Ministerstvo financií
Slovenskej republiky

EFQM
Recognised for excellence
5 star - 2012

Pokročilé bezpečnostné mechanizmy v OS Linux

RNDr. Jaroslav Janáček, PhD.
2014

KPMG
cutting through complexity™

Obsah

- Úvod
  - základný bezp. model Linux-u
- Doplnkové atribúty
- ACL
  - rozšírenie základných možností nastavenie prístupových práv
- Capabilities
  - minimalizácia privilégií procesov („rozmenenie root-a na drobné“)
- SELinux
  - DTE, RBAC, MLS/MCS

Základný bezp. model

- subjekty riadenia prístupu
  - používatelia, skupiny
  - procesy vykonávajú operácie v mene používateľa
- objekty riadenia prístupu
  - objekty súborového systému
  - zdieľaná pamäť a iné zdroje
  - procesy (posielanie signálov, ptrace)
  - globálny stav systému

Atribúty procesu

- používateľ
  - reálne (real), efektívne (effective), uložené (saved), súborové (filesystem) UID
- skupina
  - reálne (real), efektívne (effective), uložené (saved), súborové (filesystem) GID
  - zoznam doplnkových GID

Atribúty procesu

- reálne UID/GID
  - identifikuje vlastníka procesu – používateľa, ktorý ho spustil
- efektívne UID/GID
  - identifikuje používateľa/skupinu, v mene ktorého proces práve vykonáva operácie (iné ako súborové)
- súborové UID/GID
  - identifikuje používateľa/skupinu, v mene ktorého proces práve vykonáva súborové operácie
  - väčšinou rovnaké ako efektívne UID/GID

Atribúty objektu súborového systému

- vlastník (owner)
- skupina (group)
- prístupové práva (permission bits)
  - pre vlastníka, pre skupinu, pre ostatných
  - read, write, execute/search
  - set-UID, set-GID, sticky bit

Atribúty objektu súborového systému

```
$ ls -la
total 24
drwxr-xr-x  3 jerry users  4096 Oct  4 18:46 .
drwxrwxrwt 15 root  root  12288 Oct  4 18:46 ..
drwxr-xr-x  2 jerry users  4096 Oct  4 18:46 adresar
-rw-r--r--  1 jerry users     5 Oct  4 18:46 subor
```

typ
práva
vlastník
skupina

Prístupové práva

- read
  - čítanie súboru
  - čítanie adresára (zoznamu objektov)
- write
  - zápis do súboru
  - zmena obsahu adresára (vytvorenie/zmazanie/premenovanie objektu)
- execute/search
  - spustenie programu zo súboru
  - použitie adresára v ceste k objektu

Prístupové práva

- ak vlastník objektu = FSUID procesu
  - použijú sa práva pre vlastníka
- inak, ak skupina objektu = FSGID procesu alebo niektorej doplnkovej skupine procesu
  - použijú sa práva pre skupinu
- inak
  - použijú sa práva pre ostatných

Prístupové práva

- ak FSUID = 0
  - nekontrolujú sa práva pre čítanie, zápis a použitie adresára v ceste
  - spustenie programu je povolené, ak je povolené aspoň pre niekoho

Prístupové práva

- číselne vyjadrené v osmičkovej sústave
  - 1 číslica pre vlastníka/skupinu/ostatných
  - 4 = read, 2 = write, 1 = execute/search
    - 640 = `rw- r-- ---`
    - 751 = `rwx rx- --x`

Prístupové práva

- zmena prístupových práv
  - môže len vlastník (alebo root)
  - `chmod práva objekt ...`
    - práva ako číslo v osmičkovej sústave
  - `chmod KomuOpPráva[,...] objekt ...`
    - Komu: **u** – vlastník, **g** – skupina, **o** – ostatní, **a**=ugo
    - Op: **=** - nastaviť, **+** - pridať, **-** - odobrať
    - Práva: **r**, **w**, **x**, **X** (X=x, ak adresár alebo už je aspoň 1 x)
  - `chmod -R ...`
    - aplikuje rekurzívne na celý podstrom

Prístupové práva

```
$ chmod 600 a b
$ chmod u=rwx,go= d
$ ls -l
total 12
-rw------- 1 jerry jerry    6 Oct 13 10:01 a
-rw------- 1 jerry jerry    6 Oct 13 10:02 b
drwx------ 2 jerry jerry 4096 Oct 13 10:02 d
$ chmod u+x b
$ ls -l
total 12
-rw------- 1 jerry jerry    6 Oct 13 10:01 a
-rwx------ 1 jerry jerry    6 Oct 13 10:02 b
drwx------ 2 jerry jerry 4096 Oct 13 10:02 d
$ chmod go+X *
$ ls -l
total 12
-rw------- 1 jerry jerry    6 Oct 13 10:01 a
-rwx--x--x 1 jerry jerry    6 Oct 13 10:02 b
drwx--x--x 2 jerry jerry 4096 Oct 13 10:02 d
```

Prístupové práva

- zmena vlastníka
  - len root
  - `chown [-R] vlastník[:skupina] objekt ...`
- zmena skupiny
  - vlastníka len na skupinu, ktorej je členom
  - root na akúkoľvek
  - `chgrp [-R] skupina objekt ...`

Prístupové práva

- pri vytvorení nového objektu
  - vlastník = súborové UID procesu
  - skupina = súborové GID procesu alebo skupina rodičovského adresára
  - práva – určené vytvárajúcim procesom (typicky 666 pre súbory, 777 pre adresáre) a parametrom **umask** – práva nastavené v umask sa odstránia
    - umask 022 – ostatným a skupine sa vypne write
    - umask 077 – ostatným a skupine sa vypne všetko

Prístupové práva

- Set-UID bit
  - pri spustení programu sa uložené, efektívne a súborové UID nastavia na UID vlastníka súboru
    - proces beží s inými právami ako jeho vlastník
  - `chmod **u+s** súbor`
  - `chmod **4**755 súbor`

```
$ ls -la /usr/bin/passwd
-rw**s**r-xr-x 1 root root 34740 Feb 15  2011 /usr/bin/passwd
```

Prístupové práva

- Set-GID bit na súboroch
  - pri spustení programu sa uložené, efektívne a súborové GID nastavia na skupinu súboru
  - `chmod` **`g+s`** `súbor`
  - `chmod` **`2`**`755 súbor`

```
$ ls -la /usr/bin/crontab
-rwxr-sr-x 1 root crontab 30248 Dec 19  2010 /usr/bin/crontab
```

Prístupové práva

- Set-GID bit na adresároch
  - pri vytvorení nového objektu sa jeho skupina nastaví na skupinu rodičovského adresára
  - v prípade podadresára sa nastaví aj Set-GID bit

```
$ mkdir projekt
$ chgrp projekt projekt
$ chmod g+s projekt
$ touch projekt/subor
$ mkdir projekt/adresar
$ ls -la projekt
total 12
drwxr-sr-x 3 jerry projekt 4096 Oct  4 19:51 .
drwxr-xr-x 3 jerry jerry   4096 Oct  4 19:50 ..
drwxr-sr-x 2 jerry projekt 4096 Oct  4 19:51 adresar
-rw-r--r-- 1 jerry projekt    0 Oct  4 19:51 subor
```

Prístupové práva

- Sticky bit na adresároch
  - pridáva ďalšie obmedzenie pri vymazávaní a premenovávaní objektu v adresári – môže len
    - vlastník objektu
    - vlastník adresára
  - `chmod 1777 adresár`
  - `chmod +t adresár`

```
$ ls -lad /tmp
drwxrwxrwt 15 root root 12288 Oct  4 20:17 /tmp
```

Zmeny UID/GID

- ak efektívne UID = 0
  - proces si môže nastaviť všetky UID, GID a zoznam doplnkových skupín na ľubovoľné hodnoty
- inak
  - proces môže UID/GID nastavovať len na aktuálnu hodnotu niektorého UID/GID
    - typicky mení efektívne UID/GID na reálne alebo uložené
- vždy pri zmene efektívneho UID/GID sa zmení aj FSUID/FSGID
  - FSUID/FSGID je možné nastaviť aj samostatne

Doplnkové atribúty

- append only (a)
  - do súboru je možné zapisovať len spôsobom „pridávanie na koniec“
  - nie je ho možné vymazať, premenovať, zmeniť práva, vytvoriť naň (hard) link
- immutable (i)
  - súbor nie je možné zmeniť, zmazať, premenovať, zmeniť práva, ani naň vytvoriť (hard) link
- vzťahujú sa aj na root-a
  - ale ten ich môže zmeniť

Doplnkové atribúty

- zobrazenie
  - `lsattr [-R] [objekt ...]`
  - `lsattr -d adresár`
- nastavenie
  - `chattr [-R] OpAtribúty objekt ...`
    - Op: **+** - pridať, **-** - odobrať, **=** - nastaviť
    - Atribúty: **i** – immutable, **a** – append only, ...

