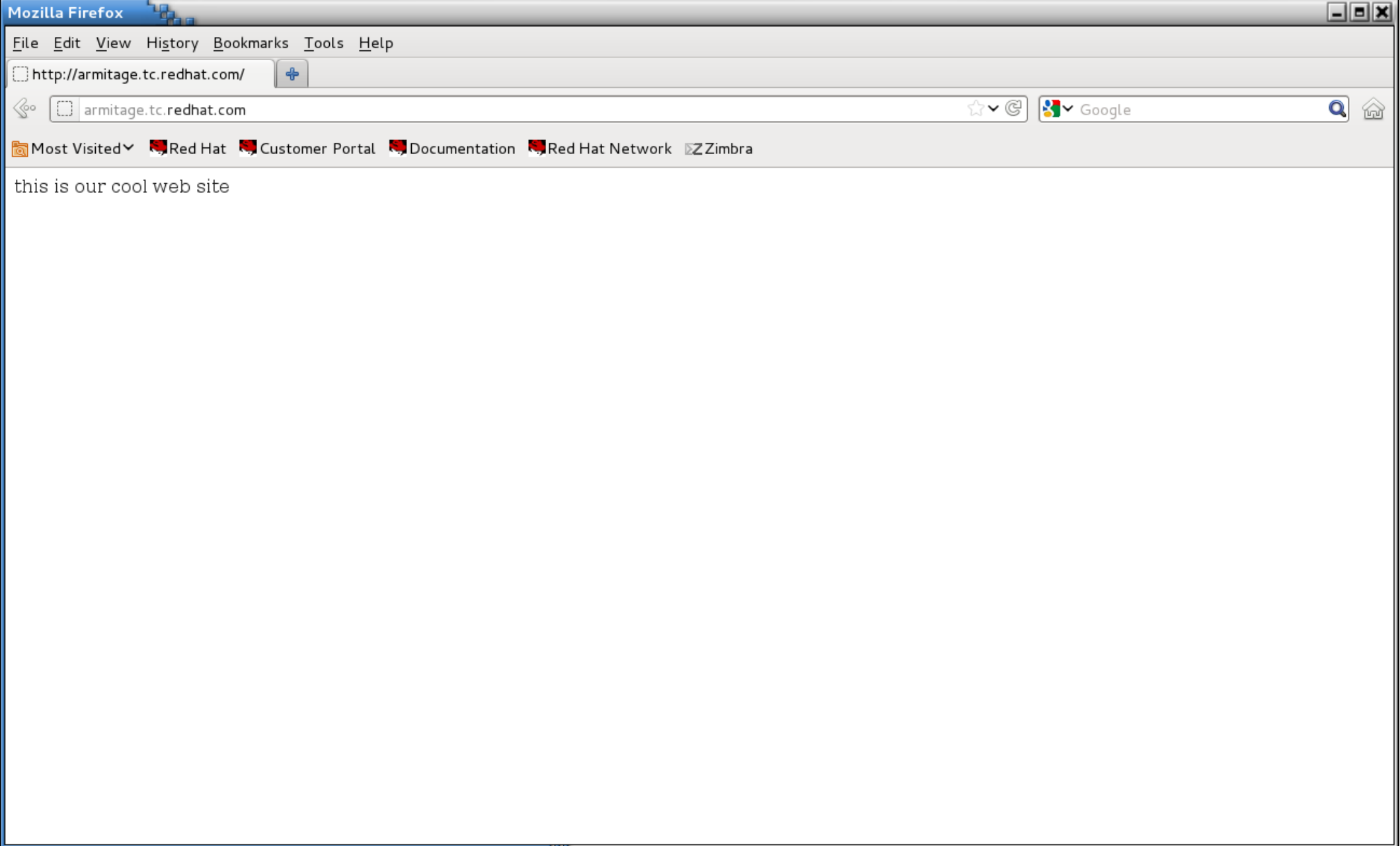
A terminal window titled 'root@armitage:~' with a menu bar (File, Edit, View, Search, Terminal, Help) and standard window controls. The command '[root@armitage ~]# chcon --reference /var/www/html/ /var/www/html/index.html' is entered at the prompt, followed by a black cursor.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# chcon --reference /var/www/html/ /var/www/html/index.html
```

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`:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# restorecon -vR /var/www/html/  
restorecon reset /var/www/html/index.html context unconfined_u:object_r:user_home_t:s0->unconfined_u:object_r:httpd_sys_content_t:s0  
[root@armitage ~]#
```



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!

```
root@armitage:~  
File Edit View Search Terminal Help  
/*      system_u:object_r:default_t:s0  
/[^/]+  --      system_u:object_r:etc_runtime_t:s0  
/a?quota\.(user|group)  --      system_u:object_r:quota_db_t:s0  
/nsr(/.*)?      system_u:object_r:var_t:s0  
/sys(/.*)?      system_u:object_r:sysfs_t:s0  
/xen(/.*)?      system_u:object_r:xen_image_t:s0  
/mnt(/[^/]* )   -l      system_u:object_r:mnt_t:s0  
/mnt(/[^/]* )?  -d      system_u:object_r:mnt_t:s0  
/bin/* system_u:object_r:bin_t:s0  
/dev/* system_u:object_r:device_t:s0  
/lib/* system_u:object_r:lib_t:s0  
/usr/* system_u:object_r:usr_t:s0  
/var/* system_u:object_r:var_t:s0  
/etc/* system_u:object_r:etc_t:s0  
/opt/* system_u:object_r:usr_t:s0  
/srv/* system_u:object_r:var_t:s0  
/tmp/* <<none>>  
/root(/.*)?      system_u:object_r:admin_home_t:s0  
/dev/[0-9]*      -c      system_u:object_r:usb_device_t:s0  
/mnt/[^/]*/*      <<none>>  
/dev/*mouse.*    -c      system_u:object_r:mouse_device_t:s0  
/rhev(/[^/]* )?  -d      system_u:object_r:mnt_t:s0  
/dev/*tty[^\/*] -c      system_u:object_r:tty_device_t:s0  
/etc/selinux/targeted/contexts/files/file_contexts
```

```
root@armitage:~  
File Edit View Search Terminal Help  
/var/www(/.*)? system_u:object_r:httpd_sys_content_t:s0  
/opt/cvs(/.*)? system_u:object_r:cvs_data_t:s0  
/var/cvs(/.*)? system_u:object_r:cvs_data_t:s0  
/etc/dcc(/.*)? system_u:object_r:dcc_var_t:s0  
/var/dcc(/.*)? system_u:object_r:dcc_var_t:s0  
/srv/git(/.*)? system_u:object_r:git_system_content_t:s0  
/etc/gpm(/.*)? system_u:object_r:gpm_conf_t:s0  
/etc/ups(/.*)? system_u:object_r:nut_conf_t:s0  
/etc/nas(/.*)? system_u:object_r:soundd_etc_t:s0  
/etc/tor(/.*)? system_u:object_r:tor_etc_t:s0  
/dev/xvc[0-9]* -c system_u:object_r:tty_device_t:s0  
/dev/dm-[0-9]+ -b system_u:object_r:fixed_disk_device_t:s0  
/dev/tpm[0-9]* -c system_u:object_r:tpm_device_t:s0  
/dev/uio[0-9]+ -c system_u:object_r:userio_device_t:s0  
/etc/ppp(/.*)? -- system_u:object_r:pppd_etc_rw_t:s0  
/usr/lib(64)?/amanda -d system_u:object_r:amanda_usr_lib_t:s0  
/usr/lib(64)?/dpkg/.+ -- system_u:object_r:bin_t:s0  
/usr/lib(64)?/sa/sa.* -- system_u:object_r:sysstat_exec_t:s0  
/usr/lib(64)?/sendmail -- system_u:object_r:sendmail_exec_t:s0  
/usr/lib(64)?/rpm/rpmd -- system_u:object_r:bin_t:s0  
/usr/lib(64)?/rpm/rpmk -- system_u:object_r:bin_t:s0  
/usr/lib(64)?/rpm/rpmv -- system_u:object_r:bin_t:s0  
/usr/lib(64)?/rpm/rpmq -- system_u:object_r:bin_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@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# mkdir -p /foo/bar  
[root@armitage ~]# ls /foo/  
bar  
[root@armitage ~]#
```

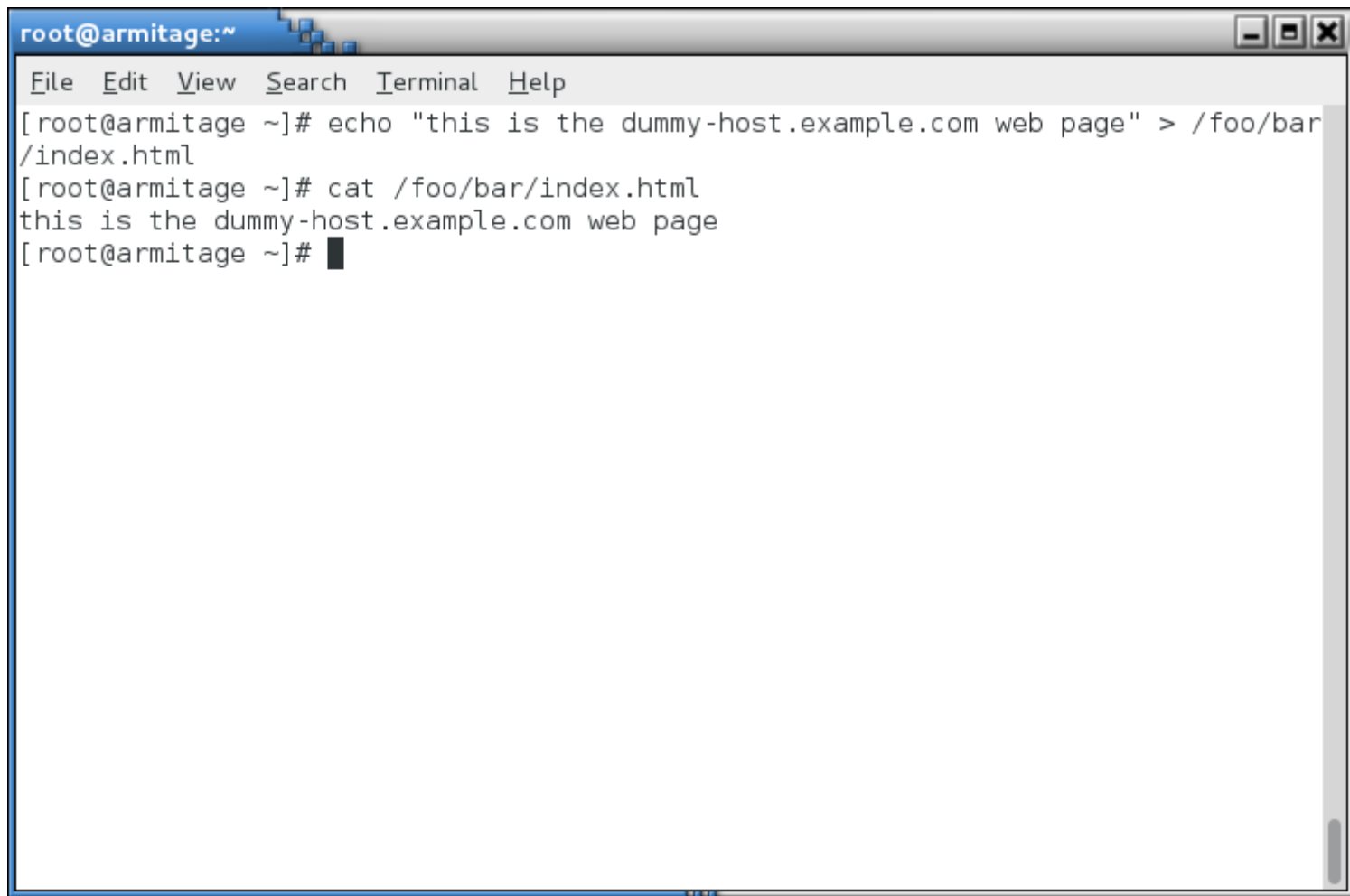
Real World Examples

- You define the virtual web site in httpd.conf:

```
root@armitage:~  
File Edit View Search Terminal Help  
#  
#  
# VirtualHost example:  
# Almost any Apache directive may go into a VirtualHost container.  
# The first VirtualHost section is used for requests without a known  
# server name.  
#  
#<VirtualHost *:80>  
#     ServerAdmin webmaster@dummy-host.example.com  
#     DocumentRoot /www/docs/dummy-host.example.com  
#     ServerName dummy-host.example.com  
#     ErrorLog logs/dummy-host.example.com-error_log  
#     CustomLog logs/dummy-host.example.com-access_log common  
#</VirtualHost>  
#  
<VirtualHost *:80>  
    ServerAdmin webmaster@dummy-host.example.com  
    DocumentRoot /foo/bar  
    ServerName dummy-host.example.com  
    ErrorLog logs/dummy-host.example.com-error_log  
    CustomLog logs/dummy-host.example.com-access_log common  
</VirtualHost>  
"/etc/httpd/conf/httpd.conf" 1017L, 34678C written
```

