Security Enhanced for Linux

Étudiant : Khaled BASBOUS

Projet de Bachelor – TE3

Session 2012

Professeur responsable Gérald LITZISTORF

28 juin 2012

Hes·so GENÈVE
Haute Ecole Spécialisée de Suisse occidentale

L'avenir est à créer
h e p i a
Haute école du paysage, d'ingénierie et d'architecture de Genève

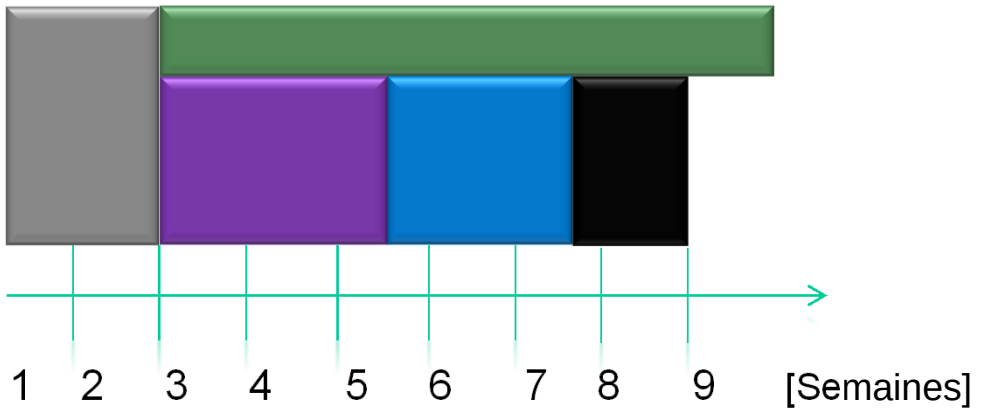
Plan de la présentation

- Cahier des charges
- Répartition du temps
- Rappel du Discretionary Access Control (DAC)
- SELinux et ces différents mécanismes de contrôles d'accès
- Démonstration
- Questions

Cahier des charges

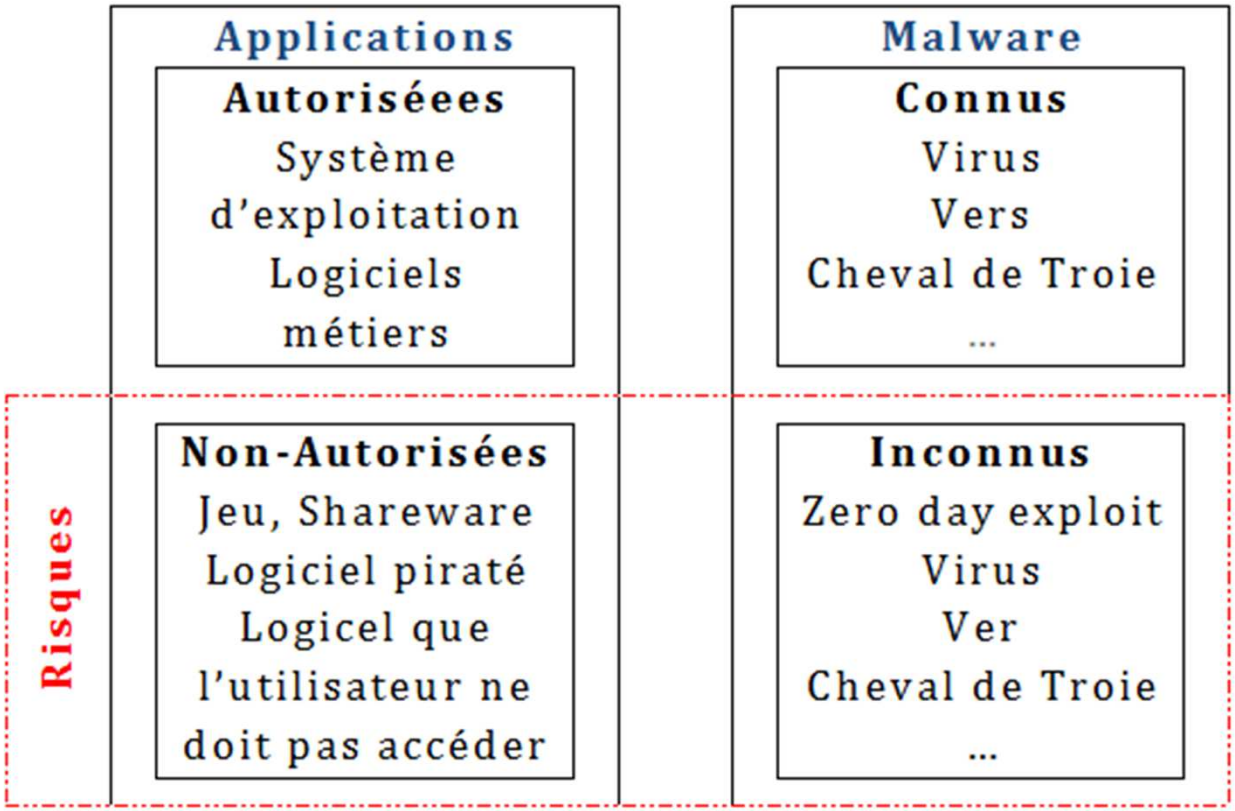
- Parcourir le livre SELinux by Example
- Comprendre le fonctionnement de SELinux dans Fedora 16
- Étudier l'architecture de SELinux
- (Configurer un exemple basé sur Bell-LaPadulla)
    - Analyse et génération module SELinux
    - Exemple qui démontre le renforcement TE

Organisation du temps



- Etude du livre SELinux by Example
- Etude de la documentation du web
- Définition et réalisations des scénarios TE, RBAC, MCS
- Rédaction du mémoire
- Définition et réalisation du scénario hacking serveur FTP et génération de module

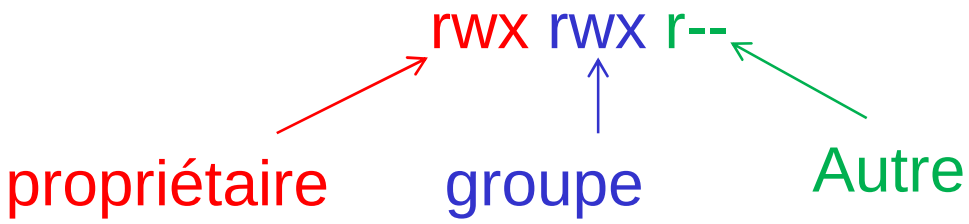
Les menaces


Applications
Malware
Autorisées
Système d'exploitation
Logiciels métiers
Connus
Virus
Vers
Cheval de Troie
...
Risques
Non-Autorisées
Jeu, Shareware
Logiciel piraté
Logicel que l'utilisateur ne doit pas accéder
Inconnus
Zero day exploit
Virus
Ver
Cheval de Troie
...

Contrôle d'accès standard de Linux

Chaque ressource possède :

