redhat.

Security Enhanced Linux

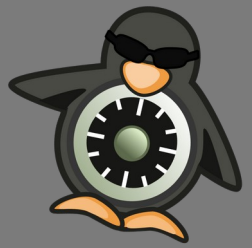
Soussan Jawad
Supervision :Dr.Mehdi Moukhafi

Ecole normale supérieure Meknès
Departement Informatique
14/12/2023

PLAN

- Introduction
- DAC vs MAC
- Modules de sécurité linux
- Politiques
- Etats et Modes de SElinux
- Fonctionnement de selinux
- Gestion des étiquettes
- Booleans

Introduction



Historique

- Dans les systèmes Linux et Unix, le contrôle d'accès discrétionnaire (DAC) permettait aux utilisateurs, d'exercer un contrôle étendu sur les autorisations des fichiers.
- Le contrôle est basé sur **la discrétion de l'utilisateur**.

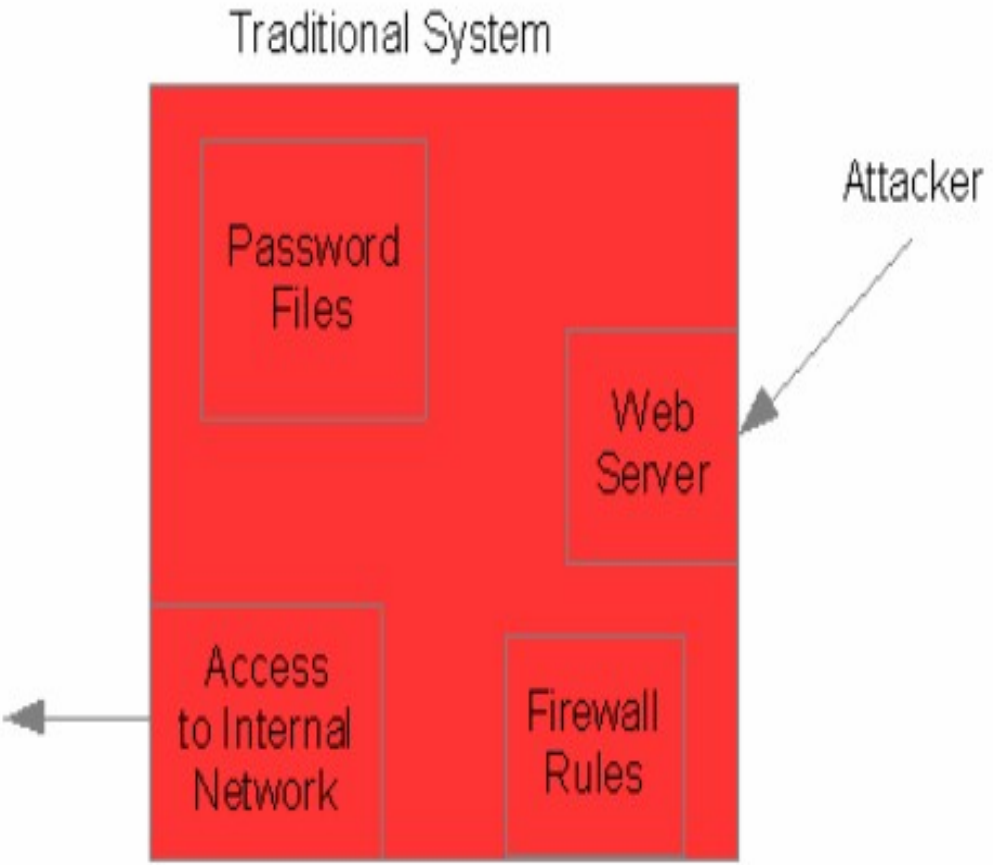
- Le DAC est un modèle de contrôle d'accès qui permet aux propriétaires de ressources de définir les droits d'accès aux objets, généralement des fichiers ou des répertoires.

- Bien que ce modèle offre une flexibilité apparente, il présente des défis majeures en termes de sécurité , il présente des vulnérabilités potentielles que les attaquants pourraient exploiter :

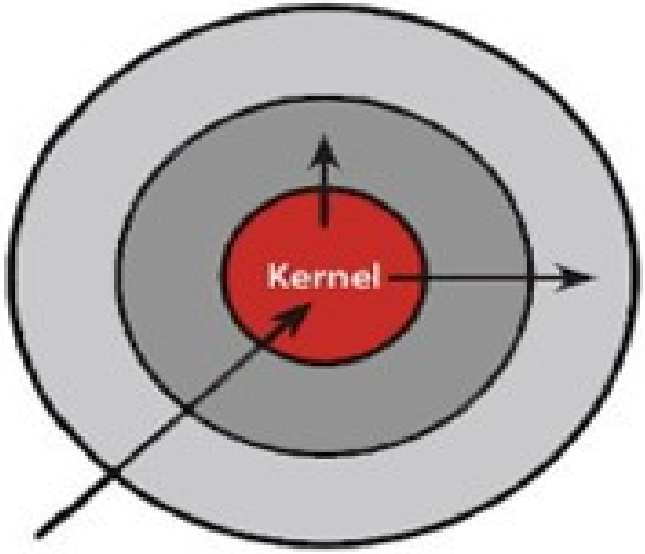
➢ Mauvaise gestion des droits

➢ Escalade de privilèges (privilege escalation)

➢ …etc .



Dès qu'une faille de sécurité touche un composant du système de privilèges, tout le système est compromis.



Discretionary Access Control

Once a security exploit gains access to priveleged system component, the entire system is compromised.

- Conscientes de cette vulnérabilité,NSA avec la contribution de **Redhat**, a lancé un projet de recherche dans les années 1990.
- Cet effort a conduit au développement de Security-Enhanced Linux (SELinux), corrigeant les failles des modèles de sécurité traditionnels.
- SELinux est un module de sécurité du noyau Linux qui fournit un mécanisme pour prendre en charge les **politiques** de sécurité du contrôle d'accès, en particulier l**es contrôles d'accès obligatoires (MAC).**

- La technologie SElinux Lancé en 2000 sous GNU GPL.
- En 2003, SELinux est devenu partie intégrante du noyau Linux.
- SELinux a été ajouté à certaines distributions Linux (***Fedora***, ***Redhat*** , ***OpenSuse*** ,**centOS**) pour encourager le développement et l'adoption de l'open source.

Ce changement a marqué une rupture avec l'approche discrétionnaire, améliorant la sécurité globale du système.

