Security Research Group, Dept. of Info. & Comm.
Gwangju Institute of Science and Technology

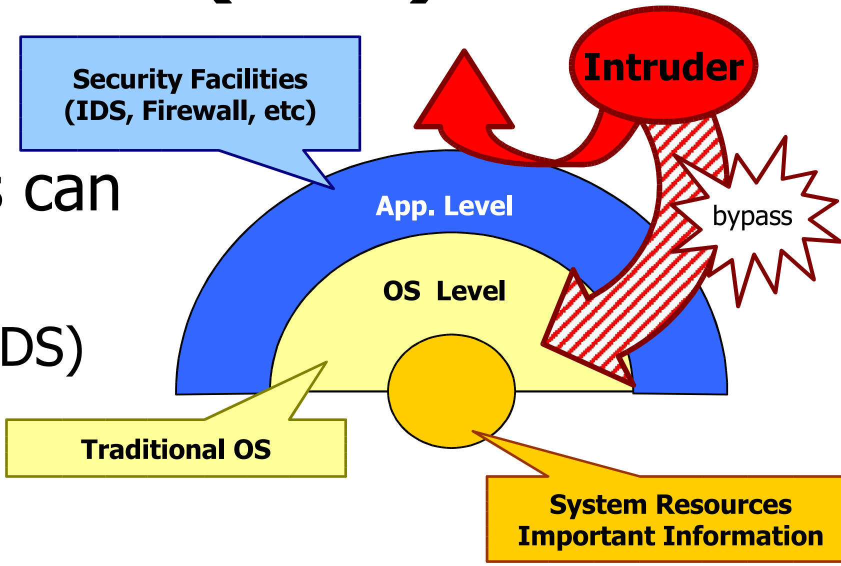
Extended Role Based Access Control for Trusted Operating Systems and its **Coloured Petri Net Model**

Gwangju Institute of Science and Technology

SHIN, Wook

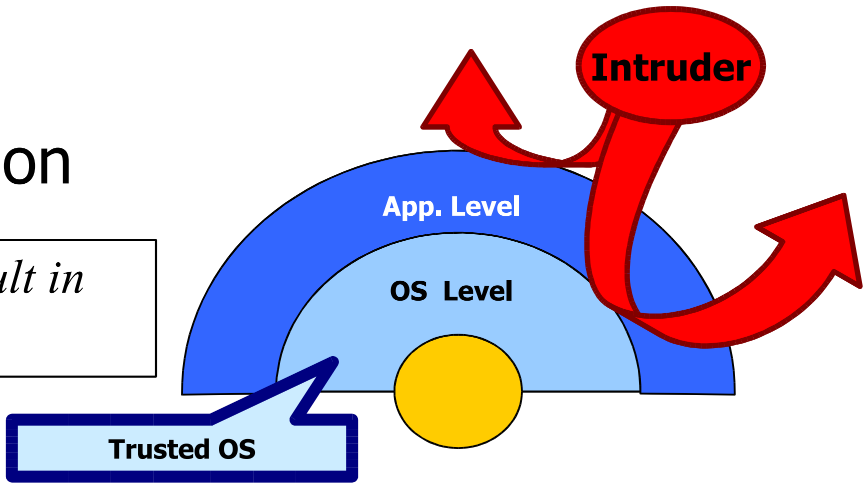
Trusted Operating System (TOS)

- App. level security solutions can be bypassed [1]
  - Intrusion Detection System (IDS) and Firewall are executed in application level