Real World Examples

- You create an index.html file:

A terminal window titled 'root@armitage:~' with standard window controls (minimize, maximize, close) in the top right. The terminal has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The command history shows: 1. '[root@armitage ~]# echo "this is the dummy-host.example.com web page" > /foo/bar/index.html' 2. '[root@armitage ~]# cat /foo/bar/index.html' 3. The output 'this is the dummy-host.example.com web page' 4. The prompt '[root@armitage ~]#' followed by a cursor. The terminal has a vertical scrollbar on the right side.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# echo "this is the dummy-host.example.com web page" > /foo/bar/index.html  
[root@armitage ~]# cat /foo/bar/index.html  
this is the dummy-host.example.com web page  
[root@armitage ~]#
```

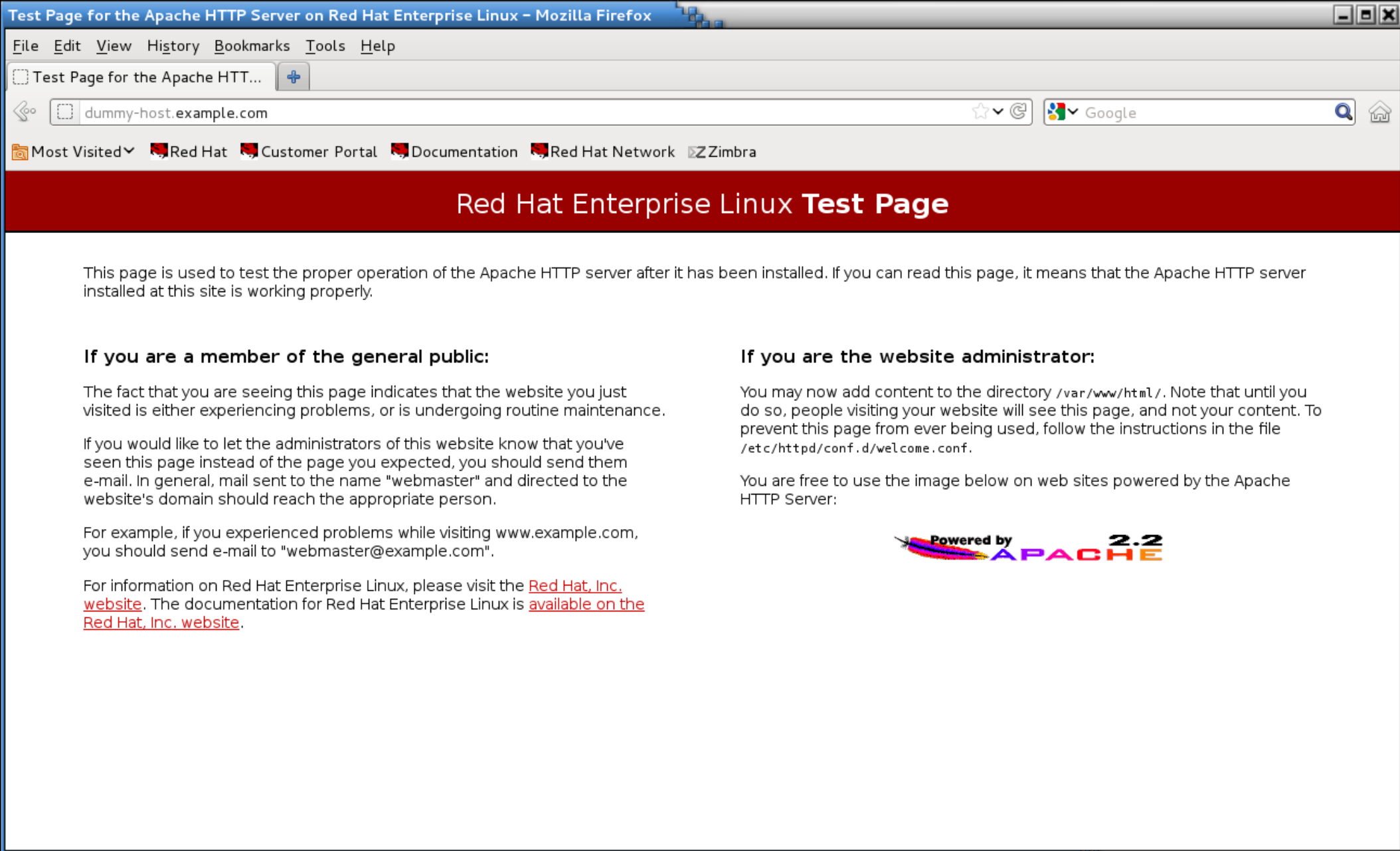
Real World Examples

- Restart the web server:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# service httpd restart  
Stopping httpd: [ OK ]  
Starting httpd: [ OK ]  
[root@armitage ~]# █
```

Real World Examples

- When you test the page...



Real World Examples

- What logfile should we check?

Real World Examples

- `/var/log/messages`

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# tail /var/log/messages  
Jun 21 12:20:21 armitage setsebool: The httpd_read_user_content policy boolean was changed to 1 by root  
Jun 21 12:20:47 armitage dbus: avc: received policyload notice (seqno=7)  
Jun 21 12:20:47 armitage dbus: [system] Reloaded configuration  
Jun 21 12:20:48 armitage setsebool: The httpd_enable_homedirs policy boolean was changed to 1 by root  
Jun 21 13:17:33 armitage setroubleshoot: Deleting alert 9f88e0bb-5f4b-4e3a-96b2-7644917fbfc4, it is allowed in current policy  
Jun 21 13:17:33 armitage setroubleshoot: Deleting alert 0feb4ad8-bfa5-4d27-ab6d-9f061ef1f162, it is allowed in current policy  
Jun 21 13:17:36 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from getattr access on the file /foo/bar/index.html. For complete SELinux message s. run sealert -l 26c7f536-5706-46d9-a149-77096e80ed2b  
Jun 21 13:17:38 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from getattr access on the file /foo/bar/index.html. For complete SELinux message s. run sealert -l 26c7f536-5706-46d9-a149-77096e80ed2b  
Jun 21 13:17:39 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from getattr access on the file /foo/bar/index.html. For complete SELinux message s. run sealert -l 26c7f536-5706-46d9-a149-77096e80ed2b  
Jun 21 13:17:40 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd from getattr access on the file /foo/bar/index.html. For complete SELinux message s. run sealert -l 26c7f536-5706-46d9-a149-77096e80ed2b  
[root@armitage ~]#
```

```
root@armitage:~  
File Edit View Search Terminal Help  
SELinux is preventing /usr/sbin/httpd from getattr access on the file /foo/bar/index.html.  
  
***** Plugin catchall_labels (83.8 confidence) suggests *****  
  
If you want to allow httpd to have getattr access on the index.html file  
Then you need to change the label on /foo/bar/index.html  
Do  
# semanage fcontext -a -t FILE_TYPE '/foo/bar/index.html'  
where FILE_TYPE is one of the following: dirsrv_config_t, httpd_mediawiki_htacce  
ss_t, fail2ban_var_lib_t, abrt_var_run_t, krb5_conf_t, udev_tbl_t, httpd_tmp_t,  
smokeping_var_lib_t, shell_exec_t, httpd_w3c_validator_htaccess_t, mysqld_etc_t,  
cvs_data_t, calamaris_www_t, dirsrvadmin_tmp_t, cobbler_etc_t, sysctl_crypto_t,  
httpd_cache_t, httpd_tmpfs_t, httpd_helper_exec_t, iso9660_t, dbusd_etc_t, dirs  
rv_share_t, var_lib_t, user_cron_spool_t, configfile, httpd_squirrelmail_t, cfe  
ngine_var_log_t, httpd_php_exec_t, httpd_nagios_htaccess_t, abrt_t, httpd_mediaw  
iki_tmp_t, lib_t, samba_var_t, dirsrv_var_log_t, zarafa_var_lib_t, abrt_helper_ex  
ec_t, net_conf_t, ld_so_t, cert_type, etc_runtime_t, git_system_content_t, dirsr  
v_var_run_t, puppet_var_lib_t, public_content_t, httpd_var_lib_t, httpd_var_run  
_t, logfile, anon_inodefs_t, sysctl_kernel_t, httpd_modules_t, user_tmp_t, httpd  
awstats_htaccess_t, httpd_dirsrvadmin_htaccess_t, textrel_shlib_t, httpd_user_ht  
access_t, chroot_exec_t, httpd_sys_content_t, public_content_rw_t, httpd_suexec  
_exec_t, application_exec_type, httpd_bugzilla_htaccess_t, httpd_cobbler_htaccess  
:
```

Real World Examples

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

```
root@armitage:~  
File Edit View Search Terminal Help  
stats_script_exec_t, httpd_dirsrvadmin_ra_content_t, httpd_dirsrvadmin_rw_content_t, krb5_host_rcache_t, httpd_apcupsd_cgi_script_exec_t, httpd_dirsrvadmin_content_t, httpd_cobbler_content_t, httpd_squid_script_exec_t, httpd_w3c_validator_ra_content_t, httpd_w3c_validator_rw_content_t, httpd_nagios_script_exec_t, nfs_t, httpd_awstats_ra_content_t, httpd_awstats_rw_content_t, httpd_awstats_content_t, httpd_user_ra_content_t, httpd_user_rw_content_t, httpd_bugzilla_script_exec_t, httpdcontent, httpd_cobbler_ra_content_t, httpd_cobbler_rw_content_t.  
Then execute:  
restorecon -v '/foo/bar/index.html'  
  
***** Plugin catchall (17.1 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:  
# grep httpd /var/log/audit/audit.log | audit2allow -M mypol  
# semodule -i mypol.pp  
  
(END)
```

Real World Examples

- What directory should we look at to get the correct context label?

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -Z /var/www/  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_script_exec_t:s0 cgi-bin  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 error  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 html  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 icons  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 manual  
drwxr-xr-x. webalizer root system_u:object_r:httpd_sys_content_t:s0 usage  
[root@armitage ~]#
```

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@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# ls -Z /var/www/  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_script_exec_t:s0 cgi-bin  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 error  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 html  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 icons  
drwxr-xr-x. root      root system_u:object_r:httpd_sys_content_t:s0 manual  
drwxr-xr-x. webalizer root system_u:object_r:httpd_sys_content_t:s0 usage  
[root@armitage ~]# semanage fcontext -a -t httpd_sys_content_t "/foo(/.*)?"  
[root@armitage ~]#
```

Real World Examples

- Or, if you're like me (lazy), you can use the -e (equals) argument to semanage fcontext:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# semanage fcontext -a -e /var/www/ /foo/  
[root@armitage ~]# cat /etc/selinux/targeted/contexts/files/file_contexts.subs  
/foo/ /var/www/  
[root@armitage ~]#
```

