IRISA

LSR
IMAG

Modélisation et raffinement d'une politique de contrôle d'accès en B

Nicolas Stouls
Vianney Darmaillacq

POTESTAT

22/03/05

Motivations

IRISA

LSR
IMAG

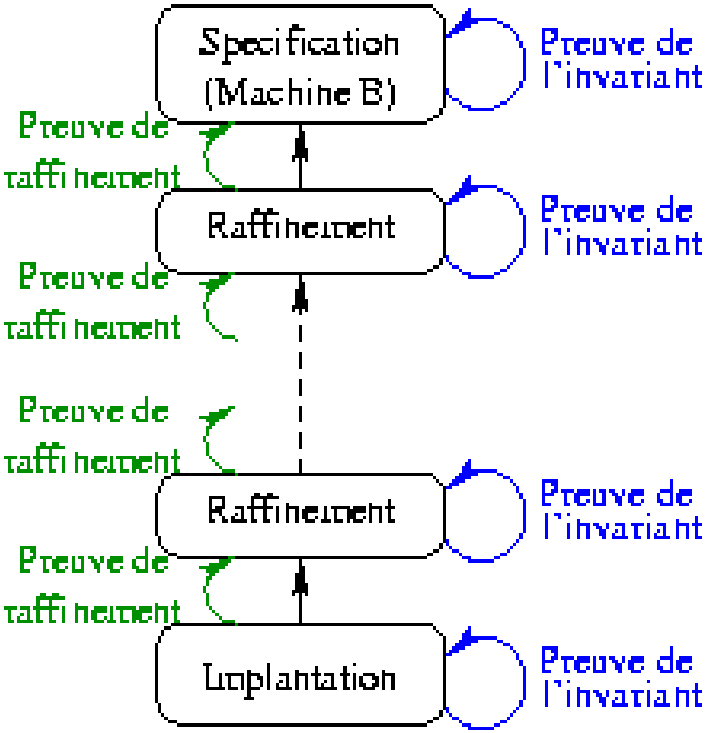
- Comment vérifier la conformité de l'implémentation d'une politique de sécurité de haut niveau sur les méchanismes matériels et logiciels ?
- $\Rightarrow$ Raffinement de la politique de sécurité de haut niveau en une politique de sécurité de niveau matériel (RAFFINEMENT)
- $\Rightarrow$ Utilisation de B pour modéliser une politique de contrôle d'accès (Cas d'étude en B)