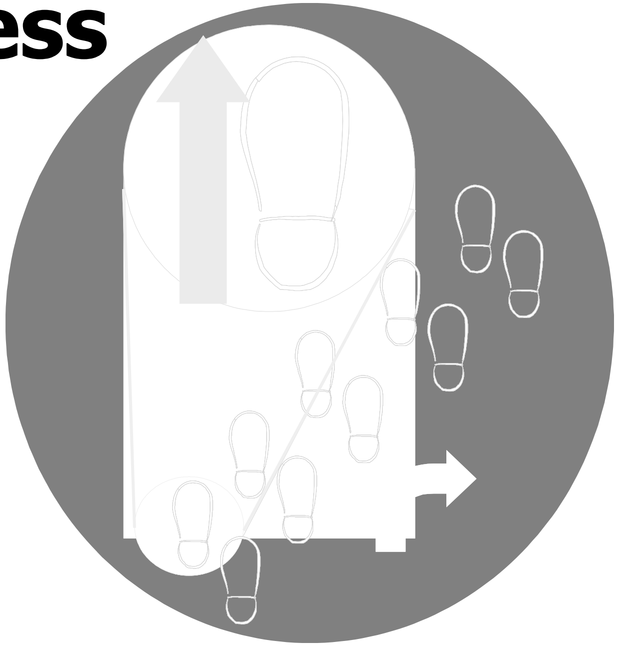


- TOS is an even more fundamental security solution

> *"Without TOS, all security efforts result in* ***Fortress built upon sand"***[2]

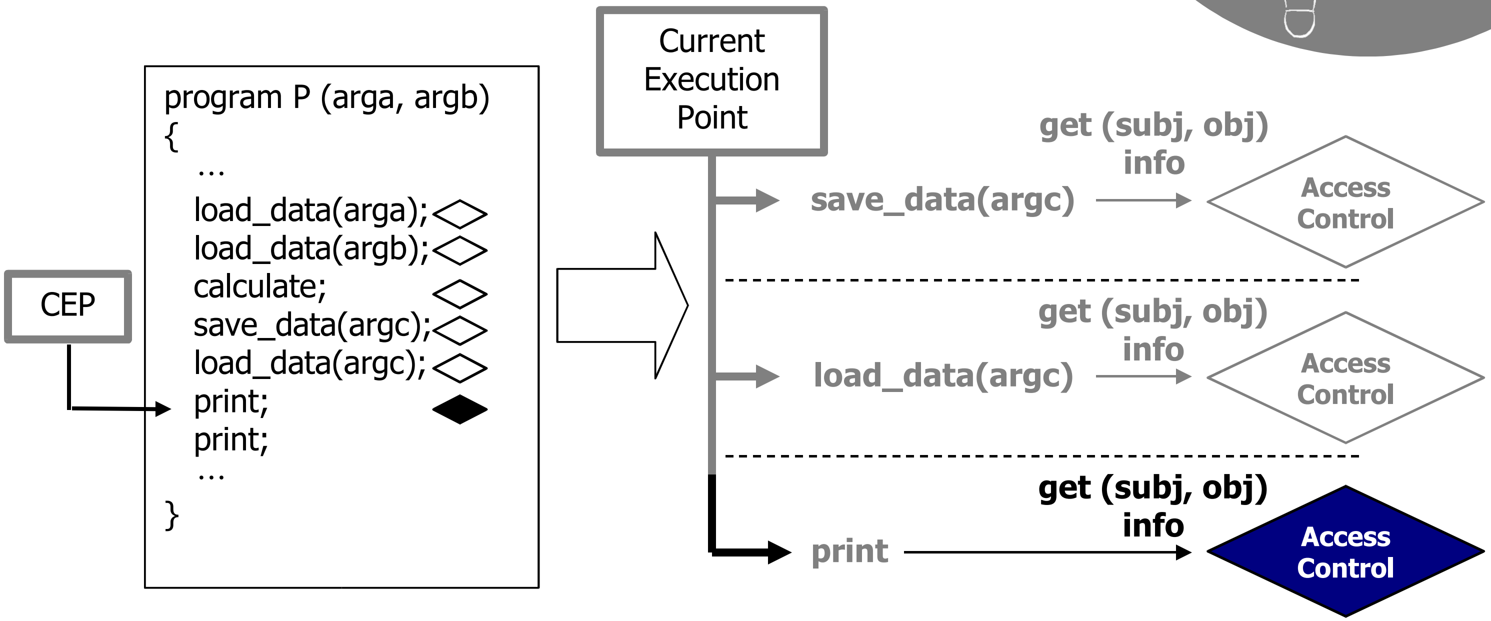


Insufficiency of Current Access Controls



- Current Access Controls
  - Control Accesses Based on Instant Access Information
  - They cannot block some kinds of attacks

