Formalisation et garantie de propriétés de sécurité système: Application à la détection d'intrusions

Jérémy Briffaut

LIFO Université d'Orléans - Equipe SDS
ENSI de Bourges

13 décembre 2007

Sommaire

Sommaire

Contexte de la thèse

- Garantir des propriétés de sécurité sur le système
- Permettre à un administrateur
  - Définir les propres propriétés de sécurité dont il a besoin
- Intégrité
  - $ssh \xrightarrow{executer} bash \xrightarrow{su} root \xrightarrow{ecrire} shadow$
  - Cas particuliers : *Biba, non-interférence, etc.*
- Confidentialité
  - $apache \xrightarrow{ecrire} tmp \xrightarrow{lire} malware \xrightarrow{signal} bash$
  - Cas particuliers : *Bell & LaPadula, flux de référence, etc.*

Travaux existants

- Contrôle d'accès et administration de politique
  - Définit politique de protection système
  - Biba [Biba 75], Bell & LaPadula [BLP 73]
  - Gestion de conflits [Bertino 96] [Moffett 93] [Sloman 94] [Cuppens 04]
- Détection d'intrusion orientée politique système
  - Détection par spécification : respect d'une politique [Ko 94]
  - Non-interférence : intégrité d'exécution [Ko Redmond 02]
  - Flux de référence : confidentalité entre domaines [Zimmermann 03]
- Cas particuliers de confidentialité et d'intégrité

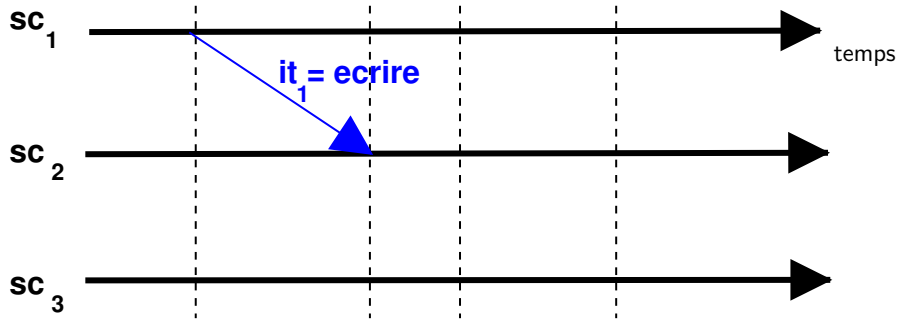
Bilan

- Pas d'approche orientée propriétés de sécurité
  - Manque d'une méthode de protection ou de détection
  - Prendre en compte toutes les propriétés nécessaires (non-supportées, supportées et des nouvelles)
- Manque un formalisme pour définir des propriétés de sécurité
  - Formaliser la dépendance entre appels système
  - Formaliser la notion de séquence