Doplnkové atribúty

```
# mkdir adresar
# touch subor
# chattr +a adresar subor
# lsattr
-----a------------- ./adresar
-----a------------- ./subor
# echo 1 > subor
bash: subor: Operation not permitted
# echo 1 >> subor
# rm subor
rm: cannot remove `subor': Operation not permitted
# mv subor subor2
mv: cannot move `subor' to `subor2': Operation not permitted
# chmod 777 subor
chmod: changing permissions of `subor': Operation not permitted
# ln subor subor2
ln: creating hard link `subor2' => `subor': Operation not permitted
```

Doplnkové atribúty

```
# touch adresar/subor
# mkdir adresar/podadresar
# mv adresar/subor adresar/subor2
mv: cannot move `adresar/subor' to `adresar/subor2': Operation not
permitted
# rm adresar/subor
rm: cannot remove `adresar/subor': Operation not permitted
# echo 1 > adresar/subor
# touch adresar/podadresar/subor
# rm adresar/podadresar/subor
# chmod 666 adresar/subor
```

Doplnkové atribúty

```
# chattr =i adresar subor
# lsattr
----i-------------- ./adresar
----i-------------- ./subor
# rm adresar/subor
rm: cannot remove `adresar/subor': Permission denied
# touch adresar/novy
touch: cannot touch `adresar/novy': Permission denied
# rm subor
rm: cannot remove `subor': Operation not permitted
# chmod 666 subor
chmod: changing permissions of `subor': Operation not permitted
# echo 1 >> subor
bash: subor: Permission denied
```

Nedostatky klasických prís t. práv

- príliš hrubé členenie subjektov
  - na pridelenie práv viacerým používateľom je potrebné vytvoriť skupinu,
  - nie je možné prideliť rôzne práva viacerým skupinám/používateľom
- nie je možné definovať práva pre nové objekty v adresári
  - je možné len vynútiť dedenie skupiny
  - práva je možné ovplyvniť len pomocou umask

Access Control List (ACL)

- rozširuje typy subjektov, pre ktoré je možné definovať práva:
  - vlastník
  - priradená skupina
  - ostatní
  - konkrétny používateľ
  - konkrétna skupina
- definuje default práva pre nové objekty v adresári

Access Control List (ACL)

- ku každému objektu súborového systému je priradený zoznam položiek
  - typ položky,
  - identifikátor používateľa/skupiny,
  - práva (read, write, execute/use)

Access Control List (ACL)

- typy položiek
  - ACL_USER_OBJ – práva pre vlastníka
  - ACL_USER – práva pre určeného používateľa
  - ACL_GROUP_OBJ – práva pre skupinu objektu
  - ACL_GROUP – práva pre určenú skupinu
  - ACL_OTHER – práva pre ostatných
  - ACL_MASK – maximálne práva pre ACL_USER, ACL_GROUP a ACL_GROUP_OBJ
- 1, 0 a viac, 0 – 1 (1 ak exist. zelená)

Access Control List (ACL)

- Vyhodnotenie práv
  - ak FSUID procesu = UID vlastníka, použijú sa práva z ACL_USER_OBJ
  - inak, ak FSUID procesu = identifikátor v niektorej položke ACL_USER, použije sa tá po AND s ACL_MASK
  - inak, sa nájdu všetky položky typu ACL_GROUP_OBJ a ACL_GROUP, ktoré zodpovedajú FSGID alebo doplnkovej skupine procesu, spraví sa OR ich práv, potom AND s ACL_MASK a výsledok sa použije
  - ak také položky neexistujú, použije sa ACL_OTHER

Access Control List (ACL)

- Vzťah k tradičným UNIX právam
  - práva pre vlastníka zodpovedajú ACL_USER_OBJ
  - práva pre ostatných zodpovedajú ACL_OTHER
  - práva pre skupinu zodpovedajú
    - ACL_GROUP_OBJ, ak neexistuje ACL_MASK
    - ACL_MASK, ak existuje ACL_MASK
  - nastavenie tradičných práv spôsobí aj zmenu ACL a naopak

Access Control List (ACL)

- Textová reprezentácia práv – dlhá verzia
  - jedna položka na riadok
  - typ:id:prava
  - typ
    - user – ACL_USER alebo ACL_USER_OBJ (ak id=““)
    - group – ACL_GROUP alebo ACL_GROUP_OBJ (ak id=““)
    - mask – ACL_MASK
    - other – ACL_OTHER
  - prava: 3 znaky z {r, w, x, -}

Access Control List (ACL)

- Textová reprezentácia – krátka verzia
  - položky oddelené čiarkou
  - typ môže byť skrátený na u, g, m, o
  - pomlčky v právach sa môžu vynechať (ale vždy musí zostať aspoň 1 znak)

Zobrazenie ACL

- `getfacl [-R] objekt`
  - -R – rekurzívne celý podstrom
  - vypíše ACL zadaných pre zadané objekty v dlhej textovej forme

Zobrazenie ACL

```
# umask
0022
# mkdir adresar
# ls -lad adresar
drwxr-xr-x 2 root root 4096 Oct  5 13:58 adresar

# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
group::r-x
other::r-x
```

Nastavenie ACL

- `setfacl [-R] -m acl objekt`
  - -R – rekurzívne pre celý podstrom
  - acl – špecifikácia v krátkej textovej forme
  - pridá/zmení uvedené položky ACL
- `setfacl [-R] -x acl objekt`
  - zmaže položky z ACL
  - acl – špecifikácia bez práv
- `setfacl [-R] -b objekt`
  - zmaže všetky ACL

Nastavenie ACL

- ďalšie parametre setfacl
  - -n – neprepočítať masku
    - normálne setfacl, ak nemá špecifikovanú masku explicitne, vypočíta a nastaví masku na hodnotu OR všetkých ACL_USER, ACL_GROUP a ACL_GROUP_OBJ
- `setfacl --restore=súbor`
  - nastaví ACL podľa obsahu súboru, ktorý je vo formáte výstupu z getfacl

Nastavenie ACL

```
# setfacl -m u:jerry:rwx,g:users:r,g::-,o::- adresar
# ls -lad adresar
drwxrwx---+ 2 root root 4096 Oct  5 13:58 adresar
# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
user:jerry:rwx
group::---
group:users:r--
mask::rwx
other::---
```

Nastavenie ACL

```
# chmod g-w adresar
# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
user:jerry:rwx          #effective:r-x
group::---
group:users:r--
mask::r-x
other::---
```

Nastavenie ACL

```
# setfacl -x g:users adresar
# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
user:jerry:rwx
group::---
mask::rwx
other::---
```

Nastavenie ACL

```
# setfacl -m g:users:rwx,m::rx adresar
# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
user:jerry:rwx          #effective:r-x
group::---
group:users:rwx         #effective:r-x
mask::r-x
other::---
```

Nastavenie ACL

```
# setfacl -n -x g:users adresar
# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
user:jerry:rwx          #effective:r-x
group::---
mask::r-x
other::---
```

Nastavenie ACL

```
# setfacl -b adresar
# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
group::---
other::---

# ls -lad adresar
drwx------ 2 root root 4096 Oct  5 13:58 adresar
```

Default ACL

- Adresár môže mať špecifikované default ACL pre nové objekty
  - ACL pre nový objekt vznikne z default ACL adresára
  - z položiek zodpovedajúcich štandardným UNIX právam sa odstránia práva, ktoré neboli pri vytvorení požadované
- Ak adresár nemá default ACL
  - ACL nového objektu bude obsahovať položky zodpovedajúce štandardným právam nastavené podľa požadovaných práv a umask

Nastavenie default ACL

- `setfacl -d -m acl adresar`
  - acl ako pri normálnych ACL
- `setfacl -m defacl adresar`
  - kde položky defacl obsahujú prefix default: alebo d:
- `setfacl -d -x acl adresar`
- `setfacl -x defacl adresar`
- `setfacl -k adresar`
  - odstráni default ACL

Nastavenie default ACL

```
# setfacl -d -m g:users:rwx,u:jerry:rwx adresar
# getfacl adresar
# file: adresar
# owner: root
# group: root
user::rwx
group::---
other::---
default:user::rwx
default:user:jerry:rwx
default:group::---
default:group:users:rwx
default:mask::rwx
default:other::---
```

Nastavenie default ACL

```
# umask
0022
# touch adresar/subor
# mkdir adresar/podadresar
# getfacl adresar/subor
# file: adresar/subor
# owner: root
# group: root
user::rw-
user:jerry:rwx          #effective:rw-
group::---
group:users:rwx         #effective:rw-
mask::rw-
other::---
```

Nastavenie default ACL

```
# getfacl adresar/podadresar/
# file: adresar/podadresar/
# owner: root
# group: root
user::rwx
user:jerry:rwx
group::---
group:users:rwx
mask::rwx
other::---
default:user::rwx
default:user:jerry:rwx
default:group::---
default:group:users:rwx
default:mask::rwx
default:other::---
```

Príklad použitia ACL

- Chceme vytvoriť adresár projekt s nasledujúcimi vlastnosťami:
  - skupina projekt má mať plné práva na celý podstrom
  - skupina staff má mať práva na čítanie a spúšťanie/použitie adresára
  - používateľ jerry má mať plné práva
  - ktokoľvek má mať právo čítať, spúšťať/použiť v podstrome projekt/pub

Príklad použitia ACL

```
# mkdir projekt
# chgrp projekt projekt
# chmod g+s projekt
# setfacl -m g::rwx,g:staff:rx,u:jerry:rwx,o:x projekt
# setfacl -d -m g::rwx,g:staff:rx,u:jerry:rwx,o:- projekt
# ls -lad projekt
drwxrws--x+ 2 root projekt 4096 Oct  5 17:54 projekt
```

Príklad použitia ACL

```
# getfacl projekt
# file: projekt
# owner: root
# group: projekt
# flags: -s-
user::rwx
user:jerry:rwx
group::rwx
group:staff:r-x
mask::rwx
other::--x
default:user::rwx
default:user:jerry:rwx
default:group::rwx
default:group:staff:r-x
default:mask::rwx
default:other::---
```

Príklad použitia ACL

```
# cd projekt; mkdir pub; getfacl pub
# file: pub
# owner: root
# group: projekt
# flags: -s-
user::rwx
user:jerry:rwx
group::rwx
group:staff:r-x
mask::rwx
other::---
default:user::rwx
default:user:jerry:rwx
default:group::rwx
default:group:staff:r-x
default:mask::rwx
default:other::---
```

Príklad použitia ACL

```
# setfacl -m o::rx,d:o::rx pub
# getfacl pub
# file: pub
# owner: root
# group: projekt
# flags: -s-
user::rwx
user:jerry:rwx
group::rwx
group:staff:r-x
mask::rwx
other::r-x
default:user::rwx
default:user:jerry:rwx
default:group::rwx
default:group:staff:r-x
default:mask::rwx
default:other::r-x
```

Príklad použitia ACL

```
# mkdir subdir pub/subdir
# touch file subdir/file pub/file pub/subdir/file
# ls -ld file subdir subdir/file
-rw-rw----+ 1 root projekt    0 Oct  5 18:05 file
drwxrws---+ 2 root projekt 4096 Oct  5 18:05 subdir
-rw-rw----+ 1 root projekt    0 Oct  5 18:05 subdir/file
# ls -ld pub/file pub/subdir pub/subdir/file
-rw-rw-r--+ 1 root projekt    0 Oct  5 18:05 pub/file
drwxrwsr-x+ 2 root projekt 4096 Oct  5 18:05 pub/subdir
-rw-rw-r--+ 1 root projekt    0 Oct  5 18:05 pub/subdir/file
```

Príklad použitia ACL

```
# getfacl file
# file: file
# owner: root
# group: projekt
user::rw-
user:jerry:rwx          #effective:rw-
group::rwx              #effective:rw-
group:staff:r-x         #effective:r--
mask::rw-
other::---
```