    Ex) race condition attacks



Additional Constraints of E-RBAC

- We propose an extended access control
  - Extend the vision and the functionality of the concept of access control based on **the sequence of operations**

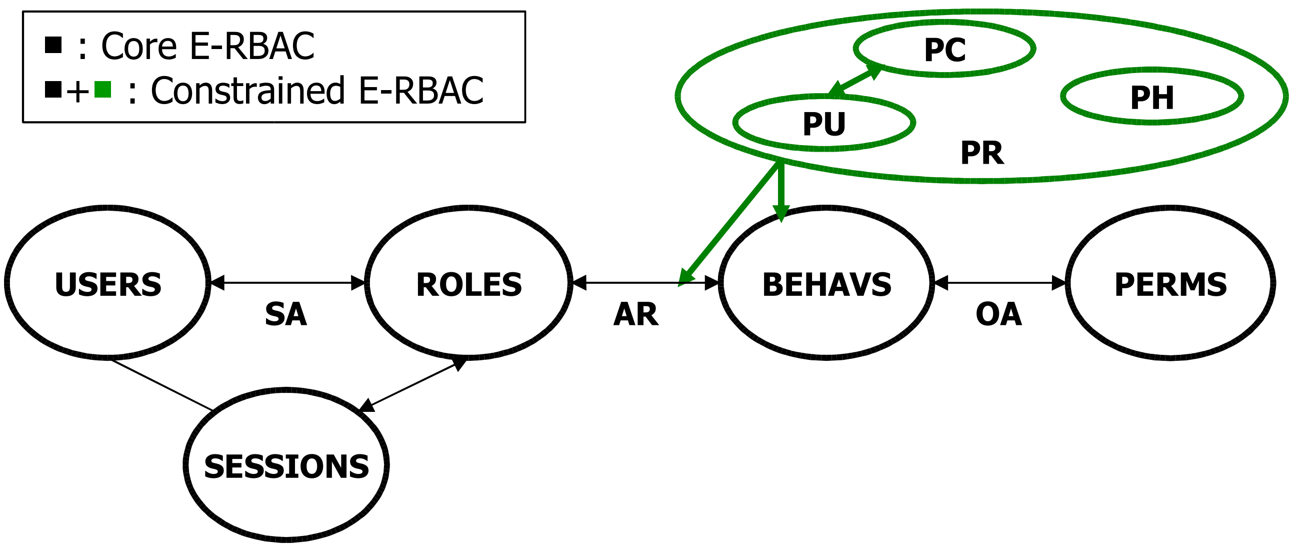
- Subject Abstraction and Object Abstraction
  - Roles: a set of users (subject-abstraction)
    Ex) Secretaries := {John, Michael, Tom}
  - Behaviors: a set of permissions (object-abstraction)
    Ex) FileOpSet := { f_open, f_close, f_read, f_write}
- Operations in E-RBAC
  - expressed in the Behavior layer
    - Permitted operations without procedural restrictions
    - Prohibited operations without procedural restrictions
    - Permitted execution sequences of operations (**Positive** procedural constraints, Positive PC)
    - Prohibited execution sequences of operations (**Negative** PC)

Newly Added Components to express "Execution Sequences"