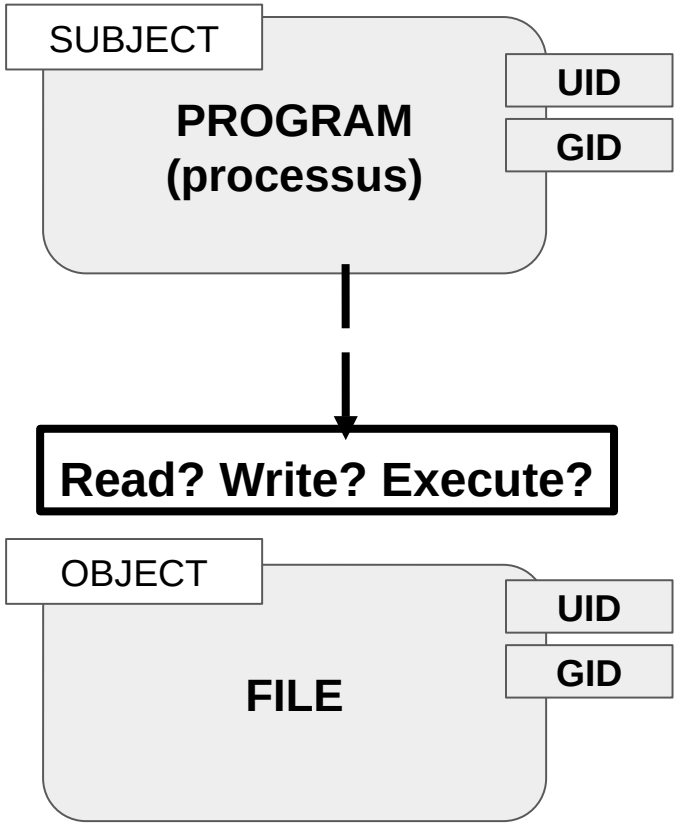
DAC vs MAC

Discretionary access control

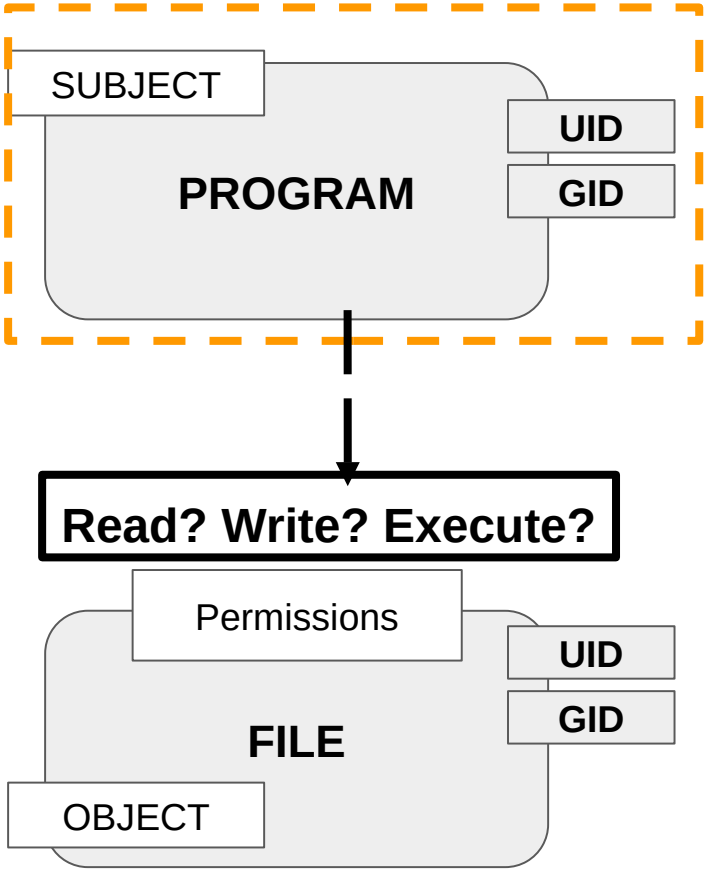
VS

Mandatory access control

DAC

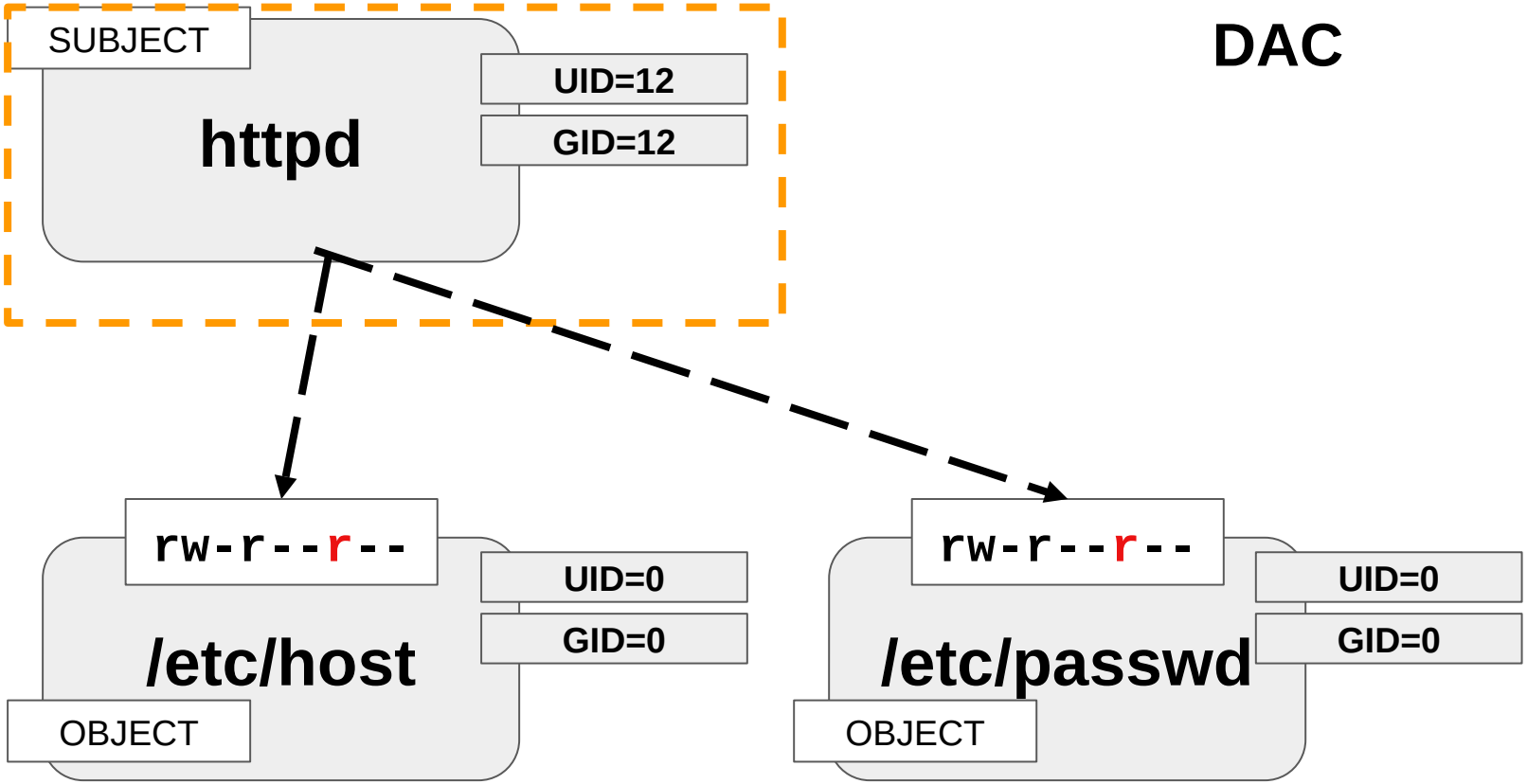


Système de sécurité traditionnel



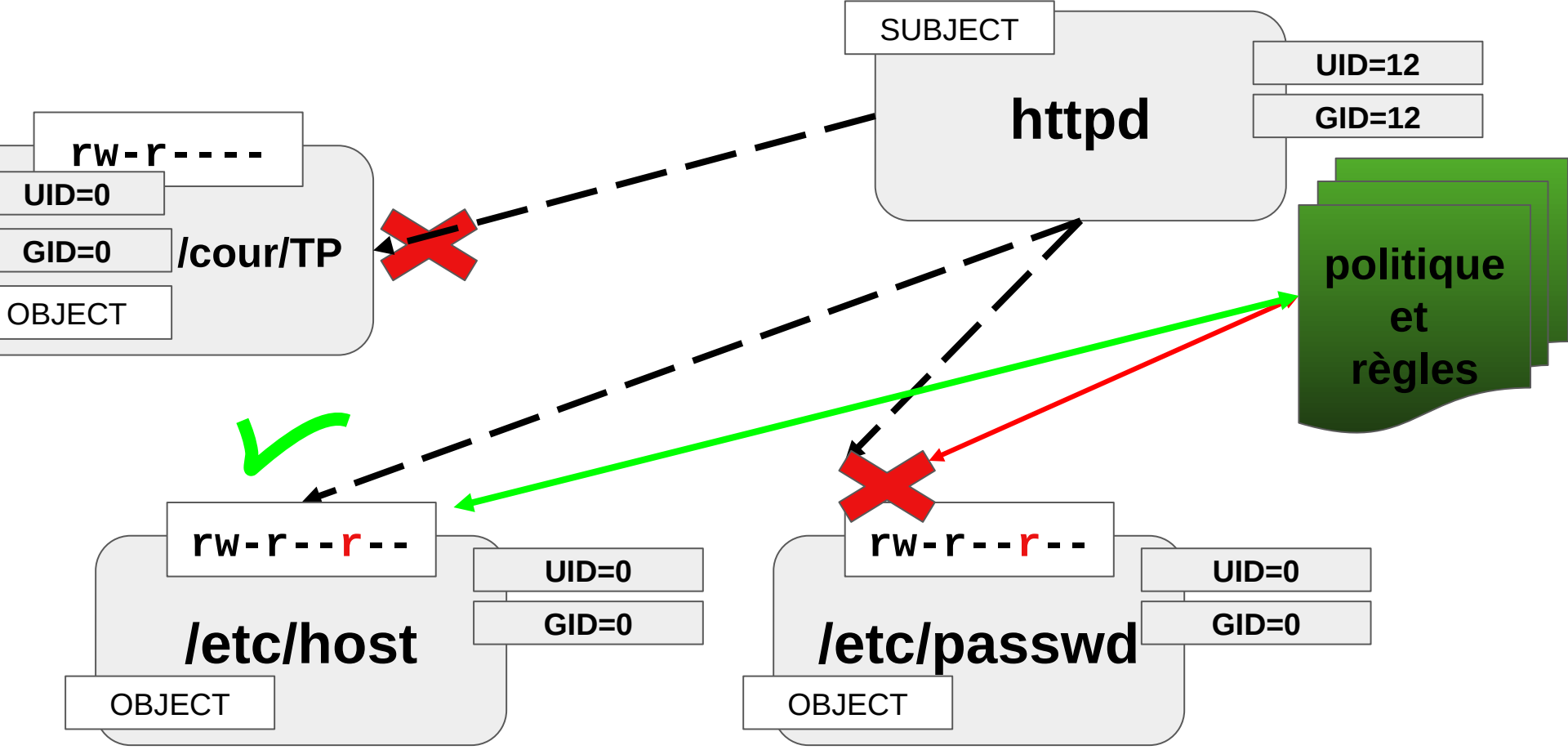
DAC

Système de sécurité traditionnel


DAC
SUBJECT
httpd
UID=12
GID=12
rw-r--r--
/etc/host
UID=0
GID=0
OBJECT
rw-r--r--
/etc/passwd
UID=0
GID=0
OBJECT

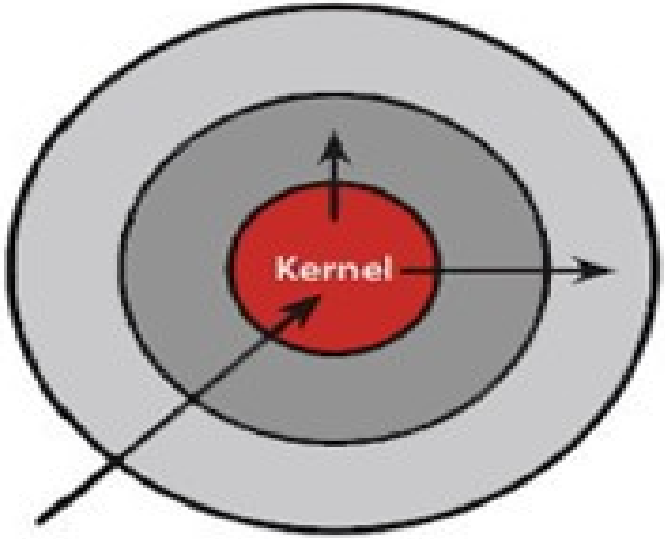
Mandatory access contrôle

- Modèle de sécurité qui restreint l'accès aux ressources système en fonction de politiques définies de manière centralisée.