- Un propriétaire
- Un groupe
- 3 ensembles de bit d'accès



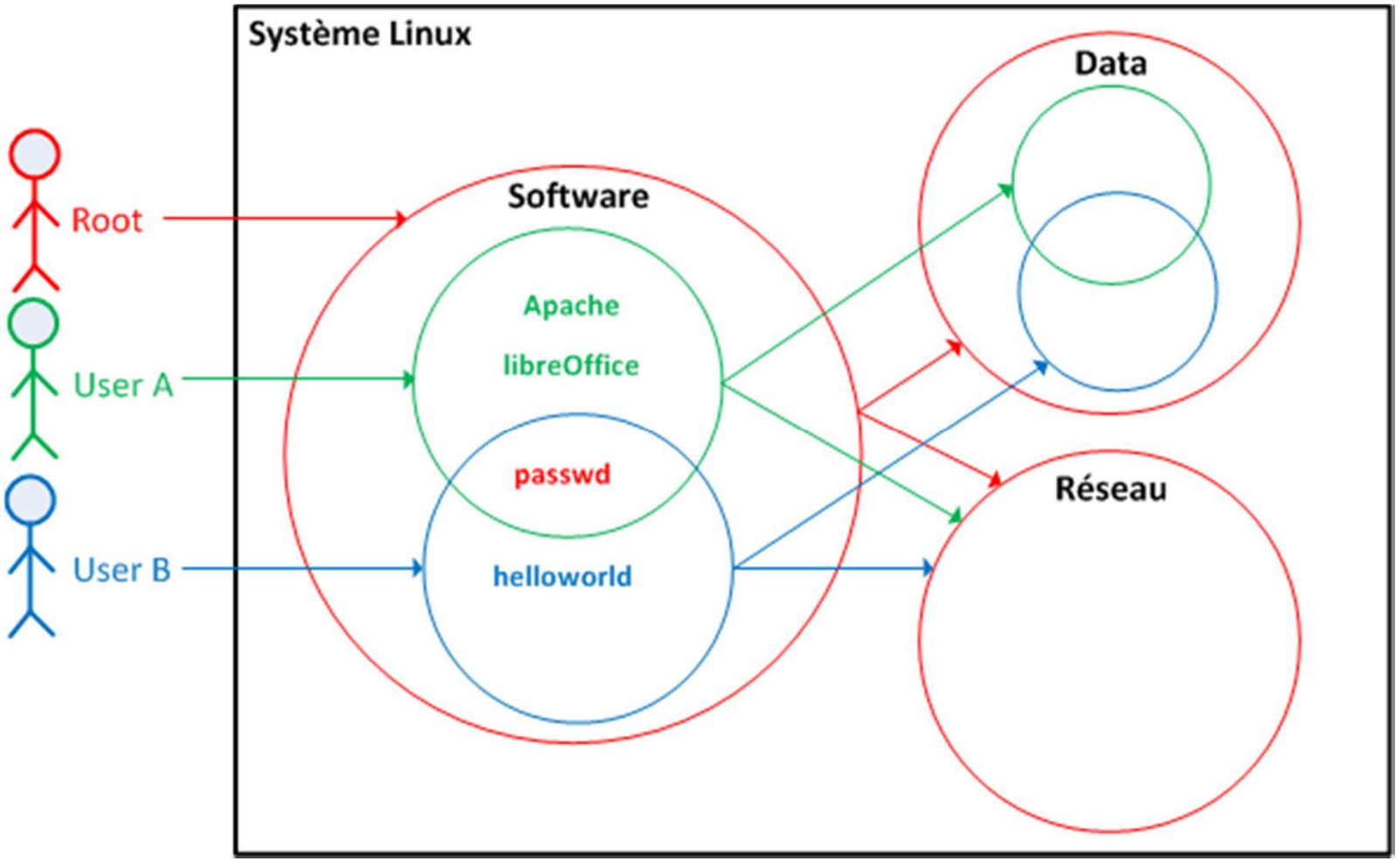
Chaque processus a un :

- Utilisateur réel
- Utilisateur effectif
- Groupe

Problème du modèle standard

- Sécurité standard trop simpliste
  - Deux niveaux de privilèges
    - User
    - Root
- Principe du moindre privilège inapplicable
- Le noyau ne peut pas distinguer entre un utilisateur et une application
- Contrôle d'accès codé en dur

