Practical SELinux: Writing Custom Application Policy

Miroslav Grepl
Lukas Vrabec
Simon Sekidde
Thursday, May 4, 10:15 AM - 12:15 PM
Agenda

- Proactive Security
- SELinux Security Policy
- Updated Userspace with Easier Policy Customization
- SELinux Awareness
- Writing SELinux Policy
- Troubleshooting Existing Policy
Proactive Security
WHEN DO PEOPLE CARE ABOUT SECURITY?
HOW DO SECURITY ISSUES AFFECT ME?
LOST/UNWITTINGLY SHARED/COMPROMISED
PERSONAL DATA
LOST/UNWITTINGLY SHARED/COMPROMISED PERSONAL DATA

ONLINE MONEY-RELATED THEFTS
LOST/UNWITTINGLY SHARED/COMPROMISED
PERSONAL DATA
ONLINE MONEY-RELATED THEFTS
MOBILE DEVICE PROTECTION
WHERE DO SECURITY ISSUES COME FROM?
I HAVE NO IDEA WHAT I'M DOING
HOW ARE THEY FIXED?
REACTIVE SECURITY
YOUR SYSTEM IS NOT PROTECTED DURING THE WINDOW OF VULNERABILITY!
PROACTIVE SECURITY
Vulnerable software released

Vulnerability announcement

Fix backported & updated released

Window of vulnerability is filled by proactive security

Timeline
PROACTIVE SECURITY HELPS TO **PROTECT** YOUR SYSTEM DURING THE WINDOW OF VULNERABILITY!
SECURITY ENHANCED LINUX IS A SECURITY MECHANISM BRINGING PROACTIVE SECURITY FOR YOUR SYSTEM.
EXPLOIT EXAMPLES WHERE SELINUX HELPED TO PROTECT YOUR SYSTEM
VENOM
VENOM

DOCKER CVE-2016-9962
VENOM

DOCKER CVE-2016-9962

SHELLSHOCK
HACKING TIME!
DEMO TIME!
CONCLUSION?
IF YOU RUN LINUX WITH SELINUX DISABLED

YOUR GONNA HAVE A BAD TIME
SELinux Security Policy
CORE COMPONENT OF SELINUX
CORE COMPONENT OF SELINUX

COLLECTION OF SELINUX POLICY RULES
CORE COMPONENT OF SELINUX
COLLECTION OF SELINUX POLICY RULES
LOADED INTO THE KERNEL BY SELINUX
USERSPACE TOOLS
ENFORCED BY THE KERNEL
ENFORCED BY THE KERNEL

USED TO AUTHORIZE ACCESS REQUESTS ON THE SYSTEM
BY DEFAULT EVERYTHING IS DENIED AND YOU DEFINE POLICY RULES TO ALLOW CERTAIN REQUESTS.
SELINUX POLICY RULES
DESCRIBE AN **INTERACTION** BETWEEN PROCESSES AND SYSTEM RESOURCES
SELINUX POLICY RULE IN HUMAN LANGUAGE
"APACHE process can READ its LOGGING FILE"
SELINUX VIEW OF THAT INTERACTION
ALLOW apache_process apache_log:FILE
READ;
apache_process  apache_log

ARE LABELS
LABELS
ASSIGNED TO PROCESSES
ASSIGNED TO PROCESSES
ASSIGNED TO SYSTEM RESOURCES
BY SELINUX SECURITY POLICY
ASSIGNED TO PROCESSES
ASSIGNED TO SYSTEM RESOURCES
BY SELINUX SECURITY POLICY

MAP REAL SYSTEM ENTITIES INTO THE SELINUX WORLD
LABELS IN REALITY
STORED IN EXTENDED ATTRIBUTES OF FILE SYSTEMS - EXT2, EXT3, EXT4 ...
# getfattr -n security.selinux /etc/passwd
getfattr: Removing leading '/' from absolute path names

# file: etc/passwd
security.selinux="system_u:object_r:passwd_file_t:s0"

# ls -Z /etc/passwd
system_u:object_r:passwd_file_t:s0 /etc/passwd
SELINUX LABELS CONSIST OF **FOUR** PARTS
<user>:<role>:<type>:<MLS/MCS>
<user>:<role>:<type>:<MLS/MCS>

Not the same as Linux users

Several Linux users can be mapped to a single SELinux user

object_u is a placeholder for Linux system resources

system_u is a placeholder for Linux processes

Can be limited to a set of SELinux roles
SELinux users can have multiple roles but only one can be active.