- Sous MAC, les autorisations d'accès sont prédéterminées par les administrateurs et ne peuvent pas être modifiées par les utilisateurs individuels.
- MAC améliore la sécurité en appliquant une politique d'accès stricte à l'échelle du système, ce qui le rend particulièrement adapté aux environnements de haute sécurité.


SUBJECT
httpd
UID=12
GID=12
rw-r----
UID=0
GID=0
/cour/TP
OBJECT
politique
et
règles
rw-r--r--
UID=0
GID=0
/etc/host
OBJECT
rw-r--r--
UID=0
GID=0
/etc/passwd
OBJECT

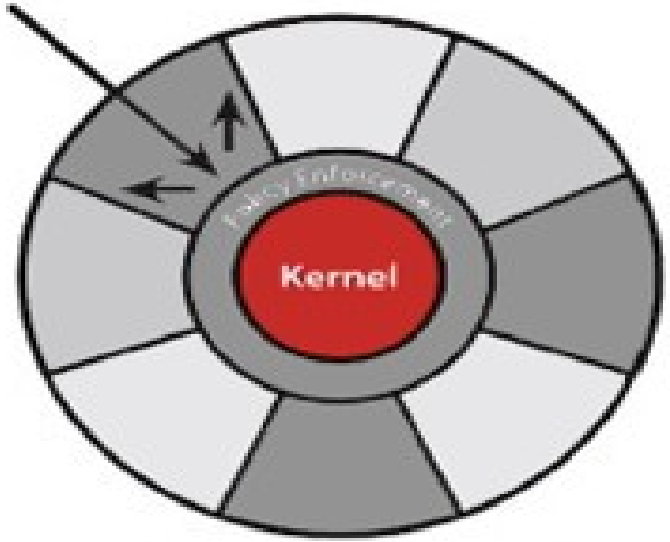
MAC

DAC **VS** **DAC + MAC**



Discretionary Access Control

Once a security exploit gains access to priveleged system component, the entire system is compromised.