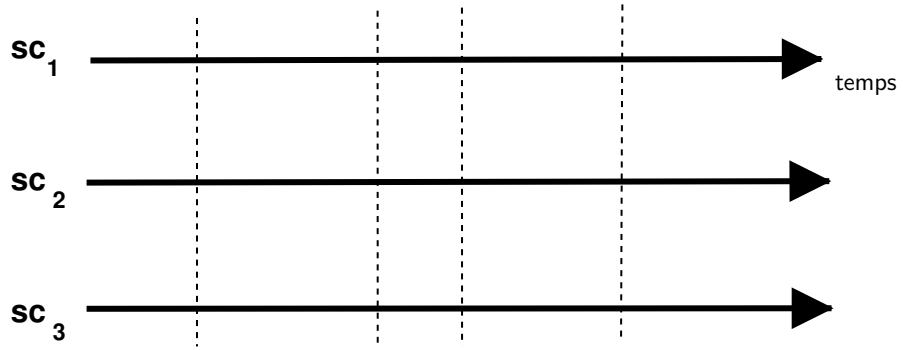
Sommaire

Dépendance causale entre Interactions

- Confidentialité : sc_3 ne doit pas obtenir d'information de sc_1

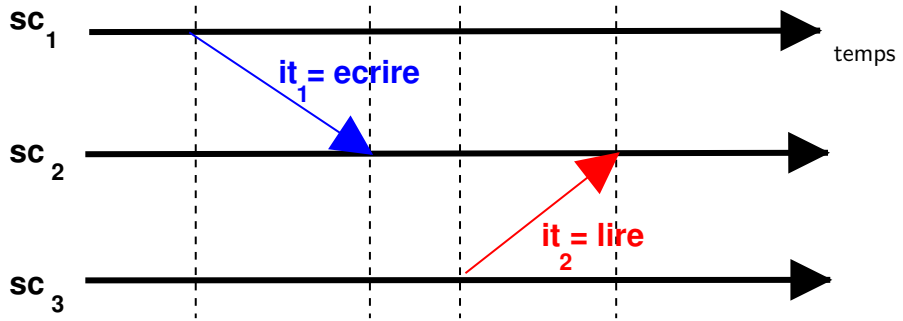


- Intégrité : sc_1 ne doit pas modifier sc_3

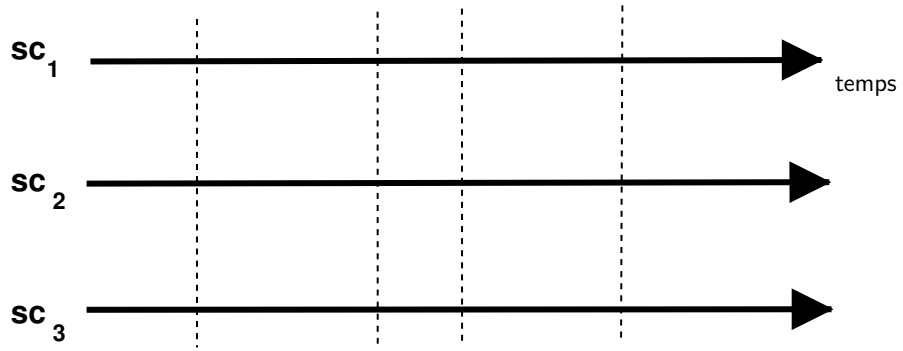


Dépendance causale entre Interactions

- Confidentialité : sc_3 ne doit pas obtenir d'information de sc_1

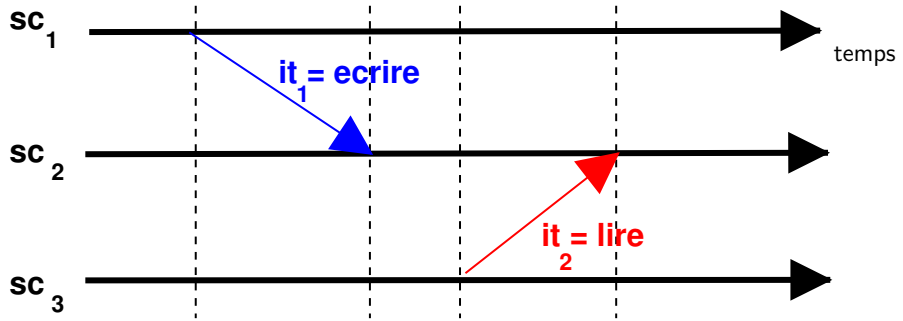


- Intégrité : sc_1 ne doit pas modifier sc_3

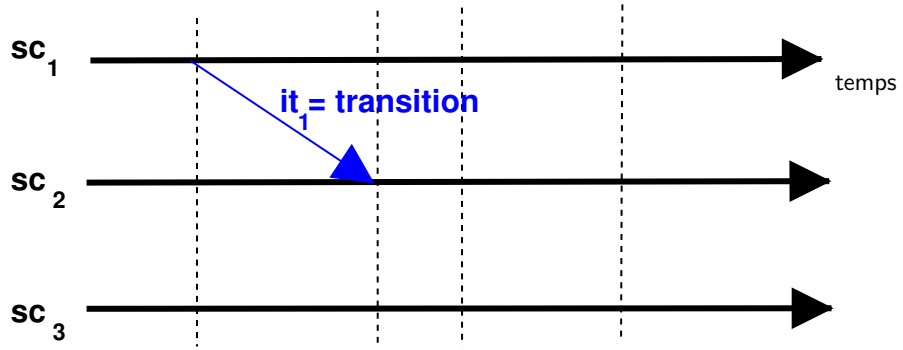


Dépendance causale entre Interactions

- Confidentialité : sc_3 ne doit pas obtenir d'information de sc_1

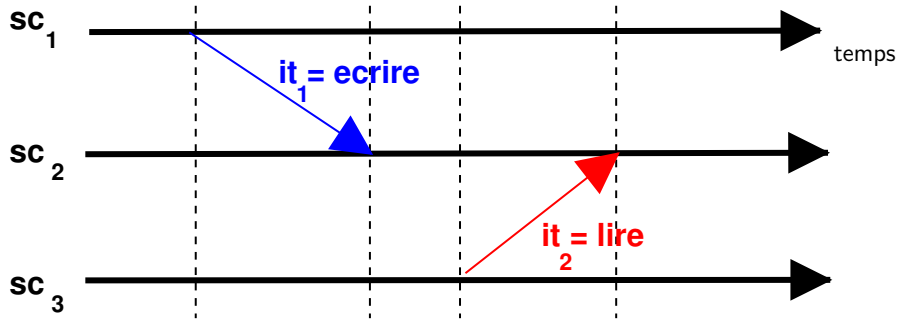


- Intégrité : sc_1 ne doit pas modifier sc_3

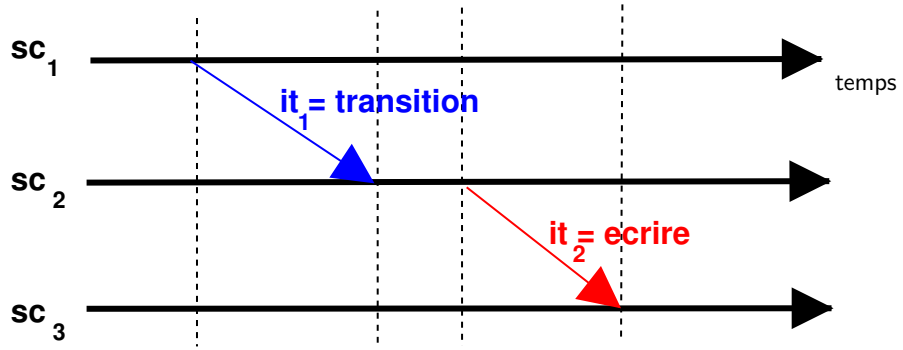


Dépendance causale entre Interactions

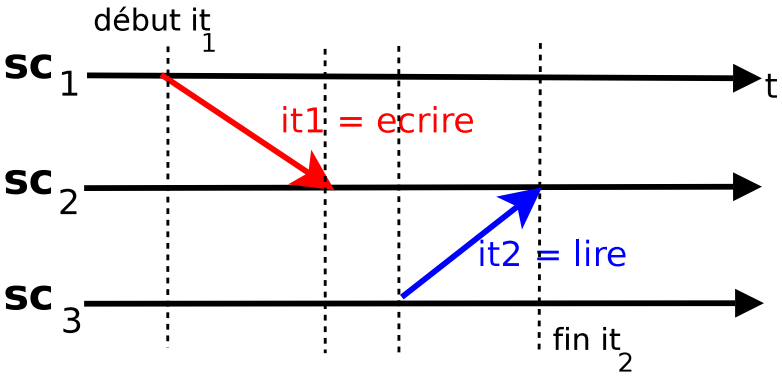
- Confidentialité : sc_3 ne doit pas obtenir d'information de sc_1

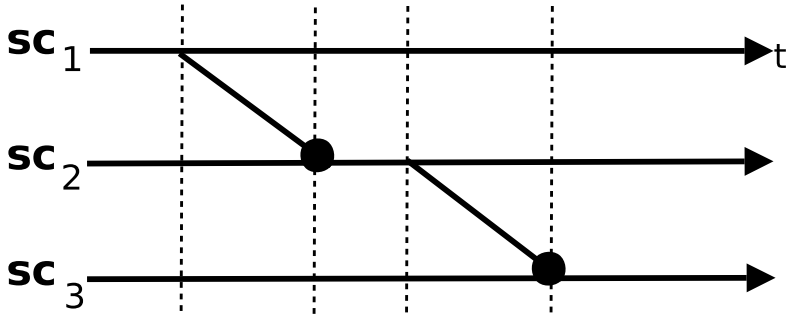


- Intégrité : sc_1 ne doit pas modifier sc_3



Dépendance causale





- Définition de la dépendance causale :

$$it_1 \twoheadrightarrow it_2 \quad \equiv_{def} \quad \begin{cases} sc_2 \in it_1, \\ sc_2 \in it_2, sc_3 \in it_2, \\ t_{sc_2}(debut(eo_1)) \leqslant t_{sc_2}(fin(eo_2)), \\ it_1 \text{ modifie l'état du contexte partagé } sc_2, \\ it_2 \text{ modifie l'état du contexte } sc_3 \end{cases}$$

Langage de description d'activités

- Séquence d'interactions $sc_{source} \Rightarrow sc_{cible}$: fermeture transitive d'interactions causalement liées

$$\{it_1, \ldots, it_n\}, \begin{cases} n \geq 2, \\ sc_{source} \in it_1, \\ sc_{cible} \in it_n, \\ \forall k = 1..n-1, it_k \twoheadrightarrow it_{k+1} \end{cases}$$

- Exemple : $ssh \Rightarrow shadow$

$(ssh \xrightarrow{executer} bash) \twoheadrightarrow (bash \xrightarrow{su} root) \twoheadrightarrow (root \xrightarrow{ecrire} shadow)$

- Langage de description d'activités manipule :
  - Interactions et Séquences (Terminaux du langage)
  - Leurs combinaisons à l'aide d'opérateurs : $\circ, \vee, \wedge$

Langage de description d'activités

- Séquence d'interactions $sc_{source} \Rightarrow sc_{cible}$: fermeture transitive d'interactions causalement liées

$$\{it_1, \ldots, it_n\}, \begin{cases} n \geq 2, \\ sc_{source} \in it_1, \\ sc_{cible} \in it_n, \\ \forall k = 1..n-1, it_k \twoheadrightarrow it_{k+1} \end{cases}$$

- Exemple : $ssh \Rightarrow shadow$

$(ssh \xrightarrow{executer} bash) \twoheadrightarrow (bash \xrightarrow{su} root) \twoheadrightarrow (root \xrightarrow{ecrire} shadow)$

- Langage de description d'activités manipule :
  - Interactions et Séquences (Terminaux du langage)
  - Leurs combinaisons à l'aide d'opérateurs : $\circ, \vee, \wedge$

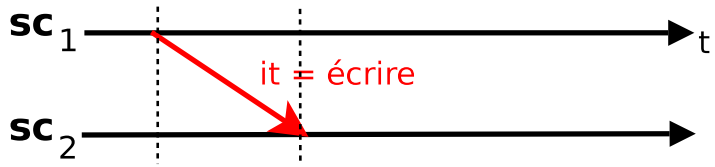
Formalisation

- Formaliser à l'aide du langage de description d'activités
- 13 propriétés définies suivant trois axes