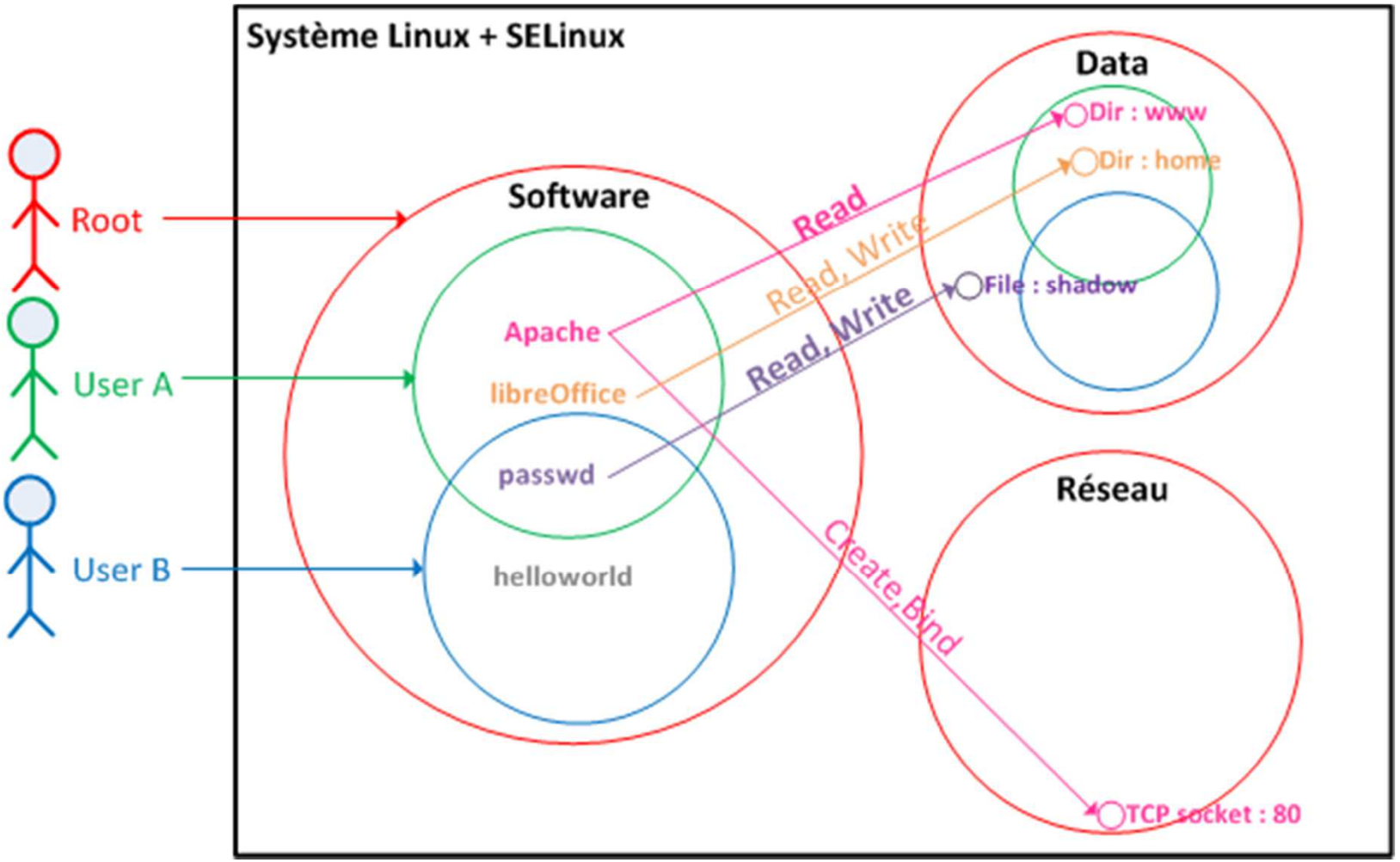
Contrôle d'accès discrétionnaire



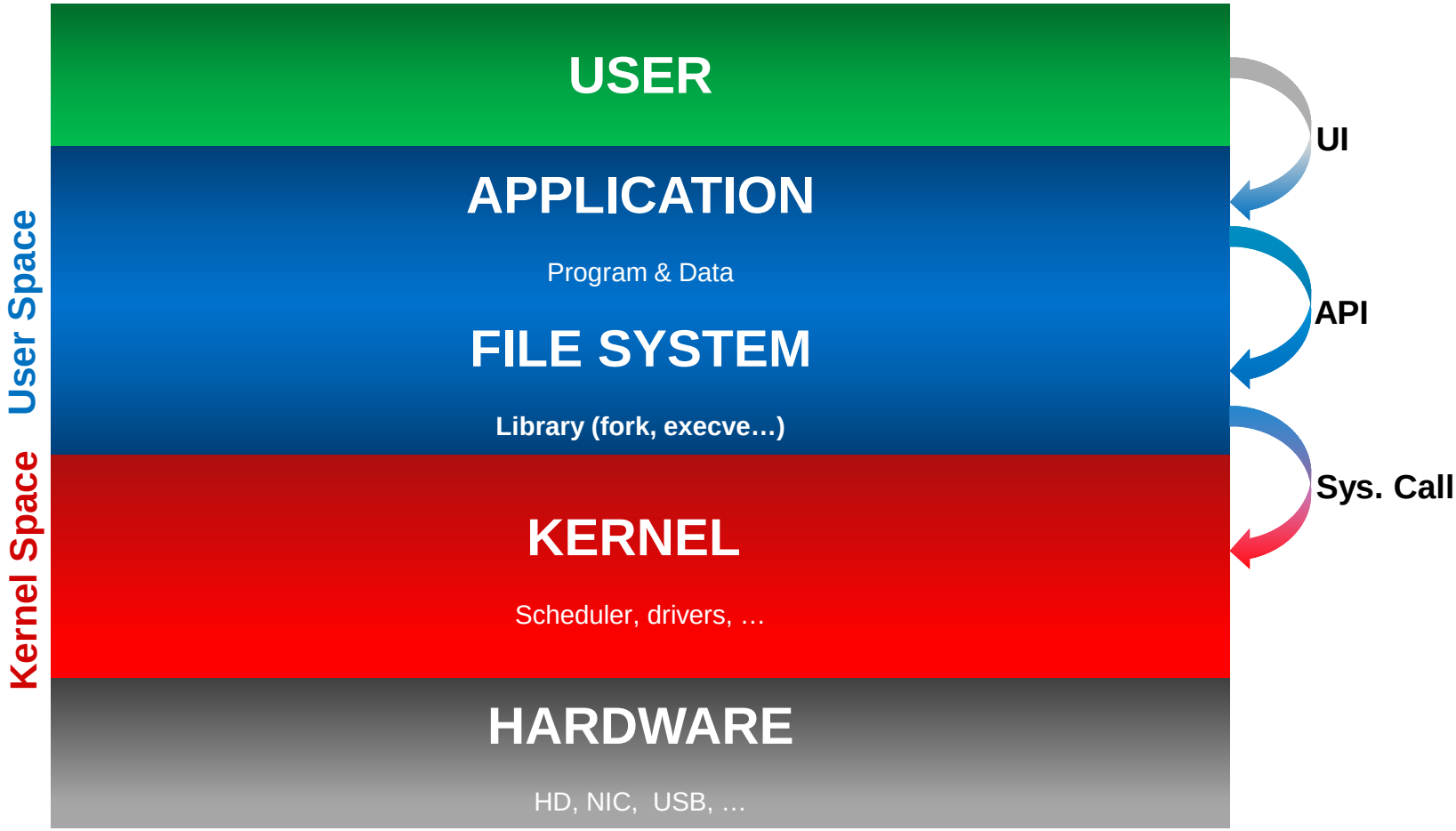
SELinux protège l'intégrité et la confidentialité du système Linux

- Ajoute des mécanismes de contrôles d'accès
  - **Type Enforcement (TE)**
  - Role Based Access Control (RBAC)
  - Multi-Level Security (MLS)
- Configurable
- Aucun accès alloué par défaut
- Transparent pour la plupart des applications
- Pas d'utilisateur tout puissant

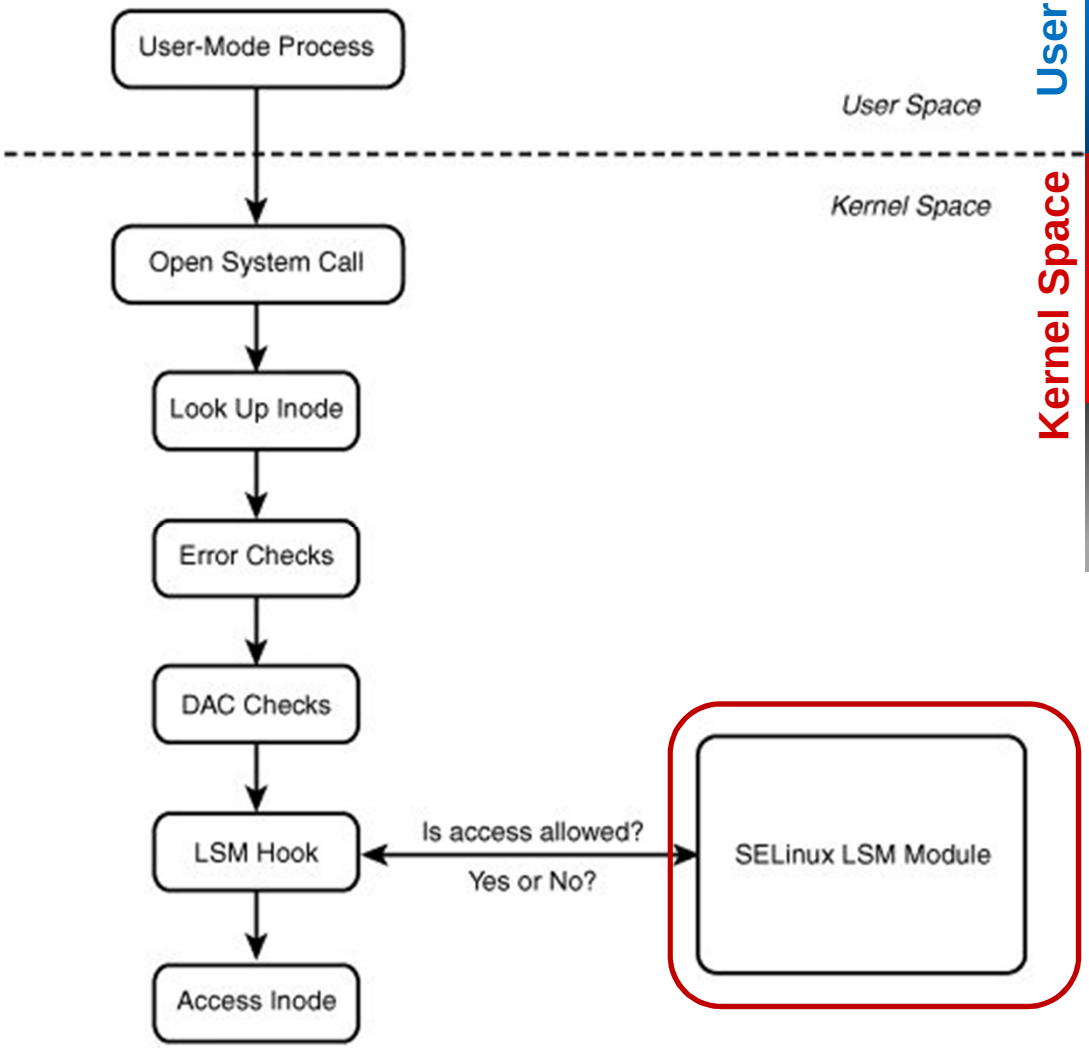
Mandatory Access Control (MAC)