Méthode B

LSR
IMAG

IRISA

Présentation

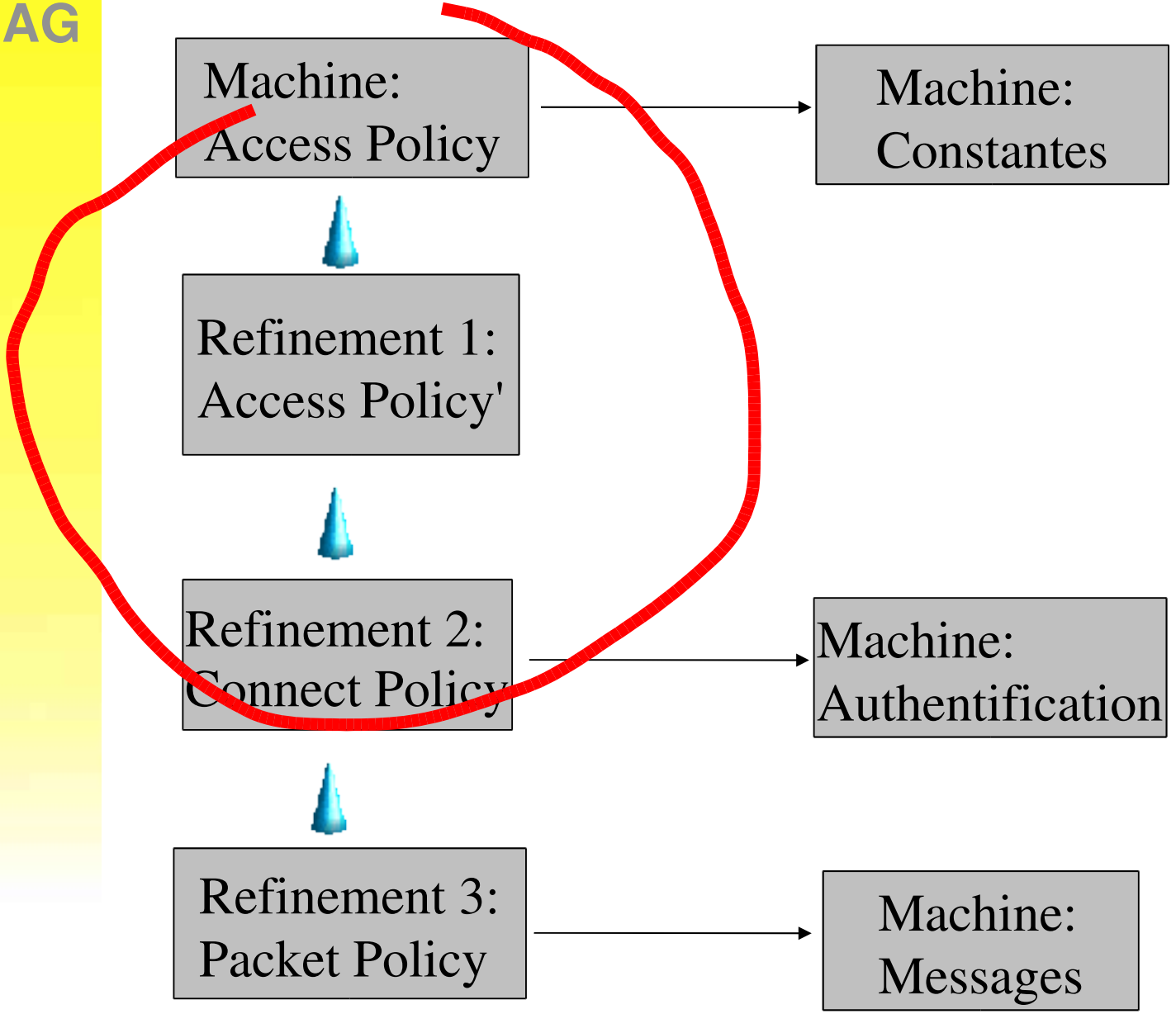


- Invariant
- Modularité
- Langage basé sur l'affectation
- Développement par raffinement
- Génération de code (C/C++/ADA)

Modèle Global

LSR
IMAG

IRISA



Access Control Policy (1)

IRISA

LSR
IMAG

Typage

- Ensembles utilisés :
  - subject, action, object, key
    - simplification : card (action) = 1
- Modélisation de la dynamique du système :
  - access $\subseteq$ (subject x object)
    - accès courants
  - auth-subject $\subseteq$ subject
    auth-object $\subseteq$ object
    - authentification
- Politique de contrôle d'accès :
  - access-right $\subseteq$ (subject x object)
    - accès autorisés

Access Control Policy (2)

IRISA

LSR
IMAG

Invariant et signature des opérations

- Invariant :
  - access $\subseteq$ access-right
    - $\neg$ (access $\subseteq$ access-right) $\Rightarrow$ violation de la politique de contrôle d'accès
  - dom (access) $\subseteq$ auth-subject
  - ran (access) $\subseteq$ auth-object
- Operations :
  - bool$\leftarrow$request-access (s, o)
  - bool$\leftarrow$authentify-subject (s, k)
  - bool$\leftarrow$authentify-object (o, k)
  - + destructeurs, observateurs

Access Control Policy (3)

LSR
IMAG

Exemple d'opération

Res←request-access (s, o) =
  PRE s ∈ subject ∧ o ∈ object THEN
    IF (s, o) ∈ access-right ∧ s ∈ auth-subject ∧ o ∈ auth-object
    THEN
      CHOICE
        access:=access ∪ {(s, o)} || Res := Ok
      OR
        Res := Ko
      END
    ELSE
      Res := Ko
    END
  END

Access Control Policy (3)

LSR
IMAG

IRISA

Exemple d'opération

Res ← authentifyUser(s, k) =
  PRE s ∈ subject ∧ k ∈ key THEN
    CHOICE
      auth-subject := auth-subject ∪ {s} ||
      Res := Ok
    OR
      Res := Ko
    END
  END

Connect Policy

LSR
IMAG

1er raffinement

- Changement de représentation de la structure des données
- Nouveaux concepts :
  - id-used $\subseteq$ id
  - connection-by $\in$ (id-used $\rightarrow$ subject)
  - connection-to $\in$ (id-used $\rightarrow$ object)
- Raffinement :
  - access = (connection-by^{-1}; connection-to)
    - la variable access disparait par raffinement

Connect Policy

LSR
IMAG

IRISA

2ème Raffinement

- Raffinement
  - subject = user
  - object = service
- Nouveaux ensembles :
  - machine, server, terminal, protocol
- Invariant :
  - used-by : terminal $\rightarrow$ user
  - connection-from $\in$ (id-used $\rightarrow$ terminal)
  - connection-proto $\in$ (id-used $\rightarrow$ protocol)
  - (connection-from^{-1}; connection-by) $\subseteq$ used-by
  - ...

Connect Policy

LSR IMAG

IRISA

Exemple d'opération

- Res←auth-subject(s, k) =
  PRE s ∈ subject ∧ k∈key THEN
  ANY term WHERE term ∈ Terminal THEN
  IF term ∉ dom(used-by) THEN
  LogOk ← TryAuth(uu,k) ;
  IF LogOk=TRUE THEN
  used-by:=used-by ∪ {(term, s)} || Res := Ok
  ELSE
  Res := Ko
  END
  ELSE
  Res := Ko
  END
  END
  END

Packet Policy

IRISA

LSR
IMAG

3ème raffinement

- Nouveaux concepts :
  - ip-address, port
- Raffinement :
  - ip-address = f (machine)
    - f : totale et bijective

Conclusion

LSR
IMAG

IRISA

Bilan

- Modèle prouvé à 100%
  - propriétés de contrôle d'accès garanties
    - raffinement des sujets et des objets
- Complexité du formalisme
  - preuve automatique difficile à obtenir
- Raffinement de la politique d'un haut niveau jusqu'à un niveau réseau
  - sujet : adresse ip
  - objet : (adresse ip, numéro de port)

Conclusion

LSR
IMAG

IRISA

Perspectives

- Ajouter un niveau de raffinement pour remplacer l'identificateur de connexion par un couple (terminal-port)
- Utiliser le dernier niveau pour dériver des configurations de méchanismes matériels ou logiciels
- Utiliser les 2 derniers niveaux comme spécification pour tester la conformité du système
  - générer des tests à partir des obligations de preuve