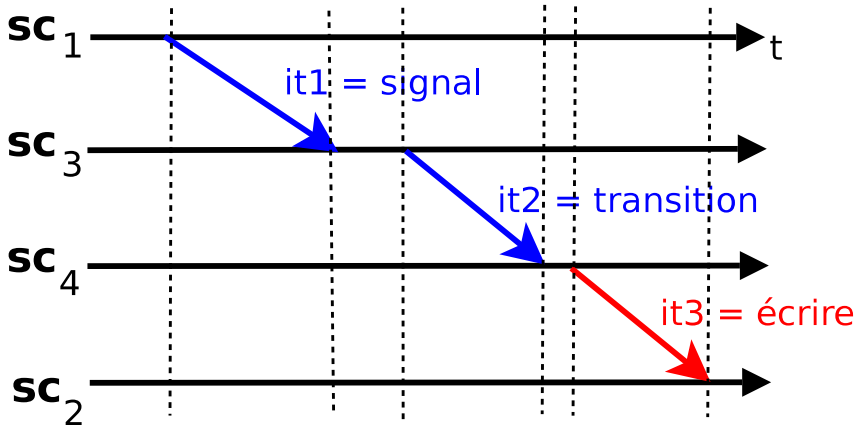
Catégorie	Propriété	Support Complet/Partiel
Intégrité	Intégrité des objets	osiris, samhain
	Biba	SELinux
	Intégrité des sujets	non-interférence
	Intégrité des domaines	TCB, SELinux
Confidentialité	Flux d'information	Slat, flux de référence
	Bell&LaPadula	
	Bell&LaPadula restrictif	SELinux
	Cohérence d'accès aux données	
Abus de Privilège	Séparation de privilèges	SELinux, grsecurity
	Absence de changement de contexte	
	Exécutables de confiances (TPE)	SELinux, grsecurity
	Respect de la politique	SELinux, grsecurity
	Respect d'une Méta-Politique	

Exemple : intégrité

- Violation d'intégrité : 1 interaction



- Violation d'intégrité : 1 séquence d'interactions

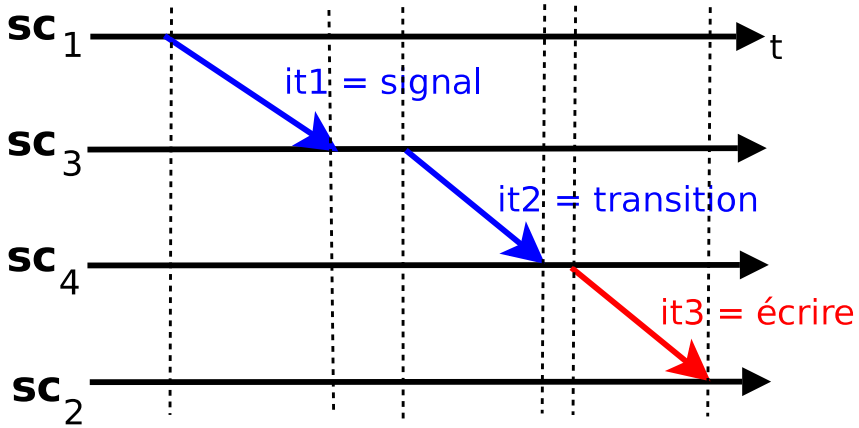


Exemple : intégrité

- Intégrité directe (*interaction*)

$$C_{int1}(sc_1, sc_2) \text{ est vraie } \textbf{ssi}$$
$$\forall eo \in IS \textbf{ t.q. } sc_1 \xrightarrow{eo} sc_2, \neg is\_write\_like(eo)$$

- Violation d'intégrité : 1 séquence d'interactions



Exemple : intégrité

- Intégrité directe (*interaction*)

$$C_{int1}(sc_1, sc_2) \text{ est vraie } \textbf{ssi}$$
$$\forall eo \in IS \textbf{ t.q. } sc_1 \xrightarrow{eo} sc_2, \neg is\_write\_like(eo)$$

- Intégrité indirecte (*séquence*)

$$C_{int2}(sc_1, sc_2) \text{ est vraie } \textbf{ssi}$$
$$\forall eo \in IS \textbf{ t.q. } sc_1 \Rightarrow_{eo} sc_2, \neg is\_write\_like(eo)$$

Exemple : intégrité

- Intégrité directe (*interaction*)

$$C_{int1}(sc_1, sc_2) \text{ est vraie } \textbf{ssi}$$
$$\forall eo \in IS \textbf{ t.q. } sc_1 \xrightarrow{eo} sc_2, \neg is\_write\_like(eo)$$

- Intégrité indirecte (*séquence*)

$$C_{int2}(sc_1, sc_2) \text{ est vraie } \textbf{ssi}$$
$$\forall eo \in IS \textbf{ t.q. } sc_1 \Rightarrow_{eo} sc_2, \neg is\_write\_like(eo)$$

- Propriété d'intégrité

$$P_{integrity}(sc_1, sc_2) \text{ est vraie } \textbf{ssi}$$
$$C_{int1}(sc_1, sc_2) \wedge C_{int2}(sc_1, sc_2)$$

Bilan

- Langage de description d'activités
  - Adapté à la formalisation des propriétés
  - Permet de définir
    - Toutes les propriétés connues
    - De les étendre
    - D'en définir de nouvelles

Permet de définir une méthode générale

Pour garantir toutes les propriétés définies dans le langage ou en détecter les violations

Bilan

- Langage de description d'activités
  - Adapté à la formalisation des propriétés
  - Permet de définir
    - Toutes les propriétés connues
    - De les étendre
    - D'en définir de nouvelles

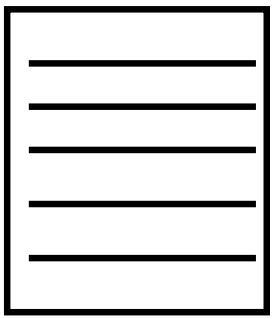
Permet de définir une méthode générale

Pour garantir toutes les propriétés définies dans le langage ou en détecter les violations

Sommaire

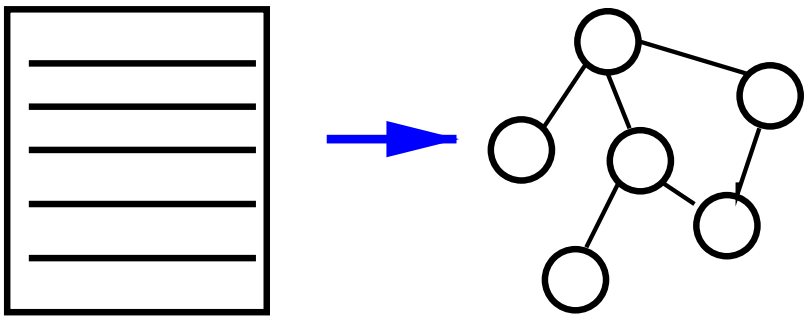
Principe général

1. Calcul d'une politique de protection système
2. Construction d'un graphe de dépendance causale
   - Contient toutes les séquences observables
3. Analyse du graphe pour extraire les activités illicites



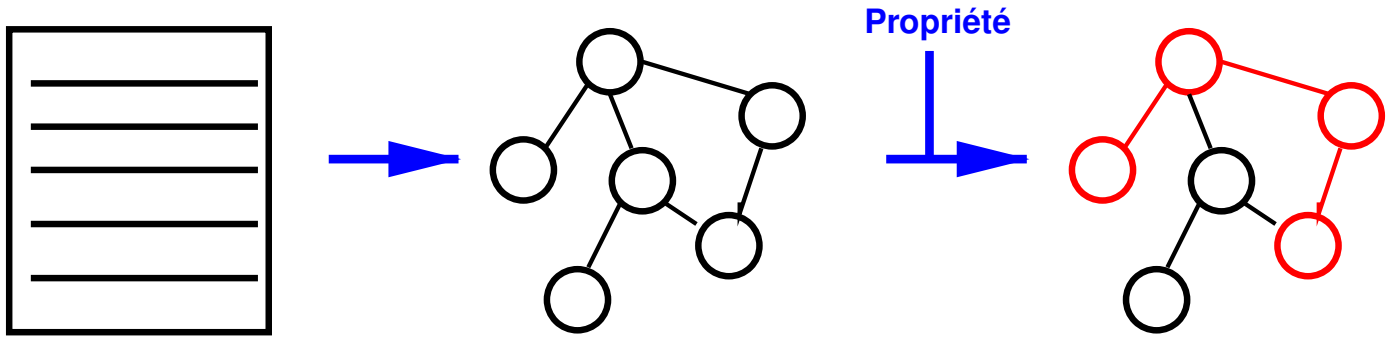
Principe général

1. Calcul d'une politique de protection système
2. Construction d'un graphe de dépendance causale
   - Contient toutes les séquences observables
3. Analyse du graphe pour extraire les activités illicites