- object_r is a placeholder for Linux system resources
- system_r is a placeholder for system processes

Can be limited to a set of SELinux types.
Security model known as **TYPE ENFORCEMENT**

In 99% you care only about TYPES

policy rules and interactions between types
Multi Level Security

Only the MCS part is used in Targeted Policy with the default s0 level

Allow users to mark resources with compartment tags \((MCS1, MCS2)\)

Used for RHEL virtualization and for container security

\(s0:c1\) can not access \(s0:c2\)
IN RHEL7 WE SHIP THE **TARGETED** SELINUX POLICY BY DEFAULT
WE MOSTLY CARE ONLY ABOUT **TYPES**
SELinux ALLOW RULE SYNTAX WITH TYPES
ALLOW TYPE1 TYPE2:OBJECT_CLASS PERMISSION;
ALLOW APACHE_T APACHE_LOG_T:FILE READ;
Updated Userspace with Easier Policy Customization
FRIENDLY SELINUX?
NEW SELINUX USERSPACE 2.5 INTRODUCED IN RHEL-7.3
NEW COMMON INTERMEDIATE LANGUAGE - CIL
"M4+COMPILATION" VS. CIL
SELinux policy rules in source policy files
*.te, *.if
*.fc files

Makefile

checkmodule
.mod

semodule_package
.pp

semodule
policy.29

*cil

semodule
policy.29
PERFORMANCE IMPROVEMENTS
PERFORMANCE IMPROVEMENTS

NEW POSSIBILITY FOR HLL
PERFORMANCE IMPROVEMENTS
NEW POSSIBILITY FOR HLL
USABILITY
LOCAL POLICY IN TWO STEPS
“I have an apache process that needs to access its log file. I would like to add SELinux policy rule reflecting the following interaction so that I am able to read important info from the apache logging file.”
# cat myapache.cil

(allow httpd_t httpd_log_t (file (open read getattr)))

#redhat #rhsummit
# semodule -i myapache.cil
HOW DO WE DO IT WITH M4 + COMPILATION?
# cat myapache.te

require {
  type httpd_t;
  type httpd_log_t;
}

allow httpd_t httpd_log_t:file { open read getattr };
# make -f /usr/share/selinux/devel/Makefile
# semodule -i myapache.pp
FUTURE
SELinux Awareness
SELINUX ENHANCING TECHNOLOGIES
SYSTEMD
SYSTEMD WORKS AS AN SELINUX ACCESS MANAGER
DO YOU REMEMBER SELINUX ALLOW RULE SYNTAX WITH TYPE ENFORCEMENT?
ALLOW TYPE1 TYPE2:OBJECT_CLASS
PERMISSION;
ALLOW HTTPD_T HTTPD_UNIT_FILE_T:SERVICE
START;
SYSTEMD

SVIRT
APPLIES MAC TO IMPROVE SECURITY WHEN USING VIRTUAL MACHINES
SELinux user:SELinux role:SELinux type:SELinux category
SELinux user:SELinux role:SELinux type:SELinux category
system_u:object_r:svirt_t:c306,c536
SELinux user:SELinux role:SELinux type:SELinux category
system_u:object_r:svirt_t:c306,c536
system_u:object_r:svirt_t:c206,c636
SYSTEMD
SVIRT
CONTAINERS
SELINUX Keeps Your Container In Its Own Space
SELinux user:SELinux role:SELinux type:SELinux category
SELinux user:SELinux role:SELinux type:SELinux category
system_u:object_r:container_t:c306,c536
SELinux user:SELinux role:SELinux type:SELinux category

system_u:object_r:container_t:c306,c536
system_u:object_r:container_t:c206,c636
SELinux user:SELinux role:SELinux type:SELinux category

system_u:object_r:container_t:c306,c536
system_u:object_r:container_t:c206,c636
system_u:object_r:container_t:c406,c736
SELINUX MANAGEMENT
SELINUX TROUBLESHOOTING IN COCKPIT
SELINUX TROUBLESHOOTING IN COCKPIT
SELINUX USERSPACE INSIDE CONTAINER
$ getenforce
Permissive

$ sudo docker run --privileged -v /sys/fs/selinux:/sys/fs/selinux:rw -v /etc/selinux:/etc/selinux:rw -v /etc/selinux:/var/lib/selinux:rw -it selinux-container:latest setenforce 1

