SELinux basis

Presented by
Zamir SUN
Fedora Community

Agenda

- Introduction of SELinux
- Context and policy basis
- Work with SELinux – basis
- Work with targeted policy

Introduction of SELinux

Brief Introduction

- Security-Enhanced Linux (SELinux) is an implementation of a flexible **mandatory access control** architecture in Linux kernel, checking for allowed operations after standard **discretionary access controls** are checked.
- SELinux can enforce rules on files and processes in a Linux system, and on their actions.

Advantages over DAC

- Processes are separated from each other by running in their own domains
- Access is only allowed if an SELinux policy rule exists that specifically allows it.
- Fine granted
- Enforced system-wide

Example

- *May <subject> do <action> to <object>?*
- Apache HTTPD (httpd_t) can acccess /var/www/html (httpd_sys_content_t)
- Apache HTTPD (httpd_t) cannot access /tmp (tmp_t) even if the file permission allows it

Modes

- /etc/sysconfig/selinux -> /etc/selinux/config
- Modes:
  - Enforcing: Enforced and denied if there is no policy for allow
  - Permissive: Not enforced, violation is recorded but still allowed
  - Disabled: SELinux won’t take any effect

Context and Policy Basis

SELinux context

- SELinux context is used on processes and files
- SELinux contexts are stored in file extended attributes.
- $ ps -eZ | grep auditd
  - system_u:system_r:kernel_t:s0 22 ? 00:00:00 kauditd
  - system_u:system_r:auditd_t:s0 658 ? 00:00:00 auditd

    ↑ user ↑ role ↑domain ↑level
- $ ls -Zd /etc/ssh
  - system_u:object_r:etc_t:s0 /etc/ssh

    ↑ user ↑ role ↑type ↑level

fedora

SELinux context

- User
  - Each Linux user is mapped to an SELinux user
    - $ id -Z
    - unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
  - Used for a specific set of roles, and for a specific MLS/MCS range.
- Role
  - Used in the Role-Based Access Control (RBAC) security model.
  - Also serves as an intermediary mapping between domains and SELinux users.

SELinux context

- Type
  - Attribute of Type Enforcement
  - Defines a domain for processes and a type for files
- Level
  - Attribute of MLS and MCS
  - Basically means a confidential level

Access control

- The targeted SELinux policy ships with 4 forms of access control:
- Type Enforcement (TE)
  - Dog is banned from eat cat chow
- Role-Based Access Control (RBAC)
  - Dog feeders can access dog chow, but not cat chow

Access control

- Multi Category Security (MCS)
  - Shepherd is banned from eat beagle's chow
- Multi Level Security (MLS)
  - Company information is public, employee only, restricted, confidential
  - Contractors can only read public info
  - Employee dealing with restricted product can read restricted info upon business need

Policy

- Targeted
  - Default policy in Fedora / RHEL which "targets" and confines selected system processes.
- Minimal
  - Originally designed for small memory devices
  - Only selected subset of the targeted policy
- Multi Level Security (MLS)
  - In addition to normal user:role:type, MLS uses “security level” to control access

Work with SELinux - basis

Disabling

- SELINUX=disabled in /etc/selinux/config
  - Can also be disabled in kernel parameter selinux=0
  - Once the init scripts noticed selinux=0 in kernel parameter will touch /.autorelabel which makes system to relabel the system next time you boot with SELinux enabled.

Permissive & Enforcing

- SELINUX=enforcing or permissive in /etc/selinux/config
- If you need to temporary change it
  - Change to permissive
    - # setenforce 0
  - Change to enforcing
    - # setenforce 1
- To check the current status
  - # getenforce

Log

- If auditd is installed and running, log is recorded in audit log with type=AVC (Access Vector Cache)
  - /var/log/audit/audit.log
    - type=AVC msg=audit(1587717890.043:863): avc: denied { relabelto } for pid=3210 comm="chcon" name="pub" dev="dm-0" ino=2626342 scontext=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 tcontext=unconfined_u:object_r:unconfined_t:s0 tclass=dir permissive=0

Log

- If setroubleshoot is installed and running, log is also in /var/log/messages
  - Jan 7 18:55:56 localhost setroubleshoot: SELinux is preventing httpd (httpd_t) "getattr" to /var/www/html/file1 (samba_share_t). For complete SELinux messages. run sealert -l de7e30d6-5488-466d-a606-92c9f40d316d

Packages

- policycoreutils-python-utils – semanage, audit2allow, audit2why
- policycoreutils – restorecon, setsebool
- libselinux-utils – getsebool, getenforce, setenforce
- setools-console - sesearch
- policycoreutils-devel - sepolicy

Work with Targeted Policy

Overview

- Targeted is the default policy for Fedora, where targeted processes run in a confined domain

Confined processes

- Runs in its own domain
- Have limited access to resources (based on SELinux policy)

Demo – httpd file

- ps -eZ | grep httpd
  - system_u:system_r:httpd_t:s0 2212 ? 00:00:00 httpd
- # echo Testpage > test.html
- # ls -lZ test.html
- -rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 0 May 6 21:50 test.html
- # chcon -t admin_home_t test.html
  - Temporary changing the context.