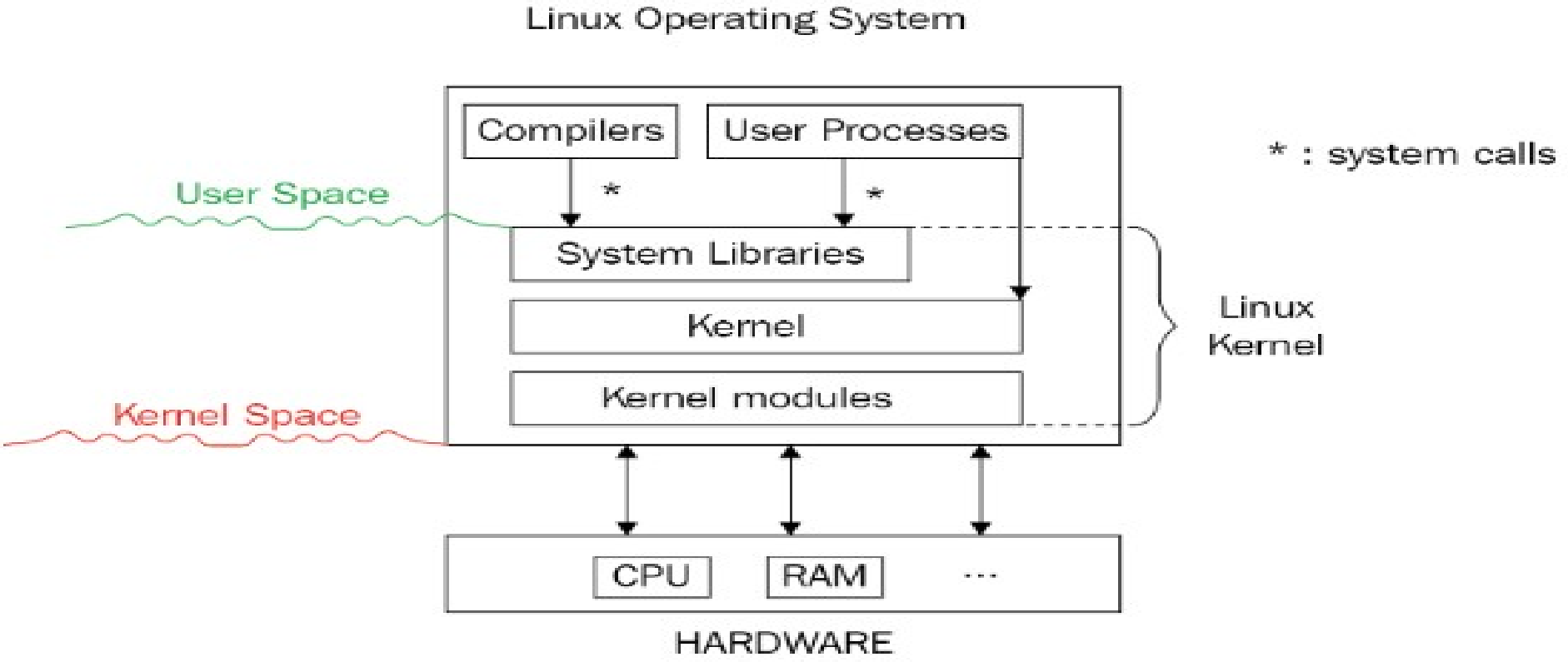
Mandatory Access Control

Kernel policy defines application rights, firewalling applications from compromising the entire system.

Modules de Sécurité Linux

Modules de noyau Linux :

- Les modules du noyau sont des blocs de code qui peuvent être chargés et déchargés dynamiquement dans le noyau Linux.
- Ils permettent d'étendre ou d'améliorer le noyau avec des fonctionnalités supplémentaires sans qu'il soit nécessaire de reconstruire ou de redémarrer l'intégralité du noyau .


Linux Operating System
Compilers
User Processes
* : system calls
User Space
*
*
System Libraries
Kernel
Linux
Kernel
Kernel modules
Kernel Space
CPU
RAM
...
HARDWARE

Modules de sécurité Linux (LSM):

- sont des infrastructures du noyau Linux qui permet aux développeurs de mettre en œuvre différentes politiques de sécurité au niveau du noyau.
- fournissent une interface standardisée permettant d'ajouter des modules de sécurité au noyau Linux sans avoir à modifier le code source du noyau lui-même.

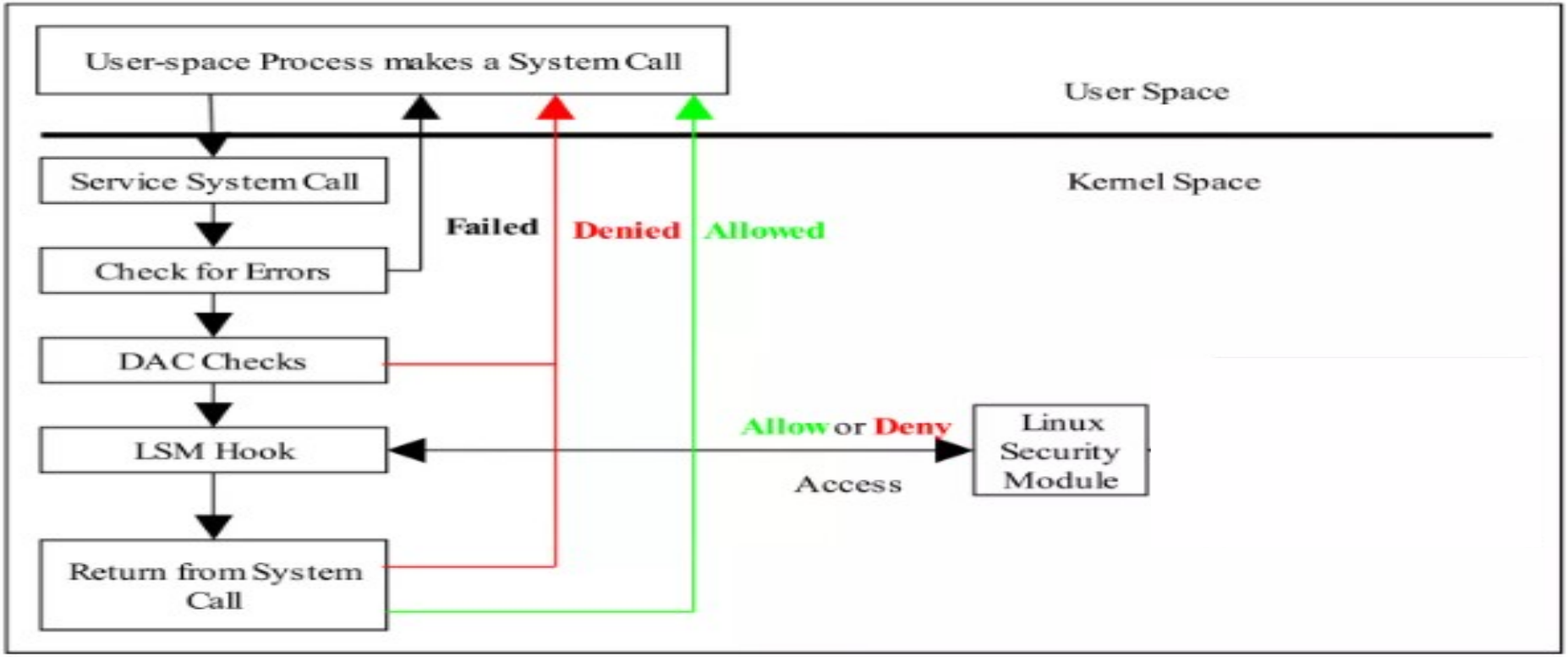


Figure 2.3: Processing a System Call – *The DAC checks are carried out first, if they pass then the Security Server is consulted for a decision.*

LSM

Modules de sécurité Linux (LSM):

Parmi les quatre LSM officiels :

- **SELinux** (Redhat ,Fedora, Opensuse)
- `AppArmor`: **Profile based model**(Debian , Ubuntu)
- Smack
- TOMOYO

SELinux se distingue en offrant des contrôles d'accès obligatoires (MAC),fournit un mécanisme pour prendre en charge les ***politiques*** de sécurité du contrôle d'accès

politique

Security-Enhanced Linux

“Toutes les actions de lecture et d'écriture de données sont contrôlées par une politique de sécurité “

Politique :

“ un plan ou un principe d'action adopté ou proposé par une organisation ou un individu.”

- c’est l'ensemble des règles d'accès aux données

- Sur un **système d'accès obligatoire MAC** , il existe une politique qui est définie et fixée administrativement
- Même si vous modifiez les paramètres **DAC** (**r-w-x**) sur votre répertoire personnel, s'il existe une politique en place empêchant un autre utilisateur ou processus d'y accéder, généralement votre système est sécurisé.