Extended-Role Based Access Control

- Extended RBAC (E-RBAC)
  - Core E-RBAC
  - Constrained E-RBAC

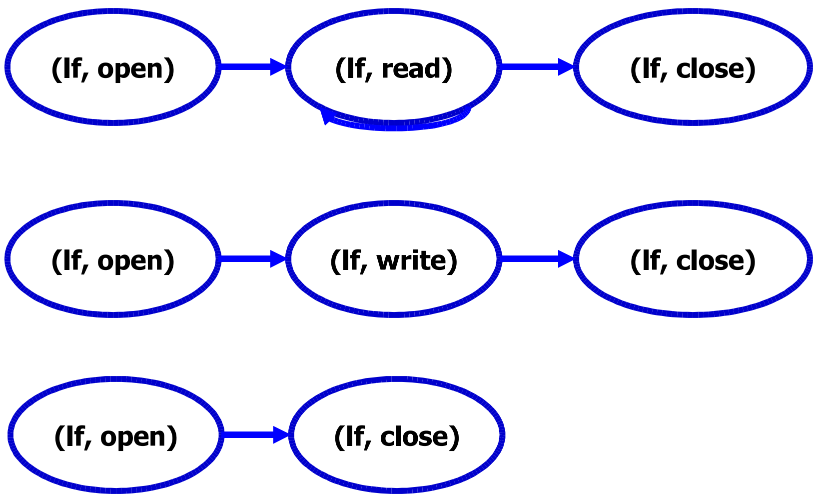
- The Conceptual Diagram



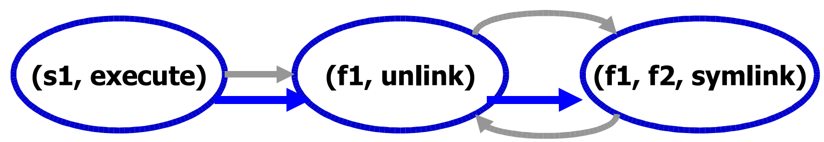
Modeling Behaviors

- Normal and Dangerous behaviors can be described

Positive	The permitted behavioral patterns of log file management



Negative	The prohibited behavioral pattern of the race condition attack



A Formal Model for E-RBAC

- We need a formal method to specify and verify security configuration of an E-RBAC system

- We define
  - a new formal model Constrained Coloured Petri Nets (CCPN)
    - based on Coloured Petri Nets (CPN) formalism
    - CCPN describes access matrix information and procedural information at the same time

Constrained CPN (CCPN)

- Main Components of Coloured Petri Net (CCPN)
  - Additional Component: Access Matrix
    - row: subjects
    - column: objects
    - entry: permissions
  - Interpretation: CPN Components are interpreted as AC entities
    - Tokens: Access Subjects
    - Places: Access Objects
    - Transitions: AEFs (Access Enforcement Function)
  - Modified Enable Condition

Testing a configuration with CPN

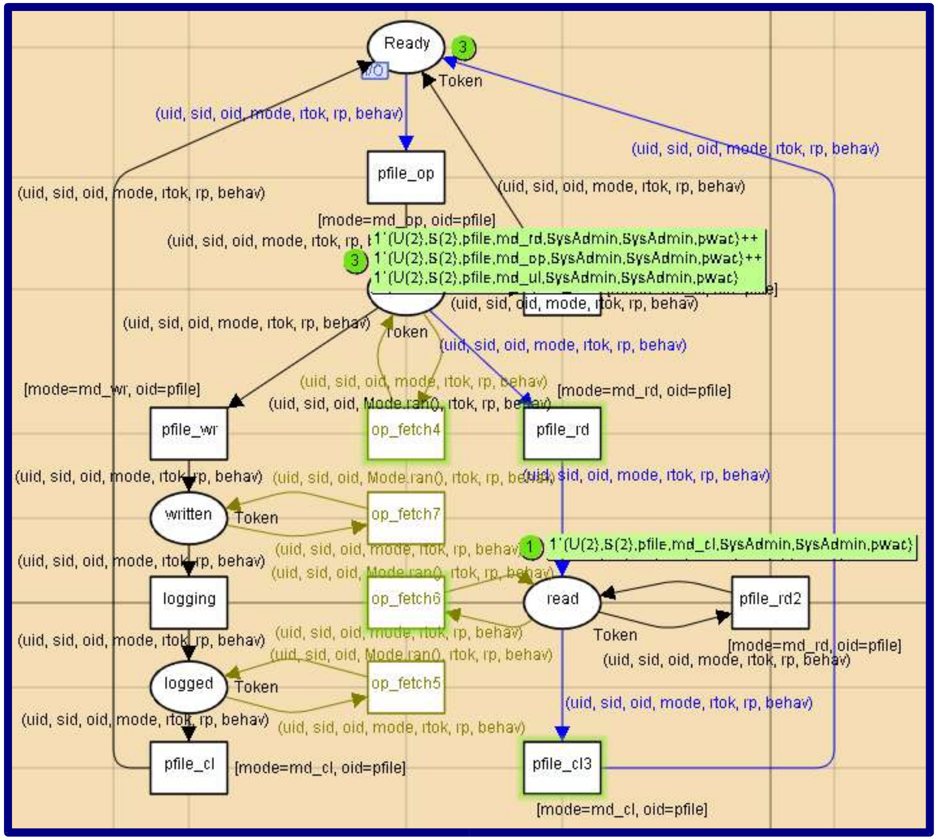
- We can test security related properties by
  - Simulation
  - Formal Analysis