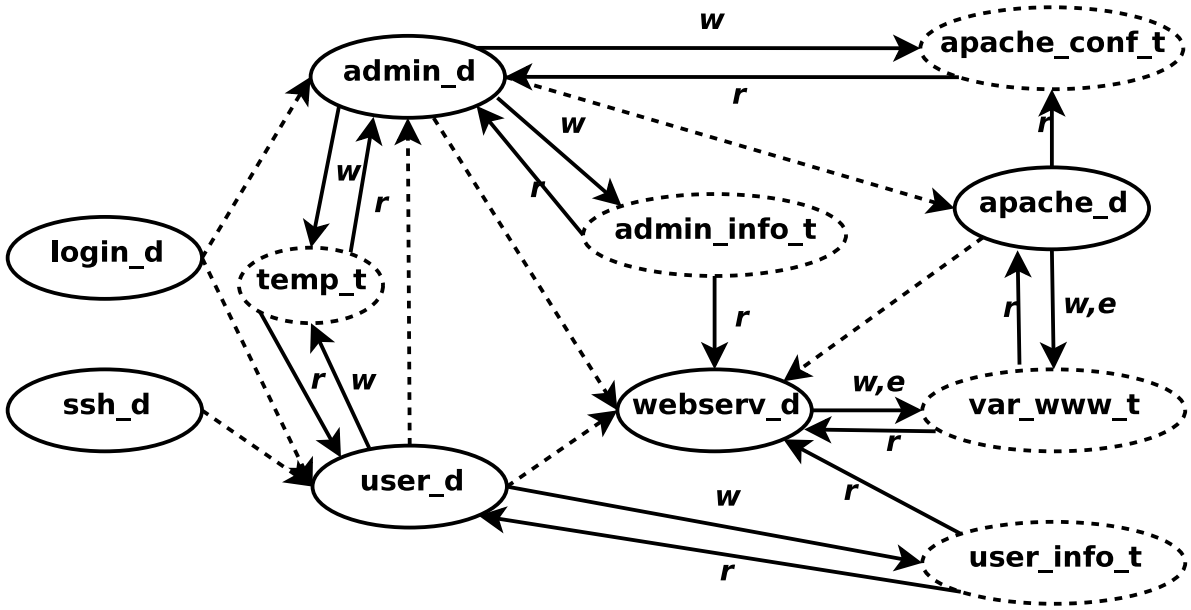
Principe général

1. Calcul d'une politique de protection système
2. Construction d'un graphe de dépendance causale
   - Contient toutes les séquences observables
3. Analyse du graphe pour extraire les activités illicites



Graphe de dépendance causale

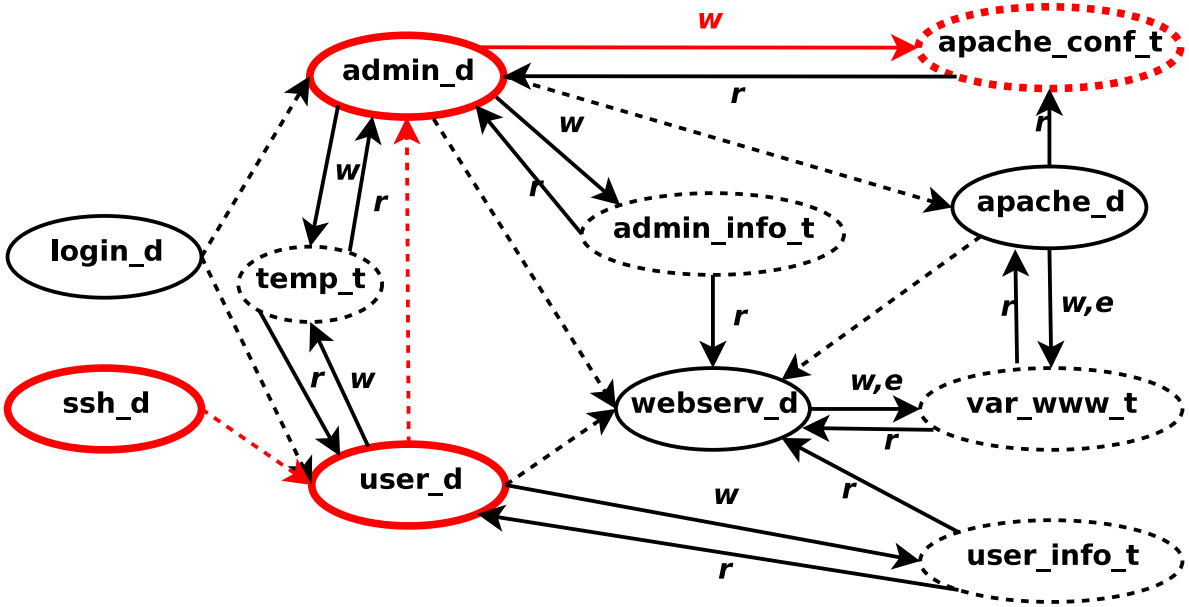
- Représente les *dépendances causales* entres interactions



- Propriété d'intégrité : integrity(*ssh_d*,*apache_conf_t*)

Graphe de dépendance causale

- Représente les *dépendances causales* entres interactions



- Propriété d'intégrité : integrity(*ssh_d*,*apache_conf_t*)

Théorème d'équivalence

Théorème d'équivalence entre chemins et séquences

- Toute séquence observable correspond à un chemin
- Tout chemin dans le graphe correspond à une séquence

Type de Graphe	Arc	Chemin
Dépendance causale	*Dépendance causale (2 arcs)* $it_1 \twoheadrightarrow it_2$	Séquence d'interactions $sc_{source} \Rightarrow sc_{cible}$
Flux d'informations	Transfert d'informations $sc_1 > sc_2$	Flux d'informations $sc_{source} \gg sc_{cible}$
Transition	Transition $sc_1 \xrightarrow{trans} sc_2$	Séquence de transitions $sc_{source} \Rightarrow_{trans} sc_{cible}$

Théorème d'équivalence

Théorème d'équivalence entre chemins et séquences

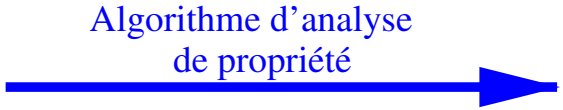
- Toute séquence observable correspond à un chemin
- Tout chemin dans le graphe correspond à une séquence

Type de Graphe	**Arc**	**Chemin**
Dépendance causale	*Dépendance causale (2 arcs)* $it_1 \twoheadrightarrow it_2$	Séquence d'interactions $sc_{source} \Rightarrow sc_{cible}$
Flux d'informations	Transfert d'informations $sc_1 > sc_2$	Flux d'informations $sc_{source} \gg sc_{cible}$
Transition	Transition $sc_1 \xrightarrow{trans} sc_2$	Séquence de transitions $sc_{source} \Rightarrow_{trans} sc_{cible}$

Analyse des propriétés de sécurité

- Un algorithme d'énumération par propriété
  - Utilise les graphes de dépendance causale
  - Énumérer toutes les activités illicites
  - Une énumération : une activité violant la propriété

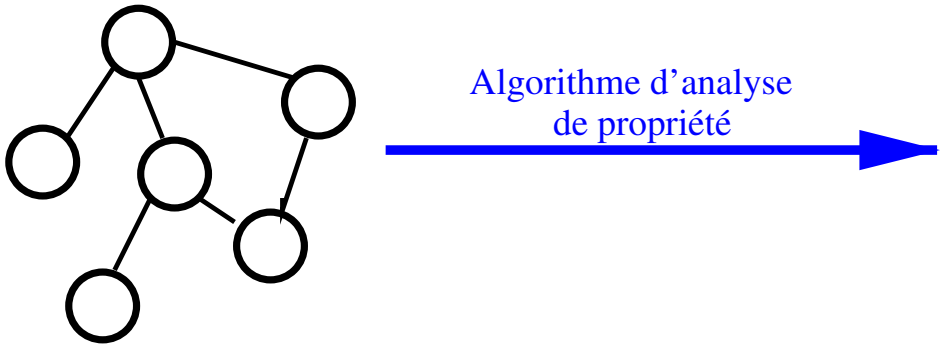
$\longrightarrow$ Algorithmes définis pour les 13 propriétés de sécurité