$ getenforce
Enforcing
SELINUX TROUBLESHOOTING COCKPIT
SELINUX USERSPACE INSIDE CONTAINER
SELINUX WITH ANSIBLE
https://github.com/cockpit-project/system-api-roles/tree/master/roles/SELinux
DEMO PLAYBOOK
$ cat > demo-playbook.yml <<EOF
---
- hosts: all
  remote_user: root
  vars:
    SElinux_type: targeted
    SElinux_mode: enforcing
    SElinux_change_running: 1
    SElinux_booleans:
      - { name: 'samba_enable_home_dirs', state: 'on' }
      - { name: 'ssh_sysadm_login', state: 'on', persistent: 'yes' }
    SElinux_file_contexts:
      - { target: '/tmp/test_dir/(.*)?', setype: 'user_home_dir_t', ftype: 'd' }
    SElinux_restore_dirs:
      - /tmp/test_dir
  roles:
    - SElinux
EOF
RUN THE PLAYBOOK USING SELINUX ROLES
$ ansible-playbook -i
grepl-rhel-73.virt,grepl-rhel-6.virt,grepl-fedora-25.virt,
demo-playbook.yml
SELINUX MODES
ENFORCING
ENFORCING

SELINUX SECURITY POLICY IS ENFORCED BY KERNEL
PERMISSIVE
PERMISSIVE

SELINUX SECURITY POLICY IS NOT ENFORCED BY KERNEL
PERMISSIVE

SELINUX SECURITY POLICY IS NOT ENFORCED BY KERNEL

ACCESSSES ARE LOGGED
LET’S SWITCH SELINUX TO PERMISSIVE TO COLLECT MORE AVC MESSAGES
AVC MESSAGES
WHERE CAN WE FIND LOGS?
# cat /var/log/audit/audit.log
# cat /var/log/audit/audit.log

# ausearch -m AVC
type=AVC  msg=audit(1226882925.714:136):  avc: denied
{  read  } for  pid=2512  comm="httpd"  name="file1"
    dev=dm-0  ino=284133
    scontext=unconfined_u:system_r:httpd_t:s0
    tcontext=unconfined_u:object_r:shadow_t:s0
    tclass=file
HOW TO PARSE AVC MESSAGES?
# sesearch
# sesearch

# audit2allow
$ ausearch -m AVC -ts recent

type=AVC msg=audit(1226882925.714:136): avc: denied { read } for
pid=2512 comm="httpd" name="shadow" dev=dm-0 ino=284133
scontext=unconfined_u:system_r:httpd_t:s0
tcontext=unconfined_u:object_r:shadow_t:s0 tclass=file

$ ausearch -m AVC -ts recent | audit2allow

#============= httpd_t ==============
allow httpd_t shadow_t:file read;
ALL NECESSARY FILES FOR WRITING CUSTOM POLICY CAN BE FOUND IN /USR/SHARE/SELINUX/DEVEL DIRECTORY
rhusummit.service
DUMMY LINUX DAEMON FOR TESTING PURPOSE
DUMMY LINUX DAEMON FOR TESTING PURPOSE

WE WILL WRITE POLICY FOR IT
DUMMY LINUX DAEMON FOR TESTING PURPOSE
WE WILL WRITE POLICY FOR IT
ACTIONS
Connecting on port 80 tcp on lvrabec-selinux.rhcloud.com
Logging messages into journal
Creating pid file
Reading /proc/meminfo
# systemctl status rhsummit

- rhsummit.service - Testing SELinux for Red Hat Summit 2017

  Loaded: loaded
  (/usr/lib/systemd/system/rhsummit.service; disabled; vendor preset: disabled)

  Active: inactive (dead)
# systemctl start rhsummit

# systemctl status rhsummit
● rhsummit.service - Testing SELinux for Red Hat Summit 2017
  Loaded: **loaded**
  (/usr/lib/systemd/system/rhsummit.service; disabled; vendor preset: disabled)
  Active: **active (running)** since Tue 2017-04-11 16:16:47 CEST; 9s ago
  Process: 2827 ExecStart=/usr/bin/rhsummit
  (code=exited, status=0/SUCCESS)
NO SELINUX POLICY FOR THE *RHSUMMIT* SERVICE MEANS THE SERVICE IS UNCONFINED SERVICE FROM SELINUX POINT OF VIEW!
# ps -efZ | grep rhsummit
system_u:system_r:unconfined_service_t:s0 root 2828 1
0 16:16 ?  00:00:00 /usr/bin/rhsummit
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
root 2849 2344  0 16:33 pts/0 00:00:00 grep
   --color=auto rhsummit
# sepolicy generate --init /usr/bin/rhsummit
Loaded plugins: product-id
Created the following files:
/root/code/policy/rhsummit.te # Type Enforcement file
/root/code/policy/rhsummit.if # Interface file
/root/code/policy/rhsummit.fc # File Contexts file
/root/code/policy/rhsummit_selinux.spec # Spec file
/root/code/policy/rhsummit.sh # Setup Script