Príklad použitia ACL

```
# getfacl subdir
# file: subdir
# owner: root
# group: projekt
# flags: -s-
user::rwx
user:jerry:rwx
group::rwx
group:staff:r-x
mask::rwx
other::---
default:user::rwx
default:user:jerry:rwx
default:group::rwx
default:group:staff:r-x
default:mask::rwx
default:other::---
```

Príklad použitia ACL

```
# getfacl pub/file
# file: pub/file
# owner: root
# group: projekt
user::rw-
user:jerry:rwx          #effective:rw-
group::rwx              #effective:rw-
group:staff:r-x         #effective:r--
mask::rw-
other::r--
```

Príklad použitia ACL

```
# getfacl pub/subdir
# file: pub/subdir
# owner: root
# group: projekt
# flags: -s-
user::rwx
user:jerry:rwx
group::rwx
group:staff:r-x
mask::rwx
other::r-x
default:user::rwx
default:user:jerry:rwx
default:group::rwx
default:group:staff:r-x
default:mask::rwx
default:other::r-x
```

Podpora ACL

- balík acl
- ACL sú ukladané v „extended attributes“ - vyžadujú podporu súborového systému
  - podporované napr. v ext2/3/4, reiserfs
  - nepodporované napr. v vfat, ntfs
- súborový systém musí byť pripojený s voľbou „acl“

```
# <file system> <mount point>   <type>  <options>   <dump>  <pass>
/dev/sda1       /               ext3    acl         0       1
```

Capabilities

- V tradičnom bezp. modeli Linuxu
  - na proces s FSUID = 0 (root) sa nevzťahuje riadenie prístupu k súborovému systému
  - privilegované operácie môže vykonávať len proces s efektívnym UID = 0 (root)
    - konfigurácia globálnych systémových parametrov
    - konfigurácia siete, prístup k nižším vrstvám siete, počúvanie na TCP/UDP portoch < 1024
    - posielanie signálov cudzím procesom, …

Capabilities

- Dôsledkom „koncentrácie moci“ je
  - veľa procesov musí bežať s efektívnym UID = 0
    - v lepšom prípade iba dočasne, kým spravia privilegovanú operáciu
  - veľa programov je Set-UID-root
    - v lepšom prípade sa opäť práv po inicializácii vzdajú
  - zneužiteľná chyba v takom programe má neobmedzený dopad na systém

Capabilities

- Riešenie
  - rozdeliť oprávnenia na menšie – capabilities
  - umožniť prideľovanie jednotlivých oprávnení programom
- Výhody
  - obmedzenie dopadu v prípade chyby
- Nevýhody
  - môže byť potrebné (pre dosiahnutie ideálneho stavu) upraviť programy

Capabilities pre riadenie prístupu k súborovému systému

- CAP_CHOWN
  - oprávnenie zmeniť vlastníka objektu
- CAP_DAC_OVERRIDE
  - oprávnenie na čítanie/zápis akéhokoľvek súboru/adresára, použitie akéhokoľvek adresára, spustenie akéhokoľvek programu, ktorý môže niekto spustiť
- CAP_DAC_READ_SEARCH
  - oprávnenie na čítanie akéhokoľvek súboru/adresára a použitie adresára
- CAP_FOWNER
  - vykonanie operácie, ktorá vyžaduje vlastníctvo objektu (chmod, sticky bit)
- CAP_FSETID
  - nastavenie Set-GID bitu na súbore cudzej skupiny, nezmazanie Set-UID a Set-GID pri zápise
- CAP_LINUX_IMMUTABLE
  - nastavenie IMMUTABLE a APPEND_ONLY atribútu
- CAP_MKNOD
  - vytváranie blokových a znakových zariadení

Capabilities pre zmenu identity, prístup k procesom, ...

- CAP_SETUID
  - nastavenie UID na ľubovoľné hodnoty
- CAP_SETGID
  - nastavenie GID a doplnkových skupín na ľubovoľné hodnoty
- CAP_KILL
  - poslanie signálu ľubovoľnému procesu
- CAP_SETFCAP
  - nastavenie oprávnení pre program
- CAP_SETPCAP
  - niektoré manipulácie s oprávneniami
- CAP_SYS_PTRACE
  - použitie ptrace na ľubovoľný proces
- CAP_SYS_NICE
  - zmena priority procesu

Capabilities pre sieť

- CAP_NET_ADMIN
  - konfigurácia siete (rozhrania, routovanie, firewall, promiskuitný mód, …)
  - povolenie multicastingu, ...
- CAP_NET_BIND_SERVICE
  - počúvanie na privilegovaných portoch (<1024)
- CAP_NET_RAW
  - prístup k RAW paketom

Capabilities pre ďalšie operácie

- CAP_SYS_ADMIN
  - ďalšie administrátorské operácie (napr. mount, privilegované operácie so zariadeniami, …)
- CAP_SYS_BOOT
  - reboot
- CAP_SYS_CHROOT
  - použitie chroot
- CAP_SYS_MODULE
  - zavedenie a odstránenie modulu do/z jadra
- CAP_SYS_RAWIO
  - oprávnenia na I/O operácie
- CAP_SYS_TIME
  - manipulácia s časom
- CAP_SYSLOG
  - konfigurácia logovania z kernelu

Capabilities

- Každý proces má niekoľko množín oprávnení:
  - permitted
    - oprávnenia, ktoré proces môže mať
    - horná hranica pre *effective*
    - horná hranica pre pridávanie do *inheritable* (neplatí, ak má CAP_SETPCAP)
    - proces si môže nejaké oprávnenie odstrániť, ale nie pridať
  - inheritable
    - oprávnenia, ktoré môže zdediť nový program (po execve)
  - effective
    - práve pridelené oprávnenia, ktoré môže proces použiť

Capabilities

- Každý spustiteľný súbor môže mať pridelené oprávnenia:
  - permitted
    - oprávnenia, ktoré proces získa do *permitted* spustením programu
  - inheritable
    - oprávnenia, ktoré sa majú procesu zachovať pri spustení programu
  - effective
    - 1 bit, určuje, či všetky po spustení bude mať proces všetky povolené oprávnenia zapnuté alebo vypnuté

Capabilities

- Proces má navyše
  - bounding_set
    - horná hranica oprávnení, ktoré môže proces získať spustením programu
    - horná hranica pre pridávanie do *inherited*
    - oprávnenia proces môže z *bounding_set* odstraňovať, ak má CAP_SETPCAP, ale nemôže ich pridávať

Capabilities pri spustení programu

- Zmena oprávnení pri spustení programu
  - P'(permitted) = (P(inheritable) & F(inheritable)) | (F(permitted) & P(bounding_set))
  - P'(effective) = ak F(effective) = 1, tak P'(permitted), inak 0
  - P'(inheritable) = P(inheritable)
  - P'(bounding_set) = P(bounding_set)

Capabilities pri spustení programu

- proces získa pri spustení programu možné oprávnenia 2 spôsobmi
  - spúšťaný program má nejaké pridelené
    - tie sa ohraničia *bounding_set* procesu
  - proces aj program majú neprázdny prienik *inheritable*
    - prienik oprávnení proces získa
- žiadne alebo všetky získané možné oprávnenia sa stanú efektívnymi

Capabilities a root

- pri spúšťaní programu, ktorý nemá nastavené žiadne oprávnenia
  - ak reálne alebo efektívne UID = 0, tak F(permitted) a F(inheritable) sa považujú za plné
    - a teda výsledkom je, že
      P'(permitted) = P(bounding_set) | P(inheritable)
  - ak efektívne UID = 0, tak F(effective) = 1, inak 0
    - teda ak efektívne UID = 0, P'(effective) = P'(permitted)
  - teda program spustený v mene roota získa plné oprávnenia limitované len zjednotením P(bounding_set) a P(inheritable)

Capabilities a root

- pri zmene UID
  - ak pôvodne bolo aspoň 1 UID = 0 a po zmene sú všetky !=0, všetky oprávnenia z *permitted* a *effective* sa odstránia
  - ak sa efektívne UID zmenilo z =0 na !=0, všetky oprávnenia z *effective* sa odstránia
  - ak sa efektívne UID zmenilo z !=0 na =0, do *effective* sa skopírujú všetky oprávnenia v *permitted*

Capabilities a root

- špeciálne správanie zohľadňujúce root-a pri spúšťaní programov a pri zmenách UID je možné vypnúť pomocou *secure bits*
  - SECBIT_NOROOT
    - zruší špeciálne správanie pri spúšťaní programu
  - SECBIT_NO_SETUID_FIXUP
    - zruší špeciálne správanie pri zmenách UID
  - SECBIT_KEEP_CAPS
    - neodstráni všetky oprávnenia z *permitted* pri zmene UID

Capabilities a root

- zmena *secure bits*
  - potrebné oprávnenie CAP_SETPCAP
  - je možné ich uzamknúť nastavením
    - SECBIT_NOROOT_LOCKED
    - SECBIT_NO_SETUID_FIXUP_LOCKED
    - SECBIT_KEEP_CAPS_LOCKED
  - uzamknutie je nevratné
    - umožňuje procesu a jeho neskorším potomkom znemožniť získanie oprávnení inak, než spustením programu, ktorý má oprávnenia explicitne nastavené

Capabilities

- Proces si môže meniť oprávnenia
  - z *permitted* môže len odoberať
  - do *inheritable* môže pridať len to, čo má v *permitted* a zároveň v *bounding_set*
    - ak má v *effective* CAP_SETPCAP, tak môže pridať čokoľvek z *bounding_set*
  - *effective* musí byť vždy podmnožinou *permitted*
  - z *bounding_set* môže len odoberať, a len ak má CAP_SETPCAP v *effective*

Capabilities – nastavenie

- `setcap popis|-r subor`
  - -r – odstráni oprávnenia
  - popis: OprOpMn …
    - Opr: zoznam oprávnení oddelených čiarkami (príp. all)
    - Op: =,+,-
      - = vymaže uvedené oprávnenia zo všetkých množín a pridá ich do uvedených
      - + pridá uvedené oprávnenia do uvedených množín
      - - odstráni uvedené oprávnenia z uvedených množín
    - Mn: e, i, p (0 až 3)
    - vždy začína z prázdnych množín

Capabilities – zobrazenie

- `getcap [-r] subor`
  - zobrazí nastavené oprávnenia pre súbor
  - -r – rekurzívne