Analyse des propriétés de sécurité

- Un algorithme d'énumération par propriété
  - Utilise les graphes de dépendance causale
  - Énumérer toutes les activités illicites
  - Une énumération : une activité violant la propriété

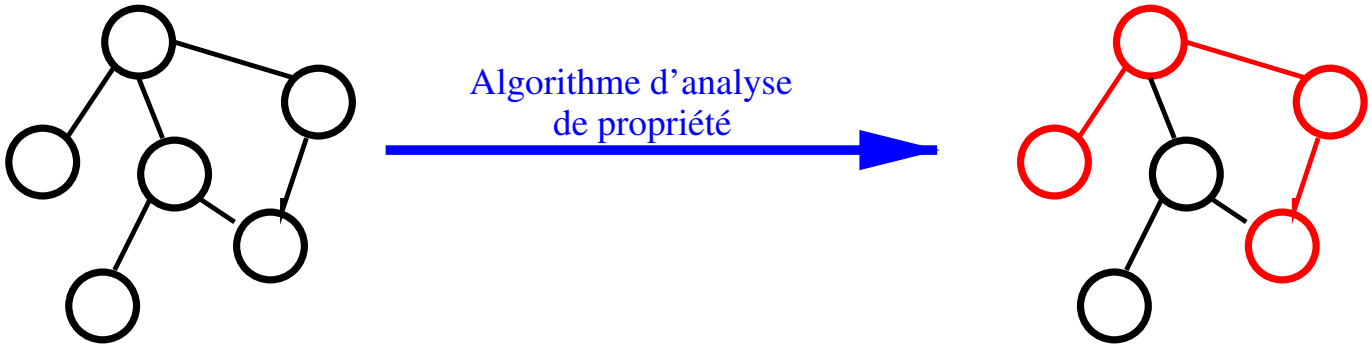
$\longrightarrow$ Algorithmes définis pour les 13 propriétés de sécurité



Analyse des propriétés de sécurité

- Un algorithme d'énumération par propriété
  - Utilise les graphes de dépendance causale
  - Énumérer toutes les activités illicites
  - Une énumération : une activité violant la propriété

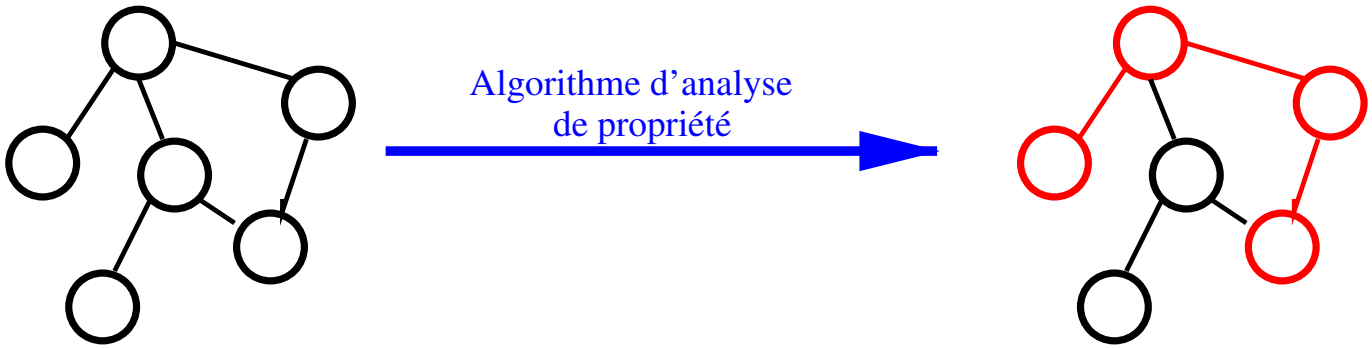
$\longrightarrow$ Algorithmes définis pour les 13 propriétés de sécurité



Analyse des propriétés de sécurité

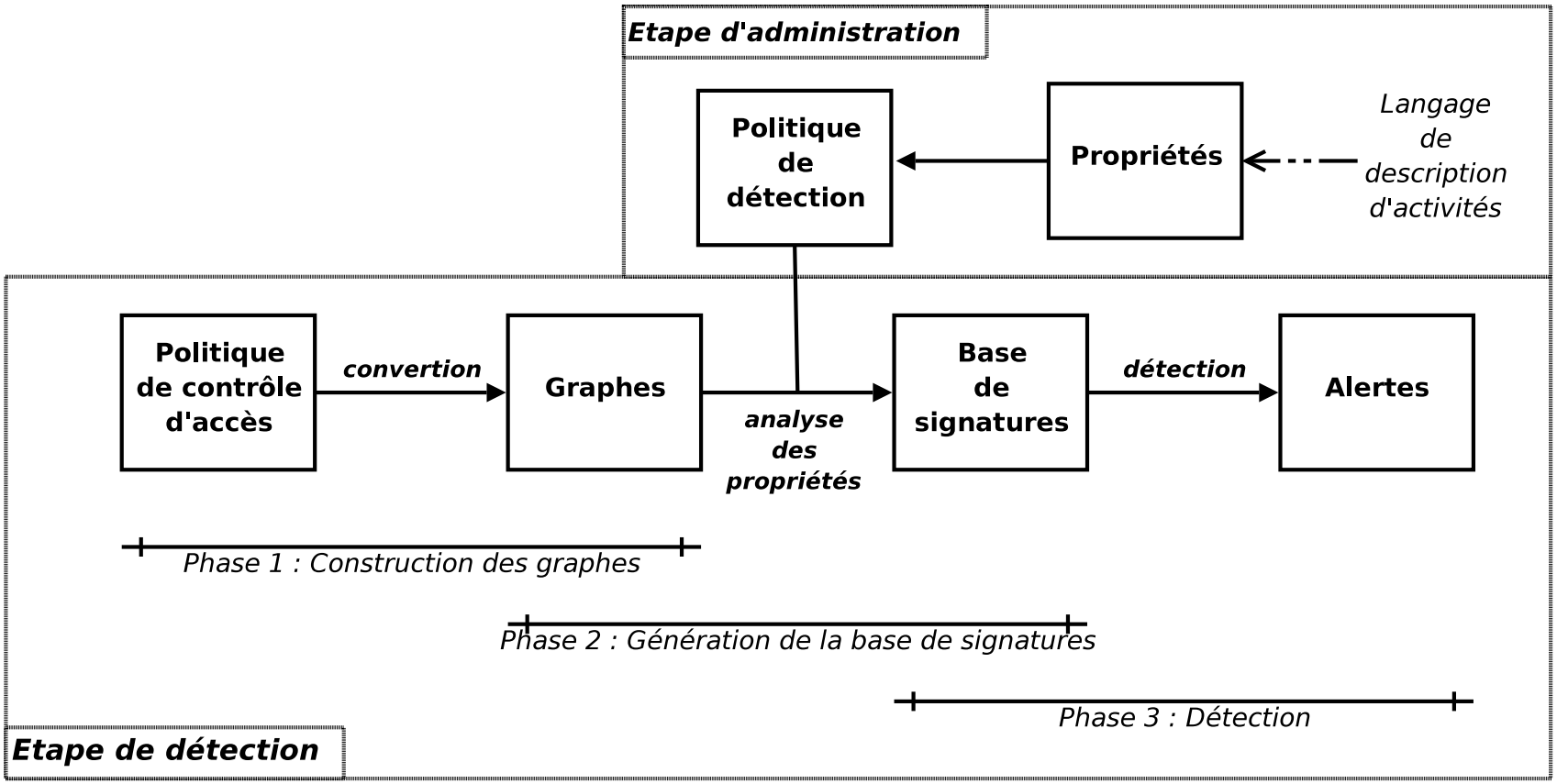
- Un algorithme d'énumération par propriété
  - Utilise les graphes de dépendance causale
  - Énumérer toutes les activités illicites
  - Une énumération : une activité violant la propriété