Demo – httpd file

- # curl http://192.168.130.10/test.html
  - <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
  - <html><head>
  - <title>403 Forbidden</title>
  - </head><body>
  - <h1>Forbidden</h1>
  - You don't have permission to access this resource.
  - </body></html>

Demo – httpd file

- # ausearch -m AVC -ts today
- ----
- time->Wed May 6 21:51:56 2020
- type=AVC msg=audit(1588773116.207:965): avc: denied { getattr } for pid=724 comm="httpd" path="/var/www/html/test.html" dev="dm-0" ino=2493829 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:admin_home_t:s0 tclass=file permissive=0

Demo – httpd file

- # restorecon -v . -R
- # curl http://192.168.130.10/test.html
  - Testpage

- # chcon -t httpd_sys_content_t test.html
- Context changed by chcon can be restored to the default context (if defined) by restorecon -v.
- For permanent changing of context, use semanage
- # semanage fcontext -a -t httpd_sys_content_t test.html

Demo – httpd port

- # grep ^Listen /etc/httpd/conf/httpd.conf
  - Listen 12345

- # systemctl restart httpd
  - Job for httpd.service failed because the control process exited with error code.
  - See "systemctl status httpd.service" and "journalctl -xe" for details.

Demo – httpd port

- # systemctl status httpd

• httpd.service - The Apache HTTP Server

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

Active: failed (Result: exit-code) since Mon 2020-06-29 16:50:36 HKT; 23s ago

...

Jun 29 16:50:36 fzug-32-vm systemd[1]: Starting The Apache HTTP Server...

Jun 29 16:50:36 fzug-32-vm httpd[384038]: AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using fe80::1c8c:f262:81ff:9371%ens3. Set the 'ServerName' d>

Jun 29 16:50:36 fzug-32-vm httpd[384038]: (13)Permission denied: AH00072: make_sock: could not bind to address [::]:12345

Jun 29 16:50:36 fzug-32-vm httpd[384038]: (13)Permission denied: AH00072: make_sock: could not bind to address 0.0.0.0:12345

...

Jun 29 16:50:36 fzug-32-vm systemd[1]: Failed to start The Apache HTTP Server.

Demo – httpd port

- # ausearch -m AVC -ts today
  - ----
  - time->Mon Jun 29 16:50:36 2020
  - type=AVC msg=audit(1593420636.777:4125): avc: denied { name_bind } for pid=384038 comm="httpd" src=12345 scontext=system_u:system_r:httpd_t:s0 tcontext=system_u:object_r:unreserved_port_t:s0 tclass=tcp_socket permissive=0
  - ----
  - time->Mon Jun 29 16:50:36 2020
  - type=AVC msg=audit(1593420636.777:4126): avc: denied { name_bind } for pid=384038 comm="httpd" src=12345 scontext=system_u:system_r:httpd_t:s0 tcontext=system_u:object_r:unreserved_port_t:s0 tclass=tcp_socket permissive=0

Demo – httpd port

- # semanage port -l | grep http

  http_cache_port_t tcp 8080, 8118, 8123, 10001-10010

  http_cache_port_t udp 3130

  http_port_t tcp 80, 81, 443, 488, 8008, 8009, 8443, 9000

  pegasus_http_port_t tcp 5988

  pegasus_https_port_t tcp 5989

-

Demo – httpd port

- # semanage port -a -t http_port_t -p tcp 12345
- # semanage port -l | grep http_port_t

  http_port_t tcp 12345, 80, 81, 443, 488, 8008, 8009, 8443, 9000

  pegasus_http_port_t tcp 5988

  # systemctl restart httpd
- # systemctl status httpd
  - httpd.service - The Apache HTTP Server

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

    Active: active (running) since Mon 2020-06-29 16:59:46 HKT; 3s ago

fedora

Unconfined processes

- Unconfined processes runs in unconfined domain
  - unconfined_service_t for services
  - kernel_t for kernel process
  - unconfined_t for services executed by unconfined user
- Default policy allows almost all access for unconfined processes