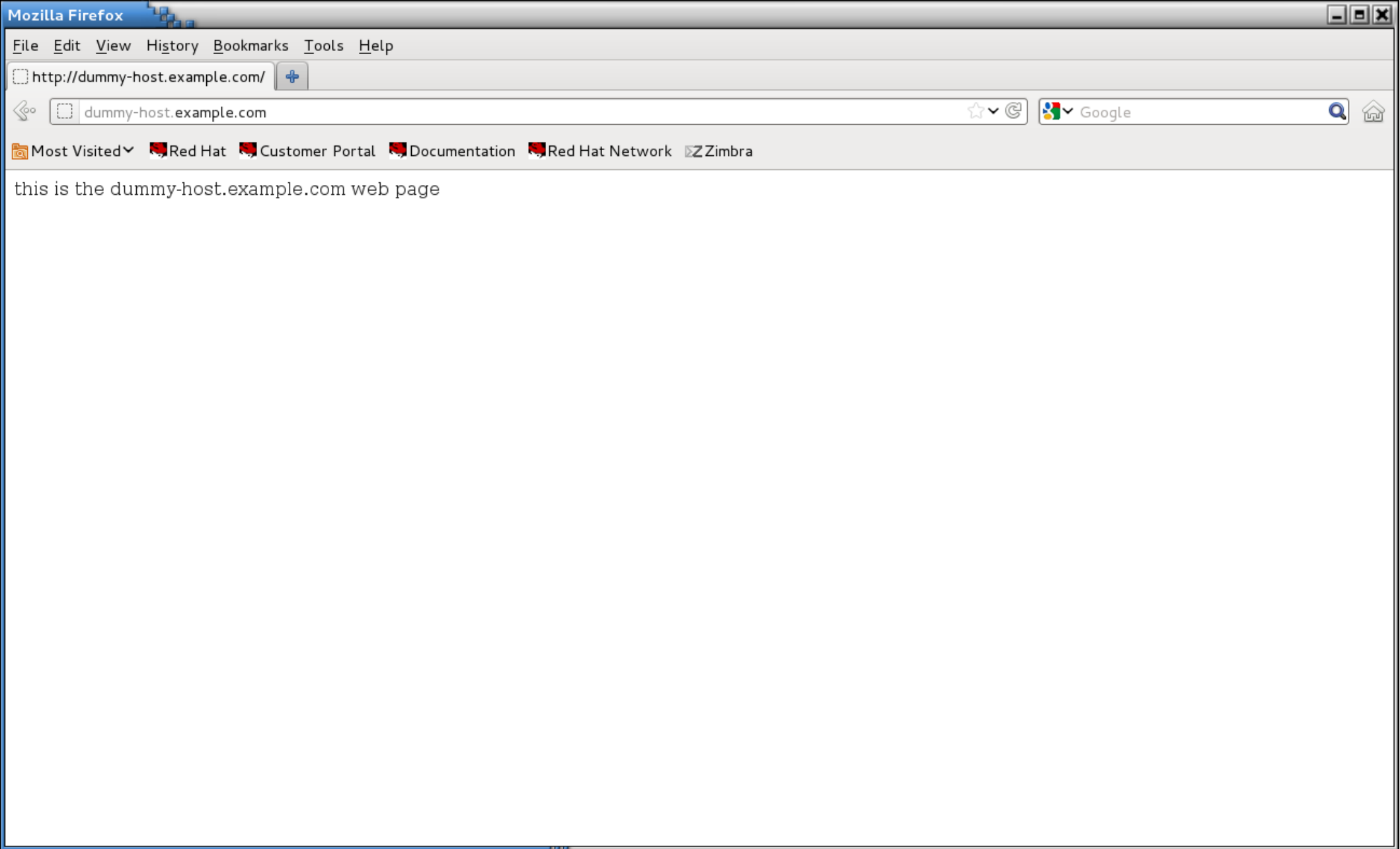
Real World Examples

- Now run restorecon against the directory:

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# 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@armitage ~]#
```

Real World Examples

- Test the site:



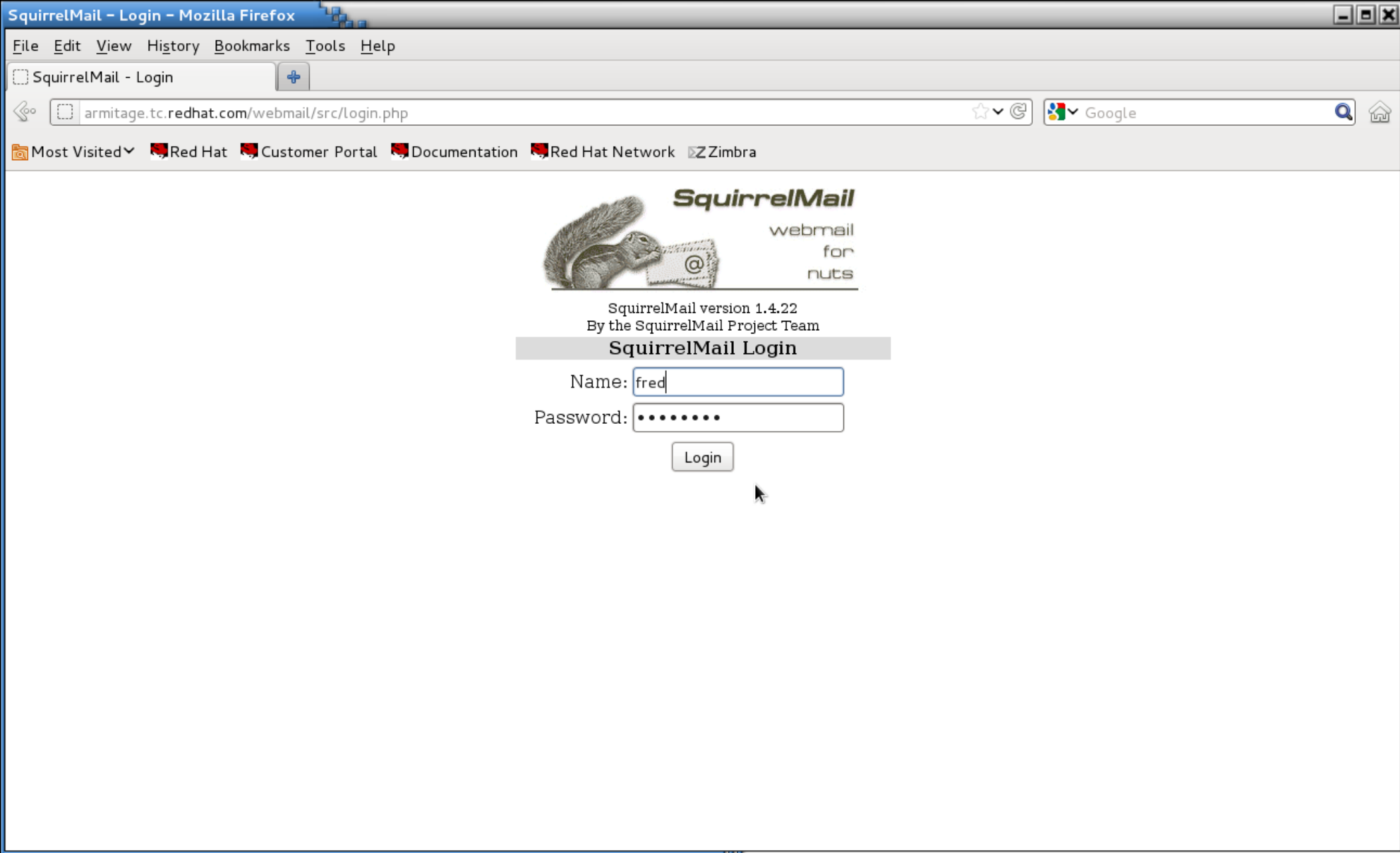
Creating Policy Modules

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 6.3 mail server.



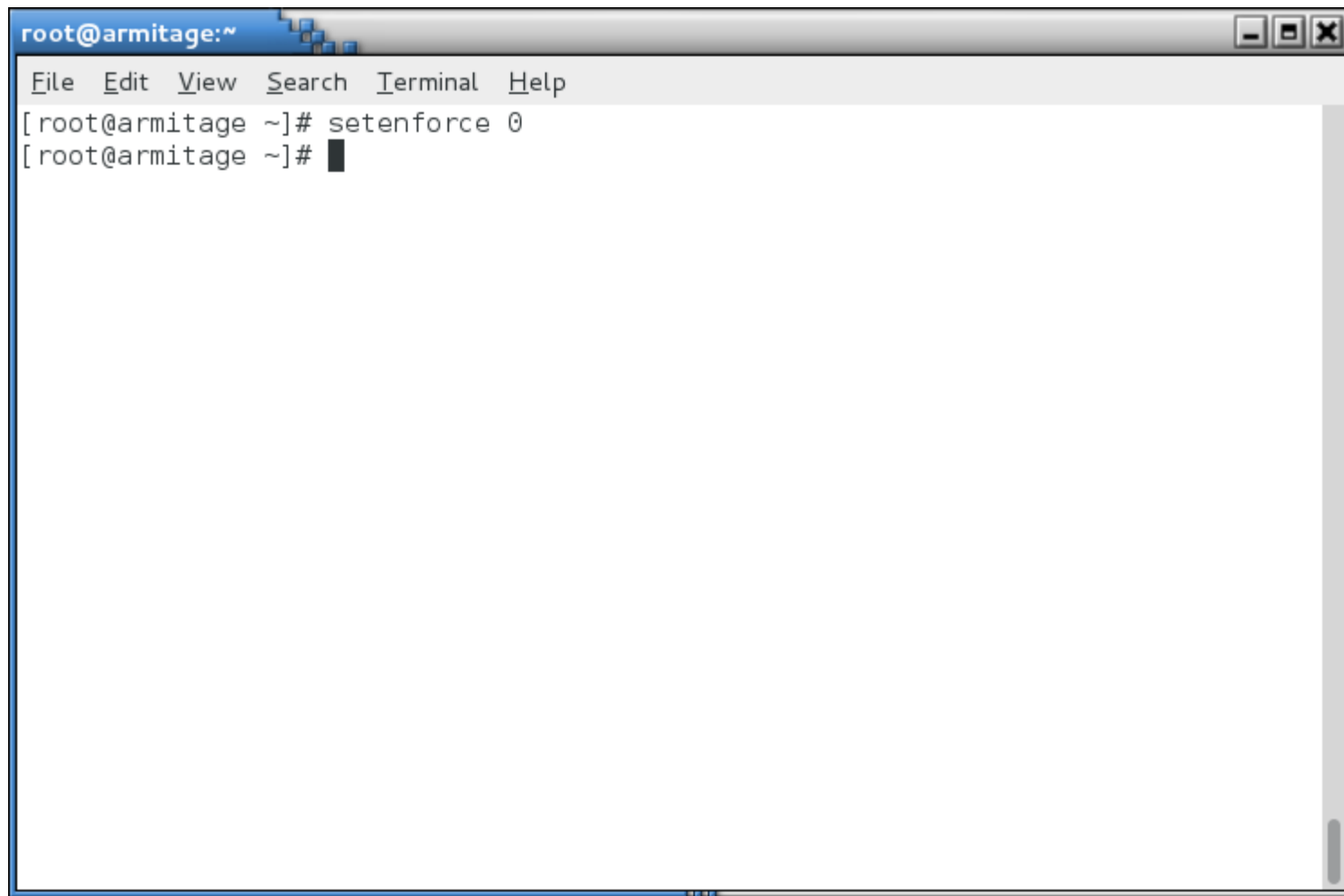
**SquirrelMail**webmail
for
nutsSquirrelMail version 1.4.22
By the SquirrelMail Project Team**ERROR**Error connecting to IMAP server: localhost.
13 : Permission denied[Go to the login page](#)

```
root@armitage:~  
File Edit View Search Terminal Help  
  
type=AVC msg=audit(1340321054.097:32692): avc: denied { name_connect } for pi  
d=3593 comm="httpd" dest=143 scontext=unconfined_u:system_r:httpd_t:s0 tcontext=  
system_u:object_r:pop_port_t:s0 tclass=tcp_socket  
type=SYSCALL msg=audit(1340321054.097:32692): arch=c000003e syscall=42 success=n  
o 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_r:httpd_t:s0 key=(  
null)  
type=AVC msg=audit(1340321054.098:32693): avc: denied { name_connect } for pi  
d=3593 comm="httpd" dest=143 scontext=unconfined_u:system_r:httpd_t:s0 tcontext=  
system_u:object_r:pop_port_t:s0 tclass=tcp_socket  
type=SYSCALL msg=audit(1340321054.098:32693): arch=c000003e syscall=42 success=n  
o exit=-13 a0=13 a1=7f0939a06250 a2=10 a3=7f093691814c items=0 ppid=3590 pid=359  
3 auid=0 uid=48 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48 tty=(no  
ne) ses=1 comm="httpd" exe="/usr/sbin/httpd" subj=unconfined_u:system_r:httpd_t:  
s0 key=(null)  
█
```

```
root@armitage:~  
File Edit View Search Terminal Help  
usr/share/setroubleshoot/plugins/catchall_boolean.py", line 76, in check_for_man  
#012 man_page = name.split("_")[0] + "_selinux"#012AttributeError: 'tuple' ob  
ject has no attribute 'split'  
Jun 21 18:23:31 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd f  
rom name_connect access on the tcp_socket . For complete SELinux messages. run s  
ealert -l f64ca3e4-4fe2-4998-85eb-de402ba79db2  
Jun 21 18:23:31 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd f  
rom name_connect access on the tcp_socket . For complete SELinux messages. run s  
ealert -l f64ca3e4-4fe2-4998-85eb-de402ba79db2  
Jun 21 18:24:15 armitage setroubleshoot: [avc.ERROR] Plugin Exception catchall_b  
oolean #012Traceback (most recent call last):#012 File "/usr/lib64/python2.6/si  
te-packages/setroubleshoot/analyze.py", line 191, in analyze_avc#012 report =  
plugin.analyze(avc)#012 File "/usr/share/setroubleshoot/plugins/catchall_boole  
an.py", line 90, in analyze#012 man_page = self.check_for_man(b)#012 File "/  
usr/share/setroubleshoot/plugins/catchall_boolean.py", line 76, in check_for_man  
#012 man_page = name.split("_")[0] + "_selinux"#012AttributeError: 'tuple' ob  
ject has no attribute 'split'  
Jun 21 18:24:15 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd f  
rom name_connect access on the tcp_socket . For complete SELinux messages. run s  
ealert -l f64ca3e4-4fe2-4998-85eb-de402ba79db2  
Jun 21 18:24:15 armitage setroubleshoot: SELinux is preventing /usr/sbin/httpd f  
rom name_connect access on the tcp_socket . For complete SELinux messages. run s  
ealert -l 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.