$\longrightarrow$ Algorithmes définis pour les 13 propriétés de sécurité

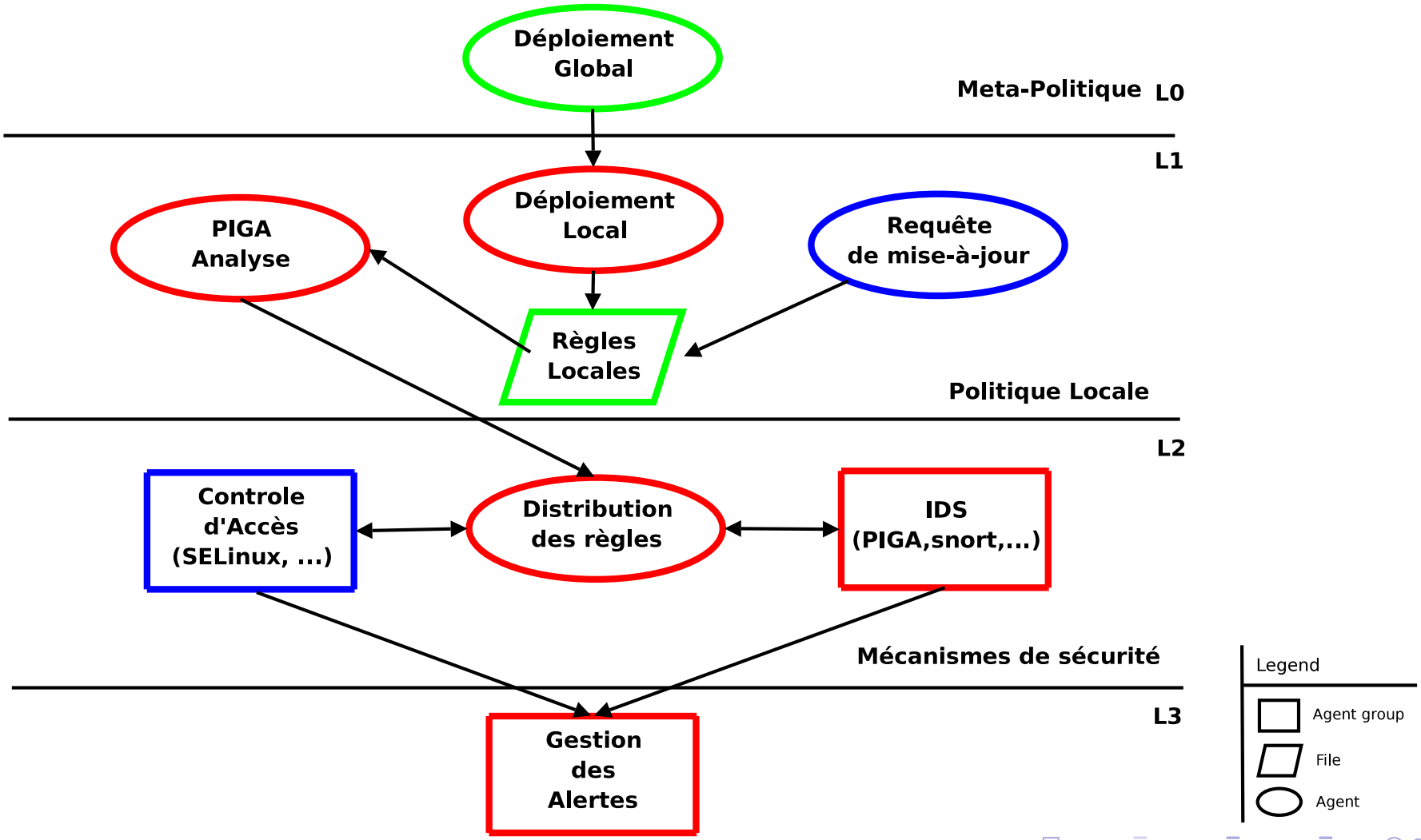


Sommaire

Modèle de détection d'intrusion



Sommaire

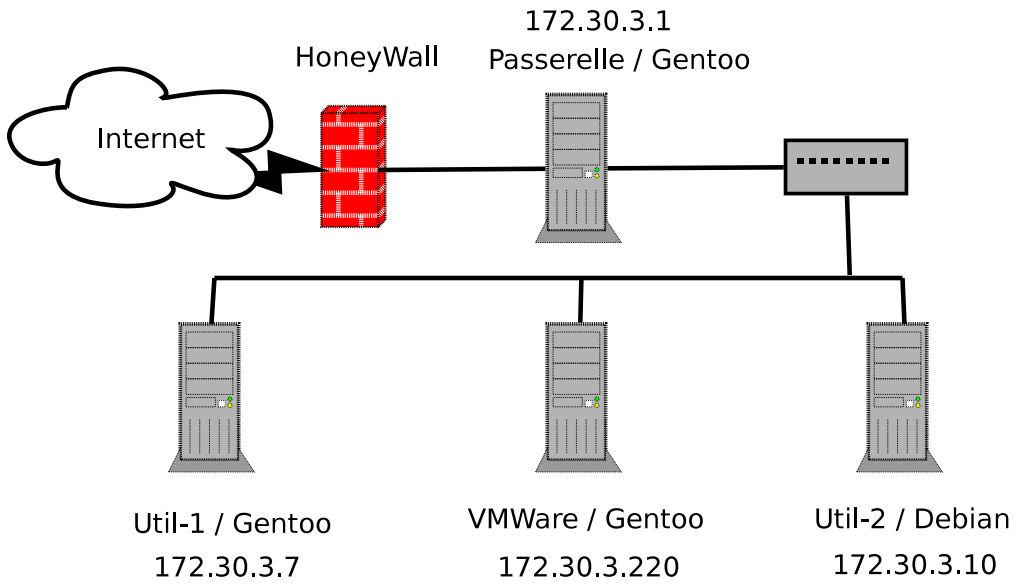

Déploiement Global
Meta-Politique
L0
L1
Déploiement Local
PIGA Analyse
Requête de mise-à-jour
Règles Locales
Politique Locale
L2
Controle d'Accès (SELinux, ...)
Distribution des règles
IDS (PIGA,snort,...)
Mécanismes de sécurité
L3
Gestion des Alertes
Legend
Agent group
File
Agent

PIGA-IDS

- Application à deux systèmes de Contrôle d'Accès Mandataire
  - SELinux et GRsecurity
- 2 Composants (implanté en Java)
  - Analyse et génération de la base de signatures (Phase 1 2)
  - Détection des activités illicites (Phase 3)

Pot-de-miel

- Plate-forme utilisée
  - Politique de détection spécifique aux pots-de-miel
  - SSH ouvert sur Internet
  - 4 machines sous SELinux
  - Durée de l'expérimentation 1 an