# ./rhsummit.sh
HOW TO CHECK AN SELINUX STATUS FOR OUR SUMMIT SERVICE?
# systemctl stop rhsummit
# systemctl start rhsummit
# ps -efZ | grep summit

# ps -efZ | grep summit

```bash
system_u:system_r:rhsummit_t:s0 root  3049  1  0 17:03 ?  00:00:00 /usr/bin/rhsummit
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 3051 2344  0 17:03 pts/0  00:00:00 grep --color=auto summit
```
RHSUMMIT SERVICE RUNS WITH RHSUMMIT LABEL WHICH MEANS THAT THE SERVICE IS CONFINED!
IMPORTANT SELINUX POLICY FILES
**rhsummit.fc** file contains SELinux security context for all service objects (files, dirs, lnk_files, sockets, ...)

**rhsummit.te** file contains rules for rhsummit_t domain

**rhsummit.if** file contains interfaces to access rhsummit_t types

**rhsummit.sh** bash script for compiling and installing SELinux policy for rhsummit service
WHY THE RHSUMMIT SERVICE RUNS AS RHSUMMIT_T DOMAIN?
SELINUX PROCESS TRANSITION RULES!
# sesearch -T -s init_t -t rhsummit_exec_t
Found 1 semantic type rules:
    type_transition init_t rhsummit_exec_t : process
    rhsummit_t;
AVCS IN AUSEARCH OUTPUTS
# ausearch -m AVC -ts recent
...
...
...

# ausearch -m AVC -ts recent > ~/avc_file
AVCS RELATED TO CREATING PID FILES
/var/run/rhsummit.pid should have custom label rhsummit_pid_t instead of var_run_t

type=AVC msg=audit(1491928595.618:342): avc: denied { write } for pid=3924 comm="rhsummit" name="rhsummit.pid" dev="tmpfs" ino=35478 scontext=system_u:system_r:rhsummit_t:s0 tcontext=unconfined_u:object_r:var_run_t:s0 tclass=file
rhsummit.te

+ type rhsummit_pid_t;

+ files_pid_file(rhsummit_pid_t)

+ manage_files_pattern(rhsummit_t, rhsummit_pid_t, rhsummit_pid_t)

+ files_pid_filetrans(rhsummit_t, rhsummit_pid_t, { file })

rhsummit.fc

+ /var/run/rhsummit.* -- gen_context(system_u:object_r:rhsummit_pid_t,s0)
AVCS RELATED TO READING /PROC
rhsummit process reads /proc/meminfo file, this access should be allowed

type=AVC msg=audit(1491928595.618:343): avc: denied { read } for pid=3924 comm="rhsummit" name="meminfo" dev="proc" ino=4026532028 scontext=system_u:system_r:rhsummit_t:s0 tcontext=system_u:object_r:proc_t:s0 tclass=file
rhsummit.te

+ kernel_read_system_state(rhsummit_t)

Interface can be found in

/usr/share/selinux/devel/include/kernel/kernel.if
AVCS RELATED TO NETWORK ACCESS
service connects to “lvrabec-selinux.rhcloud.com”, this should be allowed

```
type=AVC msg=audit(1491939758.424:387): avc: denied { name_connect } for pid=4430 comm="rhsummit" dest=80
  scontext=system_u:system_r:rhsummit_t:s0
tcontext=system_u:object_r:http_port_t:s0 tclass=tcp_socket
```
rhsummit.te

+ corenet_tcp_connect_http_port(rhsummit_t)

Interface can be found in

/usr/share/selinux/devel/include/kernel/corenetwork.if
Service resolves DNS records, so access to /etc/resolv.conf is needed.

type=AVC msg=audit(1491942725.855:436): avc: denied { read }
for pid=4874 comm="rhsummit" name="resolv.conf" dev="dm-0"
ino=5051289 scontext=system_u:system_r:rhsummit_t:s0
tcontext=system_u:object_r:net_conf_t:s0 tclass=file
rhsummit.te