- Ces politiques peuvent être très fines.
- Elles peuvent être définies pour déterminer l'accès entre :
    - ➔ users
    - ➔ files
    - ➔ directories
    - ➔ ports
    - ➔ …etc

il y a deux politiques que vous verrez généralement :

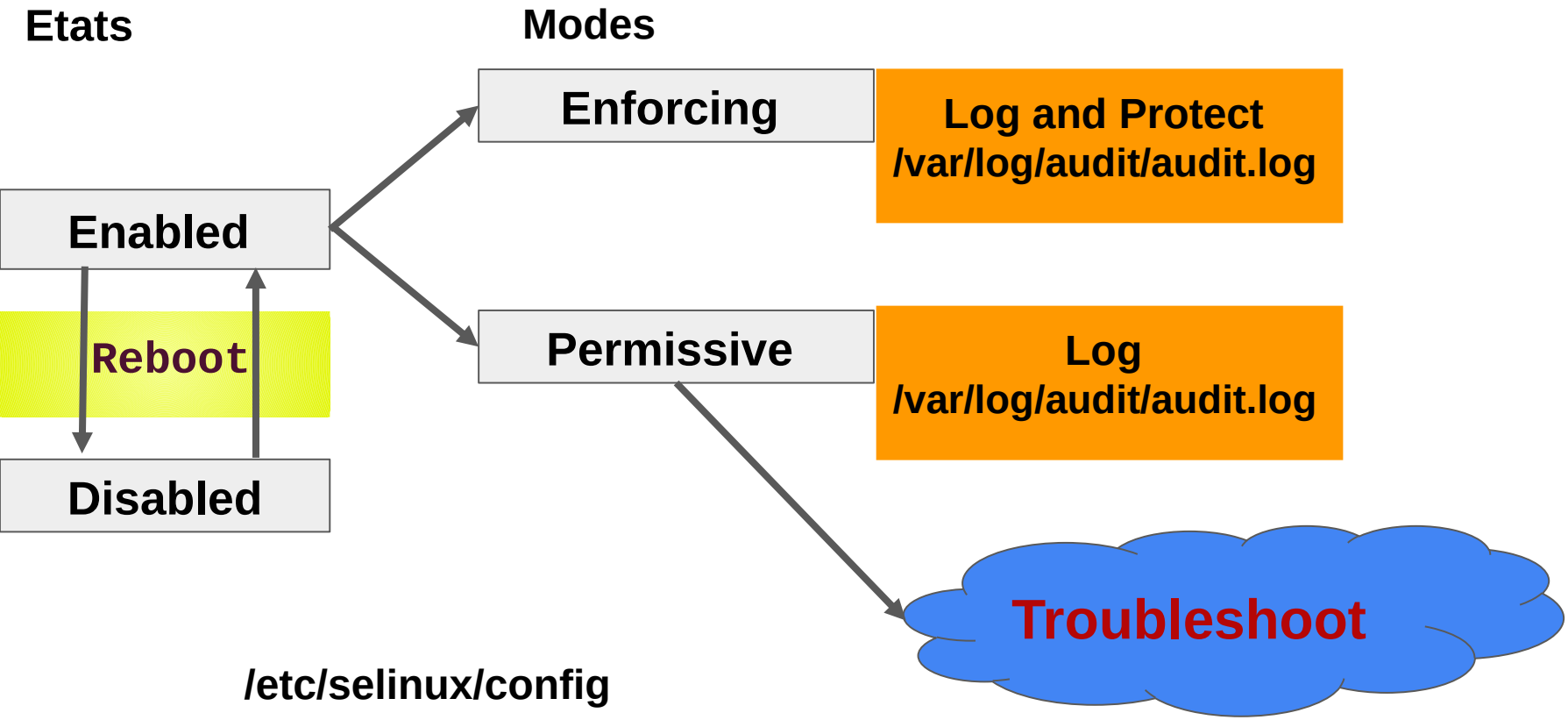
- ❑ **Targeted** - la politique par défaut, seuls les processus ciblés (il y en a des centaines) sont protégés par **selinux**. tout le reste n'est pas confiné
- ❑ **MLS** -Multi-level security
- ➔ Hors de portée de la présentation d'aujourd'hui.
- ➔ Cela peut être très compliqué.
- ➔ Couramment utilisé dans l'armée et les gouvernements.

VOUS POUVEZ DÉTERMINER QUELLE POLITIQUE VOTRE SYSTÈME EST CONFIGURÉ À UTILISER

/etc/selinux/config lié symboliquement à :
/etc/sysconfig/selinux

```
# SELINUXTYPE= can take one of these three values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are pro>
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

ETATS ET MODES DE SELINUX

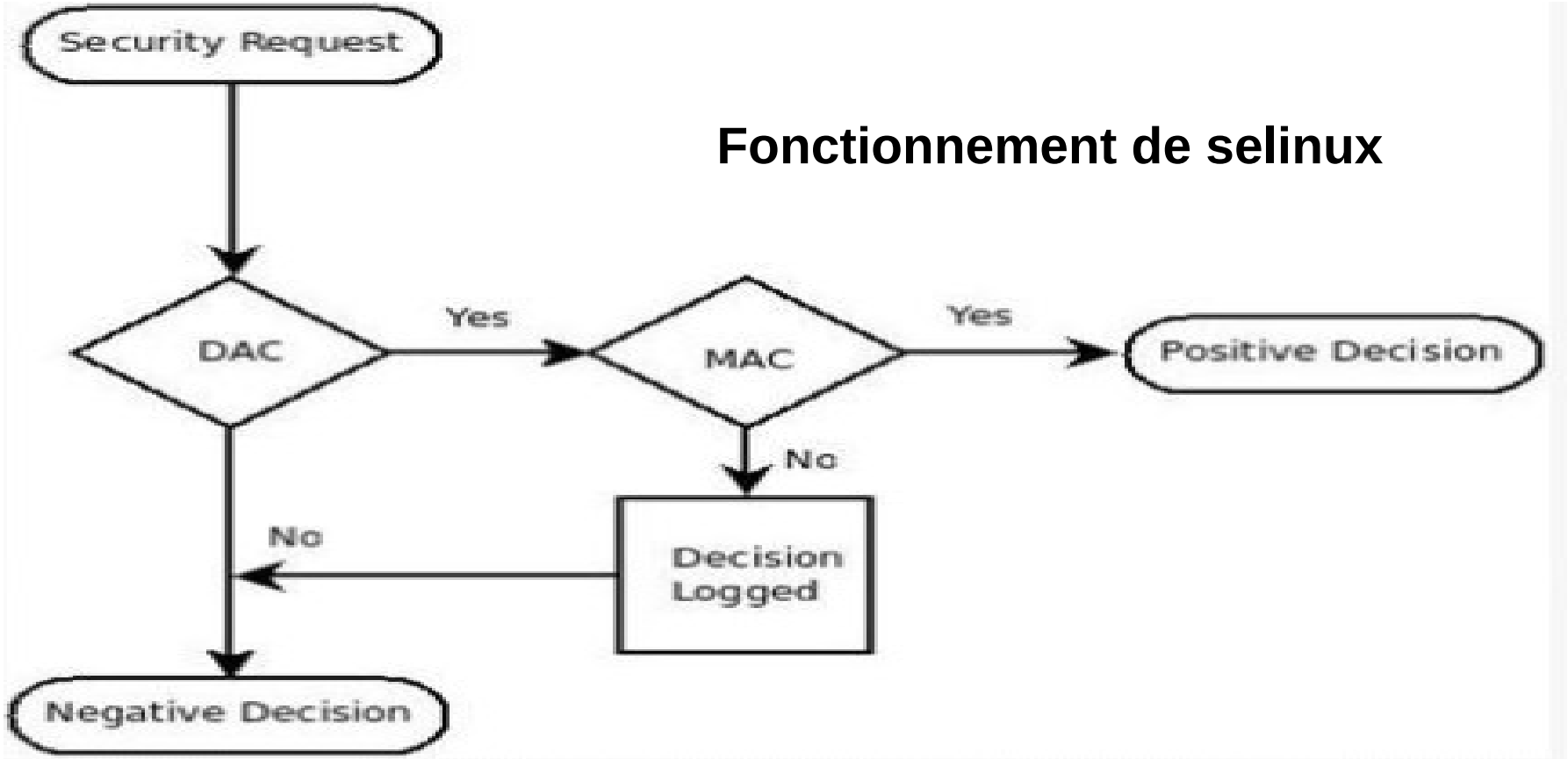

Etats
Modes
Enforcing
Log and Protect
/var/log/audit/audit.log
Enabled
Reboot
Permissive
Log
/var/log/audit/audit.log
Disabled
Troubleshoot
/etc/selinux/config

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
# See also:
# https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-selinux/#>
#
#    grubby --update-kernel ALL --args selinux=0
#
# To revert back to SELinux enabled:
#
#    grubby --update-kernel ALL --remove-args selinux
#
SELINUX=enforcing
```