A terminal window titled 'root@armitage:~' with standard window controls. The menu bar includes 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal content shows the command 'setenforce 0' being executed, with the prompt changing from '[root@armitage ~]#' to '[root@armitage ~]#'.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# setenforce 0  
[root@armitage ~]#
```

SquirrelMail 1.4.22 - Mozilla Firefox

File Edit View History Bookmarks Tools Help

SquirrelMail 1.4.22

armitage.tc.redhat.com/webmail/src/webmail.php

Most Visited Red Hat Customer Portal Documentation Red Hat Network Zimbra

Folders
Last Refresh:
Thu, 6:25 pm
(Check mail)

INBOX
Drafts
Sent
Trash

Current Folder: INBOX

[Compose](#) [Addresses](#) [Folders](#) [Options](#) [Search](#) [Help](#) [Sign Out](#) [SquirrelMail](#)

To: barney@armitage.tc.redhat.com

Cc:

Bcc:

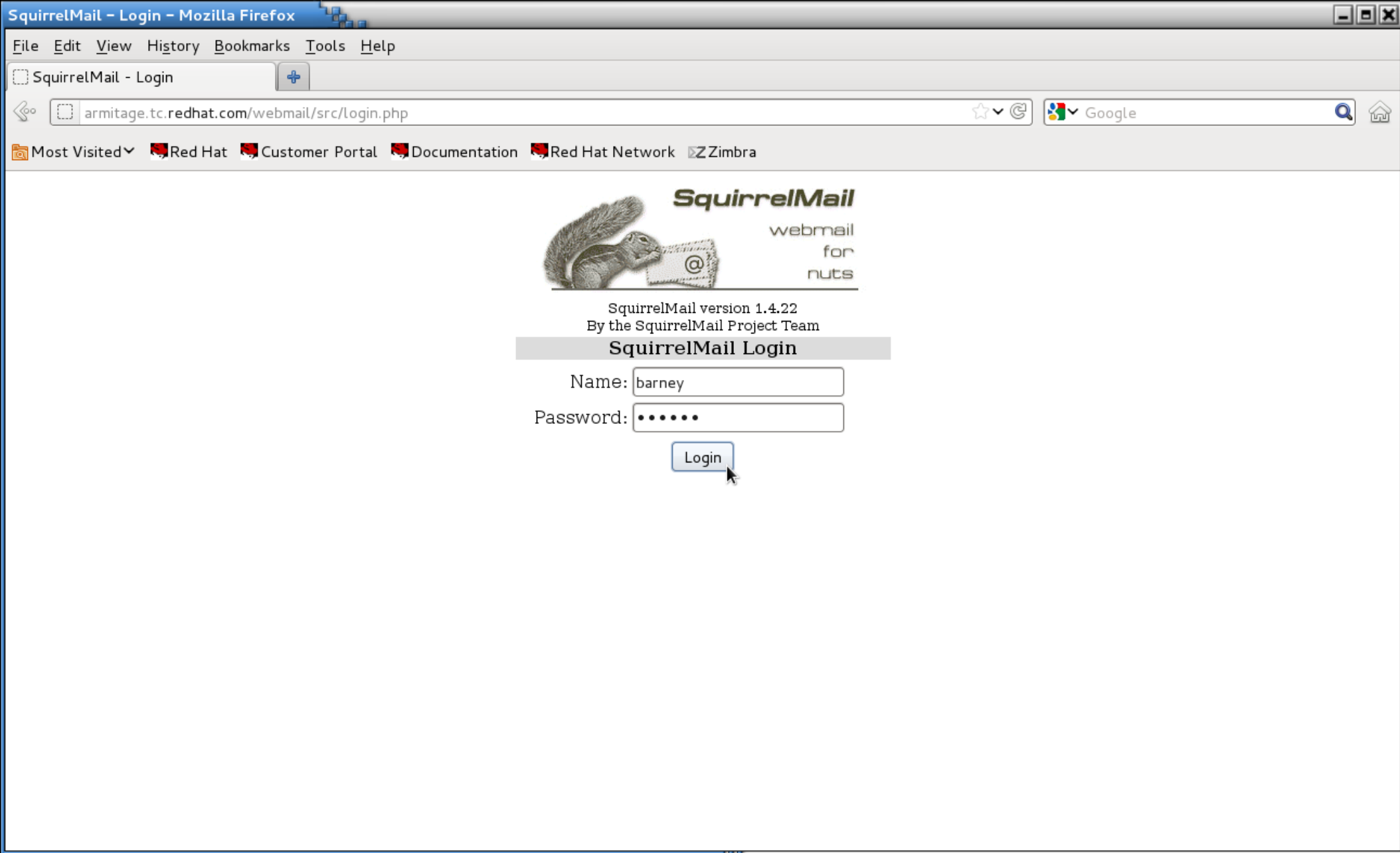
Subject: test from SquirrelMail

Priority: Normal Receipt: ☐ On Read ☐ On Delivery

Signature Addresses Save Draft Send

this is a test of sending e-mail

Send



SquirrelMail 1.4.22 - Mozilla Firefox

File Edit View History Bookmarks Tools Help

SquirrelMail 1.4.22

armitage.tc.redhat.com/webmail/src/webmail.php

Most Visited Red Hat Customer Portal Documentation Red Hat Network Zimbra

Folders
Last Refresh:
Thu, 6:26 pm
(Check mail)

INBOX (1)
Drafts
Sent
Trash

Current Folder: INBOX

[Compose](#) [Addresses](#) [Folders](#) [Options](#) [Search](#) [Help](#)

[Sign Out](#)
[SquirrelMail](#)

[Toggle All](#) Viewing Message: 1 (1 total)

Move Selected To:

INBOX Move Forward

Transform Selected Messages:
Read Unread Delete

From	Date	Subject
☐ fred@armitage.tc.redhat.com	6:07 pm	test from SquirrelMail

[Toggle All](#) Viewing Message: 1 (1 total)

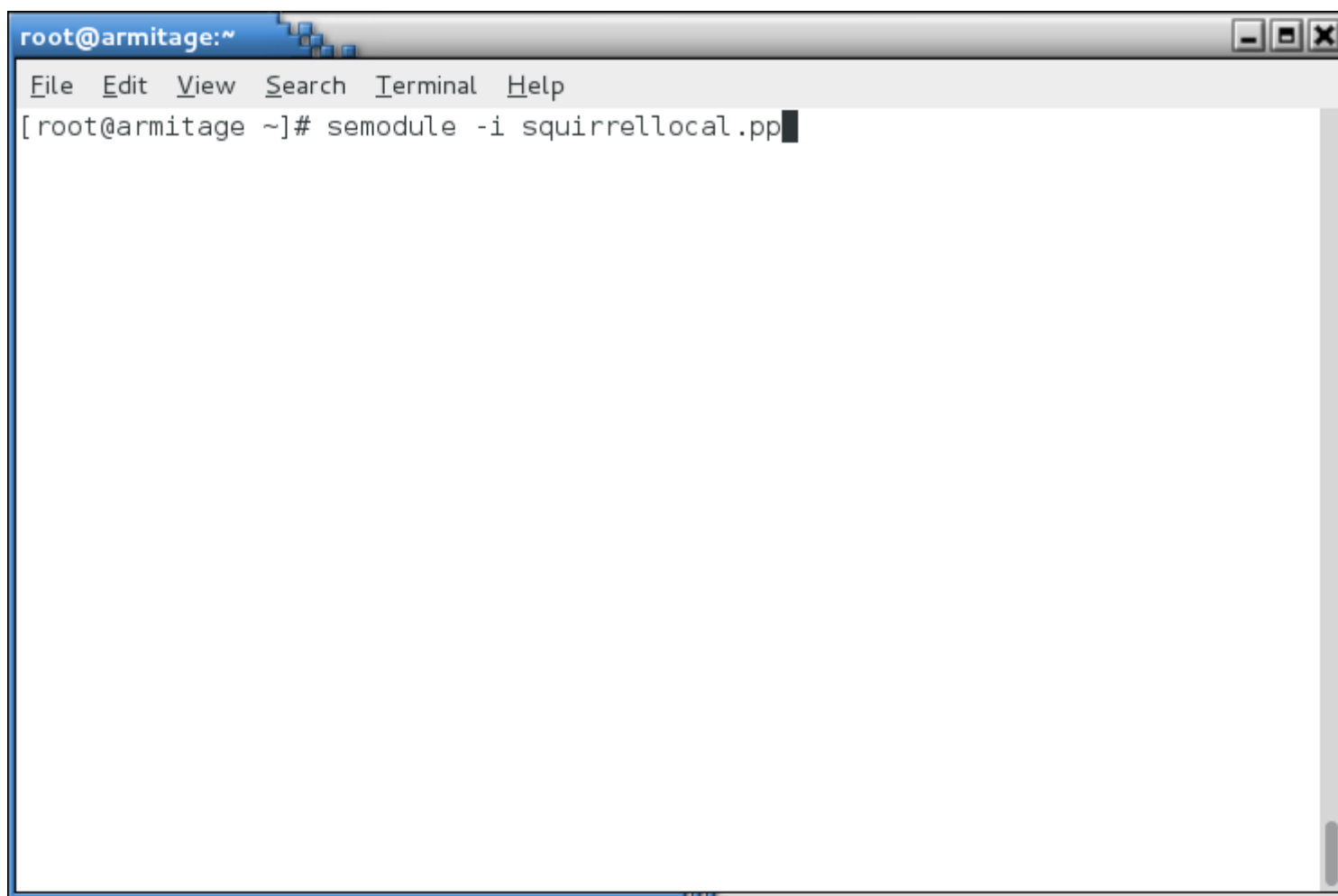
```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# sealert -l f64ca3e4-4fe2-4998-85eb-de402ba79db2  
Gtk-Message: Failed to load module "pk-gtk-module": libpk-gtk-module.so: cannot  
open shared object file: No such file or directory  
SELinux is preventing /usr/sbin/httpd 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:~  
File Edit View Search Terminal Help  
[root@armitage ~]# grep httpd /var/log/audit/audit.log | audit2allow -M squirrel  
local  
***** IMPORTANT *****  
To make this policy package active, execute:  
  