Politique de détection

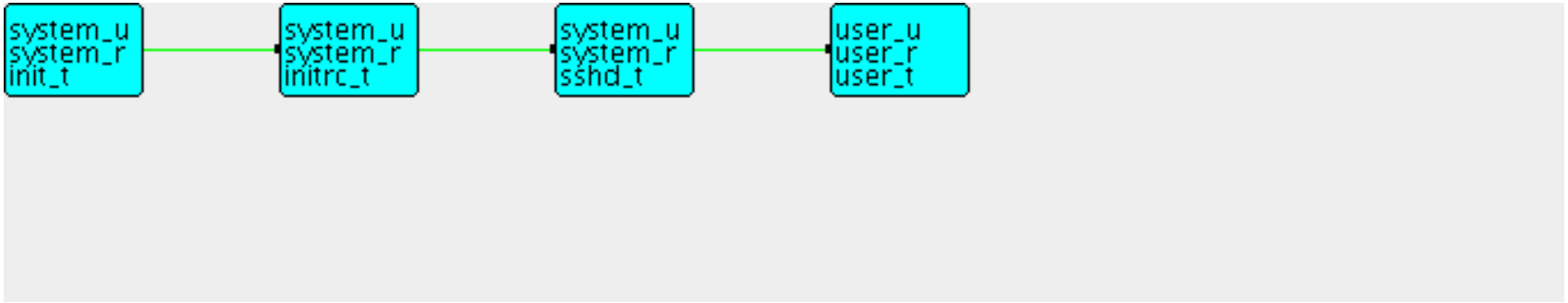
- Spécifique au cas des pots-de-miel
- Détecter les abus des utilisateurs
  - Intégrité des contextes exécutables
    - integrity($SC_{.*:.*:user.*}$,SC_{exec})
  - Intégrité du domaine utilisateur
    - int_domain($SC_{.*:.*:user.*}$)
  - Confidentialité du système vis-à-vis des utilisateurs
    - confidentiality($SC_{Systeme}$, $sc_{.*:.*:user.*}$)
  - Séparation des privilèges de modification et d'exécution
    - duties_sep($SC_{Systeme}$)
  - Transitions vers le domaine utilisateur interdites
    - bad_transition($.* : .* : user.*$, $SC_{Systeme}$)
  - Respect de la politique de contrôle d'accès
    - conformity()

Analyse des propriétés

- Phase d'analyse des politiques de contrôle d'accès

Nom		Passerelle	Util-1	VMware	Util-2
Graphe	*SC*	577	3017	624	595
	IV	17 684	314 582	21 359	18 215
Signatures	*integrity*	137	9 461	186	140
	int_domain	16 283	510 215	18 130	16 546
	confidentiality	29 510	726 842	29 510	29 510
	duties_sep	243	16 405	320	270
	bad_transition	3555	126 228	4250	3941
	Total	49 728	1 389 151	52 396	50 407
	Durée de l'analyse	47s	10min31s	1min2s	52s

Exemple d'attaque

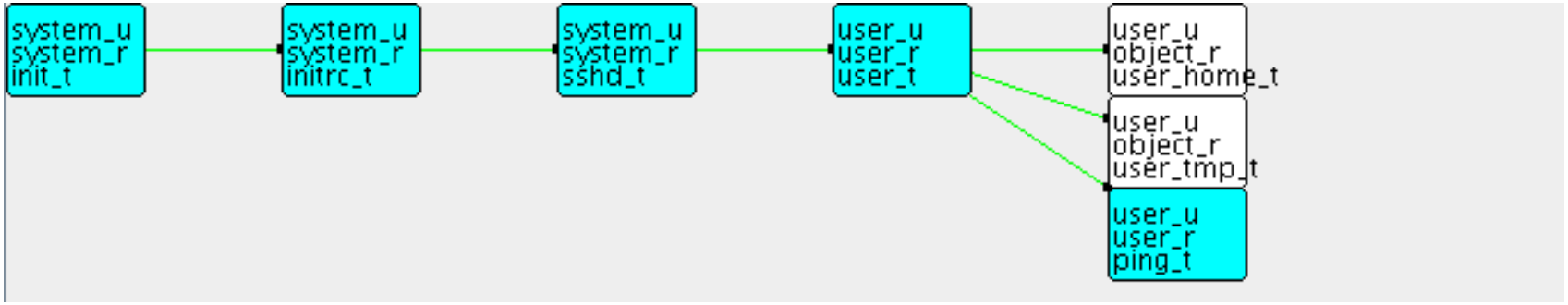


Opération de l'attaquant

17 :05 Connexion ssh

Changement de contexte interdit (domaine utilisateur)

Exemple d'attaque

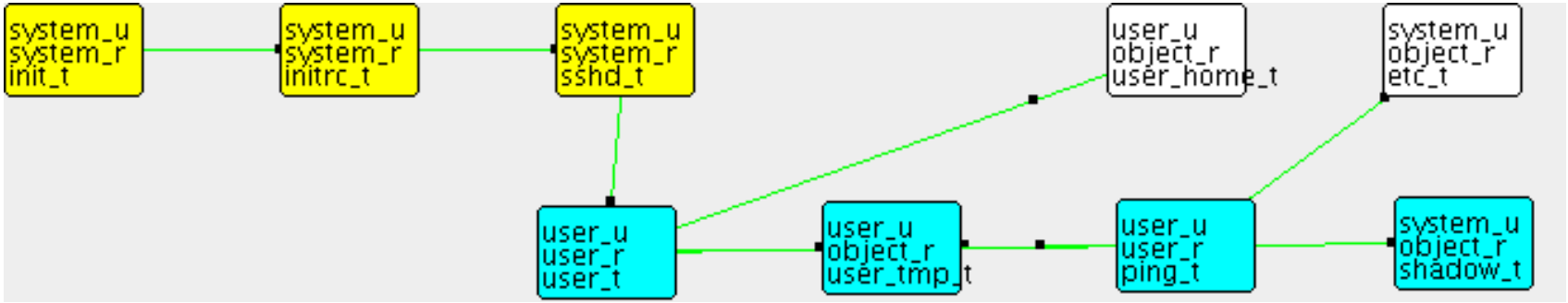


Opération de l'attaquant

17 :06 Analyse de la machine

Changement de contexte interdit (utilisation de ping)

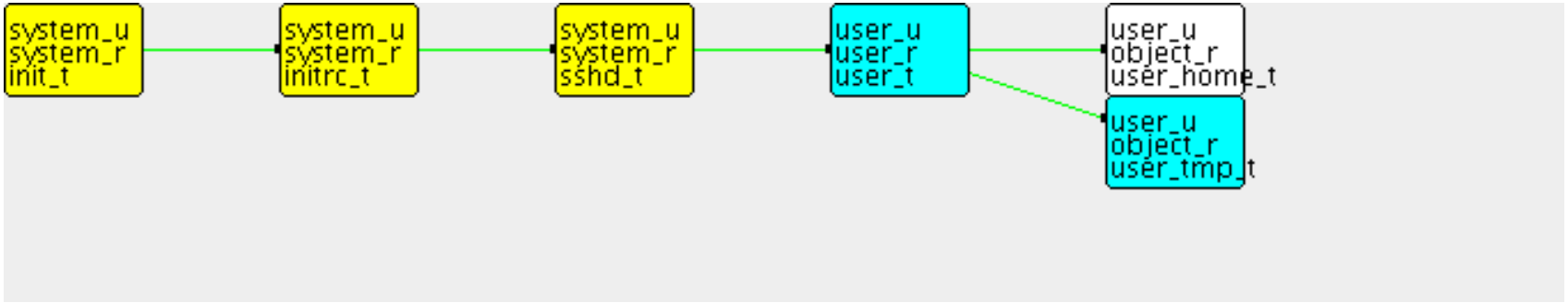
Exemple d'attaque



Opération de l'attaquant

17 :07 Attaque sur une faille de ping : obtention de */etc/shadow*
Violation de la confidentialité système