setenforce 0 → permissive mode
setenforce 1 → enforcing mode
sestatus pour Afficher le mode courant

Fonctionnement de Selinux

Fonctionnement de selinux



Fonctionnement de SElinux

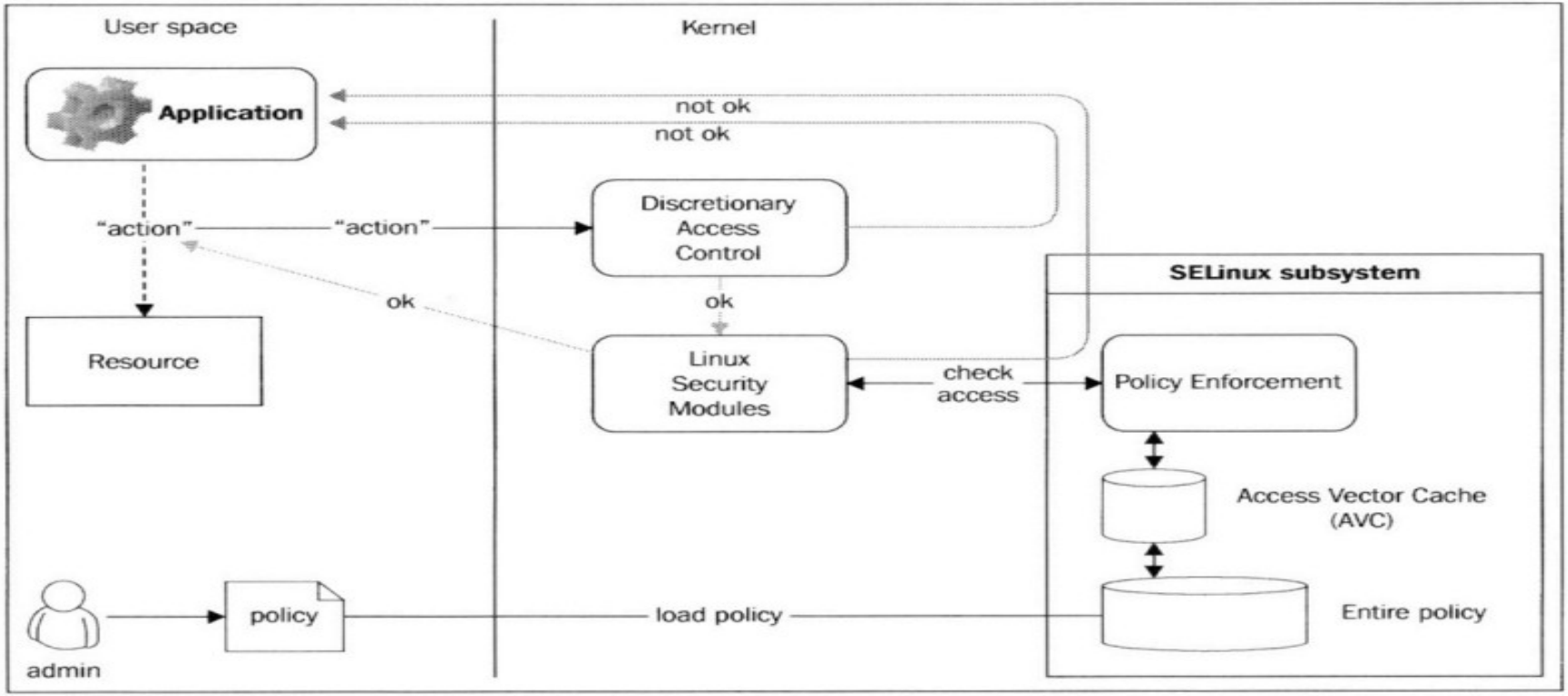


Image from "SELinux Cookbook" by Sven Vermeulen

❑ *L’avantage :*

- **<u>Protection contre les exploits Zero Day</u> :** SELinux peut améliorer la sécurité d'un système en atténuant l'impact des exploits Zero Day.

  Même si un attaquant obtient un accès non autorisé à un système, SELinux peut restreindre les actions que les processus compromis peuvent effectuer. Cela rend plus difficile pour les attaquants d’exploiter les vulnérabilités et d’élever les privilèges.