Demo – unconfined httpd

- # ls -Z $(which httpd)
  - system_u:object_r:httpd_exec_t:s0 /usr/sbin/httpd
- # chcon -t bin_t $(which httpd)
- # ls -Z $(which httpd)
  - system_u:object_r:bin_t:s0 /usr/sbin/httpd
- # systemctl start httpd
- # ps -eZ | grep httpd
  - system_u:system_r:**unconfined_service_t**:s0 2535 ? 00:00:00 httpd
- # curl http://192.168.130.10/test.html
- Testpage

fedora

Users

- Each Linux user is mapped to an SELinux user using SELinux policy.
- # semanage login -l
  - Login Name SELinux User MLS/MCS Range Service
  - __default__ unconfined_u s0-s0:c0.c1023 *
  - root unconfined_u s0-s0:c0.c1023 *
- By default users are mapped to __default__
- Confined and unconfined Linux users are both subject to executable and writeable memory checks, and are also restricted by MCS or MLS.

Users

- The SELinux policy can also define a transition from a confined user domain to its own target confined domain, in this way, the confined user's privileges are limited
- Commonly seen user/role/domain
-

User	Role	Domain	X Window System	su or sudo	Execute in home directory and /tmp/ (default)	Networking
sysadm_u	sysadm_r	sysadm_t	yes	**su and sudo**	yes	yes
staff_u	staff_r	staff_t	yes	only sudo	yes	yes
user_u	user_r	user_t	yes	no	yes	yes
guest_u	guest_r	guest_t	no	no	no	no
xguest_u	xguest_r	xguest_t	yes	no	no	Firefox only

Boolean

- Parts of SELinux policy which can be changed at run time
- # semanage boolean -l | grep samba | head
  - ...
  - samba_create_home_dirs (off , off) Allow samba to create new home directories (e.g. via PAM)
  - ...

Boolean

- # getsebool samba_enable_home_dirs
  - samba_enable_home_dirs --> off
- # setsebool samba_enable_home_dirs on
- # getsebool samba_enable_home_dirs
  - samba_enable_home_dirs --> on

Demo - boolean

- # cd /var/ftp/pub/
- # chmod 777 .
- # grep -v ^# /etc/vsftpd/vsftpd.conf | grep anon
  - anonymous_enable=YES
  - anon_upload_enable=YES

Demo - boolean

- ftp> cd pub
- 250 Directory successfully changed.
- ftp> put 30_smartcard_access.rules
- local: 30_smartcard_access.rules remote: 30_smartcard_access.rules
- 227 Entering Passive Mode (192,168,130,10,39,253).
- 553 Could not create file.

Demo - boolean

- # ausearch -m AVC -ts today | tail
  - time->Wed May 6 23:00:17 2020
  - type=AVC msg=audit(1588777217.614:1011): avc: denied { write } for pid=20180 comm="vsftpd" name="pub" dev="dm-0" ino=2367528 scontext=system_u:system_r:ftpd_t:s0-s0:c0.c1023 tcontext=system_u:object_r:public_content_t:s0 tclass=dir permissive=0

Demo - boolean

- # semanage boolean -l | grep ftp
  - ....
  - ftpd_anon_write (on , off) Determine whether ftpd can modify public files used for public file transfer services. Directories/Files must be labeled public_content_rw_t.
  - ....

Demo - boolean

- # setsebool ftpd_anon_write on
- # chcon -t public_content_rw_t .
  - ftp> put 30_smartcard_access.rules
  - local: 30_smartcard_access.rules remote: 30_smartcard_access.rules
  - 227 Entering Passive Mode (192,168,130,10,41,136).
  - 150 Ok to send data.
  - 226 Transfer complete.
  - 423 bytes sent in 3.8e-05 secs (11131.58 Kbytes/sec)

Modular policy

- SELinux Type Enforcement policy can be managed in a modular basis.
- System administrator can add their own module for customized access

Demo - audit2allow

- # echo Testpage > test.html
- # chcon -t cupsd_var_run_t test.html
- # ls -Z
  - unconfined_u:object_r:admin_home_t:s0 test.html
- # curl http://192.168.130.10/test.html -I
  - HTTP/1.1 403 Forbidden
  - Date: Thu, 07 May 2020 09:36:05 GMT
  - Server: Apache/2.4.41 (Fedora)
  - Content-Type: text/html; charset=iso-8859-1