Capabilities – príklad

```
$ ls -l ping
-rwxr-xr-x 1 root root 31104 Oct  7 13:43 ping
$ ./ping 127.0.0.1
ping: icmp open socket: Operation not permitted


# setcap CAP_NET_RAW+pe ping
# getcap ping
ping = cap_net_raw+ep


$ ./ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_req=1 ttl=64 time=0.049 ms
^C
--- 127.0.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.049/0.049/0.049/0.000 ms
```

Capabilities – použitie

- zníženie oprávnení programov
  - mnohé potrebujú len niektoré oprávnenia
    - napr. CAP_NET_BIND_SERVICE, CAP_NET_RAW
- umožnenie použitia niektorých programov bežným používateľom (alebo vybranej skupine)
  - napr. tcpdump pre administrátorov bez potreby prihlásiť sa ako root

Podpora pre capabilities

- balík libcap2-bin
  - setcap, getcap, capsh
- balík libcap-ng-utils
  - netcap, filecap, pscap, captest

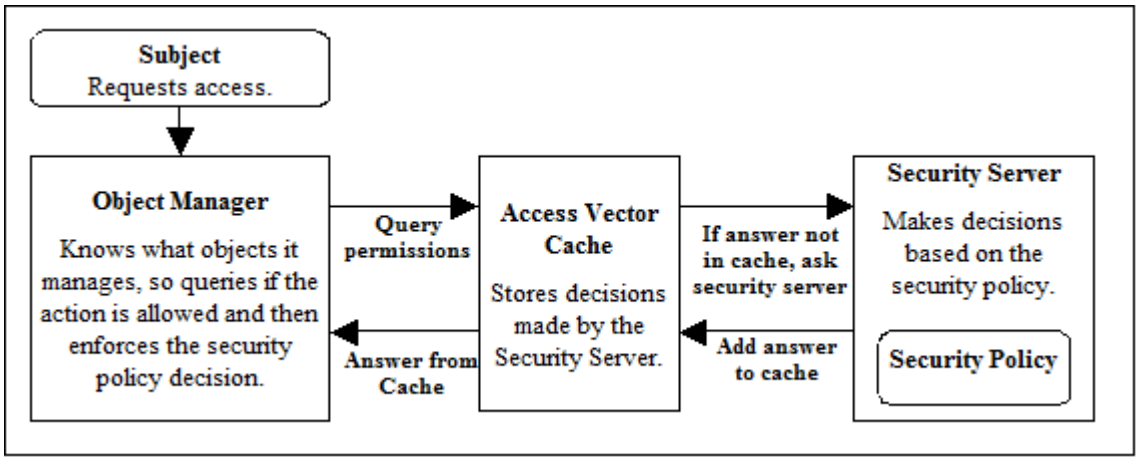
SELinux

- bezpečnostný modul jadra OS Linux
- povinné riadenie prístupu (MAC)
- implementuje
  - variant modelu DTE
  - Role-based Access Control
  - voliteľne MLS (napr. Bell – La Padula alebo Biba)
    - pravidlá sú konfigurovateľné

SELinux – architektúra

- Security Server
  - komponent, ktorý vyhodnocuje pravidlá politiky
- Security Policy
  - politika – súbor pravidiel
- Access Vector Cache
  - cache rozhodnutí pre urýchlenie operácií
- Object Manager
  - systém, ktorý realizuje požiadavky subjektov
  - napr. kernel, ale aj niektoré programové subsytémy

SELinux – architektúra



SELinux RBAC

- **používatelia** majú pridelené **roly**
  - SELinux nepoužíva bežné Linux-ové identity používateľov ale vlastný systém používateľov
    - Linux-oví používatelia majú priradenú SELinux identitu
    - aj viac Linux-ových používateľov môže mať rovnakú SELinux identitu – potom nie sú z pohľadu SELinux-u odlíšení (napr. „bežný používateľ“, „administrátor“)
- roly majú pridelené povolené domény
  - proces vždy beží v mene používateľa konajúceho v niektorej role a môže bežať len v doméne, ktorú má táto rola povolenú

SELinux DTE

- subjekty (procesy) a objekty (napr. objekty súborového systému, procesy ako cieľ operácie, sieťové pakety, ...) systému majú priradený **typ**
  - v prípade subjektov sa nazýva aj **doména**
- rozhodovanie o tom, či sa subjektu povolí operácia s objektom je riadené **politikou**
  - pravidlami na základe typu (domény) subjektu, typu objektu a **triedy** objektu (súbor, adresár, proces, …)
  - obmedzeniami, ktoré musia byť dodržané

SELinux – bezp. kontext

- každý subjekt a objekt má priradený **bezpečnostný kontext (security context)**
  - user:role:type[:mls_range]
    - user – SELinux používateľ
    - role – aktuálna rola
      - objekty majú typicky rolu object_r
    - type – typ/doména
    - mls_range – rozsah (low – high) MLS značiek

SELinux – bezp. kontext

- príklad:
  - user_u:user_r:user_t:s0-s0:c0.c10
    - používateľ user_u
    - rola user_r
    - typ/doména user_t
    - spodná hranica (aktuálna úroveň) MLS rozsahu s0
      - úroveň s0, množina kategórií prázdna
    - horná hranica MLS rozsahu s0:c0.c10
      - úroveň s0, kategórie c0 až c10
      - s0:c0,c2,c3 = kategórie c0, c2 a c3

SELinux – bezp. kontext

- bezp. kontext pre objekty súborového systému
  - uložený v „extended attributes“
  - určený politikou
    - napr. pre súb. systémy bez podpory ext. attr.
  - odvodený od vytvárajúceho procesu
    - pre pseudosúborové systémy (sockfs, tmpfs, devfs, ...)
- bezp. kontext pre subjekty a iné objekty
  - odvodený od vytvárajúceho subjektu

SELinux – bezp. kontext

- zobrazenie

```
admin@debian:~$ id -Z
staff_u:staff_r:staff_t:s0

admin@debian:~$ ps axZ
LABEL                             PID TTY      STAT   TIME COMMAND
staff_u:staff_r:staff_t:s0       4993 pts/1    Ss     0:00 -bash
staff_u:staff_r:staff_t:s0       5158 pts/1    R+     0:00 ps axZ

root@debian:~# ls -laZ /etc/passwd /etc/shadow /bin/bash /bin/cp
-rwxr-xr-x. 1 root root   system_u:object_r:shell_exec_t:s0 941252 Jan  1  2013 /bin/bash
-rwxr-xr-x. 1 root root   system_u:object_r:bin_t:s0       124804 Jan 26  2013 /bin/cp
-rw-r--r--. 1 root root   system_u:object_r:etc_t:s0         1110 Oct  8 18:22 /etc/passwd
-rw-r-----. 1 root shadow system_u:object_r:shadow_t:s0      1091 Oct  8 21:35 /etc/shadow
```

SELinux – politika

- typy a atribúty typov
- povoľovacie a auditovacie pravidlá
- pravidlá pre odvodzovanie typov
- roly a povolené domény
- používatelia a povolené roly
- obmedzenia
- MLS značky
- ...

SELinux – typy a atribúty

- `type id_t,attr1, attr2 ;`
  - definuje typ id_t a priradí mu atribúty attr1 a attr2
- `attribute attr1;`
  - definuje atribút attr1
  - atribúty slúžia na označenie množiny typov
- `typeattribute id_t attr1, attr2 ;`
  - dodatočne typu id_t pridá atribúty

SELinux – povoľovacie pravidlá

- `allow zdroj ciel : trieda prava ;`
  - povolí subjektu v doméne **zdroj** vykonať operácie prava s objektom typu **ciel** a triedy **trieda**
- `dontaudit zdroj ciel : trieda prava ;`
  - nebude generovať auditný záznam o zamietnutí
- `auditallow zdroj ciel : trieda prava ;`
  - bude generovať auditný záznam aj pri povolení
- `neverallow zdroj ciel : trieda prava ;`
  - znemožní, aby také allow pravidlo bolo v politike

SELinux – povoľovacie pravidlá

- typ môže byť určený
  - menom typu alebo atribútu, cieľový môže byť aj `self`
  - môže ich byť viac `{typ1_t typ2_t attr1}`
  - je možné vylúčiť typ z množiny `{attr1 -typ1_t}`
- trieda je určená menom triedy
  - môže ich byť viac `{file dir}`
- práva môžu byť určené
  - menom, ak ich je viac, tak v {}: `{read write search}`
  - `*` (všetky práva danej triedy)
  - `~read` – všetky práva danej triedy okrem uvedeného

SELinux – triedy objektov a práva

- SELinux podporuje veľké množstvo tried objektov
  - v kerneli v súčasnosti cca 50
  - napr. file, dir, filesystem, process, packet, security, capability, ...
- každá trieda má definované práva
  - v `/sys/fs/selinux/class/meno_triedy/perms` je možné vidieť všetky podporované práva danej triedy

SELinux – triedy objektov a práva

```
root@debian:/sys/fs/selinux/class# ls process/perms
dyntransition  getattr     noatsecure  setexec       setsched       sigkill
execheap       getcap      ptrace      setfscreate   setsockcreate  signal
execmem        getpgid     rlimitinh   setkeycreate  share          signull
execstack      getsched    setcap      setpgid       sigchld        sigstop
fork           getsession  setcurrent  setrlimit     siginh         transition

root@debian:/sys/fs/selinux/class# ls file/perms
append        execmod       ioctl       open          relabelto  unlink
audit_access  execute       link        quotaon       rename     write
create        execute_no_trans  lock    read          setattr
entrypoint    getattr       mounton     relabelfrom   swapon

root@debian:/sys/fs/selinux/class# ls dir/perms
add_name      execmod   link    quotaon      remove_name  search   write
append        execute   lock    read         rename   setattr
audit_access  getattr   mounton relabelfrom  reparent     swapon
create        ioctl     open    relabelto    rmdir    unlink

root@debian:/sys/fs/selinux/class# ls filesystem/perms
associate  mount     quotamod     relabelto  transition
getattr    quotaget  relabelfrom  remount    unmount
```