semodule -i squirrellocal.pp  
[root@armitage ~]#
```

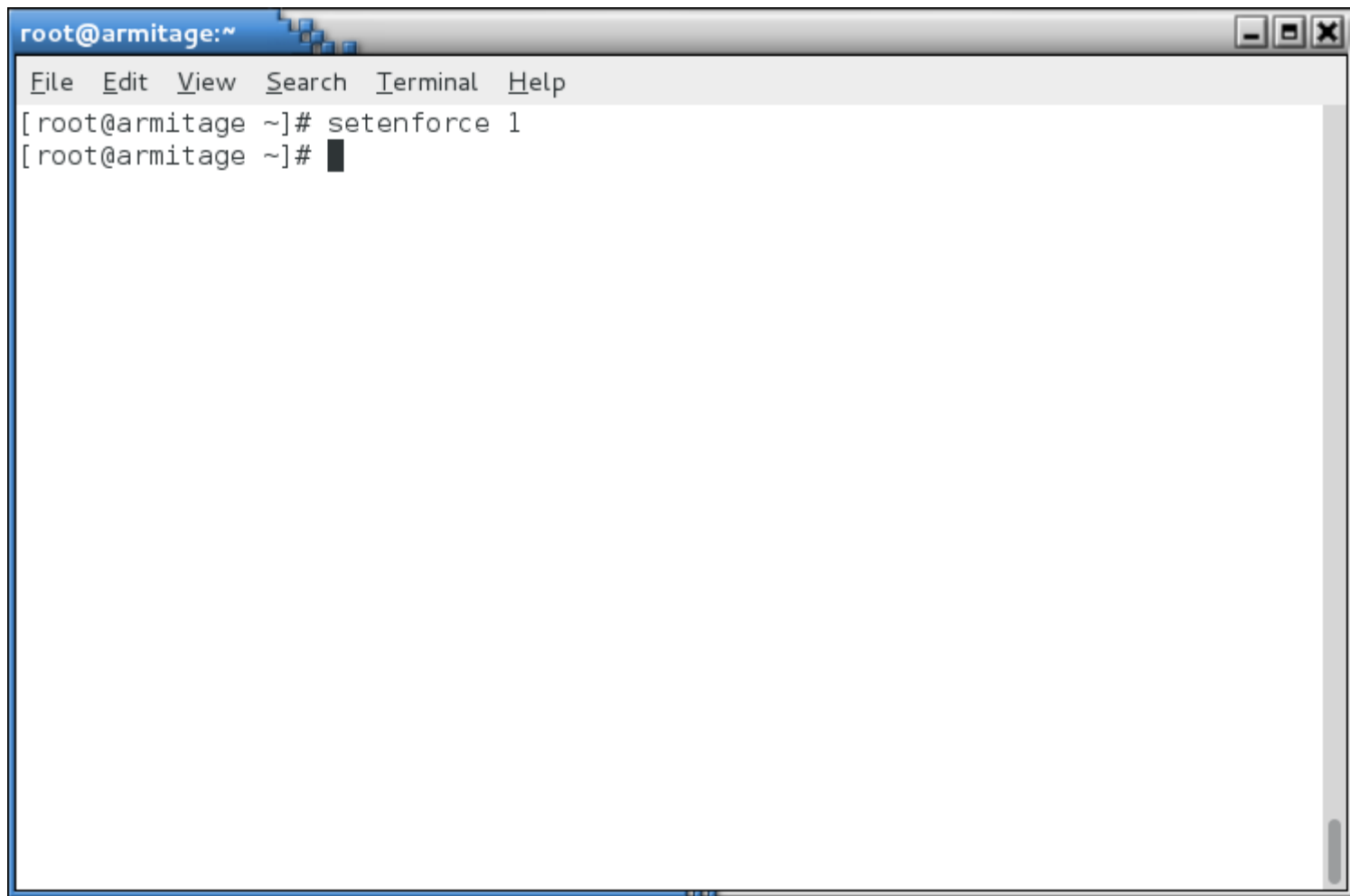
Note

- Actually, this error **could** be fixed by setting a boolean. I am just creating a policy module so you can see it being done.

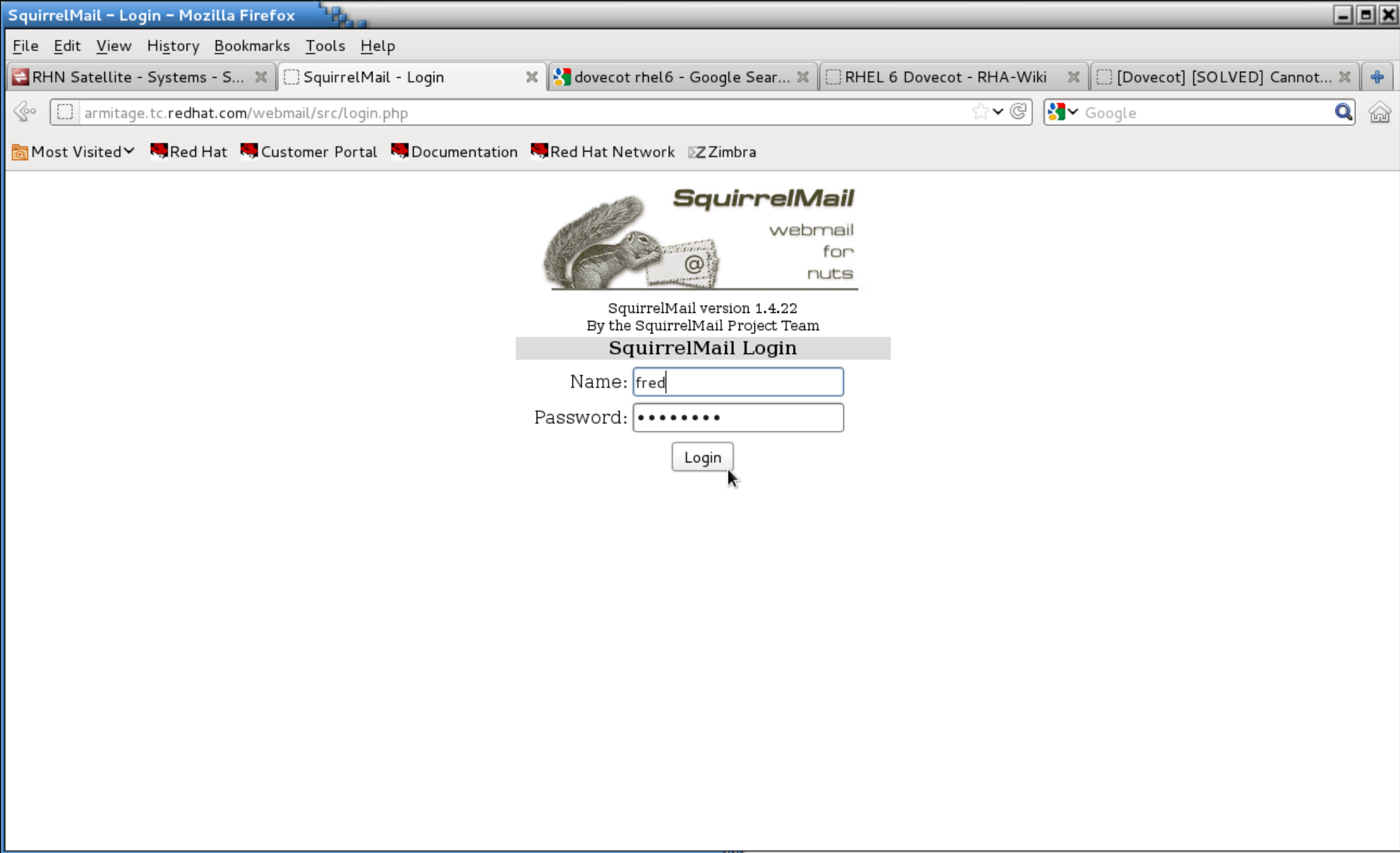
```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# cat squirrellocal.te  
  
module squirrellocal 1.0;  
  
require {  
    type httpd_t;  
    type smtp_port_t;  
    type pop_port_t;  
    class tcp_socket name_connect;  
}  
  
#===== httpd_t =====  
#!!!! This avc can be allowed using one of the these booleans:  
#    httpd_can_sendmail, allow_ybind, 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_ybind, httpd_can_network_connect  
  
allow httpd_t smtp_port_t:tcp_socket name_connect;  
[root@armitage ~]#
```

A terminal window titled 'root@armitage:~' with standard window controls (minimize, maximize, close) in the top right corner. The menu bar includes 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The command prompt shows '[root@armitage ~]# semodule -i squirrellocal.pp' with a black cursor at the end of the line.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# semodule -i squirrellocal.pp
```

A terminal window titled 'root@armitage:~' with standard window controls. It features a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The command history shows '[root@armitage ~]# setenforce 1' followed by a new prompt '[root@armitage ~]#' with a cursor.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# setenforce 1  
[root@armitage ~]#
```



SquirrelMail 1.4.22 - Mozilla Firefox

File Edit View History Bookmarks Tools Help

RHN Satellite - Systems - S... SquirrelMail 1.4.22 dovecot rhel6 - Google Sear... RHEL 6 Dovecot - RHA-Wiki [Dovecot] [SOLVED] Cannot...

armitage.tc.redhat.com/webmail/src/webmail.php

Most Visited Red Hat Customer Portal Documentation Red Hat Network Zimbra

Folders
Last Refresh:
Thu, 6:18 pm
(Check mail)

INBOX
Drafts
Sent
Trash

Current Folder: INBOX [Sign Out](#)
[Compose](#) [Addresses](#) [Folders](#) [Options](#) [Search](#) [Help](#) [SquirrelMail](#)

Move Selected To: INBOX Move Forward Transform Selected Messages: Read Unread Delete

From	Date	Subject
THIS FOLDER IS EMPTY		

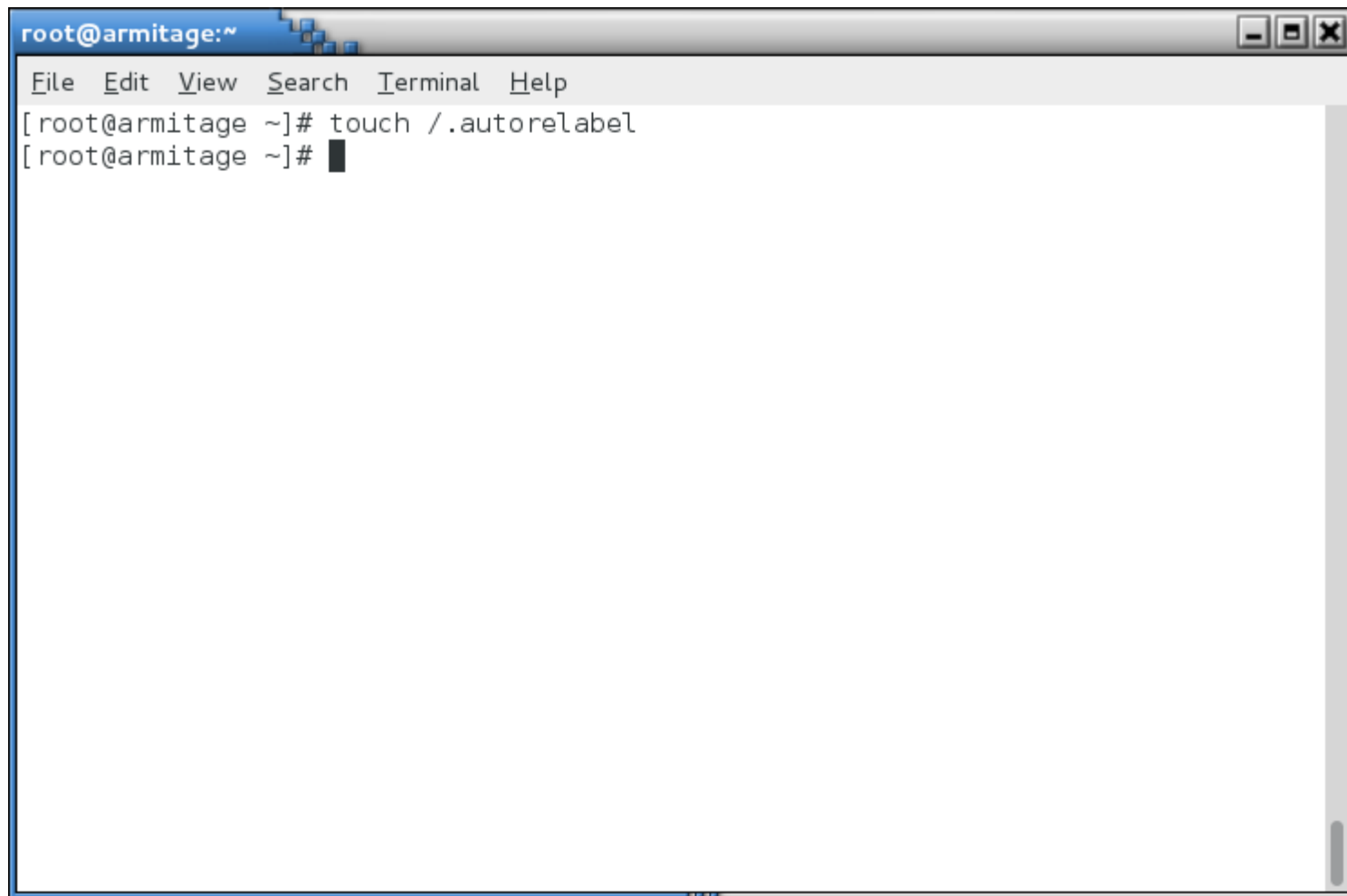
Enabling SELinux

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.

Enabling SELinux

- Then create a file in the root of the filesystem called `.autorelabel`

A terminal window titled 'root@armitage:~' with standard window controls. The menu bar includes 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The command history shows '[root@armitage ~]# touch /.autorelabel' followed by a new prompt '[root@armitage ~]#'.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# touch /.autorelabel  
[root@armitage ~]#
```

Enabling SELinux

- Reboot, and the system will relabel the filesystem.

```

[ OK ]
Checking filesystems
/dev/vda3: clean, 34262/1234576 files, 348839/4935424 blocks
/dev/vda1: clean, 38/51200 files, 34415/204800 blocks
[ OK ]
Remounting root filesystem in read-write mode: [ OK ]
Mounting local filesystems: [ OK ]
Enabling local filesystem quotas: [ OK ]
Welcome to Red Hat Enterprise Linux Server
Starting udev: [ OK ]
Setting hostname localhost: [ OK ]
Setting up Logical Volume Management: No volume groups found
[ OK ]
Checking filesystems
/dev/vda3: clean, 34262/1234576 files, 348839/4935424 blocks
/dev/vda1: clean, 38/51200 files, 34415/204800 blocks
[ OK ]
Remounting root filesystem in read-write mode: [ OK ]
Mounting local filesystems: [ OK ]
Enabling local filesystem quotas: [ OK ]

*** Warning -- SELinux targeted policy relabel is required.
*** Relabeling could take a very long time, depending on file
*** system size and speed of hard drives.
*****_

```

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.

```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# fixfiles relabel  
  
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 [N]? y  
Cleaning out /tmp  
[root@armitage ~]# init 6
```