- Example System Configuration
  - USERS = {u_1, …, u_i}
  - ROLES = {SysAdmin, User, r_1, …, r_j}
  - Objects = {logfile, mail_prg, $file_1$, …, $file_k$}
  - Modes = {read, write, open, close, execute, link, unlink}
  - Behaviors = {ExecuteMailProgram, AccessLogFiles, b_1, …, b_n}

- Using the formal method, we can correct security configuration errors

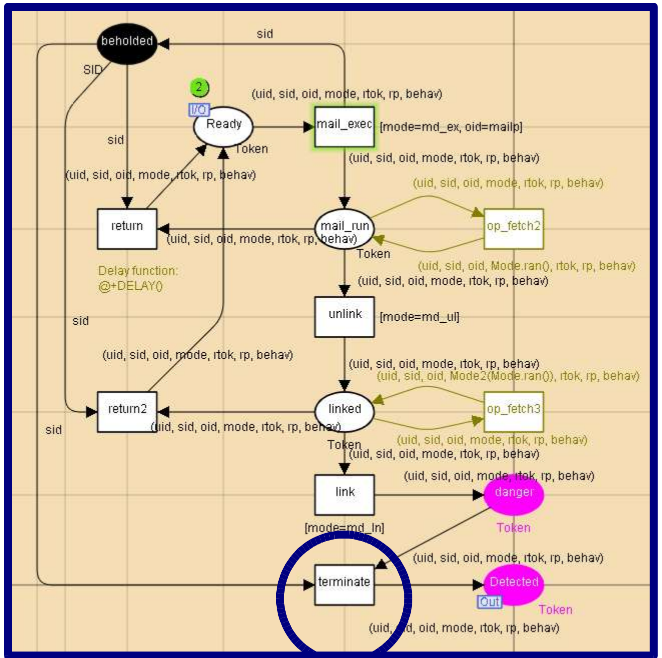
Simulation Example

- Analysis by Simulation: A Positive PC Example
  - The sets of execution sequences are performed well
    - {open-read*-close} or {open-write*-close}



Formal Analysis

- Analysis by Formalism
  - Liveness
    - Liveness check for the transition of attack detection
    - We can find and remove the possibility of being attacked



Liveness check of this transition

Formal analysis results

```
Liveness Properties
------------------------------------------------
Dead Markings:  None
Dead Transitions Instances:

NegativeRestriction1'link 1
NegativeRestriction1'mail_exec 1
NegativeRestriction1'op_fetch2 1
NegativeRestriction1'op_fetch3 1
NegativeRestriction1'return 1
NegativeRestriction1'return2 1
NegativeRestriction1'terminate 1
NegativeRestriction1'unlink 1
Overview'GenOPs 1
Overview'dummy 1
PositiveRestriction1'logging 1
PositiveRestriction1'op_fetch5 1
PositiveRestriction1'op_fetch6 1
PositiveRestriction1'op_fetch7 1
PositiveRestriction1'pfile_cl 1
PositiveRestriction1'pfile_cl2 1
PositiveRestriction1'pfile_cl3 1
PositiveRestriction1'pfile_rd 1
PositiveRestriction1'pfile_rd2 1
PositiveRestriction1'pfile_wr 1
Live Transitions Instances:

PositiveRestriction1'op_fetch4 1
```