SELinux – triedy objektov a práva

```
root@debian:/sys/fs/selinux/class# ls security/perms/
check_context  compute_member   load_policy  setcheckreqprot
compute_av     compute_relabel  read_policy  setenforce
compute_create  compute_user    setbool      setsecparam

root@debian:/sys/fs/selinux/class# ls capability/perms/
audit_control  ipc_lock     net_bind_service  sys_admin  sys_rawio
audit_write    ipc_owner    net_broadcast     sys_boot   sys_resource
chown          kill         net_raw           sys_chroot sys_time
dac_override   lease        setfcap           sys_module sys_tty_config
dac_read_search  linux_immutable  setgid       sys_nice
fowner         mknod        setpcap           sys_pacct
fsetid         net_admin    setuid            sys_ptrace
```

SELinux – odvodzovanie typov

- `type_transition zdroj ciel : trieda novy ;`
- zmena domény (trieda process)
  - keď proces v doméne **zdroj** spustí súbor typu **ciel**, nový typ procesu bude **novy**
- vytvorenie nového objektu
  - keď proces v doméne **zdroj** vytvorí nový objekt v adresári typu **ciel**, typ nového objektu bude **novy**

SELinux – odvodzovanie typov

- pri zmene domény sú potrebné povolené práva
  - execute medzi pôvodnou doménou a súborom
  - entrypoint medzi novou doménou a súborom
  - transition medzi pôvodnou doménou a novou doménou
- ak sa pri spustení súboru doména nemení, musí mať doména k súboru minimálne práva
  - execute
  - execute_no_trans

SELinux – odvodzovanie typov

- `type_change zdroj ciel : trieda novy ;`
  - slúži podporným programom (napr. pri prihlasovaní používateľa) na určenie typu niektorých objektov, ktorých typ je potrebné prispôsobiť

SELinux – roly

- `role id_r ;`
  - definuje novú rolu
- `role id_r types dom_t ;`
  - definuje novú rolu a/alebo povolí jej doménu `dom_t`
- `role id_r types {typ1 attr2 -typ2} ;`
  - povolených domén môže byť aj viac
- `allow stara_r nova_r ;`
  - povoľuje prechod z jednej roly do druhej
- `role_transition stara_r typ_t nova_r ;`
  - určuje prechod z jednej roly do druhej pri spustení programu

SELinux – používatelia

- `user id_u roles id_r ;`
  - definuje používateľa id_u a povolí mu rolu id_r
- `user id_u roles {id1_r id2_r} ;`
  - definuje používateľa id_u a povolí mu roly id1_r a id2_r
- `user id_u roles id_r level mls_level range mls_range;`
  - pri použití MLS politiky definuje aj defaultnú úroveň a rozsah MLS úrovní pre používateľa

SELinux – obmedzenia

- `constrain trieda prava vyraz ;`
  - pri kontrole práv na objekt danej triedy overuje, či je splnená dodatočná podmienka **vyraz**
  - vyraz je logický výraz, v ktorom je možné porovnávať používateľa/rolu/typ subjektu a objektu navzájom alebo so zoznamom

```
constrain dir_file_class_set { create relabelto relabelfrom }
(
        u1 == u2
        or t1 == can_change_object_identity
);
```

SELinux – MLS značky

- `sensitivity sens_id;`
  - definuje identifikátor MLS úrovne
- `dominance {s0 s1 s2}`
  - definuje usporiadanie úrovní
- `category cat_id;`
  - definuje identifikátor kategórie
- `level sens_id:cat_ids;`
  - priradí úrovni množinu povolených kategórií

SELinux – MLS obmedzenia

- `mlsconstrain trieda prava vyraz ;`
  - pri kontrole práv na objekt danej triedy kontroluje dodatočnú podmienku **vyraz** podobne ako constrain, ale navyše môže testovať dominanciu MLS značiek subjektu a objektu

```
mlsconstrain file { read ioctl lock execute execute_no_trans }
  (( h1 dom h2 ) or ( t1 == mcsreadall ) or ( t2 == domain ));

mlsconstrain file { write setattr append link rename }
  ((( h1 dom h2 ) and ( l1 domby l2 )) or ( t1 == mcswriteall )
or (t2 == mcstrustedobject) or ( t2 == domain ));
```

SELinux – značkovanie portov

- `portcon protocol port_num context`
  - priradí príslušnému portu daný bezp. kontext
  - umožňuje následne definovať v politike, ktoré sieťové porty smie doména použiť

```
portcon tcp 22 system_u:object_r:ssh_port_t:s0
```

SELinux – referenčná politika

- pokrýva veľa bežných služieb a aplikácií
- modulárna
  - aktuálne vyše 300 modulov, 4M riadkov
  - modul typicky pokrýva jednu službu / aplikáciu
  - moduly sa dajú individuálne pridávať a odoberať
  - modul definuje
    - svoje typy, atribúty, roly, pravidlá
    - interface pre iné moduly
      - parametrizovanú šablónu, ktorú môžu iné moduly použiť na prístup k objektom politiky definovaným týmto modulom
    - bezp. kontext pre súbory

SELinux – referenčná politika

- roly
  - user_r – rola pre bežného používateľa
  - staff_r – neprivilegovaná rola pre administrátora
  - sysadm_r – rola pre administrátora
  - system_r – rola pre systémové procesy
  - unconfined_r – rola pre neobmezdeného používateľa
    - táto sa používa v tzv. *targeted* móde, kedy sú obmedzené len systémové služby a niektoré aplikácie, no nie bežní používatelia a root

SELinux – referenčná politika

- používatelia
  - system_u – špeciálny používateľ pre systémové procesy a objekty
    - povolené roly: system_r
  - user_u – bežný používateľ
    - povolené roly: user_r
  - staff_u – používateľ, ktorý si môže zmeniť rolu na administrátora
    - povolené roly: staff_r, sysadm_r
  - sysadm_u – „čistý“ administrátor
    - povolené roly: sysadm_r
  - unconfined_u – neobmedzený používateľ
    - povolené roly: unconfined_r, system_r
  - root – obmedzený root
    - povolené roly: staff_r, sysadm_r, system_r

SELinux – správa používateľov

- mapovanie používateľov Linux – SELinux
  - `semanage login oper [opts] login_name`
    - oper
      - -a – pridanie
      - -m – zmena
      - -d – vymazanie
      - -l – zoznam
      - -D – vymazanie všetkých zmien
    - opts
      - -s selinux_name – SELinux používateľ
      - -r mls_range – rozsah MLS značiek

SELinux – správa používateľov

```
root@debian:~# semanage login -a -s staff_u  admin
root@debian:~# semanage login -a -s root -r s0-s0:c0.c10 root2

root@debian:~# semanage login -l
Login Name                  SELinux User                MLS/MCS Range
__default__                 unconfined_u                s0-s0:c0.c1023
admin                       staff_u                     s0
luser                       user_u                      s0
root                        unconfined_u                s0-s0:c0.c1023
root2                       root                        s0-s0:c0.c10
system_u                    system_u                    s0-s0:c0.c1023
```

SELinux – správa používateľov

- SELinux používatelia a pridelené roly
  - `semanage user oper [opts] selinux_name`
    - -L – default MLS úroveň pre používateľa
    - -P – prefix používaný pre značkovanie domovských adresárov
      - väčšinou rovnaký ako rola bez _r
    - -r – rozsah MLS značiek
    - -R – povolená rola
  - umožňuje definovať používateľov namiesto ich pevnej definície v politike

SELinux – správa používateľov

```
root@debian:~# semanage user -a -P user -R user_r test_u

root@debian:~# semanage user -l

                Labeling   MLS/       MLS/
SELinux User    Prefix     MCS Level  MCS Range                      SELinux Roles

root            sysadm     s0         s0-s0:c0.c1023                 staff_r sysadm_r system_r
staff_u         staff      s0         s0-s0:c0.c1023                 staff_r sysadm_r
sysadm_u        sysadm     s0         s0-s0:c0.c1023                 sysadm_r
system_u        user       s0         s0-s0:c0.c1023                 system_r
test_u          user       s0         s0                             user_r
unconfined_u    unconfined s0         s0-s0:c0.c1023                 system_r unconfined_r
user_u          user       s0         s0                             user_r
```

SELinux – konfigurácia

- /etc/selinux/config
  - názov politiky
  - mód
    - permissive – politika nie je vynucovaná, len generuje auditné záznamy
    - enforcing – politika je vynucovaná
- /etc/selinux/*meno_politiky*/
  - konfigurácia konkrétnej politiky
  - čiastočne spravovaná pomocou `semanage`

SELinux - konfigurácia

- contexts/default_contexts
- contexts/users/*selinux_user*
  - pre jednotlivé spôsoby prihlásenia (definované rolou a doménou prihlasovacieho programu) popisuje možné kontexty (rola, doména, MLS) pre používateľa

```
system_r:crond_t:s0       unconfined_r:unconfined_t:s0 sysadm_r:sysadm_t:s0 staff_r:staff_t:s0 user_r:user_t:s0
system_r:local_login_t:s0 unconfined_r:unconfined_t:s0 sysadm_r:sysadm_t:s0 staff_r:staff_t:s0 user_r:user_t:s0


staff_r:staff_su_t:s0           unconfined_r:unconfined_t:s0 sysadm_r:sysadm_t:s0 staff_r:staff_t:s0 user_r:user_t:s0
sysadm_r:sysadm_su_t:s0         unconfined_r:unconfined_t:s0 sysadm_r:sysadm_t:s0 staff_r:staff_t:s0 user_r:user_t:s0
user_r:user_su_t:s0       unconfined_r:unconfined_t:s0 sysadm_r:sysadm_t:s0 staff_r:staff_t:s0 user_r:user_t:s0


#
# Uncomment if you want to automatically login as sysadm_r
#
#system_r:sshd_t:s0       unconfined_r:unconfined_t:s0 sysadm_r:sysadm_t:s0 staff_r:staff_t:s0 user_r:user_t:s0
```

SELinux – konfigurácia

- contexts/default_type
  - definuje defaultnú doménu pre jednotlivé roly

```
auditadm_r:auditadm_t
secadm_r:secadm_t
sysadm_r:sysadm_t
staff_r:staff_t
unconfined_r:unconfined_t
user_r:user_t
```

SELinux – prepnutie roly

- `newrole [-r rola] [-t domena] [-l mls_range]`
  - spustí shell so zmenenou rolou, doménou a MLS rozsahom
    - ak to politika umožňuje

```
luser@debian:~$ id
uid=1001(luser) gid=1001(luser) groups=1001(luser)
context=user_u:user_r:user_t:s0

luser@debian:~$ newrole -r sysadm_r
user_u:sysadm_r:sysadm_t:s0 is not a valid context
```