Enabling SELinux

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

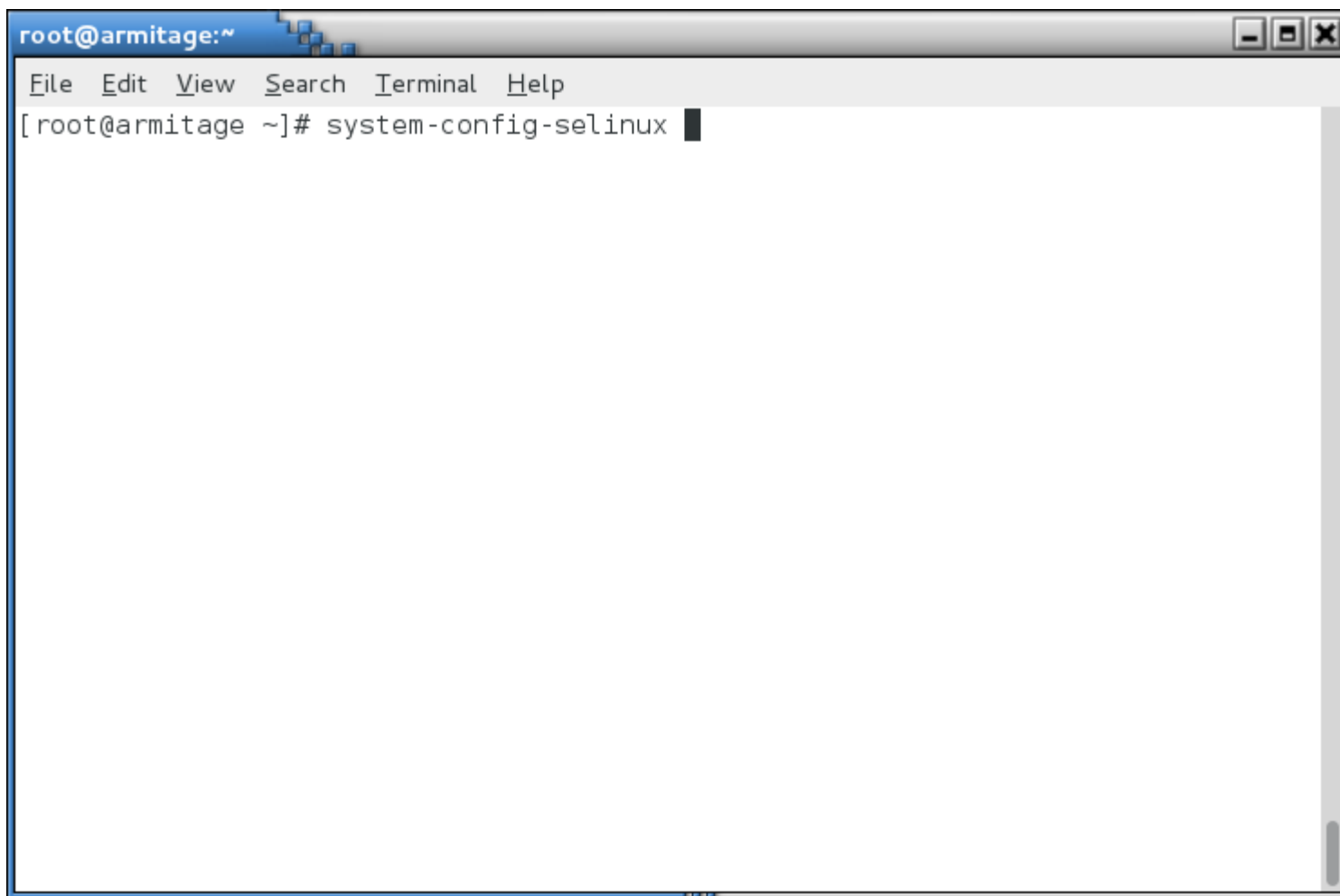
Graphical Tools

Graphical Tools

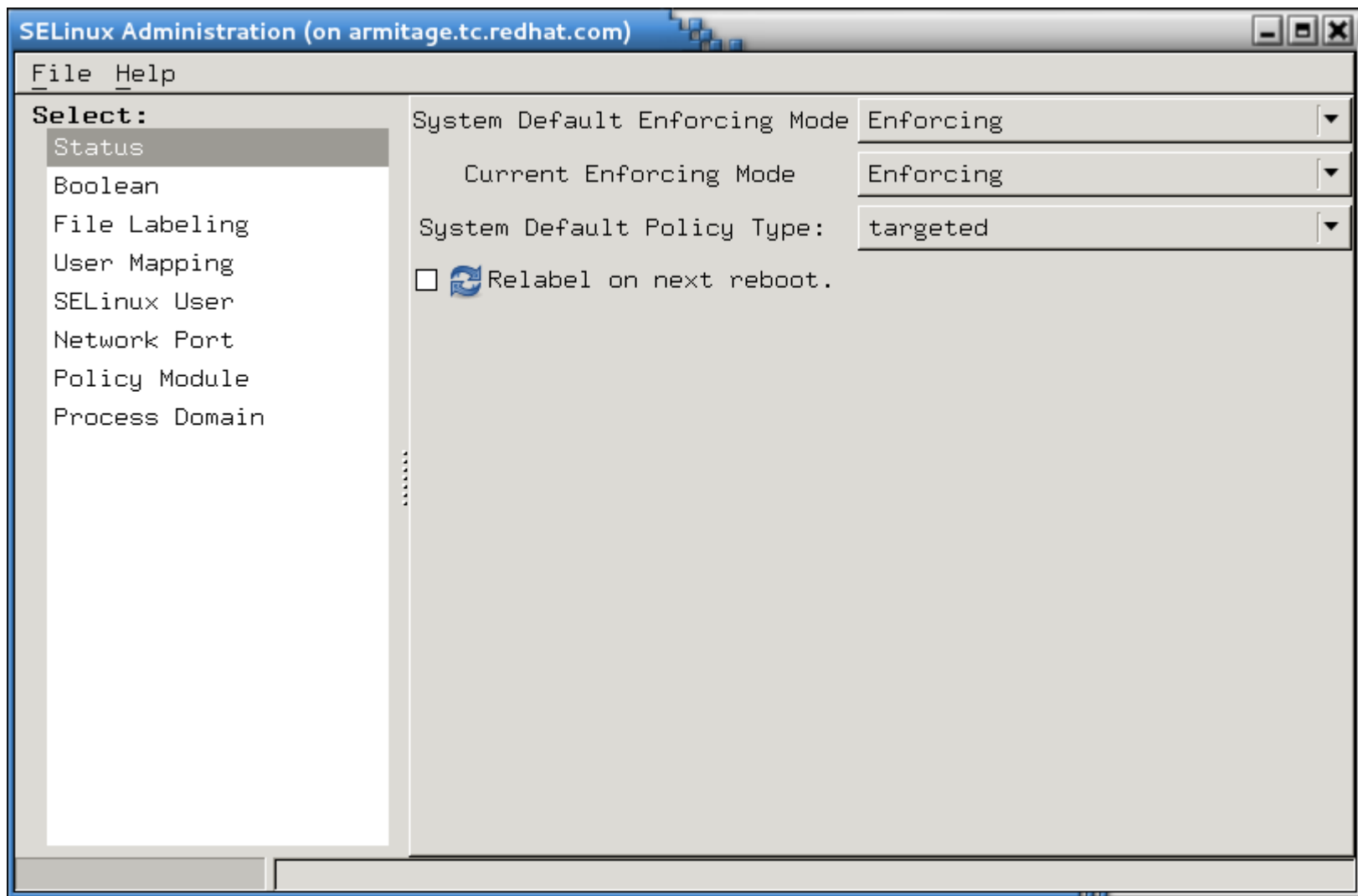
- This stuff is so easy, even a Windows admin can do it!
 - Install xorg-x11-xauth, a font (I like bitmap-fixed-fonts, or you can do `yum groupinstall fonts`), and `polycoreutils-gui`. and you can `ssh -X` into the box and run `system-config-selinux`

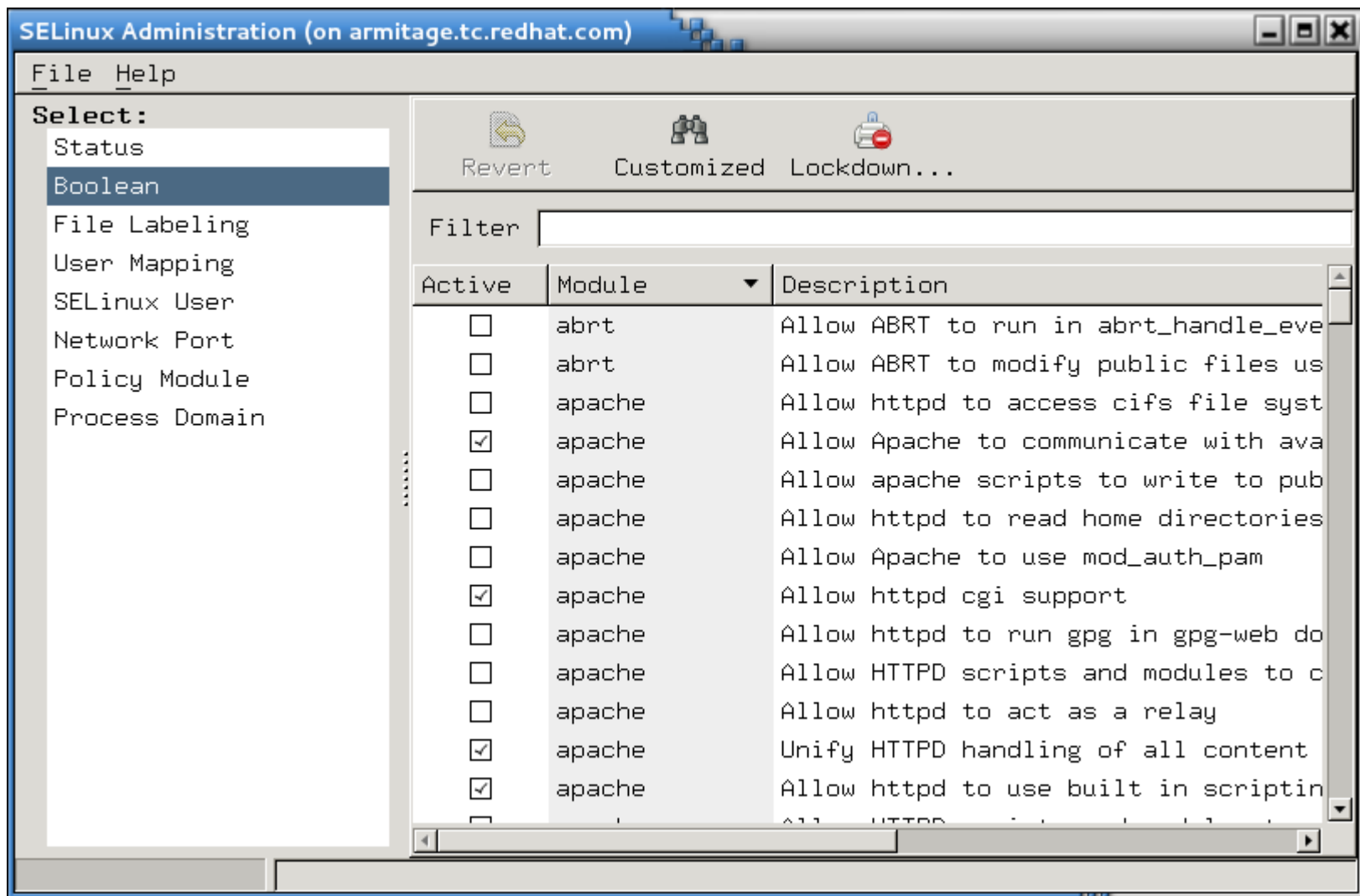
```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# yum install xorg-x11-xauth policycoreutils-gui bitmap-fixed-fonts  
Loaded plugins: product-id, rhnplugin, security, subscription-manager  
Updating certificate-based repositories.  
Unable to read consumer identity  
Setting up Install Process  
Resolving Dependencies  
--> Running transaction check  
--> Package bitmap-fixed-fonts.noarch 0:0.3-15.el6 will be installed  
--> Processing Dependency: fontpackages-filesystem for package: bitmap-fixed-fonts-0.3-15.el6.noarch  
--> Package policycoreutils-gui.x86_64 0:2.0.83-19.24.el6 will be installed  
--> Processing Dependency: gtkhtml2 for package: policycoreutils-gui-2.0.83-19.24.el6.x86_64  
--> Processing Dependency: gnome-python2-gtkhtml2 for package: policycoreutils-gui-2.0.83-19.24.el6.x86_64  
--> Processing Dependency: setools-console for package: policycoreutils-gui-2.0.83-19.24.el6.x86_64  
--> Processing Dependency: usermode-gtk for package: policycoreutils-gui-2.0.83-19.24.el6.x86_64  
--> Processing Dependency: gnome-python2-gnome for package: policycoreutils-gui-2.0.83-19.24.el6.x86_64  
--> Package xorg-x11-xauth.x86_64 1:1.0.2-7.1.el6 will be installed  
--> Processing Dependency: libXmu.so.1()(64bit) for package: 1:xorg-x11-xauth-1
```