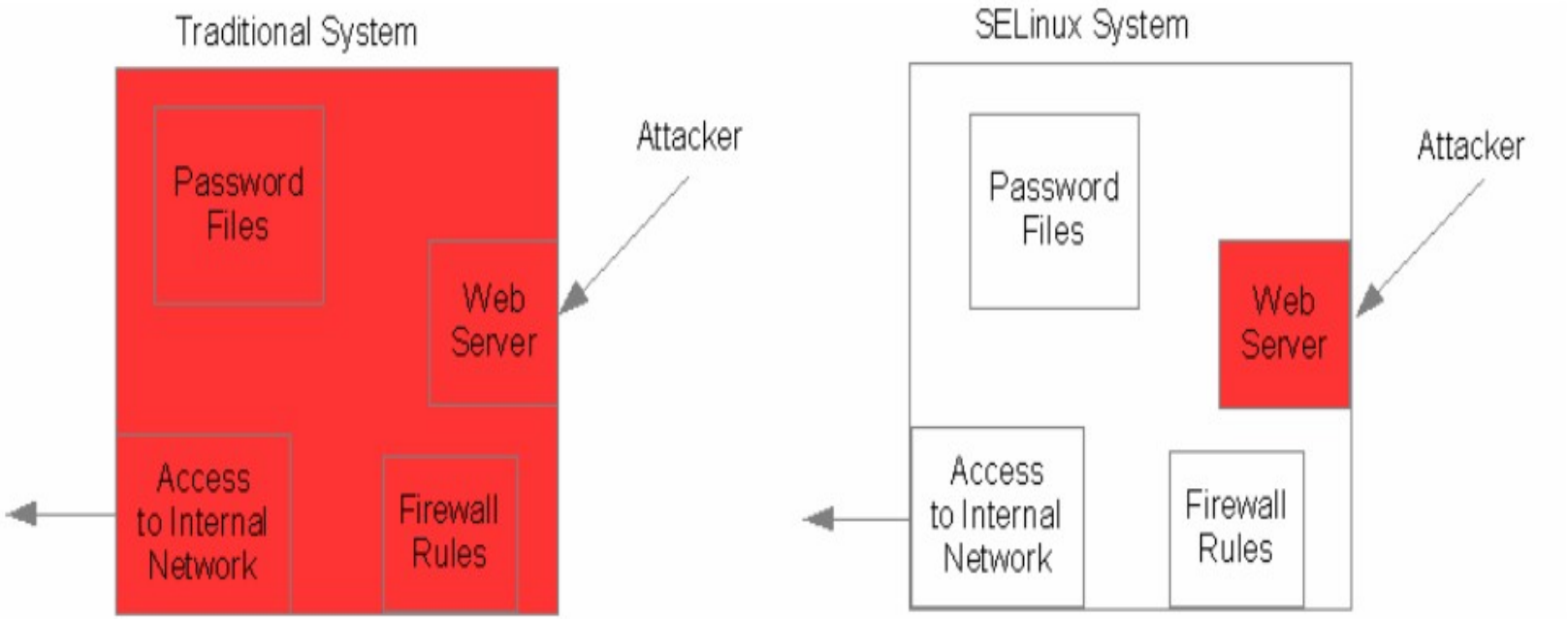


圖 2、傳統 Linux 與 SELinux 安全機制比較圖

❑ *Le but de SElinux :*

- SELinux n'a pas pour objectif d'empêcher l'attaque elle-même. Le but est de placer le processus sous contrôle d'accès obligatoire (MAC) et de limiter le comportement même si le processus est compromis ou si le privilège root est pris.

Alors, comment fonctionne Selinux ?

Deux concepts importants à comprendre avec selinux sont :

- **Labeling** (étiquetage/ type)
- **Type enforcement**(application du type)

❑ **Labeling** (Type)

➔ **Toutes les fichiers, processus,port ... sont étiquetés avec un contexte selinux.**

➔ **Pour les fichiers et répertoires, ces étiquettes sont stockées sur le système de fichiers.**

/etc/selinux/targeted/contexts/files/

➔ **Pour les processus et les ports ..etc, le noyau gère ces étiquettes.**

les étiquettes sont au format :

[User : Role : Type/Domain : level /Sensitivity range]

- **Labeling** (Type): **command+ (-Z)**

Context

[User : Role : Type/Domain : level /Sensitivity range]

ce qui nous intéresse dans cette présentation, c'est le Type

❑ **Labeling** (Type)

User_u : Role_r : Type/Domain_t : level /Sensitivity range]

EXEMPLES

❑ **Labeling** (Type)

★ Nous examinons bien un service assez complexe, qui fournit un accès depuis le réseau et a potentiellement accès à l'ensemble du système de fichiers.

★ **Apache Web server** n'est pas nécessairement non sécurisé, il est simplement très varié dans son accès.

★ **Apache** dispose d'un exécutable binaire qui se lance depuis **/usr/sbin**

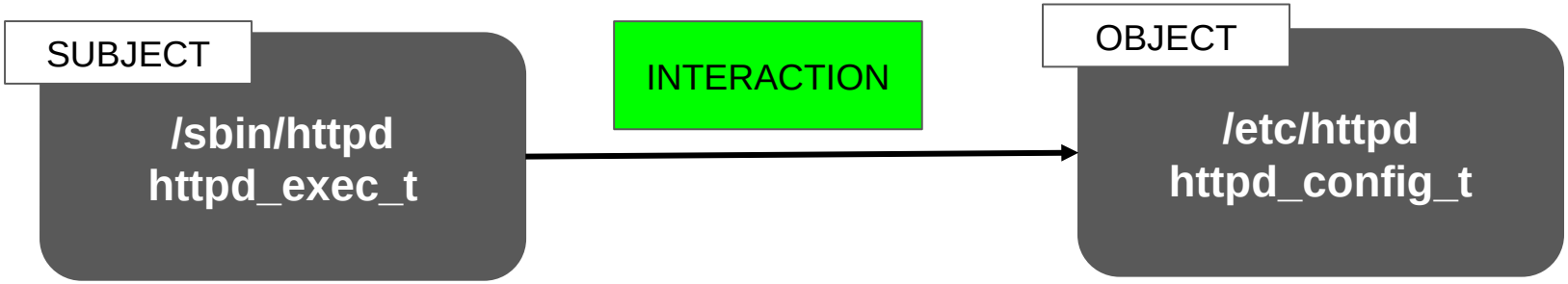
★ quand vous regardez le contexte de ce fichier, vous voyez que son type est **httpd_exec_t**

❑ **Labeling** (Type)

Apache

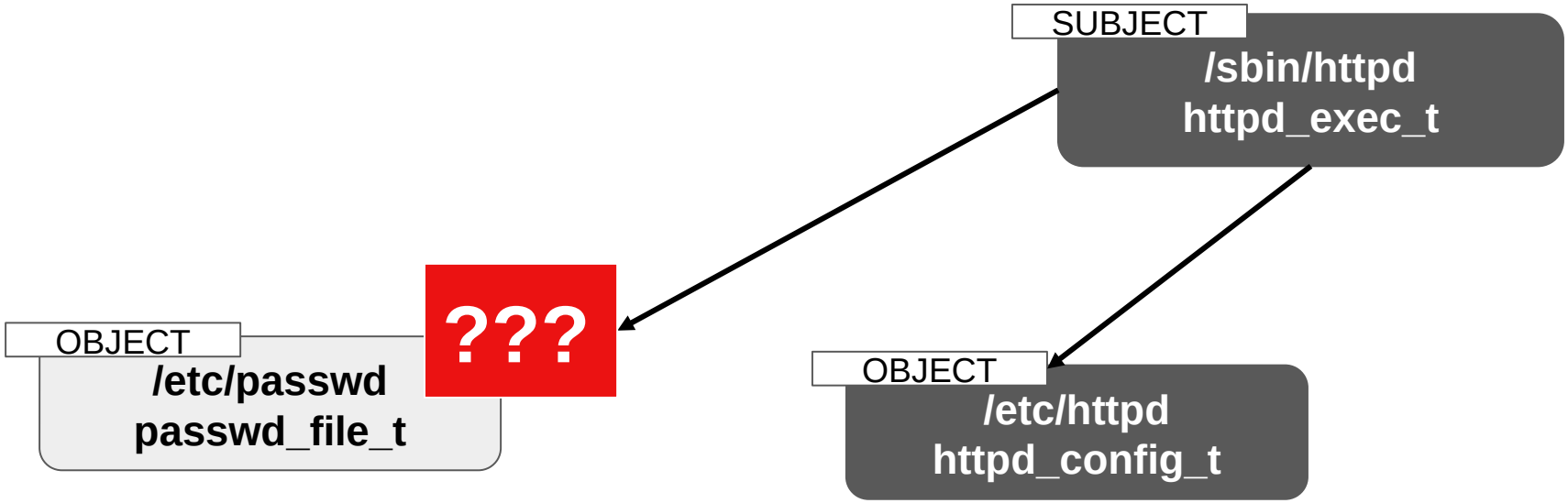
- ★ configuration répertoire est de type **httpd_config_t**
- ★ log file (fichier journal) est de type **httpd_log_t**
- ★ content (contenu) **httpd_sys_content_t**
- ★ startup-script **httpd_initrc_exec_t**
- ★ Pendant que le serveur Web s'exécute, son processus est de type **httpd_t**

