

# F8 Security

Steve Grubb  
Red Hat

# SE Linux Part 2

# Booleans

- Switches on and off certain policy checks
- The default is reasonable for general install
- Must be tweaked to be most secure



# Booleans

- [root ~]# getsebool -a | grep ' on'
- allow\_daemons\_dump\_core --> on
- allow\_daemons\_use\_tty --> on
- allow\_execmem --> on
- allow\_execstack --> on
- allow\_gadmin\_exec\_content --> on
- allow\_gssd\_read\_tmp --> on
- allow\_kerberos --> on
- allow\_mouton\_anydir --> on
- allow\_postfix\_local\_write\_mail\_spool --> on
- allow\_staff\_exec\_content --> on
- allow\_sysadm\_exec\_content --> on
- allow\_unconfined\_exec\_content --> on
- allow\_unlabeled\_packets --> on
- allow\_user\_exec\_content --> on
- allow\_xserver\_execmem --> on
- allow\_zebra\_write\_config --> on
- browser\_confine\_xguest --> on
- httpd\_builtin\_scripting --> on
- httpd\_enable\_cgi --> on
- httpd\_enable\_homedirs --> on
- httpd\_tty\_comm --> on
- httpd\_unified --> on
- nfs\_export\_all\_ro --> on
- nfs\_export\_all\_rw --> on
- read\_default\_t --> on
- samba\_run\_unconfined --> on
- spamd\_enable\_home\_dirs --> on
- use\_nfs\_home\_dirs --> on
- user\_ping --> on



# Setroubleshooter

- This program was created to be able to solve simple problems.
  - <http://fedoraproject.org/wiki/SELinux/Troubleshooting/AVCDecisions>
- Could the fault tree be automated?



# Setroubleshooter

- Can be used to analyze problems on other machines.
  - Use ausearch to extract the log files
  - Import the file into setroubleshoot



# Improved Roles

- Role based access control means that there are a collective set of domain transitions and access rights that is associated with a user.
- To make roles air tight strict policy was merged with targeted.
- Now have concept of confined users and unconfined users



# New Roles

- `guest_r` - Minimal priv login user for tty login.
- `webadm_r` - Root user administrating only web apps
- `logadm_r` - Root user administrating only logging facilities
- `xguest_r` - Minimal privilege X windows login type
- 
- Ported from strict policy into targeted:
- `staff_r` - Fully privileged user which can become `sysadmin_r`
- `user_r` - Fully privileged user which cannot become `sysadmin_r`
- `unconfined_r` - A completely unconfined role
- `sysadmin_r` - Able to system management
- `auditadm_r` - Allows audit system management



# Guest\_u

- `guest_u` SELinux user is a user account
  - Can not run any Setuid applications
  - Can not communicate on the network
  - Can not execute files in its home directory or `/tmp`
  - Can not use Xwindows.
- Uses: where you allow a user to ssh and configure their `public_html` account, or only run a single application like a git user account.



# Xguest\_u

- xguest\_u is similar to guest\_u except it can work on an Xwindows machine.
  - useful in a kiosk environment.
  - set up a xguest account to allow users to login without a password.
  - Can not run setuid apps
  - Can not connect to the network, except through Firefox.
  - Can not execute files in the home directory.



# Roles

- `guest_u` and `xguest_u` serve as basis for custom roles
  - if you want to setup a guest account that can only send mail
    - use the tool to generate policy for an `xguestmail_u` user account
    - add the ability to connect to port 25.
  - If you wanted `xguest` to be able to chat
    - define a user account `xguest_xchat_u`
    - Add ability to connect to port 194.



# Xguest package

- will setup a machine to run as a kiosk.
  - Sets up the xguest\_u account described above
  - additional safeguards
    - such as a temporary home directory and temp directories
    - They are destroyed on logout
    - Prevents someone from reading information left by you, or leaving an application to watch the next person to use the kiosk.
  - May be good for a livecd image which could be used at libraries, universities, coffee shops, ...



# Writing policy

- New tool to help with simple policy generation
- Answer a few questions
- Generates initial policy
- Run the app
- Regenerate improved policy
- Repeat



# Rsyslog

# Backwards compatible

- Based on sysklogd source
- Can read same config file



# New options

- Regex file splitting
- Execute commands
- TCP connection
- Database backend
- PHP report viewer



# Syslog rules

- Rsyslog uses the old rule format, selector + action
- `openlog("auditd", LOG_PID, LOG_DAEMON);`
- `syslog(LOG_WARNING, "test");`
- Selectors are `facility.level`.
- In the above, `daemon.warning`



# Syslog rules

- Action goes on right side:
  - \*.info;mail.none;authpriv.none;cron.none /var/log/messages
  - This take any info or higher messages and sends them to /var/log/messages with the exception of mail, authpriv, and cron.
  - Simply putting a filename means log to this file



# New Selectors

- :property, [!]compare-operation, "value"
  - Colon must be column in 0
  - Property can be: msg, HOSTNAME, FROMHOST, syslogtag, programname, PRI (numeric), PRI-text, syslogfacility(numeric), syslogfacility-text, syslogseverity, syslogseverity -text, ...
  - Compare ops: contains, isequal, startswith, regex.



# Actions

- Regular file `/var/log/messages`
- Named Pipes `/var/run/my-daemon`
- Tty or console `/dev/console`
- Remote machine `@192.168.1.2`
- Users `root`
- Everyone logged in `*`



# New Actions

- Database table >dbhost,dbname,dbuser,dbpassword;dbtemplate
- Discard ~
- Shell execute ^/bin/my-responder
- Supports templates for actions to make it more readable
  - \$template hostsecure, “/var/log/hosts/%HOSTNAME%/secure/%\$NOW%”
  - /var/log/hosts/myhostname.mydomain.com/secure/2007-07-21



# Examples

\$template hostsecure, “/var/log/hosts/%HOSTNAME%/secure/%\$NOW%”

authpriv.\* -?hostsecure

:msg, contains, “glibc detected” /var/log/glibc.log

:msg, contains, “media error”, ^/bin/my-script

my-script: echo "\$1" | mail -s "test syslogd" someone@example.net



# Questions?

Email: [sgrubb@redhat.com](mailto:sgrubb@redhat.com)

Web page: <http://people.redhat.com/sgrubb/audit>