+ sysnet_read_config(rhsummit_t)

Interface can be found in

/usr/share/selinux/devel/include/system/sysnetwork.if
AVCS RE-CHECK FOR THE RHSUMMIT SERVICE
# ausearch -m AVC -ts recent

<no matches>
SELINUX POLICY FOR THE RHSUMMIT SERVICE IS READY!
Troubleshooting Existing Policy
MISLABELED SYSTEM I.E LABELS ON OBJECTS ARE WRONG
# restorecon -Rv /

or

# fixfiles onboot
# reboot
SELINUX MODIFICATIONS OF THE DISTRO POLICY VIA SELINUX USERSPACE TOOLING
SOMETIMES IT’S NOT NECESSARY TO CREATE CUSTOM SELINUX POLICY, LOCAL MODIFICATION CAN FIX IT.
APACHE HTTP SERVER WITH CHANGES IN THE DEFAULT CONFIGURATION
httpd service configured to listen on tcp port 3131 instead of port 80
httpd service configured to listen on tcp port 3131 instead of port 80
document root will be /var/test_www/ instead of /var/www/
Change in /etc/httpd/conf/httpd.conf

Listen 80 -> Listen 3131
Change in /etc/httpd/conf/httpd.conf

```
# sed -i 's_Listen 80_Listen 3131_' /etc/httpd/conf/httpd.conf
```
APPLY CHANGES IN THE CONFIGURATION AND SEARCH FOR AVC DENIALS
# systemctl restart httpd
# ausearch -m AVC -ts recent
....
httpd service trying to bind on port 3131 instead of 80, this should be changed in SELinux policy

```
type=AVC msg=audit(1491948261.488:599): avc: denied { name_bind } for
    pid=5920 comm="httpd" src=3131
tcontext=system_u:system_r:httpd_t:s0
tclass=tcp_socket
```
# sesearch -A -s httpd_t -t http_port_t -c tcp_socket -p name_bind
Found 1 semantic av rules:
    allow httpd_t http_port_t : tcp_socket name_bind ;

# semanage port -a -t http_port_t -p tcp 3131
# semanage port -l | grep http_port_t
http_port_t       tcp    3131, 80, 81, 443, 488, 8008, 8009, 8443, 9000
httpd service DocumentRoot is in /var/test_www/html and this directory has wrong label

type=AVC msg=audit(1491949594.146:622): avc: denied { read } for pid=6094 comm="httpd" name="index.html" dev="dm-0" ino=13485999 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:var_t:s0 tclass=file
# matchpathcon /var/test_www/html/index/html
/var/test_www/html/index/html  system_u:object_r:var_t:s0

# matchpathcon /var/www/html/index.html
/var/www/html/index.html  system_u:object_r:httpd_sys_content_t:s0

# semanage fcontext -a -t httpd_sys_content_t "/var/test_www(/.*)?"
# restorecon -Rv /var/
# semanage fcontext -l | grep httpd_sys_content_t | grep www
/var/www(/.*)? all files
system_u:object_r:httpd_sys_content_t:s0
/var/test_www(/.*)? all files
system_u:object_r:httpd_sys_content_t:s0

....

....
HTTPD SERVICE SELINUX DENIALS ARE FIXED WITHOUT WRITING CUSTOM POLICY!
LOCAL POLICY MODULE WRITTEN IN CIL
When system interface cannot be used (e.g: there is no such interface), it’s possible to create local module in CIL language.

For example we have following AVC:

```plaintext
type=AVC msg=audit(1491942725.855:436): avc: denied { read }
for pid=4874 comm="rhsummit" name="resolv.conf" dev="dm-0"
  ino=5051289 scontext=system_u:system_r:rhsummit_t:s0
tcontext=system_u:object_r:net_conf_t:s0 tclass=file
```
# cat module.cil
(allow source_context target_type (class (permissions)))

# cat local_module.cil
(allow rhssummary_t net_conf_t (file (read open getattr)))

# semodule -i local_module.cil
# semodule -1full | grep local_module
400 local_module cil
QUESTIONS?

Miroslav Grepl’s blog  https://mgrepl.wordpress.com/
Paul Moore’s blog  http://www.paul-moore.com/
Lukas Vrabec’s blog  https://lvrabec-selinux.rhcloud.com/
Dan Walsh’s blog  http://danwalsh.livejournal.com/
THANK YOU
LEARN. NETWORK. EXPERIENCE OPEN SOURCE.