❑ **Labeling** (Type)



❑ **Type enforcement**(application du type)

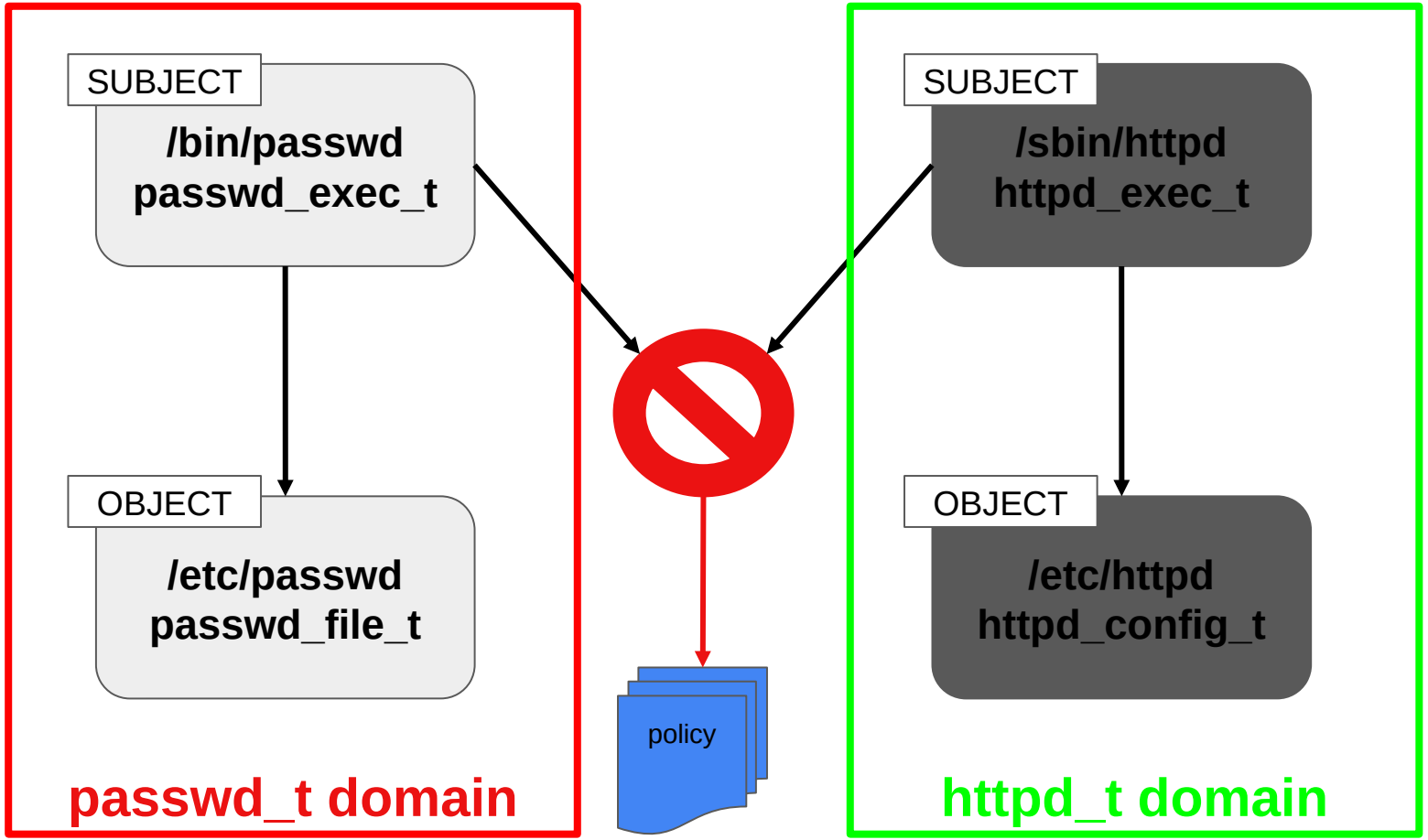
pensez-vous qu'il est logique qu'un processus exécuté avec **httpd_t** context puisse interagir avec un fichier avec **passwd_file_t** context ?



❑ **Type enforcement**(application du type)

la réponse est NON

L'APPLICATION DE TYPE est la partie de la politique qui dit, "Seul un processus étiqueté **exemple_t** peut avoir un accès en lecture à un fichier étiqueté **exemple_config_t** “


SUBJECT
/bin/passwd
passwd_exec_t
OBJECT
/etc/passwd
passwd_file_t
passwd_t domain
policy
SUBJECT
/sbin/httpd
httpd_exec_t
OBJECT
/etc/httpd
httpd_config_t
httpd_t domain

Gestion des étiquettes

❑ Gestion des étiquettes

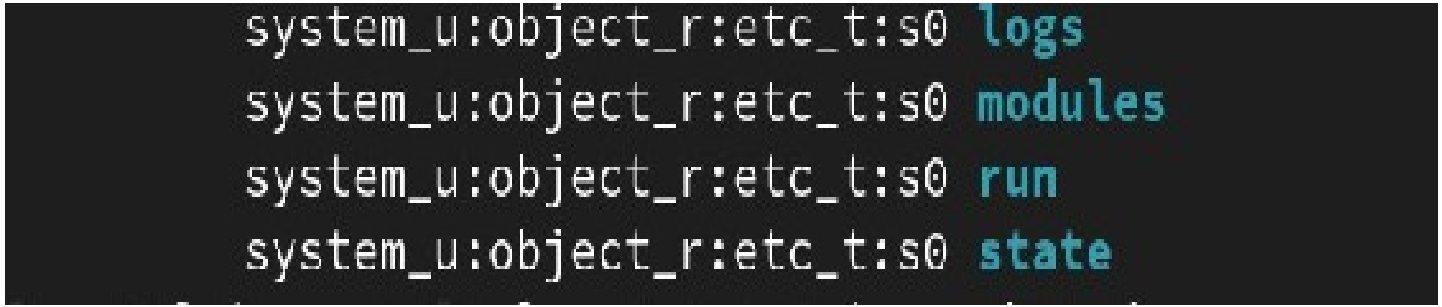
l'ARGUMENT -Z

De nombreuses commandes acceptent cet argument

- ❑ **ls -Z**
- ❑ **id -Z**
- ❑ **ps -Z**
- ❑ **netstat -Z**
- ❑ **cp -Z**
- ❑ **mkdir -Z**

❑ Gestion des étiquettes

- vous pouvez utiliser des commandes selinux comme **chcon** , **restorecon** pour changer le contexte d'un fichier.
- le contexte est défini lors de la création des fichiers, en fonction de leur répertoire parent (à quelques exceptions près).
- ls -Z /etc



❑ Gestion des étiquettes

(L'exception)

Transition de fichier (file transition) :

- **Défini par la politique .**
- **Si une application X_t crée un fichier dans le répertoire étiqueté Y_t , la politique peut exiger une transition pour que le fichier soit créé avec l'étiquette Z_t**

❑ Gestion des étiquettes

Transition de fichier (file transition) :

- **EXEMPLE :** processus , **dhclient** en cours d'exécution de type **dhclient_t ,** créer un fichier **resolve.conf** de type **net_conf_t** dans le rep **/etc** de type **etc_t** .
- Sans la **transition de fichier /etc/resolve.conf** *aurait hérité* **etc_t .**

Gestion des étiquettes : cp vs mv

Booleans

booleans

- *les booléens ne sont que des paramètres on/off pour Selinux*
- *des choses simples comme permettre à FTP d'accéder au répertoire personnel*
- *pour afficher tous les booléens, exécutez **getsebool -a***

booleans

End .