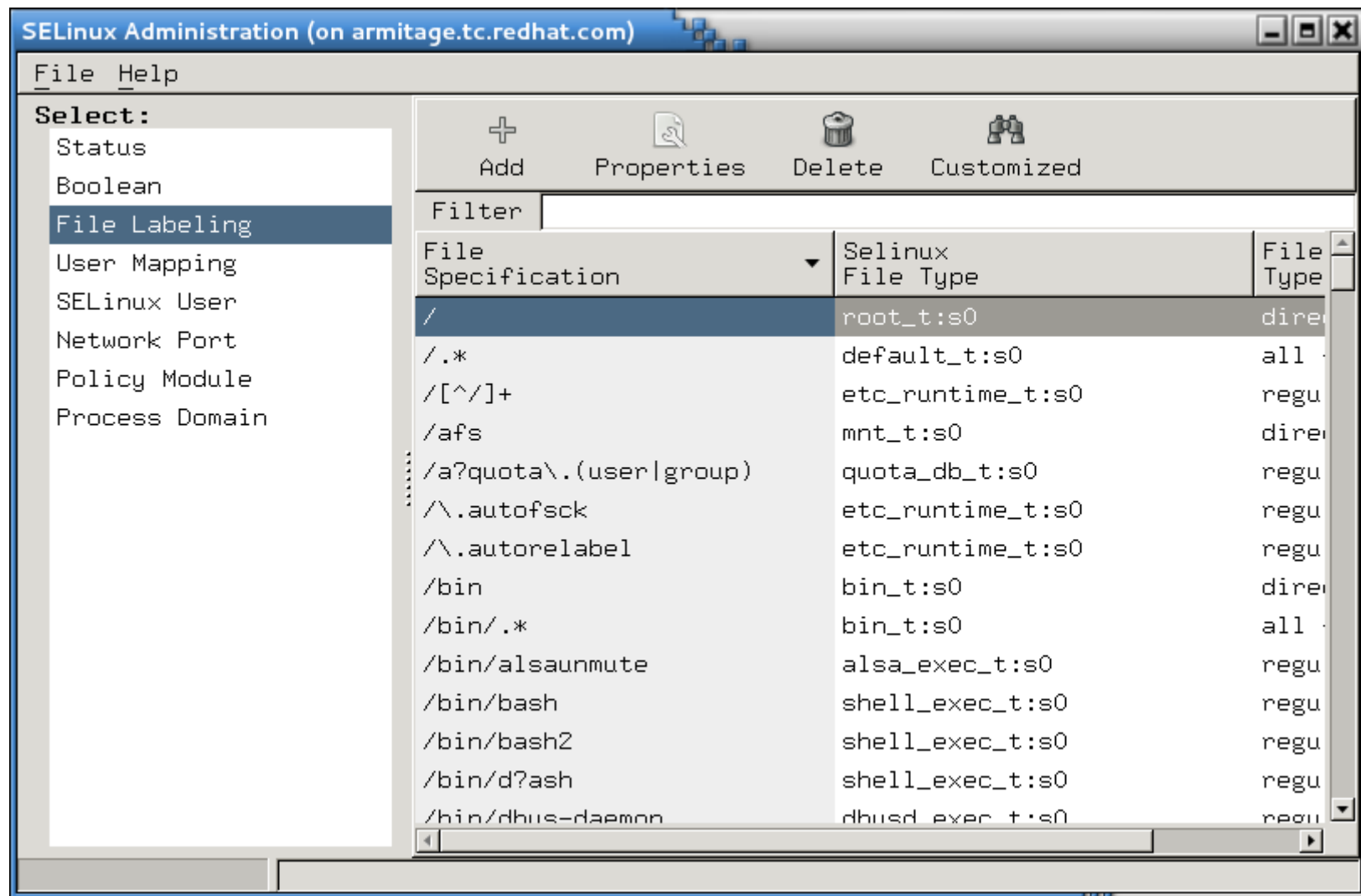
```
root@armitage:~  
File Edit View Search Terminal Help  
gnome-icon-theme.noarch 0:2.28.0-2.el6  
gnome-python2-extras.x86_64 0:2.25.3-20.el6  
gnome-python2-gnome.x86_64 0:2.28.0-3.el6  
gnome-python2-gnomevfs.x86_64 0:2.28.0-3.el6  
gnome-python2-gtkhtml2.x86_64 0:2.25.3-20.el6  
gnome-themes.noarch 0:2.28.1-6.el6  
gnome-vfs2.x86_64 0:2.24.2-6.el6  
gtk2-engines.x86_64 0:2.18.4-5.el6  
gtkhtml2.x86_64 0:2.11.1-7.el6  
libXmu.x86_64 0:1.0.5-1.el6  
libXt.x86_64 0:1.0.7-1.el6  
libbonobo.x86_64 0:2.24.2-5.el6  
libbonoboui.x86_64 0:2.24.2-3.el6  
libdaemon.x86_64 0:0.14-1.el6  
libgnome.x86_64 0:2.28.0-11.el6  
libgnomeui.x86_64 0:2.24.1-4.el6  
setools-console.x86_64 0:3.3.7-4.el6  
shared-mime-info.x86_64 0:0.70-4.el6  
system-gnome-theme.noarch 0:60.0.2-1.el6  
system-icon-theme.noarch 0:6.0.0-2.el6  
usermode-gtk.x86_64 0:1.102-3.el6  
  
Complete!  
[root@armitage ~]#
```

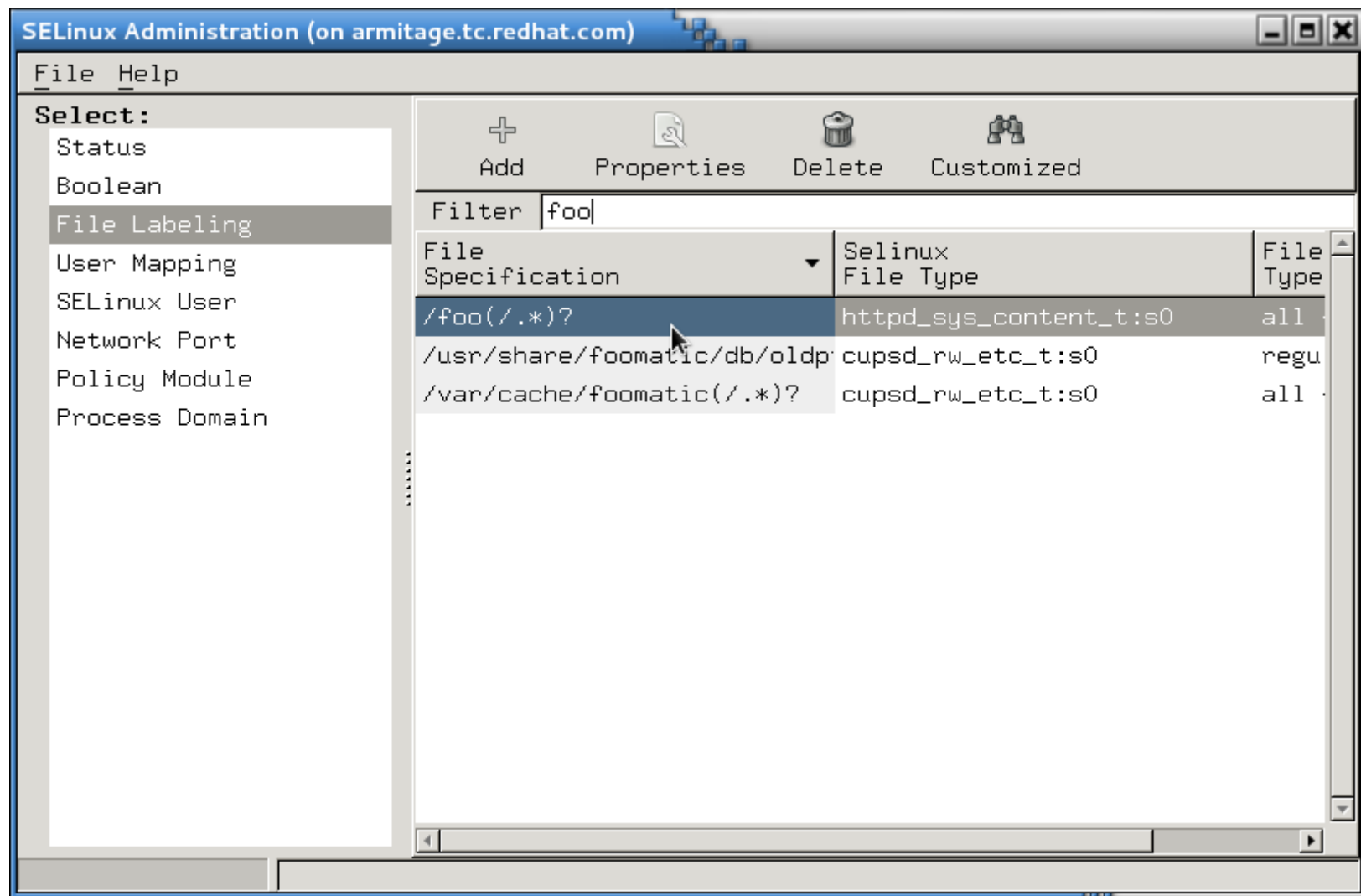


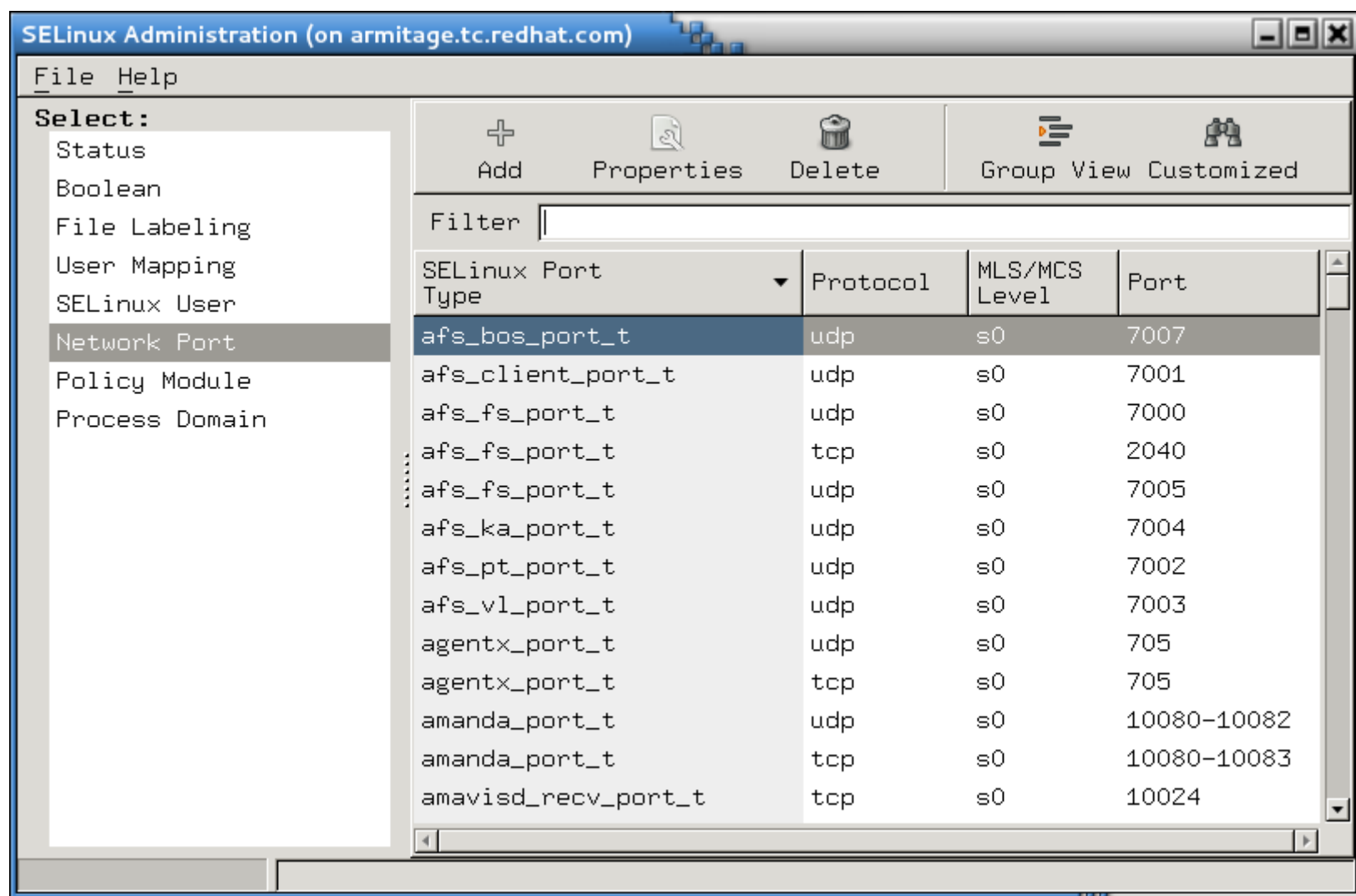
```
root@armitage:~  
File Edit View Search Terminal Help  
[root@armitage ~]# system-config-selinux
```