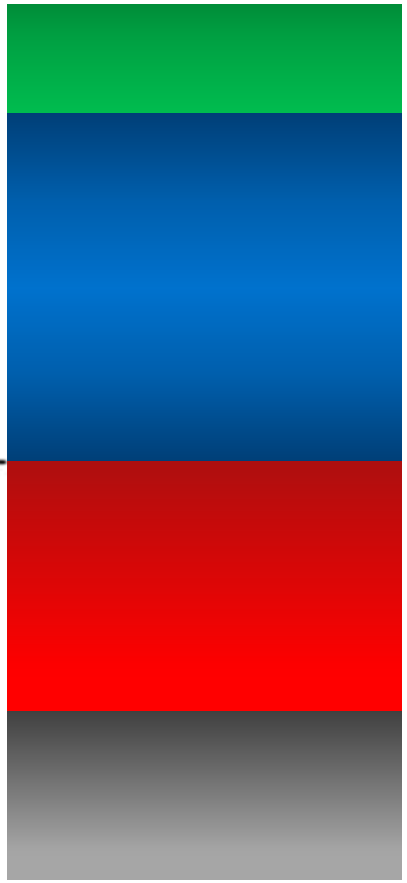


Modèle en couche de Linux

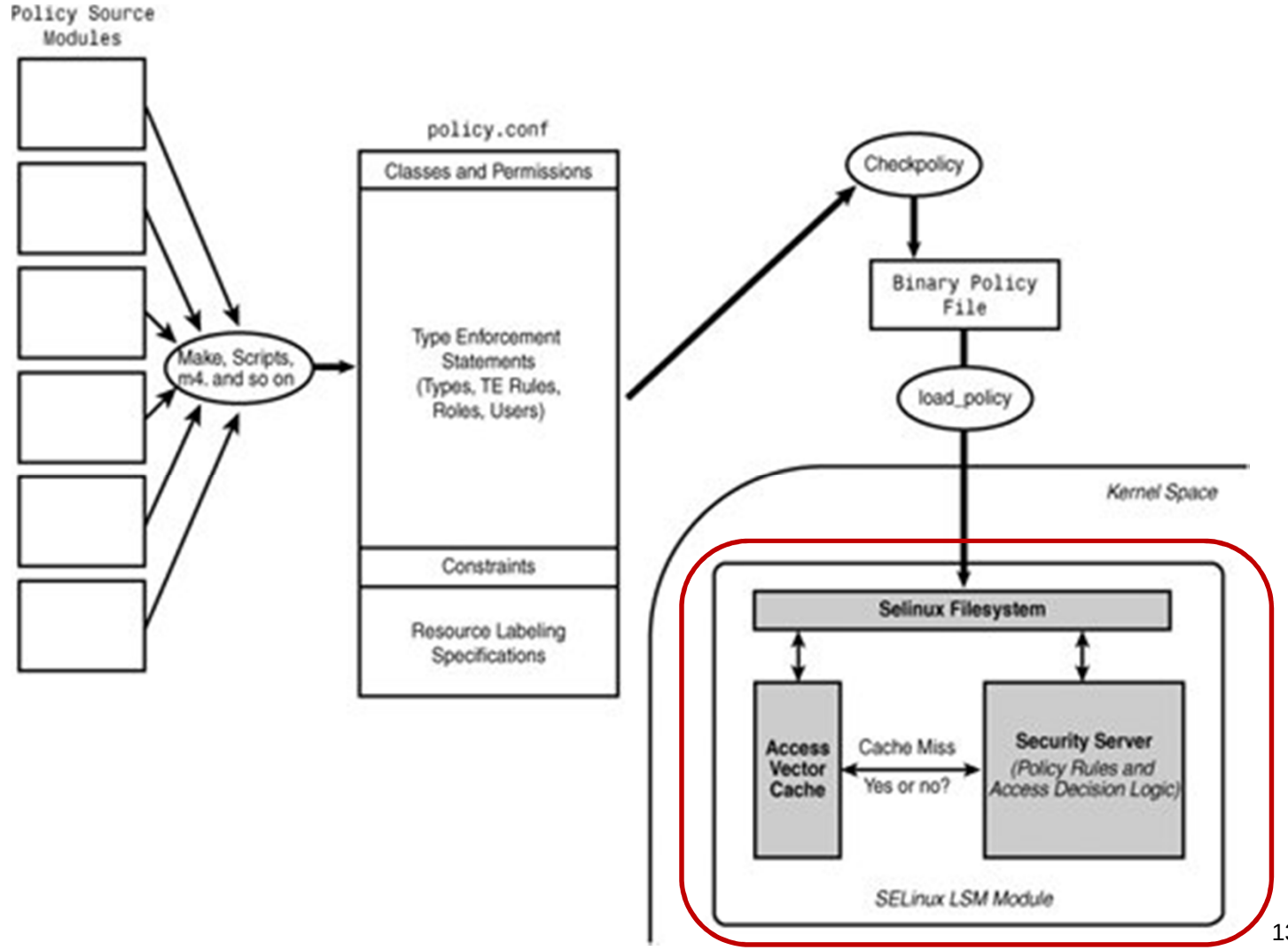


Architecture

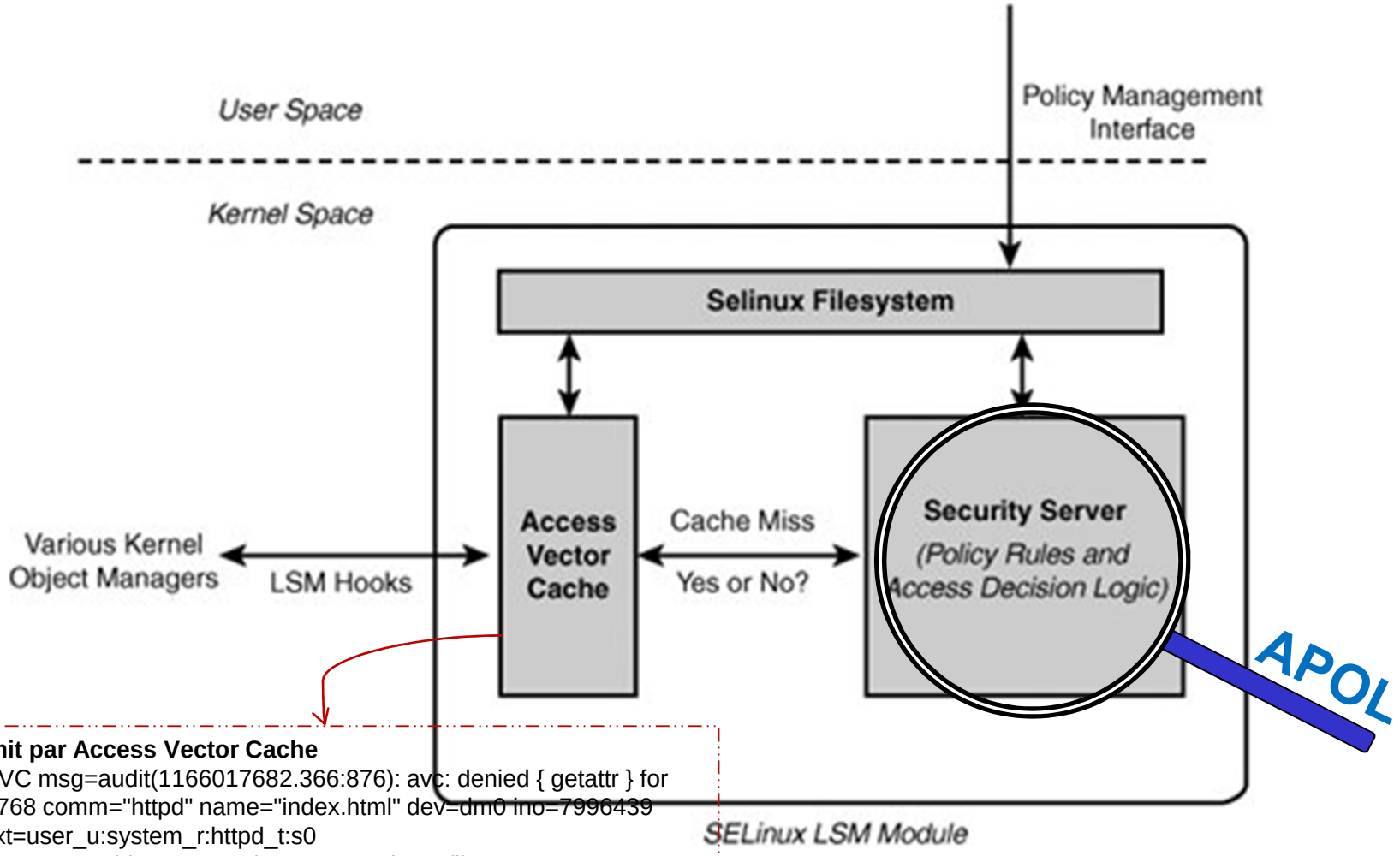




Architecture



Architecture



Log émit par Access Vector Cache
type=AVC msg=audit(1166017682.366:876): avc: denied { getattr } for
pid=23768 comm="httpd" name="index.html" dev=dm0 ino=7996439
scontext=user_u:system_r:httpd_t:s0
tcontext=user_u:object_r:user_home_t:s0 tclass=file