SELinux – prepnutie roly

```
root@debian:/home# id
uid=0(root) gid=0(root) groups=0(root)
context=root:staff_r:staff_t:s0:c0-s0:c0.c10

root@debian:/home# ls -la
ls: cannot access luser: Permission denied
ls: cannot access jerry: Permission denied
ls: cannot access admin: Permission denied
total 8
drwxr-xr-x.  5 root root 4096 Oct  8 18:22 .
drwxr-xr-x. 22 root root 4096 Oct 12 19:40 ..
d??????????  ? ?    ?       ?            ? admin
d??????????  ? ?    ?       ?            ? jerry
d??????????  ? ?    ?       ?            ? luser
```

SELinux – prepnutie roly

```
root@debian:/home# newrole -r sysadm_r
Password:
root@debian:/home# id
uid=0(root) gid=0(root) groups=0(root)
context=root:sysadm_r:sysadm_t:s0:c0-s0:c0.c10

root@debian:/home# ls -la
total 20
drwxr-xr-x.  5 root  root  4096 Oct  8 18:22 .
drwxr-xr-x. 22 root  root  4096 Oct 12 19:40 ..
drwxr-xr-x.  5 admin admin 4096 Oct  8 22:57 admin
drwxr-xr-x.  2 jerry jerry 4096 Oct  5 18:59 jerry
drwxr-xr-x.  2 luser luser 4096 Oct  8 18:33 luser
```

SELinux – zmena kontextu súboru

- `chcon [-R] kontext subor`
- `chcon [-R] [-u user] [-r role] [-t type] [-l range] subor`
  - zmení kontext súboru (-R = rekurzívne)
- `restorecon [-R] subor`
  - nastaví kontexty podľa špecifikácie v politike
- `semanage fcontext oper [opts] subor`
  - -r – rola
  - -s – SELinux používateľ
  - -t – typ
  - upraví konfiguráciu politiky, ktorú používa restorecon

SELinux – zmena kontextu súboru

```
root@debian:/home/luser# id
uid=0(root) gid=0(root) groups=0(root) context=staff_u:sysadm_r:sysadm_t:s0
root@debian:/home/luser# mkdir aaa
root@debian:/home/luser# ls -ldZ aaa
drwxr-xr-x. 2 root root staff_u:object_r:user_home_t:s0 4096 Oct 14 12:45 aaa
root@debian:/home/luser# chown luser:luser aaa


luser@debian:~$ cd aaa
-bash: cd: aaa: Permission denied


root@debian:/home/luser# ls -ldZ aaa
drwxr-xr-x. 2 luser luser staff_u:object_r:user_home_t:s0 4096 Oct 14 12:45 aaa
root@debian:/home/luser# chcon -u user_u aaa
root@debian:/home/luser# ls -ldZ aaa
drwxr-xr-x. 2 luser luser user_u:object_r:user_home_t:s0 4096 Oct 14 12:45 aaa

luser@debian:~/aaa$ ls -laZ
total 8
drwxr-xr-x. 2 luser luser user_u:object_r:user_home_t:s0     4096 Oct 14 12:45 .
drwxr-xr-x. 3 luser luser user_u:object_r:user_home_dir_t:s0 4096 Oct 14 12:45 ..
```

SELinux – nastavenie kontextu

- `runcon kontext prikaz`
- `runcon [-u user] [-r role] [-t type] [-l range] prikaz`
  - spustí uvedený príkaz v novom kontexte, ak je taký prechod politikou povolený
    - napr. môže zmeniť MLS rozsah
      - defaultne je to povolené len administrátorom, ale nie je problém to ľahko zmeniť

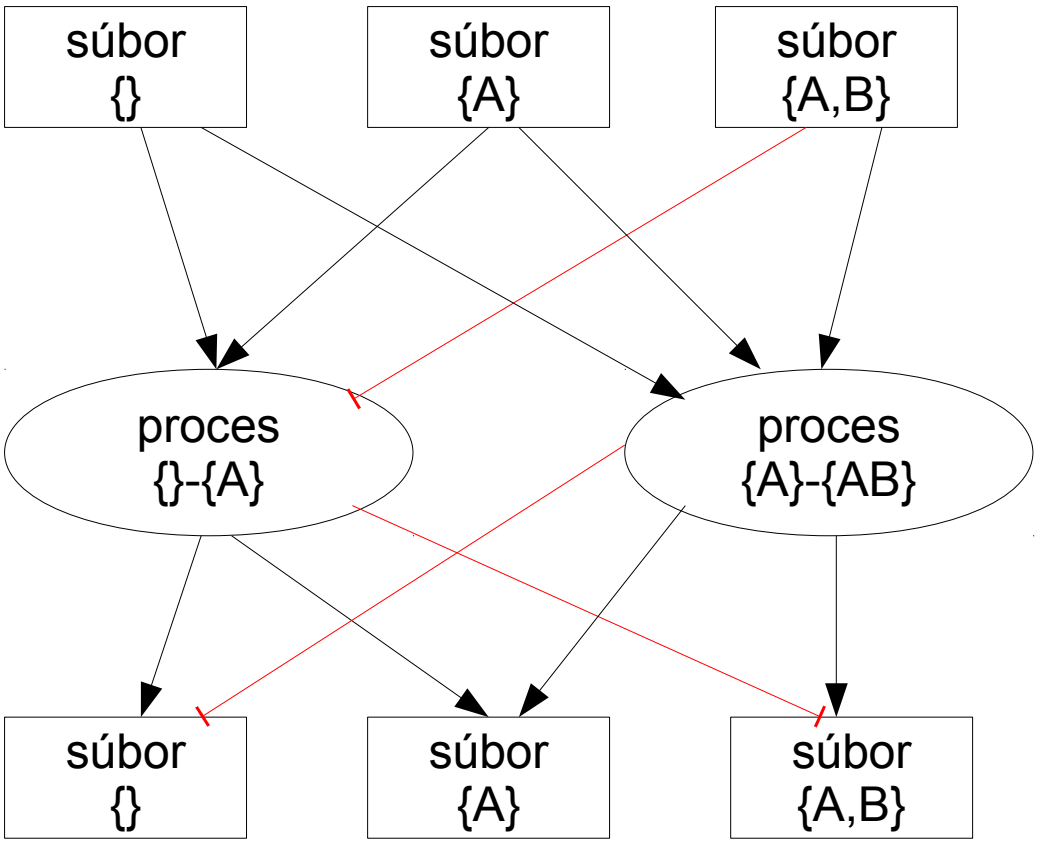
SELinux – MCS

- využívajú sa kategórie v MLS rozsahoch
  - len 1 úroveň (s0)
- objekty majú v kontexte uvedenú množinu kategórií
  - objekty musia mať dolnú a hornú hranicu rovnakú
- subjekty majú v kontexte dolnú a hornú hranicu kategórií
- A **dominuje** B práve vtedy, keď cat(A) $\supseteq$ cat(B)
  - A dom B, B domby A

SELinux – MCS

- čítanie súboru / adresára
  - $S_{horná}$ dom O
  - teda proces môže čítať len z objektov, ktorým dominuje svojou hornou hranicou
- zápis do súboru, zmena atribútov, premenovanie, vymazanie
  - $S_{horná}$ dom O a $S_{dolná}$ domby O
  - čiže subjekt môže zapisovať len do súborov, ktorým dominuje svojou hornou hranicou, a ktoré zároveň dominujú dolnej hranici subjektu
  - nové súbory môže vytvoriť v adresári, ktorému dominuje svojou hornou hranicou

SELinux – MCS


súbor
{}
súbor
{A}
súbor
{A,B}
proces
{}-{A}
proces
{A}-{AB}
súbor
{}
súbor
{A}
súbor
{A,B}

SELinux – MCS

```
admin@debian:~/mcstest$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:staff_t:s0-s0:c0.c10
admin@debian:~/mcstest$ chcon -l s0:c0 a
admin@debian:~/mcstest$ ls -lZ
total 8
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:s0:c0 2 Oct 14 16:04 a
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:s0    2 Oct 14 16:04 b
admin@debian:~/mcstest$ runcon -l s0 bash
admin@debian:~/mcstest$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:staff_t:s0
admin@debian:~/mcstest$ cat a
cat: a: Permission denied
admin@debian:~/mcstest$ cat b
B
admin@debian:~/mcstest$ echo a >> a
bash: a: Permission denied
admin@debian:~/mcstest$ echo a >> b
admin@debian:~/mcstest$ runcon -l s0:c0 bash
runcon: bash: Permission denied
admin@debian:~/mcstest$ exit
```

SELinux – MCS

```
admin@debian:~/mcstest$ runcon -l s0:c0 bash
admin@debian:~/mcstest$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:staff_t:s0:c0
admin@debian:~/mcstest$ cat a
A
admin@debian:~/mcstest$ cat b
B
a
admin@debian:~/mcstest$ echo a >> a
admin@debian:~/mcstest$ echo a >> b
bash: b: Permission denied
admin@debian:~/mcstest$ echo a >> c
admin@debian:~/mcstest$ ls -lZ
total 12
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:s0:c0 4 Oct 14 16:06 a
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:s0    4 Oct 14 16:05 b
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:s0:c0 2 Oct 14 16:09 c
admin@debian:~/mcstest$ rm b
rm: remove write-protected regular file `b'? y
rm: cannot remove `b': Permission denied
```

SELinux – MCS

- /etc/selinux/*meno_politiky*/setrans.conf
  - umožňuje definovať mená pre rôzne MLS značky
  - preklad zabezpečuje služba mcstransd
    - po zmene konfigurácie je ju potrebné reštartovať

```
s0:c0=sukromne
s0:c1=pracovne
```

```
admin@debian:~/mcstest$ ls -lZ
total 12
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:sukromne 4 Oct 14 16:06 a
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:s0       4 Oct 14 16:05 b
-rw-r--r--. 1 admin admin staff_u:object_r:user_home_t:sukromne 2 Oct 14 16:09 c
```

SELinux – MCS

- `chcat cat subor`
  - nastaví súboru kategóriu cat
- `chcat +cat subor`
  - pridá súboru kategóriu
- `chcat -- -cat subor`
  - odoberie súboru kategóriu
- `chcat -d subor`
  - odoberie všetky kategórie