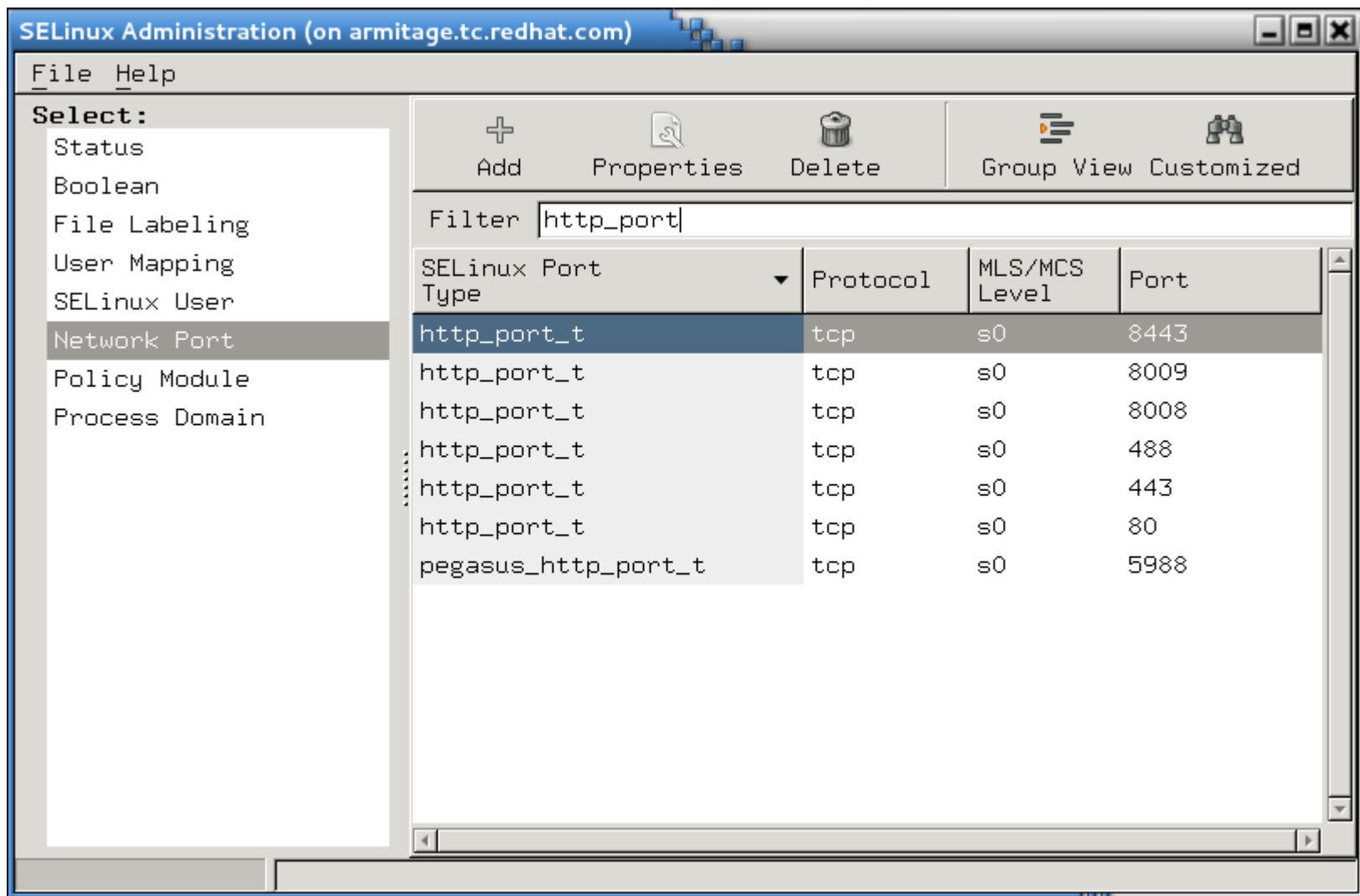


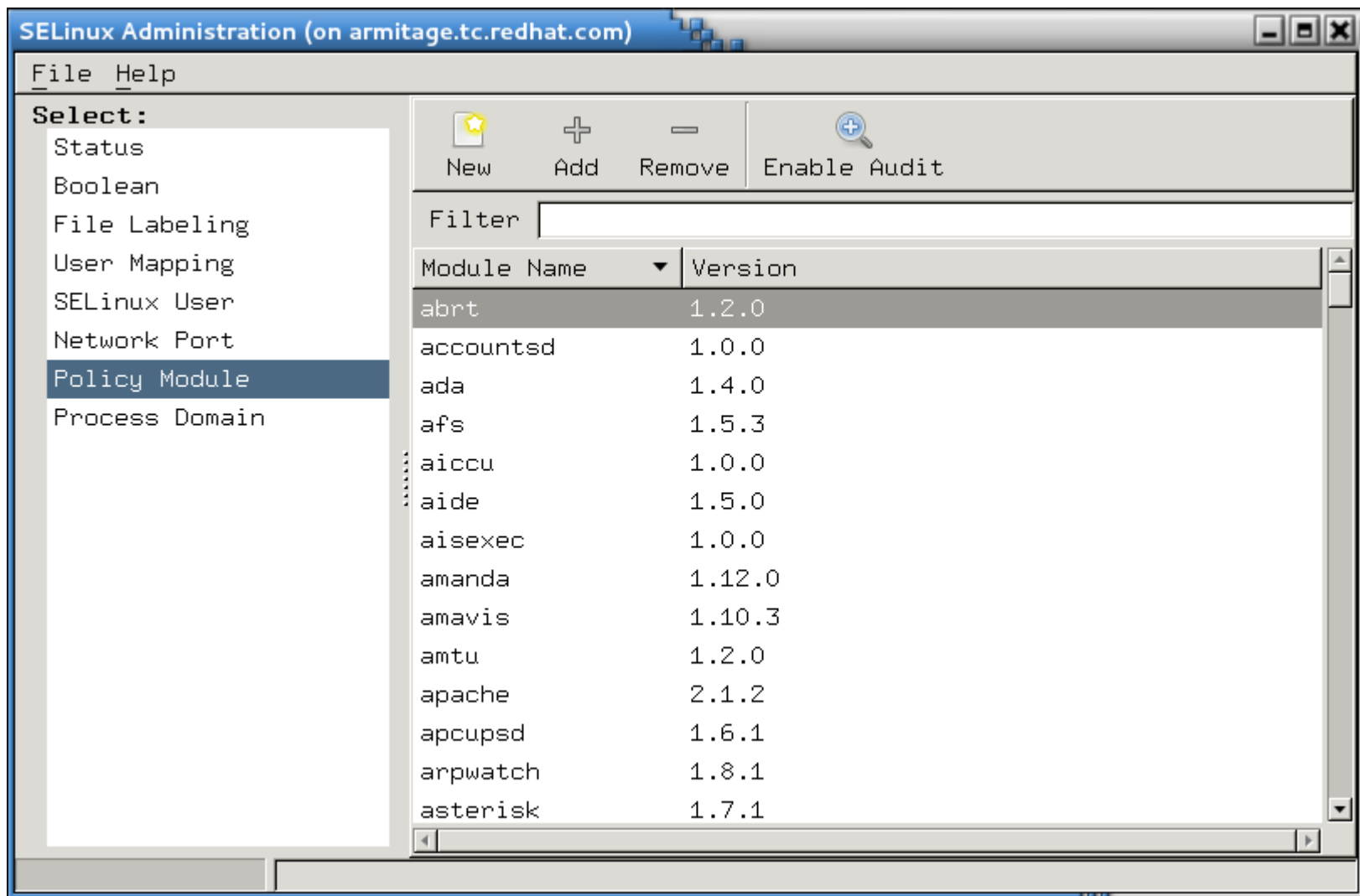


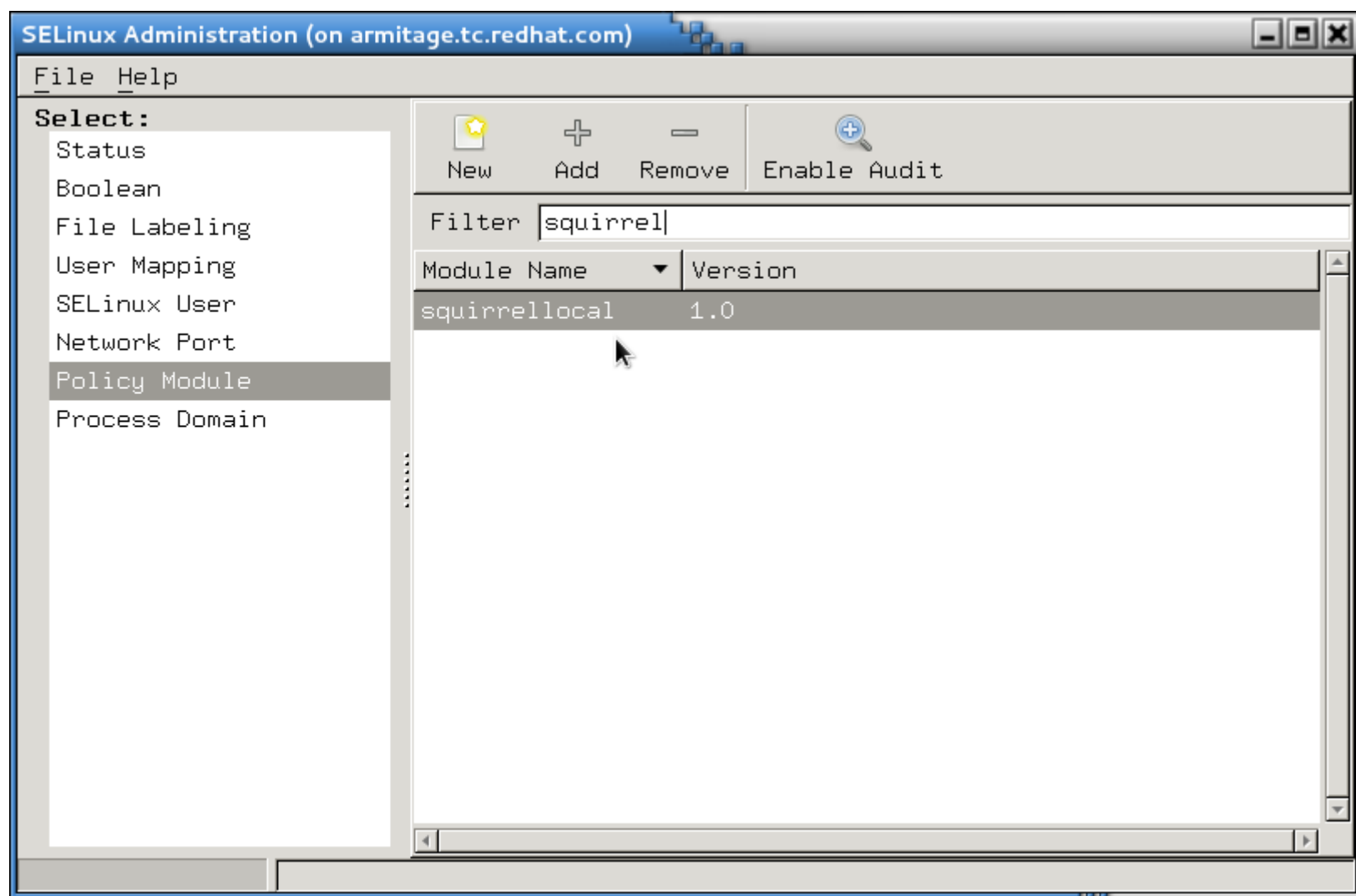












And That's It!

- Hopefully, you now feel like:



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 months ago
- NSA grade security is available at no extra cost - use it!

Thank You!

- If you liked today's presentation, please let us know!

More Information

- SELinux Guide: http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security-Enhanced_Linux/index.html
- Fedora Project SELinux Docs: <http://fedoraproject.org/wiki/SELinux>
- fedora-selinux-list (mailing list):
 - <https://www.redhat.com/mailman/listinfo>
- Red Hat Training - Red Hat Enterprise SELinux Policy Administration: <http://www.redhat.com/training>

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/>

Questions?



Title here

Text with no bullets

- Bullets layer one
 - Bullets layer two
 - Bullets layer three