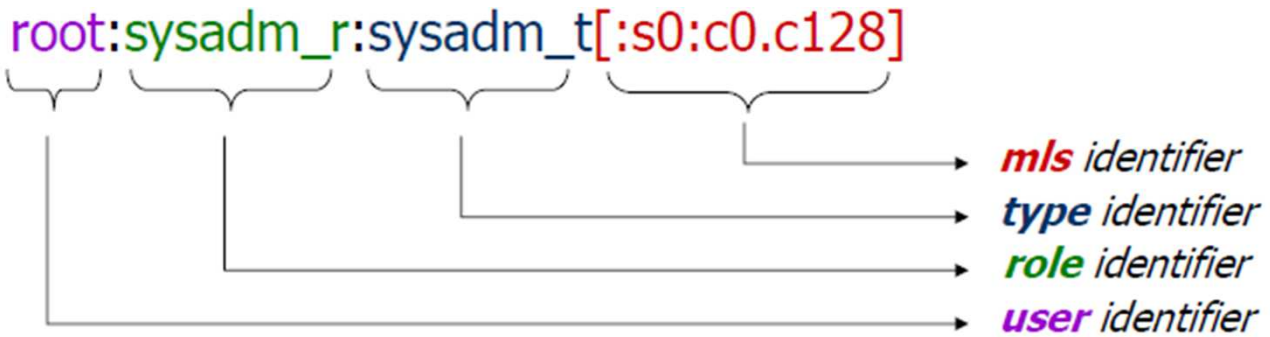
Fondement

Classe d'objet défini pour chaque ressource présente dans Linux (fichier, répertoire, processus, socket, pipe...)

Chaque classe a une liste de permissions possible :

- fichier : lecture, écriture, exécuter, point d'entrée,…
- socket : création, connexion, bind, lire, écrire,…

Chaque ressource lui est attribué un contexte de sécurité :



Contexte de sécurité

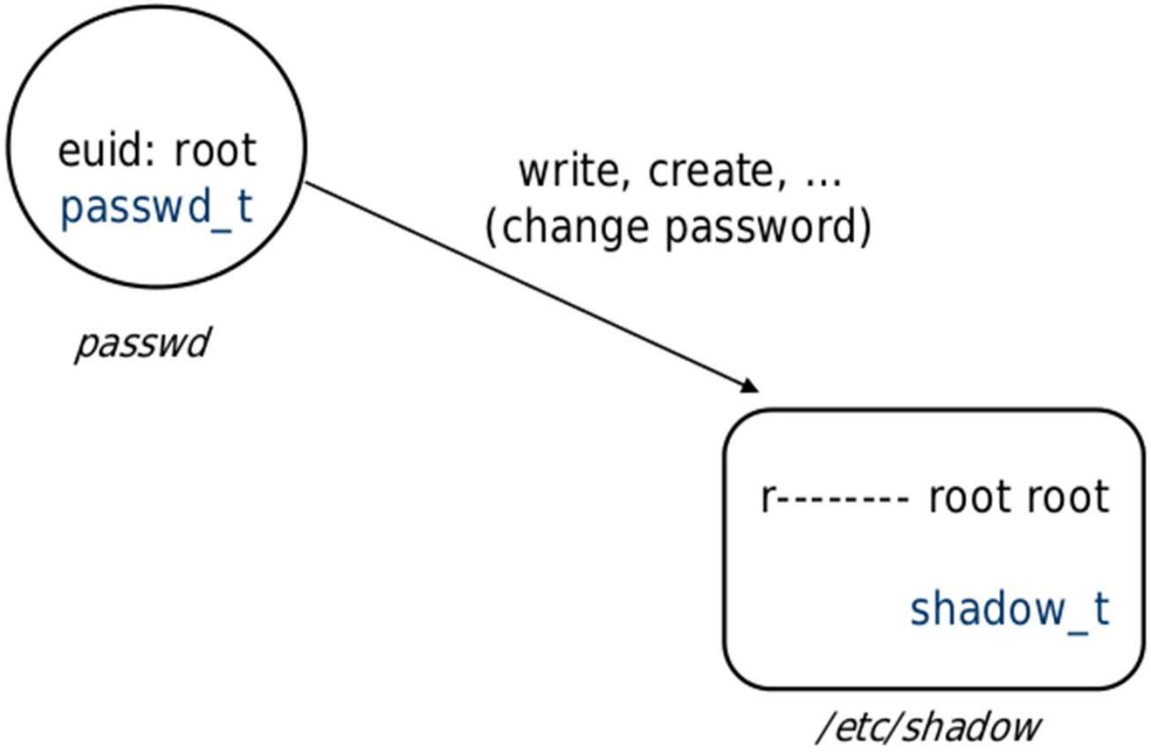
Plusieurs commandes Linux modifiées pour SELinux :

- Ajout de l'option "-Z" pour visualité le contexte de sécurité d'un objet

Exemples:

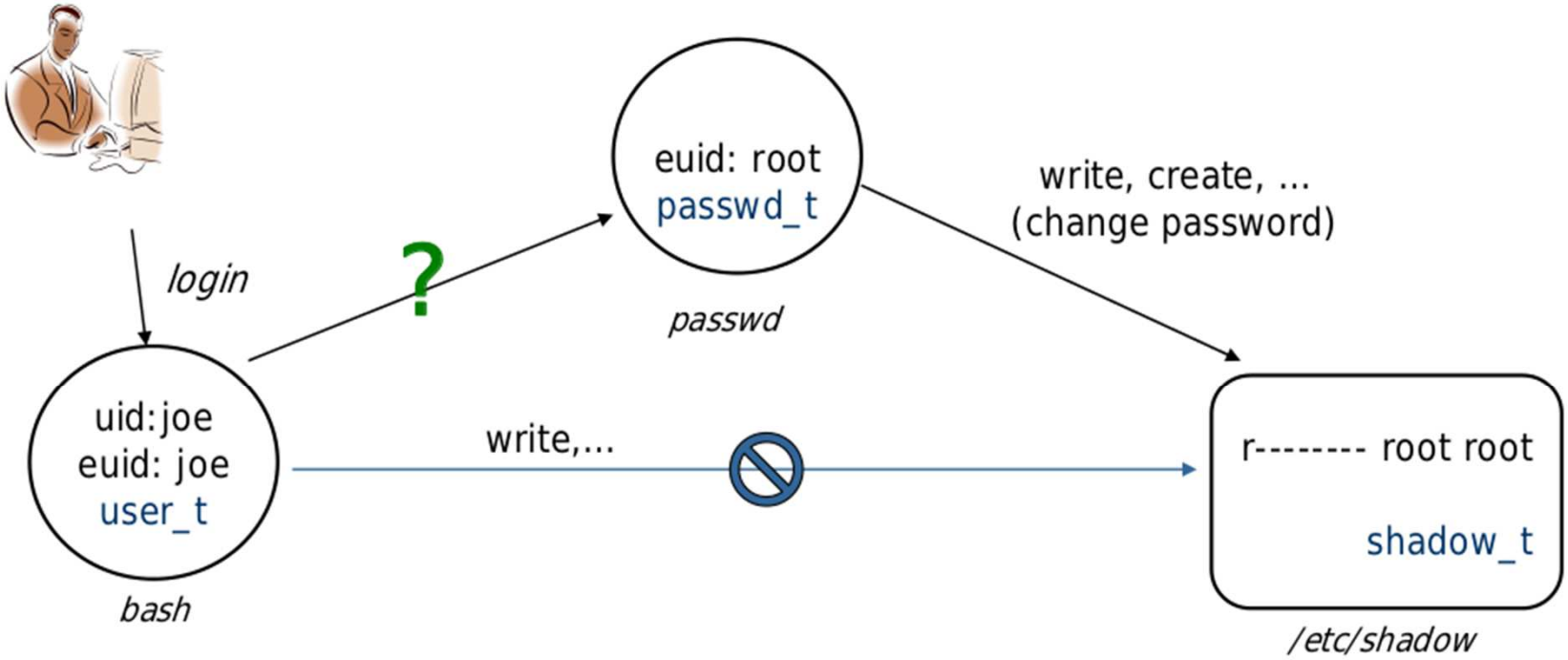
```
$ ls -lZ /home/test/ | grep Music

    drwxr-xr-x. test test user_u:object_r:user_home_t  Music

$ ps -eZ | grep passwd

       user_u:user_r:passwd_t 13212 pts/1 00:00:00 passwd
```

