SELinux Kernel Internals and Architecture

James Morris
jmorris@namei.org

FOSS.IN/2005
Bangalore, India
SELinux Overview

• Mandatory Access Control (MAC)
• Policy-flexible
• Kernel mediation of access between subjects and objects
• Several security models
  - Type Enforcement (TE)
  - Role Based Access Control (RBAC)
  - Multilevel Security (MLS)
Interesting Files

Files related to this talk.

Kernel (typically under /usr/src/linux):

  LSM API:
  
  include/linux/security.h
  security/ (top level)

  SELinux:
  
  security/selinux/ (everything under here)
  security/selinux/hooks.c (especially)

Policy (Fedora-specific):

  /etc/selinux/targeted/

  Install package selinux-policy-targeted-sources (or equivalent).
Object Labeling

• Important objects in the OS are labeled
  - Processes, files, inodes, superblocks etc.
• Files persistently labeled via extended attributes
• Labels are called security contexts
• They contain all SELinux security information:

```
user_u:object_r:tmp_t:unclassified
```

Identity  Role  Type  MLS Label
SELinux Policy

- Labeling rules
  - Describe how objects are to be labeled
- Access rules
  - Describe how subjects access objects (and subjects)
- Compiled into binary form and loaded into kernel
- Enforced by the kernel
Kernel Hooks

- SELinux hooks into security-critical operations
  - System calls, network flow etc.
- Linux Security Modules (LSM)
- Netfilter
Hook Example – LSM

```
fs/ioctl.c:
asmlinkage long sys_ioctl(unsigned int fd, unsigned int cmd, unsigned long arg) {
    struct file * filp;
    int error = -EBADF;
    int fput_needed;

    filp = fget_light(fd, &fput_needed);
    if (!filp)
        goto out;

    error = security_file_ioctl(filp, cmd, arg);
    if (error)
        goto out_fput;

    error = vfs_ioctl(filp, fd, cmd, arg);
    out_fput:
        fput_light(filp, fput_needed);
    out:
        return error;
}
```
Hook Example - SELinux

security/selinux/hooks.c

static int selinux_file_ioctl(struct file *file, unsigned int cmd,
                               unsigned long arg)
{
    int error = 0;

    switch (cmd) {
        case FIONREAD: case FIBMAP: case FIGETBSZ:
            case EXT2_IOC_GETFLAGS: case EXT2_IOC_GETVERSION:
                error = file_has_perm(current, file, FILE__GETATTR);
                break;
        [...] 

        case EXT2_IOC_SETFLAGS: case EXT2_IOC_SETVERSION:
            error = file_has_perm(current, file, FILE__SETATTR);
            break;
        [...] 

        default:
            error = file_has_perm(current, file, FILE__IOCTL);
    }

    return error;
}
Hook Example - Policy

Domain of sysadm_t type is allowed to perform “getattr” on object of type shadow_t and class “file”:

allow sysadm_t shadow_t:file getattr;

Types:

# ls -Z /etc/shadow
-r-------- root root system_u:object_r:shadow_t /etc/shadow

# id -Z
root:system_r:sysadm_t
Logging

• AVC denials are logged by default
• Integrated with CAPP Audit subsystem
• auditallow
• auditdeny
**selinuxfs**

- Mechanism to control and monitor SELinux

```
# tree /selinux
/selinux
|-- access
  |-- avc
    |-- cache_stats
    |-- cache_threshold
    `-- hash_stats
  `-- booleans
    |-- NetworkManager_disable_trans
    `-- allow_execmem
[...]`
  |-- checkreqprot
  |-- commit_pending_bools
  |-- context
  |-- create
  |-- disable
  |-- enforce
  |-- load
  |-- member
  |-- mls
  |-- null
  |-- policyvers
  |-- relabel
  `-- user
```
Procattr API

- Refers to /proc/[PID]/attr
- Per-task extended security API
- Can be re-used by other security modules

Under SELinux:

```
# tree /proc/self/attr

/proc/self/attr
|-- current
|-- exec
`-- fscreate
   `-- prev
```
Netlink Notifications

- Netlink sockets are for kernel-user communication
- SELinux sends event notifications to userspace
  - Policy load with serial number
  - Set enforcing mode
- Synchronize state with userspace security servers
  - DBUS, X
Resources

• /usr/src/linux/security/selinux

• NSA SELinux Pages
  http://www.nsa.gov/selinux/

• Mailing Lists (see above)

• IRC: irc.freenode.net #selinux

• SELinux Symposium 2006 (Feb/Mar)
  http://www.selinux-symposium.org/