SELinux – MCS

- `chcat -l … login`
  - nastaví kategórie používateľa (hornú hranicu rozsahu)
  - tiež možno spraviť pomocou `semanage login`

SELinux – správa modulov

- `semodule opt`
  - -l – vypíše zoznam aktuálnych modulov
  - -i modul.pp – nainštaluje nový modul
  - -r modul – odstráni modul
  - -d modul – zakáže modul
  - -e modul – povolí modul
- moduly sú typicky v `/usr/share/selinux/`
- po zmene je často potrebné nechať preznačkovať súborový systém

SELinux – zapnutie/vypnutie enforcing módu

- enforcing vs. permissive mód je možné prepnúť, ak to politika umožňuje

```
root@debian:~# getenforce
Enforcing
root@debian:~# setenforce 0
root@debian:~# getenforce
Permissive
root@debian:~# setenforce 1
root@debian:~# getenforce
Enforcing
```

SELinux – audit2allow, audit2why

- SELinux pri zamietnutí prístupu (prípadne aj pri povolení) generuje auditné záznamy
- `audit2why`
  - prečíta záznamy (zo vstupu, z dmesg (-d), ...) a vypíše na výstup vysvetlenie
- `audit2allow`
  - vypíše pravidlá, ktoré zakázané prístupy povolia
  - `-m modul -o modul.te`
    - vytvorí minimalistický modul
  - `-M modul`
    - vytvorí a skompiluje minimalistický modul

SELinux – audit2why, audit2allow

```
root@debian:~/allowshadow# id
uid=0(root) gid=0(root) groups=0(root) context=root:sysadm_r:sysadm_t:s0-s0:c0.c10

root@debian:~/allowshadow# dmesg -C

root@debian:~/allowshadow# cat /etc/shadow
cat: /etc/shadow: Permission denied

root@debian:~/allowshadow# dmesg
[188606.149450] audit_printk_skb: 24 callbacks suppressed
[188606.149454] type=1400 audit(1413233421.953:3029): avc:  denied  { read } for
pid=13562 comm="cat" name="shadow" dev=vda1 ino=131710
scontext=root:sysadm_r:sysadm_t:s0-s0:c0.c10 tcontext=system_u:object_r:shadow_t:s0
tclass=file

root@debian:~/allowshadow# audit2why -d
[188606.149454] type=1400 audit(1413233421.953:3029): avc:  denied  { read } for
pid=13562 comm="cat" name="shadow" dev=vda1 ino=131710
scontext=root:sysadm_r:sysadm_t:s0-s0:c0.c10 tcontext=system_u:object_r:shadow_t:s0
tclass=file
    Was caused by:
        Missing type enforcement (TE) allow rule.

        You can use audit2allow to generate a loadable module to allow this access.
```

SELinux – audit2why, audit2allow

```
root@debian:~/allowshadow# audit2allow -d -M allowshadow
******************** IMPORTANT ***********************
To make this policy package active, execute:

semodule -i allowshadow.pp

root@debian:~/allowshadow# cat allowshadow.te

module allowshadow 1.0;

require {
    type sysadm_t;
    type shadow_t;
    class file read;
}

#============= sysadm_t ==============
allow sysadm_t shadow_t:file read;
```

SELinux – audit2why, audit2allow

```
root@debian:~/allowshadow# semodule -i allowshadow.pp
libsepol.check_assertion_helper: neverallow violated by allow
sysadm_t shadow_t:file { read };
libsemanage.semanage_expand_sandbox: Expand module failed
semodule:  Failed!
```

SELinux – audit2why, audit2allow

```
root@debian:~# setenforce 0
root@debian:~# dmesg -C
```

Teraz ako staff_t prečítame nejaký súbor v /usr/src (sú typu src_t), v logu bude zaznamenaných veľa prístupov, ktoré by boli zamietnuté.

```
root@debian:~# audit2allow -d

#============= staff_t ==============
allow staff_t src_t:dir { read getattr open search };
allow staff_t src_t:file { read getattr open };
allow staff_t src_t:lnk_file { read getattr };
root@debian:~# audit2allow -d -M staff-src
******************** IMPORTANT ***********************
To make this policy package active, execute:

semodule -i staff-src.pp

root@debian:~# semodule -i staff-src.pp
root@debian:~# setenforce 1
```

SELinux – audit2why, audit2allow

```
admin@debian:/usr/src/selinux-policy-src$ cat VERSION
2.20110726

root@debian:~# semodule -r staff-src

admin@debian:/usr/src/selinux-policy-src$ cat VERSION
cat: VERSION: Permission denied
```

SELinux – ďalšie utility

- seinfo
  - umožňuje zistiť informácie o typoch, atribútoch, obmedzeniach, rolách, používateľoch, …
- sesearch
  - umožňuje hľadať pravidlá v politike

```
root@debian:~# sesearch -A -s user_t -c process -p transition
Found 18 semantic av rules:
   allow user_t user_sudo_t : process { transition sigchld sigkill sigstop
signull signal } ;
   allow user_t newrole_t : process transition ;
   allow user_t user_su_t : process { transition sigchld signal getattr } ;
   allow user_t chkpwd_t : process { transition getattr } ;
   allow user_t passwd_t : process transition ;
   allow user_t chfn_t : process transition ;
   allow user_t ssh_t : process { transition signal getattr } ;
  ...
```

SELinux – vlastný modul

- zdrojový text modulu sa skladá z
  - modul.te
    - definícia typov, atribútov, vlastných pravidiel, …
  - modul.if
    - definícia interface-ov, ktoré môže použiť iné moduly
  - modul.fc
    - pravidlá pre označkovanie súborov

SELinux – vlastný modul

- kompilácia zdrojových súborov
  - najjednoduchšie pomocou `make`
  - Makefile si skopírujeme z `/usr/share/doc/selinux-policy-dev/examples/Makefile`
  - `make` skompiluje modul do výsledného binárneho formátu, ktorý sa dá importovať do politiky pomocou `semodule`

SELinux – vlastný modul

- Skúsme vytvoriť modul *myapp*, ktorý vytvorí vlastnú doménu:
  - bude v nej môcť spúšťať bežné programy
  - bežní používatelia budú môcť spustiť programy v tejto doméne
  - nebude mať prístup k používateľským dátam
  - bude mať prístup k vlastným dátam v /mydata
  - bežní používatelia nebudú mať prístup k /mydata

SELinux – vlastný modul

myapp.te

```
policy_module(myapp, 1.0.0)

type myapp_t;
type myapp_exec_t;
type myapp_files_t;

# povolime zakladne nalezitosti aplikacnej domeny
# myapp_t oznaci ako aplikacnu domenu (napr. jej povoli pouzivat
# kniznice)
# myapp_exec_t oznaci ako jej "entrypoint" a spravi ho spustitelny

application_domain(myapp_t, myapp_exec_t)
```

SELinux – vlastný modul

```
root@debian:/home/admin/myapp# make
Compiling default myapp module
...
Creating default myapp.pp policy package
/usr/bin/semodule_package -o myapp.pp -m tmp/myapp.mod -f
tmp/myapp.mod.fc
rm tmp/myapp.mod tmp/myapp.mod.fc

root@debian:/home/admin/myapp# semodule -i myapp.pp
```

SELinux – vlastný modul

```
root@debian:/usr/local/bin# ls -lZ
total 964
-rwxr-xr-x. 1 root staff system_u:object_r:bin_t:s0   4862 Oct 14 19:14 hw
-rwxr-xr-x. 1 root staff system_u:object_r:bin_t:s0 941252 Oct 15 18:07 mybash
-rwxr-xr-x. 1 root staff system_u:object_r:bin_t:s0  34396 Oct 14 18:36 myid

root@debian:/usr/local/bin# chcon -t myapp_exec_t *
root@debian:/usr/local/bin# ls -lZ
total 964
-rwxr-xr-x. 1 root staff system_u:object_r:myapp_exec_t:s0   4862 Oct 14 19:14 hw
-rwxr-xr-x. 1 root staff system_u:object_r:myapp_exec_t:s0 941252 Oct 15 18:07 mybash
-rwxr-xr-x. 1 root staff system_u:object_r:myapp_exec_t:s0  34396 Oct 14 18:36 myid
```

SELinux – vlastný modul

```
admin@debian:/usr/local/bin$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:staff_t:s0-s0:c0.c10
admin@debian:/usr/local/bin$ ./myid
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:staff_t:s0-s0:c0.c10
admin@debian:/usr/local/bin$ ./hw
Hello World


admin@debian:/usr/local/bin$ runcon -t myapp_t ./hw
runcon: invalid context: staff_u:staff_r:myapp_t:s0-s0:c0.c10:
Invalid argument
```

SELinux – vlastný modul

myapp.te

```
policy_module(myapp, 1.0.1)

type myapp_t;
type myapp_exec_t;
type myapp_files_t;

# povolime zakladne nalezitosti aplikacnej domeny
# myapp_t oznaci ako aplikacnu domenu (napr. jej povoli pouzivat
kniznice)
# myapp_exec_t oznaci ako jej "entrypoint" a spravi ho spustitelny

application_domain(myapp_t, myapp_exec_t)

require{
  role staff_r;
}

role staff_r types myapp_t;
```

SELinux – vlastný modul

```
root@debian:/home/admin/myapp# make
root@debian:/home/admin/myapp# semodule -i myapp.pp
```

```
admin@debian:/usr/local/bin$ runcon -t myapp_t ./hw
runcon: ./hw: Permission denied

root@debian:/usr/local/bin# dmesg
[363612.964329] type=1400 audit(1413408428.769:7327): avc:
denied  { transition } for  pid=23633 comm="runcon"
path="/usr/local/bin/hw" dev=vda1 ino=280838
scontext=staff_u:staff_r:staff_t:s0-s0:c0.c10
tcontext=staff_u:staff_r:myapp_t:s0-s0:c0.c10 tclass=process
```

SELinux – vlastný modul

/usr/share/selinux/default/include/support/misc_patterns.spt

```
define(`spec_domtrans_pattern',`
        allow $1 self:process setexec;
        domain_transition_pattern($1,$2,$3)

        allow $3 $1:fd use;
        allow $3 $1:fifo_file rw_fifo_file_perms;
        allow $3 $1:process sigchld;
')


define(`domain_transition_pattern',`
        allow $1 $2:file { getattr open read execute };
        allow $1 $3:process transition;
        dontaudit $1 $3:process { noatsecure siginh rlimitinh };
')
```