Règle allow (TE)



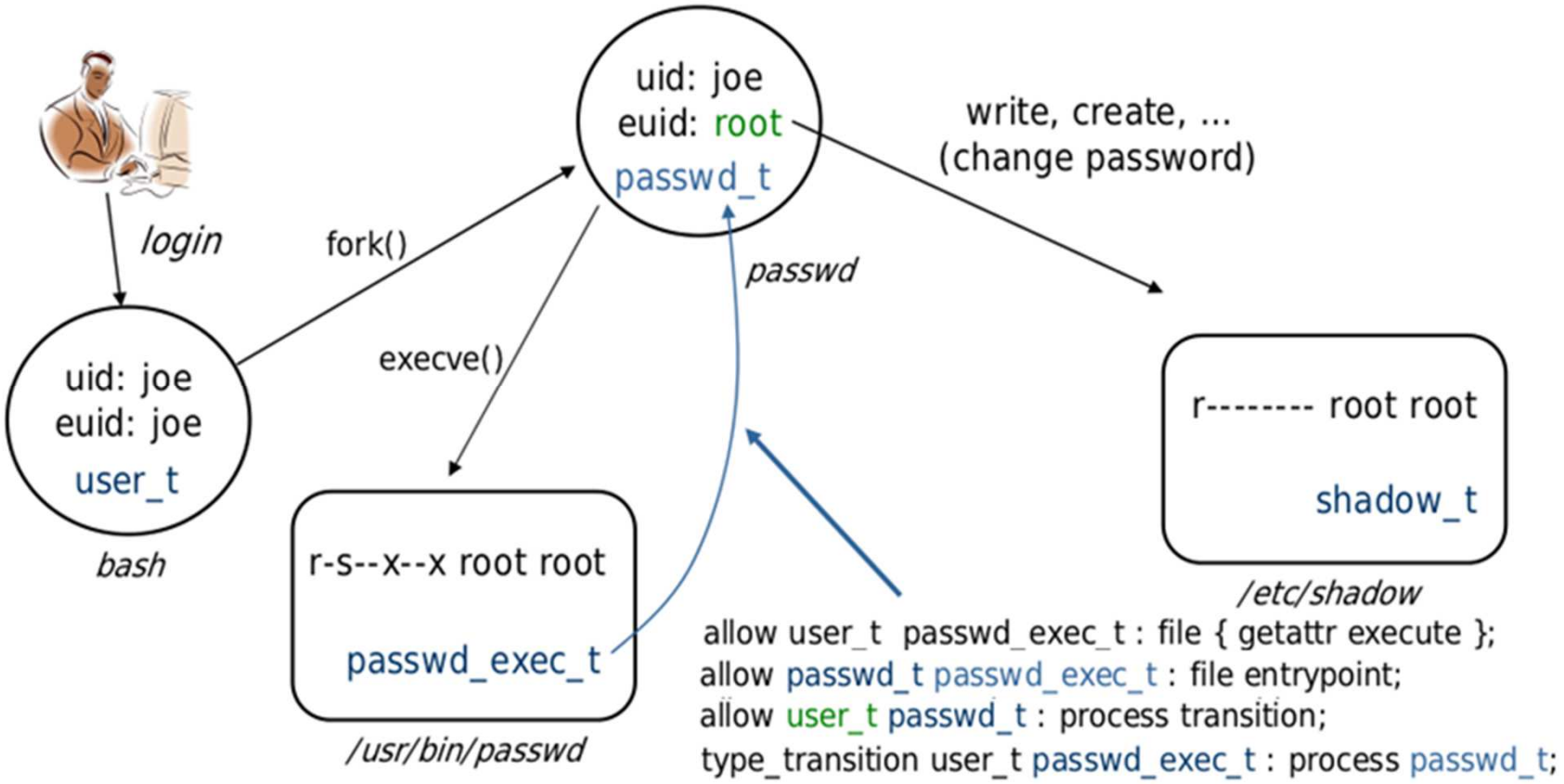

```
allow  passwd_t  shadow_t :  file
            {  read getattr write setattr append };
```

Règle type transition (TE)

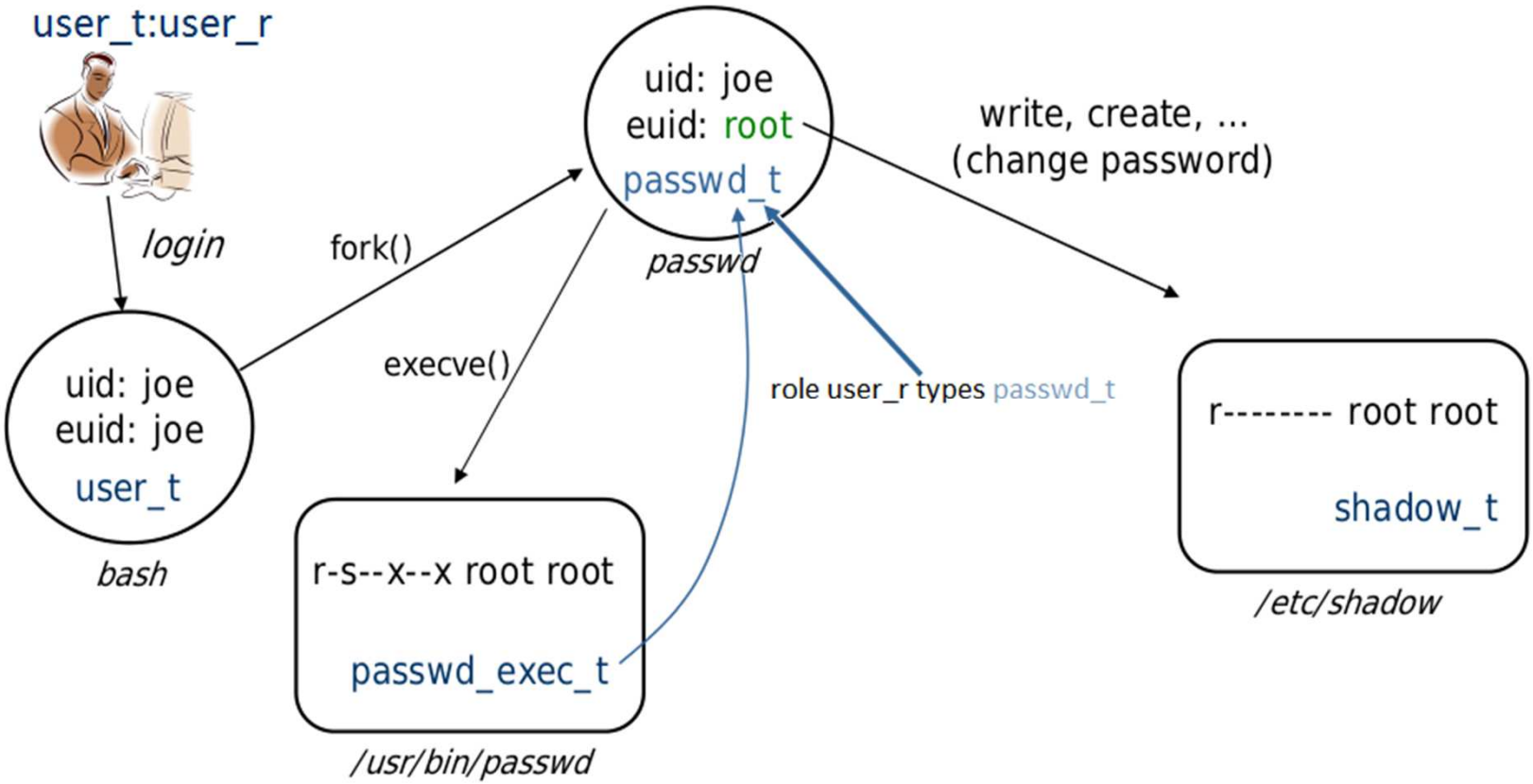



```
allow  passwd_t  shadow_t :  file
                { read getattr write setattr append };
```