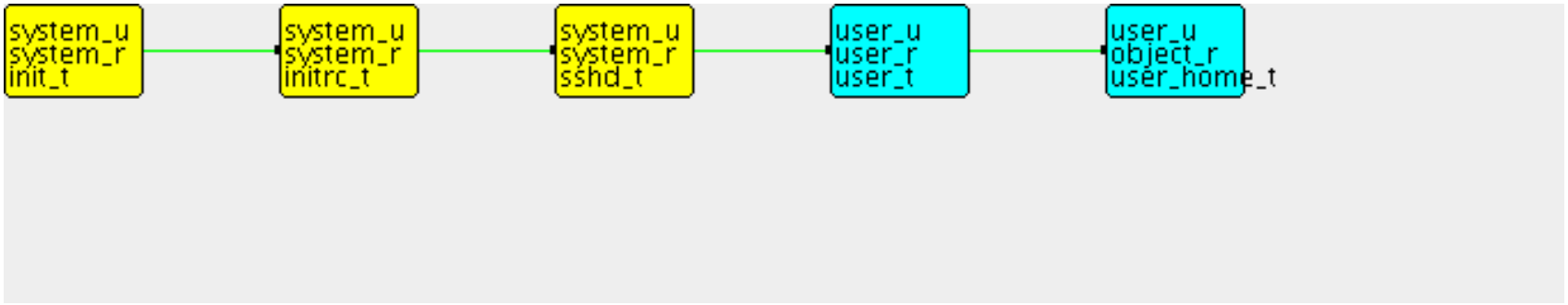
Exemple d'attaque



Opération de l'attaquant

17 :10 Téléchargement et installation d'un outil de scan
Violation de séparation de privilèges

Exemple d'attaque

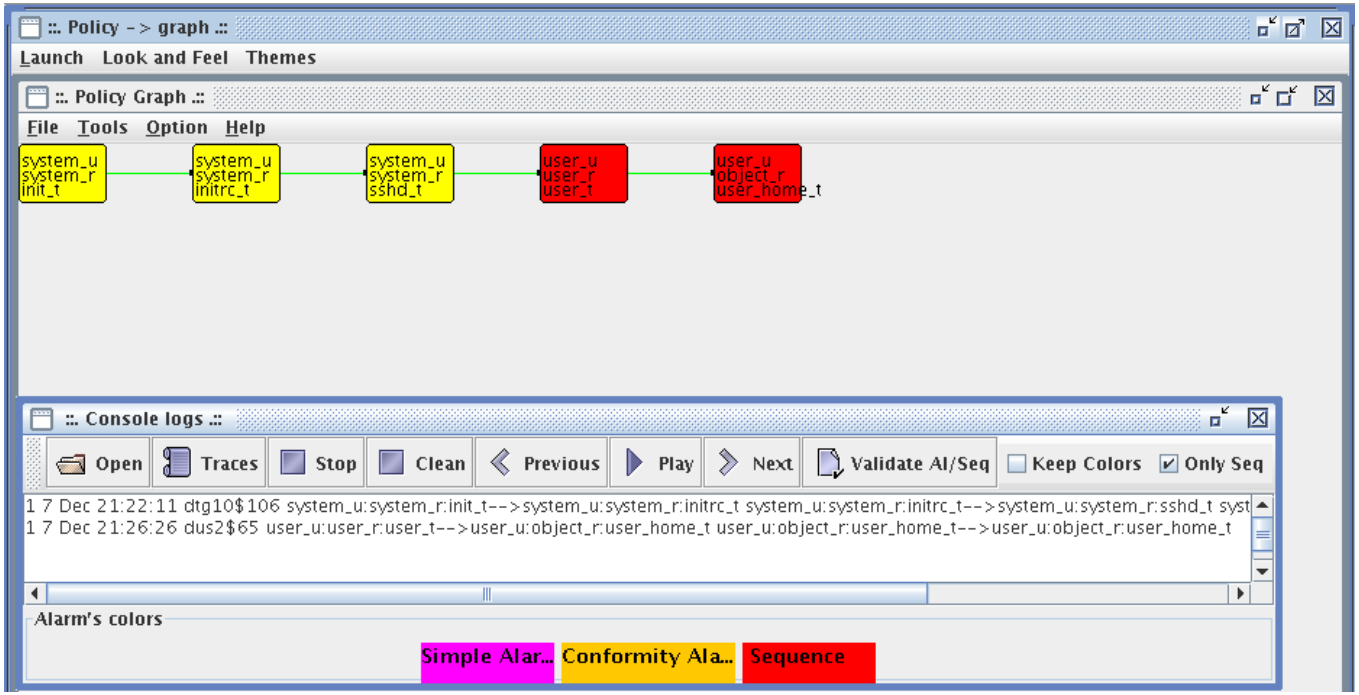


Opération de l'attaquant

22 :00 Récupération des résultats du scan

Violation de confidentialité

Exemple d'attaque



Opération de l'attaquant

22 :00 Récupération des résultats du scan

Violation de confidentialité

4 étapes de l'attaque

1. Connexion ssh *(changement de contexte interdit)*

$$init\_t \xrightarrow{trans} initrc\_t \xrightarrow{trans} sshd\_t \xrightarrow{trans} user\_t$$

2. Analyse de la machine *(changement de contexte interdit)*

$$init\_t \xrightarrow{trans} initrc\_t \xrightarrow{trans} sshd\_t \xrightarrow{trans} user\_t \xrightarrow{trans} ping\_t$$

3. Installation d'un outil de scan *(séparation de privilèges)*

$$(user\_t \xrightarrow{execute} user\_tmp\_t) \circ (user\_t \xrightarrow{write} user\_tmp\_t)$$

4. Récupération du résultat du scan *(confidentialité)*

$$(init\_t \xrightarrow{trans} initrc\_t \xrightarrow{trans} sshd\_t \xrightarrow{trans} user\_t) \wedge (user\_tmp\_t > user\_t)$$

Session d'attaque

- Signature correspondant à l'installation d'outil de scan

```
installation_scanner_ssh =
(recuperation_des_resultats   ○  installation_d_un_outil )
                              ○
(analyse_du_systeme           ○   connexion)
```

- Propriété de sécurité complexe

```
empecher_installation_malware =
(confidentiality      ○ duties_sep )
                      ○
(bad_transition       ○ bad_transition)
```

Résultats

- 45 590 connections en 1 an
  - 92% violations d'intégrité, 6% de confidentialité
  - 2219 installations d'outil
    - Robot IRC, outil de scan
- Mise en évidence
  - Des sessions d'attaques
  - De propriétés complexes

Conclusion

- Nouveau langage de description des activités
- Analyse de propriétés de sécurité
- Approche de détection d'intrusions

Conclusion

- Nouveau langage de description des activités
- Analyse de propriétés de sécurité
- Approche de détection d'intrusions

Conclusion

- Nouveau langage de description des activités
- Analyse de propriétés de sécurité
- Approche de détection d'intrusions

Travaux complémentaires

- Plateforme Multi-Agent pour la détection d'intrusions
  - Extension de la Méta-Politique
    - Prise en compte des propriétés de sécurité
    - Extension du langage pour les IDS
  - Implantation de l'architecture Multi-Niveaux
    - 4 niveaux pour répartir et garantir une Méta-Politique
    - Coopération entre IDS et contrôle d'accès
    - Distribution et répartition des règles entre les niveaux

$\longrightarrow$ Déploiement et répartition d'une Méta-Politique

Travaux complémentaires

- Plateforme Multi-Agent pour la détection d'intrusions
  - Extension de la Méta-Politique
    - Prise en compte des propriétés de sécurité
    - Extension du langage pour les IDS
  - Implantation de l'architecture Multi-Niveaux
    - 4 niveaux pour répartir et garantir une Méta-Politique
    - Coopération entre IDS et contrôle d'accès
    - Distribution et répartition des règles entre les niveaux

$\longrightarrow$ Déploiement et répartition d'une Méta-Politique

Perspectives

- Vers une garantie des propriétés
  - Application à la protection système
  - Contrôler/Bloquer les activités illicites
  - Performances ?
- Corrélation et Apprentissage des propriétés
  - Caractériser automatiquement les sessions d'attaques
  - Caractériser automatiquement les propriétés de sécurité

Perspectives

- Application aux politiques dynamiques
- Compilateur du langage de description d'activités
- Extension du langage de description d'activités

Publications

- Détection orientée propriétés de sécurité
  - Conférences internationales
    - CTS 2005, POLICY 2006, ColSec 2006
  - Conférences nationales
    - CRiSIS 2005
- Architecture de deploiement de Méta-Politique de détection
  - Conférences internationales
    - CTS 2005, SAR 2005, PSACE 2006, MASC 2006

Questions

- Questions ?