Demo - audit2allow

- # ausearch -m AVC
  - type=AVC msg=audit(1588844837.810:213): avc: denied { read } for pid=1212 comm="httpd" name="test.html" dev="dm-0" ino=2493829 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:admin_home_t:s0 tclass=file permissive=1
  - type=AVC msg=audit(1588844837.810:214): avc: denied { open } for pid=1212 comm="httpd" path="/var/www/html/test.html" dev="dm-0" ino=2493829 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:admin_home_t:s0 tclass=file permissive=1
  - type=AVC msg=audit(1588844837.810:215): avc: denied { map } for pid=1212 comm="httpd" path="/var/www/html/test.html" dev="dm-0" ino=2493829 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:admin_home_t:s0 tclass=file permissive=1
  - time->Thu May 7 20:05:20 2020
  - type=AVC msg=audit(1588853120.369:284): avc: denied { getattr } for pid=1277 comm="httpd" path="/var/www/html/test.html" dev="dm-0" ino=2493829 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:admin_home_t:s0 tclass=file permissive=0

Demo - audit2allow

- With all needed permission
- # cat auditlog | audit2why
  - type=AVC msg=audit(1588844313.867:201): avc: denied { getattr } for pid=728 comm="httpd" path="/var/www/html/test.html" dev="dm-0" ino=2493829 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:admin_home_t:s0 tclass=file permissive=0
  - ...
  - Was caused by:
  - Missing type enforcement (TE) allow rule.
  - You can use audit2allow to generate a loadable module to allow this access.

fedora

Demo - audit2allow

- # cat auditlog | audit2allow | grep -v ^$
  - module myhttpd 1.0;
  - require {
  - type httpd_t;
  - type admin_home_t;
  - class file { getattr map open read };
  - }
  - #============= httpd_t ==============
  - #!!!! This avc can be allowed using the boolean 'domain_can_mmap_files'
  - allow httpd_t admin_home_t:file map;
  - allow httpd_t admin_home_t:file { getattr open read };

Demo - audit2allow

- # cat auditlog | audit2allow -a -M myhttpd
  - ****************** IMPORTANT **********************
  - To make this policy package active, execute:
  -
  - semodule -i myhttpd.pp
- # semodule -i myhttpd.pp
- # curl http://192.168.130.10/test.html
  - Testpage
- # semodule -r myhttpd

Transition

- How does a file/process get into a certain context?
  - Inherit from parent process
  - When a process executed another file with certain context which matches the predefined rule
    - This is called a transition rule

Transition

- # sesearch -T -s motion_t
  - type_transition motion_t abrt_helper_exec_t:process abrt_helper_t;
  - ...
- Domain transition (as the :process class shows)
- A process with motion_t executed another process whose context is abrt_helper_exec_t will result in a running process with abrt_helper_t.

Transition

- # sesearch -T -s motion_t
  - type_transition motion_t var_t:dir motion_data_t;
  - type_transition motion_t var_t:file motion_data_t;
  - ...
- Object transition (as the file/dir class shows)
- A process with motion_t creates a file in the directory of var_t will result in a new file with motion_data_t.

Demo – transition rule

- # ausearch -ts recent -m AVC,USER_AVC
- ......
- time->Sat Feb 1 22:04:57 2020
- type=AVC msg=audit(1580565897.496:1797): avc: denied { write } for pid=61249 comm="ml1" name="camera" dev="md127" ino=882229921 scontext=system_u:system_r:motion_t:s0 tcontext=unconfined_u:object_r:default_t:s0 tclass=dir permissive=0
- ----
- time->Sat Feb 1 22:04:57 2020
- type=AVC msg=audit(1580565897.496:1798): avc: denied { write } for pid=61249 comm="ml1" name="camera" dev="md127" ino=882229921 scontext=system_u:system_r:motion_t:s0 tcontext=unconfined_u:object_r:default_t:s0 tclass=dir permissive=0

Demo – transition rule

- # sesearch -T -s motion_t
- type_transition motion_t abrt_helper_exec_t:process abrt_helper_t;
- type_transition motion_t var_log_t:dir motion_log_t;
- type_transition motion_t var_log_t:file motion_log_t;
- type_transition motion_t var_run_t:dir motion_var_run_t;
- type_transition motion_t var_run_t:file motion_var_run_t;
- **type_transition motion_t var_t:dir motion_data_t;**
- **type_transition motion_t var_t:file motion_data_t;**
- type_transition motion_t zoneminder_exec_t:process zoneminder_t;

Questions?

Contact:

zsun@fedoraproject.org

References

- General introduction
  - https://selinuxproject.org/page/BasicConcepts
  - https://wiki.centos.org/HowTos/SELinux
  - https://fedoraproject.org/wiki/SELinux/Policies
- Detailed introduction
  - https://docs.fedoraproject.org/en-US/Fedora/25/html/SELinux_Users_and_Administrators_Guide/index.html

References

- Hands-on
  - https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/using_selinux/index
- Access control
  - https://people.redhat.com/duffy/selinux/selinux-coloring-book_A4-Stapled.pdf

Slides

fedora