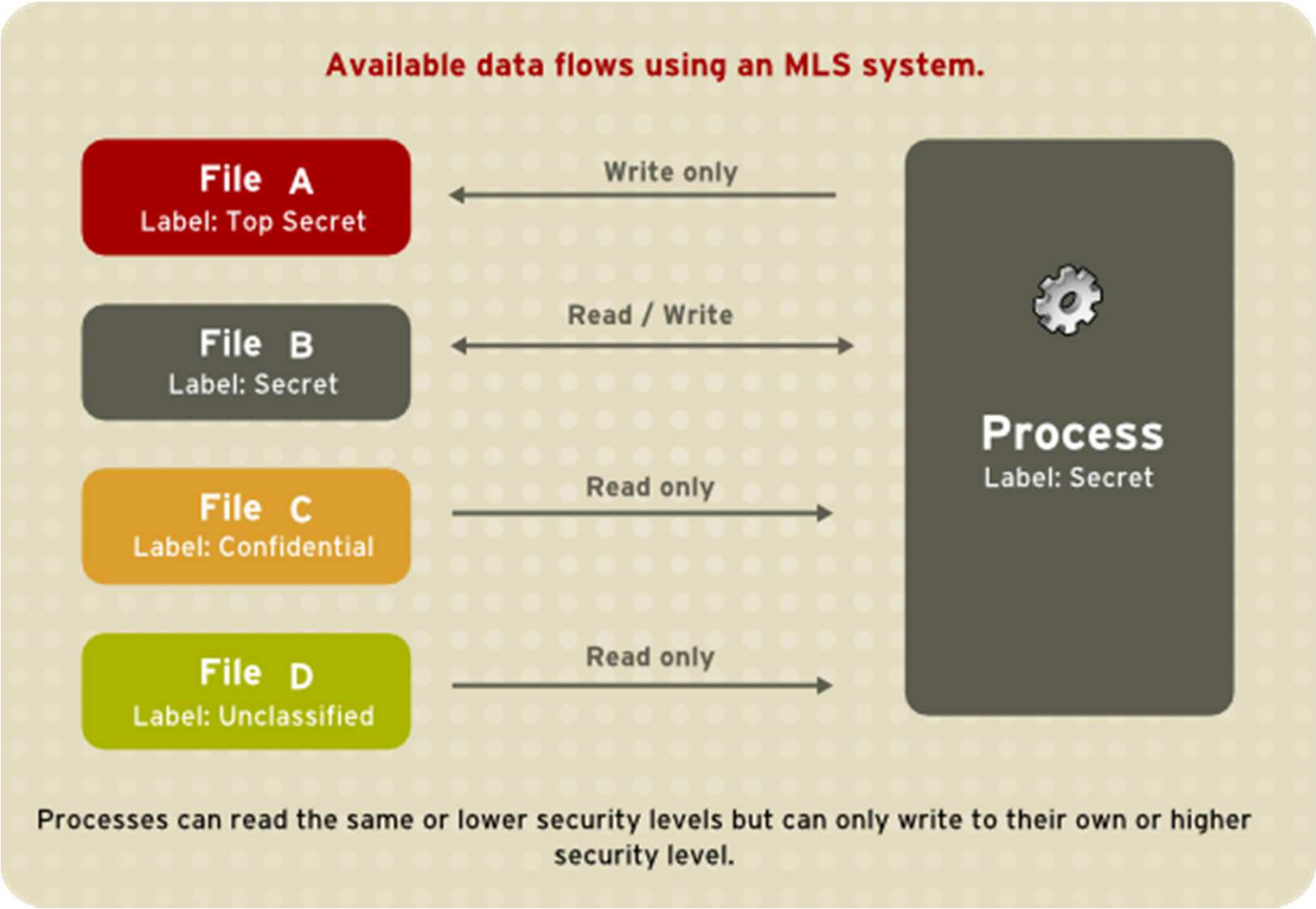
Règles changement de domaine (TE)



Contrôle d'accès basé sur le rôle (RBAC)

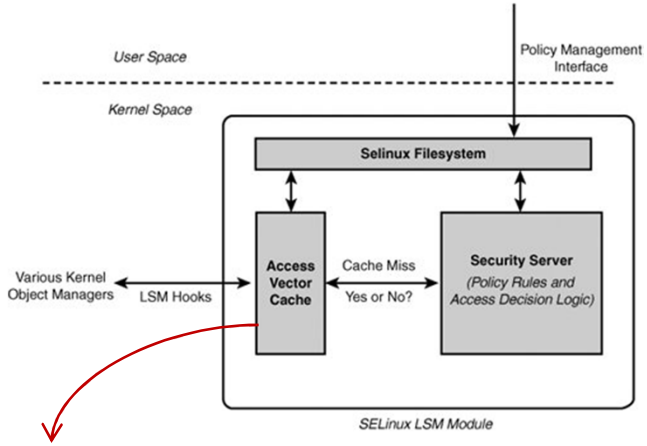


Contrôle d'accès multi-niveaux (MLS)



Contrôle d'accès multi-niveaux (MLS)

AVC Access Vector Cache



messages in /var/log/messages or /var/log/audit/audit.log

type=AVC msg=audit(1166017682.366:876): avc: **denied { getattr }** for pid=23768 **comm="httpd"** name="index.html" dev=dm0 ino=7996439 **scontext=user_u:system_r:httpd_t:s0** **tcontext=user_u:object_r:user_home_t:s0** **tclass=file**

Audit2allow est un outil qui génère les règles SELinux à partir du log pour autoriser l'accès

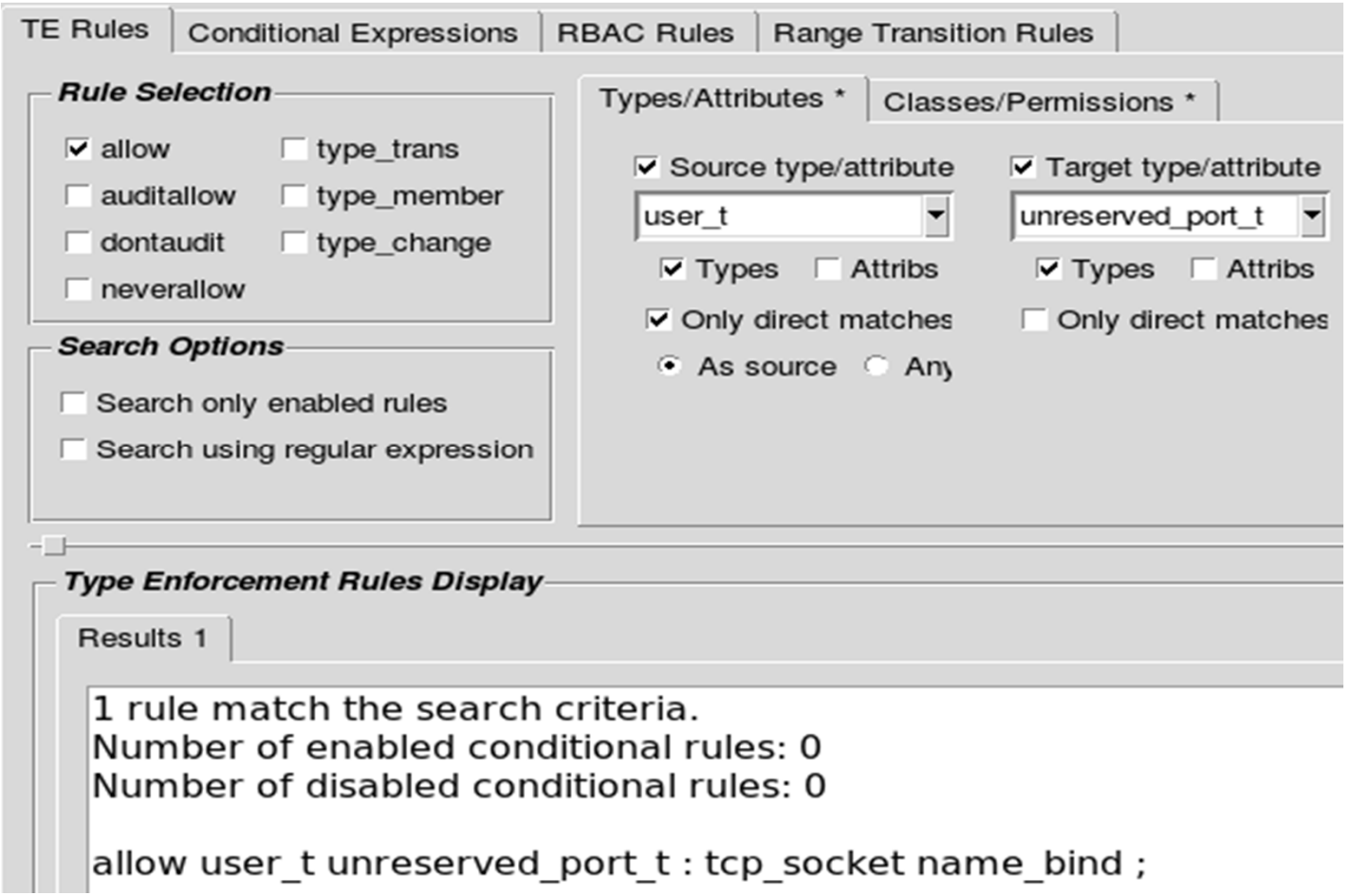
Gestion des modules SELinux

```
# ausearch -m AVC -x dropbox -ts today | audit2allow -M dropbox
```

```
module dropbox 1.0;
require {
    type unreserved_port_t;
    type user_t;
    class tcp_socket name_bind;
}
#============= user_t ==============
#!!!! This avc can be allowed using one of the these booleans:
#     user_tcp_server, allow_ypbind
allow user_t unreserved_port_t:tcp_socket name_bind;
```

```
# semodule -i dropbox.pp
```