Original configuration

Remove a dangerous operation

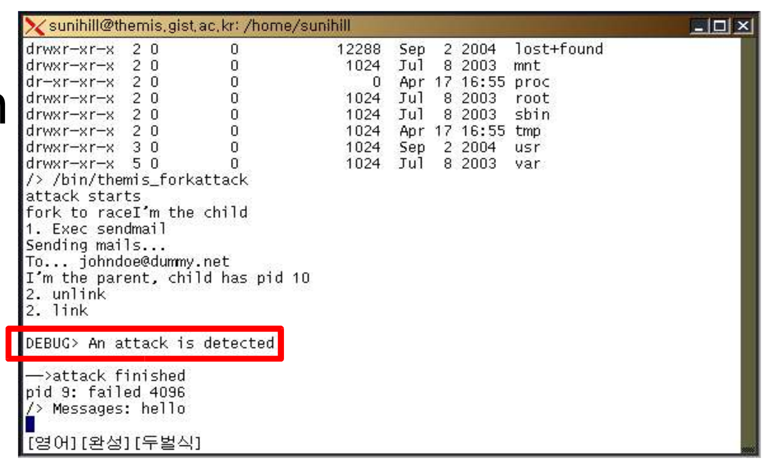
```
Liveness Properties
------------------------------------------------
Dead Markings:  6 [359,221,169,144,109,...]
Dead Transitions Instances:

NegativeRestriction1'link 1
NegativeRestriction1'terminate 1
Overview'GenOPs 1
Overview'dummy 1
Live Transitions Instances: None
```

Modified configuration
(Prohibit the unlink operation)

An Implementation

- The Implementation Environment
  - IFC-ETK100: An Embedded Board
    - CPU: SE3208(32 bit EISC Processor)
    - Memory:
      - 4M ROM, 4M Flash, 16M SDRAM
  - OS: uClinux-2.4.19

- The Implementation Result
  - Successfully detects race condition attacks



Performance Test

- Performance Measurement
  - Time costs of the execution of a simple program
  - Time costs of the execution of a file copy (512bytes)
  - Time costs of the execution of a simple program that have procedural constraints
- Results

Our system: 10 % overhead

Overhead of other systems
-A current TOS implementation (SELinux): 5%
-A current application level IDS solution (Snort): 10%

Conclusion

- The achievements
  - Extended RBAC Model
    - The vision and function of access control are extended
    - The attacks which consist of ordinary operations are denied
  - CPN Model for E-RBAC
    - Hybrid model for access control
    - Helpful for security administration
  - Trusted Embedded OS
    - E-RBAC can be implemented with reasonable overheads

Bibliography

[1] T. Ptacek and T. Newsham, "Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection", 1998.

[2] D. Baker, "Fortresses built upon sand", In Proceedings of the New Security Paradigms Workshop, 1996.

[3] D. Gollmann, "Computer Security", John Wiley & SONS, 1999.

[4] CPN Tools: http://wiki.daimi.au.dk/cpntools/

[5] M.Gasser, "Building a Secure Computer System", van Nostrand Reinhold, 1988.

[6] M. Bishop, “Computer Security: Art and Science”, Addison Wesley Professional, 2003

[7] K. Jansen, “Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use”, Vol. 1-2, Springer Verlag, 1992

[8] Department of Defense (U.S.), "Department of Defense Trusted Computer System Evaluation Criteria", Department of Defense Standard(Dod 5200.28-STD), Library Number S225, 711, December 1985.

[9] D. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn, and R. Chandramouli, "Proposed standard for Role Based Access Control,“ ACM Transactions on Information and System Security, vol. 4, no. 3 (August, 2001) - draft of a consensus standard for RBAC.