The Confinement Problem

James Campbell

**Lampson, Butler. "A Note on the Confinement Problem"
Communications of the ACM. 10/73 p. 613-5**

Summary

- **Definition of the Problem**
- **Principles**
- **Alternative Conceptions of the Problem**
- **Solutions and Proofs**

Definition

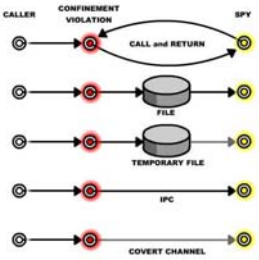
- The problem of restricting information granted to a process by its caller to that process

Principles

- Total isolation
  - Confined processes may not call any other processes
  - Impractical
- Transitivity
  - Confined processes may not call unconfined processes/facilities, unless trusted

Principles

- Channels
  - Vectors by which the Confinement property is violated
  - Types
    - Persistent memory
    - Permanent storage
    - Temporary storage
    - IPC
    - Other



Principles

- Channel Properties
  - Type
    - Storage
    - Legitimate
    - Covert

Principles

- Masking
  - Calling process may restrict inputs/outputs of the Confined process
- Enforcement
  - Confinement property enforced by the supervisor

Examples

- Persistent Memory
  - Caller information retained and relayed on subsequent call by collaborator example
  - Example solutions: prohibiting IPC, loading procedure into caller memory space
- Persistent storage
  - Caller information saved to disk and accessed later by collaborator
  - Example solution: object access/ownership controls

Examples

- Temporary storage
  - Caller information relayed to collaborator during execution through temporary files
  - Example solutions: same
- IPC
  - Caller information relayed directly
  - Example solutions: masking, auditing

Examples

- Covert channels
  - Timing
    - Execution time, delays, etc. used as vector [give timing example]
    - Example solutions: eliminating exact timing (impractical), auditing
  - Resource availability
    - CPU, storage, etc. denial used as vector
    - Example solution: Virtualization
  - System features
    - Paging, file presence/access tests, etc.
    - Example solutions: separate memory, minimize bandwidth

Other approaches

- Principles
  - * property
    - Confined processes cannot write to a less restricted channel
  - High-water mark
    - All writes are subject to the most restrictive security of any channel accessed
- Channels
  - Type
    - Static (semipermanent)
    - Time-decaying
  - Utility
    - Ease-of-use
    - Bandwidth

Solutions

- Lampson
  - Enumerate all security vulnerabilities
  - Close all vectors
- Loepere
  - Minimization

Bandwidth	Correction
100+ bps	Remove/Reduce
10-100 bps	Remove/Reduce/Audit
1-10 bps	Remove/Reduce/Audit/Document
<1 bps	Remove/Reduce/Audit/Document/Ignore

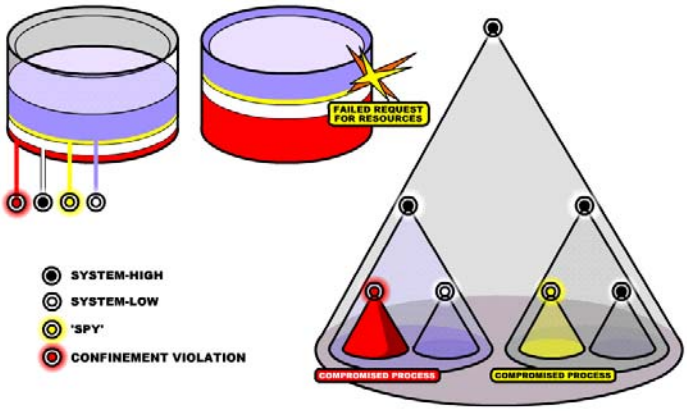
Remove	Close channel
Reduce	Reduce bandwidth
Audit	Audit object behavior for suspicious activity
Document	Document channel
Ignore	Ignore

Solutions

- Virtualization
  - Resources are abstracted
  - Eliminates the shared objects which form communication channels

Solutions

- Virtualization (example)



Solutions

- Formalizing information channels

à information channel
a source
b destination
← path

$|\triangleright^{à}$ conveyance over operations A

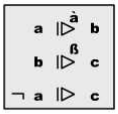
example:
à: if c then b ← a

If we want to prevent information leakage from a to b

$\neg a \; |\triangleright^{à} \; b$

apply contraint

$\neg c$



Problem

$a \; |\triangleright^{à} \; b \;\wedge\; b \; |\triangleright^{ß} \; c \Rightarrow a \; |\triangleright^{àß} \; c$

Solution

$a \; |\triangleright^{à} \; b \;\wedge\; b \; |\triangleright^{ß} \; c \not\Rightarrow a \; |\triangleright^{àß} \; c$

$\neg a \; |\triangleright \; c \rightarrow \neg b \; |\triangleright \; c$

(Flow Property)

Solutions

- Limiting the size of the security kernel
  - Robinson, et al assert that e.g. MULTICS, HYDRA are too large to ‘prove’ secure
  - PDP-11 'proof' of security kernel cites confinement principles above as essential to the security objective.
    - e.g. reference monitor or security kernel, tamperproof property, interpose itself on all object accesses

Proving the effectiveness of the system

- Establish Confinement Property for least restricted level & access controls
- It follows that less privileged processes are also Confined

Relationships with other types of security issues

- Access to files/resources
- Masking
- Exploitability of security mechanisms

Conclusions

- Enforcement Principle implies the necessity of a dedicated security kernel that controls object access
- Confinement solutions have strong parallels with those of other security issues

References

- Lampson, Butler. "A Note on the Confinement Problem" Communications of the ACM. 10/73 p. 613-5
- Lipner, Steven. "A Comment on the Confinement Problem" Proceedings of the fifth ACM symposium on Operating systems principles. 11/75 p. 192-6
- Loepere, Kieth. "Resolving Covert Channels Within a B2 Class System" ACM SIGOPS Operating Systems Review . 7/85 p. 9-28
- Rushby, J.M. "Design and verification of secure systems" Proceedings of the eighth ACM symposium on Operating systems principles. 12/81 p. 12-21
- Cohen, E. "Information Transmission in Computational Systems" Proceedings of Sixth ACM Symposium on Operating Systems Principles. 11/97 p. 133-139
- Cohen, E. and Jefferson, D. “Protection in The Hydra Operating System” Proceedings of the Fifth ACM Symposium on Operating Systems Principles, 11/75 p. 141-160.