APOL


TE Rules
Conditional Expressions
RBAC Rules
Range Transition Rules
Rule Selection
allow
auditallow
dontaudit
neverallow
type_trans
type_member
type_change
Search Options
Search only enabled rules
Search using regular expression
Types/Attributes *
Classes/Permissions *
Source type/attribute
user_t
Types
Attribs
Only direct matches
As source
Any
Target type/attribute
unreserved_port_t
Types
Attribs
Only direct matches
Type Enforcement Rules Display
Results 1
1 rule match the search criteria.
Number of enabled conditional rules: 0
Number of disabled conditional rules: 0
allow user_t unreserved_port_t : tcp_socket name_bind ;

Statistique APOL de Fedora 16

Number of Classes and Permissions:	
Object Classes:	82
Common Permissions:	5
Permissions:	241
Number of Types and Attributes:	
Types:	3651
Attributes:	297
Number of Type Enforcement Rules:	
allows:	91628
auditallows:	103
dontaudits:	7002
neverallows:	N/A
type_transitions:	14467
type_members:	62
type_changes:	46
Number of Roles:	
Roles:	13
Number of RBAC Rules:	
allows:	23
role_transitions:	290
Number of Users:	
Users:	9
Number of Booleans:	
Booleans:	218
Number of MLS Components:	
Sensitivities:	1
Categories:	1024
Number of MLS Rules:	
range_transitions:	3110
Number of Initial SIDs:	
SIDs:	27
Number of OContexts:	
PortCons:	446
NetIfCons:	0
NodeCons:	0
GenFSCons:	84
fs_use statements:	24

SELinux dans Fedora 16

- Enforcing par défaut (activé et renforce)
- Targeted Policy
  - Politique où seules les applications les plus exposées aux attaques sont restreintes dans des domaines
  - Plus de 90'000 règles allow
  - Plus de 14'000 règles de transition de domaine
  - 9 utilisateurs
  - 13 rôles
  - MCS

Administration

Courante :

- Utilisateurs SELinux
- Rôles
- Règles conditionnelles
- Correction contexte de sécurité (ex. www)
- Installation modules automatique
- Génération modules audit2allow
- Multi-Category Security

Poussée :

- Écriture de règles adaptées aux logiciels internes et externes

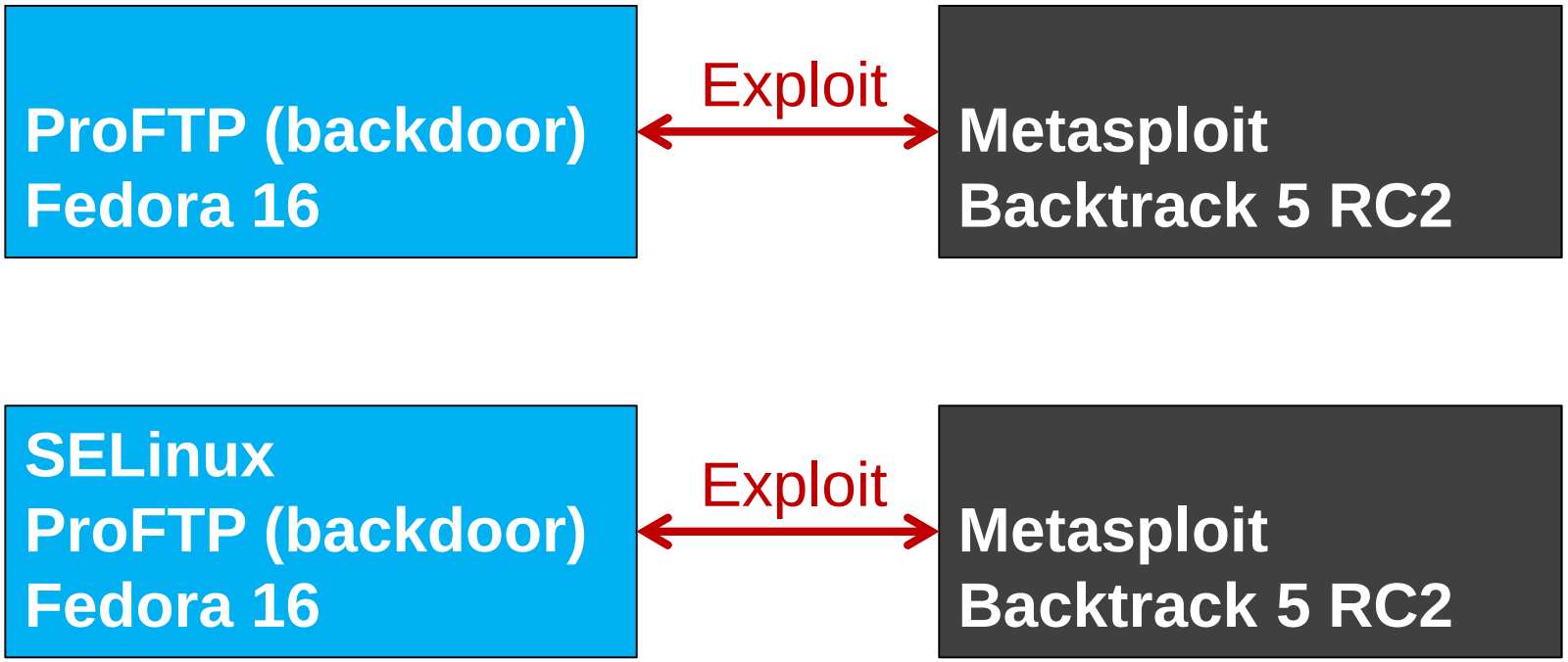
En conclusion

- Règles de référence créées et entretenues par l'entreprise Tresys
- Seuls quelques experts écrivent toutes les règles et utilitaires
- Solution stable et m'a convaincue de son apport
- Poursuite du projet dans un programme exploratoire financé par la Hes-so

Démonstration

- Exploitation d'un backdoor ajouté les sources du serveur proFTP
- Étude règles appliquées au serveurs FTP présent dans Fedora 16
- À quoi faut faire attention avant de compiler le serveur
- Utilisation d'APOL et des outils CLI

Comparaison



Questions

Liens et références

Figures extraite du livre « SELinux by Example »

Figures MLS extraite du guide de déploiement Red Hat :
http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/

Figure de la slide menaces reprise de l'entreprise SecureWave

Modèle en couche Linux de Frédéric RAYNAL et Damien AUMAITRE :
http://ws.edu.isoc.org/data/2007/147211836047d8efbe7b18d/linux-handout.pdf