SELinux – vlastný modul

myapp.te

```
policy_module(myapp, 1.0.2)

type myapp_t;
type myapp_exec_t;
type myapp_files_t;

application_domain(myapp_t, myapp_exec_t)

require{
  role staff_r;
  type staff_t;
}

role staff_r types myapp_t;
spec_domtrans_pattern(staff_t,myapp_exec_t,myapp_t)
```

SELinux – vlastný modul

```
admin@debian:/usr/local/bin$ runcon -t myapp_t ./hw
admin@debian:/usr/local/bin$

root@debian:/usr/local/bin# dmesg
[364748.489238] type=1400 audit(1413409564.293:10303359): avc:
denied  { read write } for  pid=23853 comm="hw" name="2" dev=devpts
ino=5 scontext=staff_u:staff_r:myapp_t:s0-s0:c0.c10
tcontext=staff_u:object_r:user_devpts_t:s0 tclass=chr_file
[364748.489293] type=1400 audit(1413409564.293:10303360): avc:
denied  { use } for  pid=23853 comm="hw" path="/dev/pts/2"
dev=devpts ino=5 scontext=staff_u:staff_r:myapp_t:s0-s0:c0.c10
tcontext=system_u:system_r:sshd_t:s0-s0:c0.c1023 tclass=fd
```

SELinux – vlastný modul

myapp.te

```
policy_module(myapp, 1.0.3)
type myapp_t;
type myapp_exec_t;
type myapp_files_t;
application_domain(myapp_t, myapp_exec_t)

# povolime pristup na zdedene fd (vstup/vystup)
domain_use_interactive_fds(myapp_t)

# povolime pristup na terminal pouzivatela
userdom_use_inherited_user_terminals(myapp_t)

require{
  role staff_r;
  type staff_t;
}

role staff_r types myapp_t;
spec_domtrans_pattern(staff_t,myapp_exec_t,myapp_t)
```

SELinux – vlastný modul

```
admin@debian:/usr/local/bin$ runcon -t myapp_t ./hw
Hello World


admin@debian:/usr/local/bin$ runcon -t myapp_t ./myid
uid=1002 gid=1002 groups=1002
```

SELinux – vlastný modul

```
# povolime citanie beznych suborov z /etc
files_read_etc_files(myapp_t)

# povolime pristup potebny na zistenie, ci je aktivny SELinux
selinux_get_fs_mount(myapp_t)

# povolime pristup k suborom pre lokalizaciu prostredia
miscfiles_read_localization(myapp_t)

# povolime si spustanie zakladnych programov v /bin a pod.
corecmd_exec_bin(myapp_t)

# shell bude potrebovat setpgid na nastavenie vytvorenie procgroup

allow myapp_t self:process setpgid;
```

SELinux – vlastný modul

```
admin@debian:/usr/local/bin$ runcon -t myapp_t ./hw
Hello World
admin@debian:/usr/local/bin$ runcon -t myapp_t ./myid
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:myapp_t:s0-s0:c0.c10
admin@debian:/usr/local/bin$ runcon -t myapp_t ./mybash
mybash: /home/admin/.bashrc: Permission denied
admin@debian:/usr/local/bin$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:myapp_t:s0-s0:c0.c10
```

SELinux – vlastný modul

```
admin@debian:/usr/local/bin$ ls -l /
ls: cannot access /initrd.img: Permission denied
ls: cannot access /root2: Permission denied
ls: cannot access /vmlinuz: Permission denied
ls: cannot access /media: Permission denied
ls: cannot access /mnt: Permission denied
ls: cannot access /boot: Permission denied
ls: cannot access /root: Permission denied
ls: cannot access /home: Permission denied
ls: cannot access /lost+found: Permission denied
ls: cannot access /selinux: Permission denied
ls: cannot access /tmp: Permission denied
total 36
drwxr-xr-x.  2 root root 4096 Oct  5 18:49 bin
d??????????  ? ?    ?       ?            ? boot
drwxr-xr-x. 13 root root 3020 Oct 11 18:27 dev
drwxr-xr-x. 78 root root 4096 Oct 15 23:57 etc
d??????????  ? ?    ?       ?            ? home
l??????????  ? ?    ?       ?            ? initrd.img
drwxr-xr-x. 13 root root 4096 Sep 24 14:55 lib
```

SELinux – vlastný modul

myapp.te

```
policy_module(myapp, 1.0.5)
type myapp_t;
type myapp_exec_t;
type myapp_files_t;
application_domain(myapp_t, myapp_exec_t)
domain_use_interactive_fds(myapp_t)
userdom_use_inherited_user_terminals(myapp_t)
files_read_etc_files(myapp_t)
selinux_get_fs_mount(myapp_t)
miscfiles_read_localization(myapp_t)
corecmd_exec_bin(myapp_t)
allow myapp_t self:process setpgid;

files_type(myapp_files_t)
allow myapp_t myapp_files_t:dir *;
allow myapp_t myapp_files_t:file *;

...
```

SELinux – vlastný modul

```
root@debian:/# mkdir mydata
root@debian:/# chcon -t myapp_files_t mydata
root@debian:/# chown admin mydata
root@debian:/# ls -ldZ mydata
drwxr-xr-x. 2 admin root staff_u:object_r:myapp_files_t:s0 4096 Oct
16 00:30 mydata
```

SELinux – vlastný modul

```
admin@debian:/usr/local/bin$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:staff_t:s0-s0:c0.c10
admin@debian:/usr/local/bin$ ls -la /mydata
ls: cannot access /mydata: Permission denied

admin@debian:/usr/local/bin$ runcon -t myapp_t ./mybash
mybash: /home/admin/.bashrc: Permission denied
admin@debian:/usr/local/bin$ ls -la /mydata
total 8
drwxr-xr-x.  2 admin root 4096 Oct 16 00:30 .
drwxr-xr-x. 24 root  root 4096 Oct 16 00:30 ..
admin@debian:/usr/local/bin$ echo yes > /mydata/file
admin@debian:/usr/local/bin$ ls -lZ /mydata/
total 4
-rw-r--r--.  1 admin admin staff_u:object_r:myapp_files_t:s0    4 Oct 16 00:34 file
```

SELinux – vlastný modul

/usr/share/selinux/default/include/support/misc_patterns.spt

```
define(`domtrans_pattern',`
        domain_auto_transition_pattern($1,$2,$3)

        allow $3 $1:fd use;
        allow $3 $1:fifo_file rw_fifo_file_perms;
        allow $3 $1:process sigchld;
')

define(`domain_auto_transition_pattern',`
        domain_transition_pattern($1,$2,$3)
        type_transition $1 $2:process $3;
')
```

SELinux – vlastný modul

myapp.te

```
policy_module(myapp, 1.0.6)
type myapp_t;
type myapp_exec_t;
type myapp_files_t;
...
files_type(myapp_files_t)
allow myapp_t myapp_files_t:dir *;
allow myapp_t myapp_files_t:file *;


require{
  role staff_r;
  type staff_t;
}

role staff_r types myapp_t;
domtrans_pattern(staff_t,myapp_exec_t,myapp_t)
```

SELinux – vlastný modul

```
admin@debian:/usr/local/bin$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:staff_t:s0-s0:c0.c10

admin@debian:/usr/local/bin$ ./myid
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:myapp_t:s0-s0:c0.c10

admin@debian:/usr/local/bin$ ./mybash
mybash: /home/admin/.bashrc: Permission denied
admin@debian:/usr/local/bin$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
context=staff_u:staff_r:myapp_t:s0-s0:c0.c10
admin@debian:/usr/local/bin$ exit
```

SELinux – vlastný modul

myapp.te

```
policy_module(myapp, 1.0.7)

type myapp_t;
type myapp_exec_t;
type myapp_files_t;

application_domain(myapp_t, myapp_exec_t)
domain_use_interactive_fds(myapp_t)
userdom_use_inherited_user_terminals(myapp_t)
files_read_etc_files(myapp_t)
selinux_get_fs_mount(myapp_t)
miscfiles_read_localization(myapp_t)
corecmd_exec_bin(myapp_t)
allow myapp_t self:process setpgid;

files_type(myapp_files_t)
allow myapp_t myapp_files_t:dir *;
allow myapp_t myapp_files_t:file *;
```

SELinux – vlastný modul

myapp.if

```
interface(`myapp_role',`

  gen_require(`
    type myapp_t;
  ')

  role $1 types myapp_t;

')

interface(`myapp_autotrans',`

  gen_require(`
    type myapp_t;
    type myapp_exec_t;
  ')

  domtrans_pattern($1, myapp_exec_t, myapp_t)
')
```

SELinux – vlastný modul

myapp.fc

```
/usr/local/bin/hw   gen_context(system_u:object_r:myapp_exec_t,s0)
/usr/local/bin/my.* gen_context(system_u:object_r:myapp_exec_t,s0)
```

myapp_trans.te

```
policy_module(myapp_trans, 1.0.0)

require{
  role staff_r;
  role sysadm_r;
  attribute userdomain;
}

myapp_role(staff_r)
myapp_role(sysadm_r)
myapp_autotrans(userdomain)
```

SELinux – vlastný modul

```
root@debian:/home/admin/myapp# semanage fcontext -l | grep myapp
/usr/local/bin/hw    all files    system_u:object_r:myapp_exec_t:s0
/usr/local/bin/my.*  all files    system_u:object_r:myapp_exec_t:s0

root@debian:/home/admin/myapp# semanage fcontext -a -t myapp_files_t -f -d /mydata
root@debian:/home/admin/myapp# semanage fcontext -a -t myapp_files_t '/mydata/.*'

root@debian:/home/admin/myapp# semanage fcontext -l -C
SELinux fcontext    type               Context

/mydata             directory          system_u:object_r:myapp_files_t:s0
/mydata/.*          all files          system_u:object_r:myapp_files_t:s0
```

SELinux – inštalácia

- balíky
  - selinux-basics
    - cez závislosti nainštaluje niekoľko ďalších
      - checkpolicy, policycoreutils, selinux-utils
  - selinux-policy-default
    - obsahuje default referenčnú politiku
  - selinux-policy-dev (pre vývoj vlastných modulov)
- selinux-activate
  - upraví konfiguráciu boot-loader-a a pam
  - po reboot-e sa označkuje celý súborový systém
- http://www.selinuxproject.org/