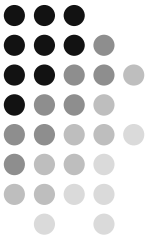
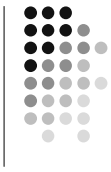
Access Control

Current Approaches and New Challenges

Carlisle Adams
School of Information Technology and Engineering
University of Ottawa

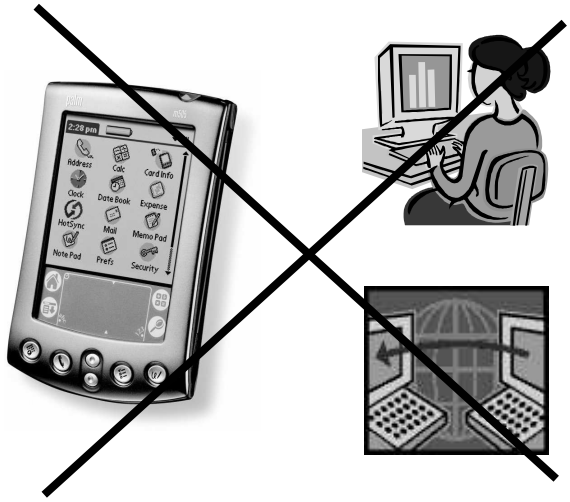


Roadmap



- Past
  - The birth of a problem and early solutions
- Present
  - The birth of connectivity and current approaches
- Future
  - The birth of ubiquity and next steps
- Conclusions

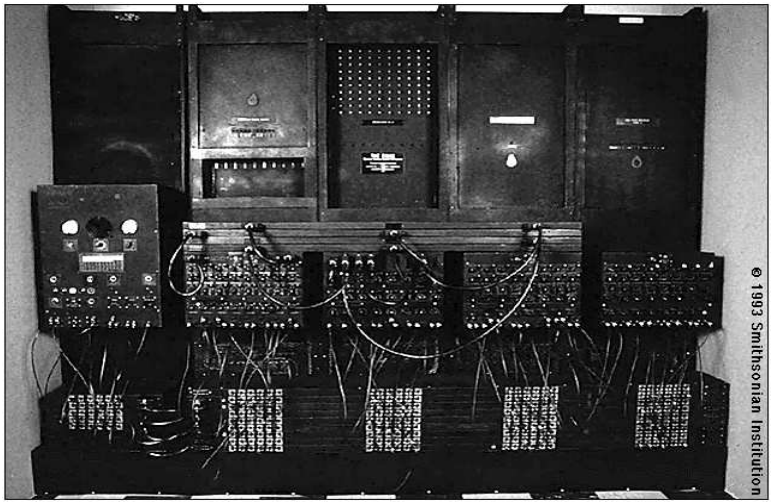
Early Days of Computing



Early Days of Computing (cont'd)

- At the beginning of the computer era
  - No operating systems: humans manipulated mechanical switches and plugged in patch cords to input programs / data (in binary)

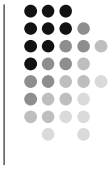
Early Days of Computing (cont'd)



Early Days of Computing (cont'd)

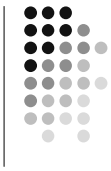
- Then, "executives" invented to assist user
  - Early operating systems
  - Helped with tasks of linking, loading, access to compiler, and so on
  - Sat in background waiting to be called by user

The Birth of a Problem

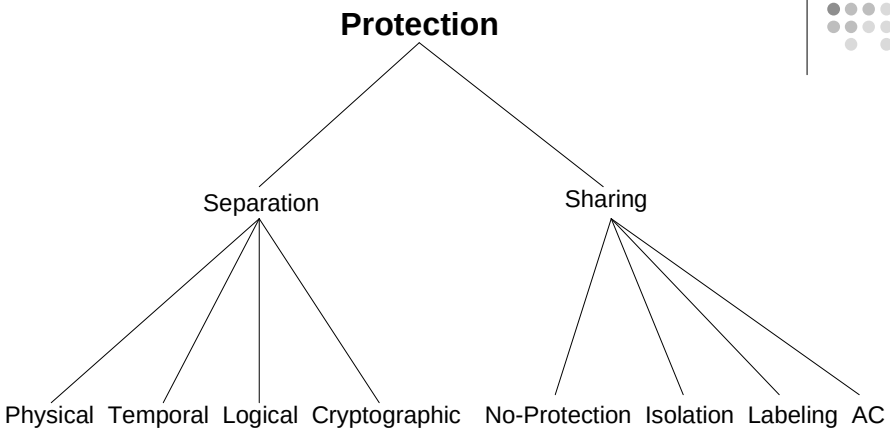


- Multiprogramming
  - Two or more users simultaneously on a machine (generally: multi-user, single-machine env.)
  - Need to interleave access to machine resources (scheduling, sharing, parallel use)
- "Executive" → "Monitor"
  - More active, authoritarian role
  - Software in control of machine (not user!)
    - Needed for fairness in scheduling and sharing
    - Needed for protection

Protected Objects



- "Monitor" needs to protect
  - Memory
  - Sharable I/O devices (e.g., disks)
  - Serially reusable I/O devices (e.g., printers, tape drives)
  - Sharable programs and sub-procedures
  - Sharable data (e.g., files)



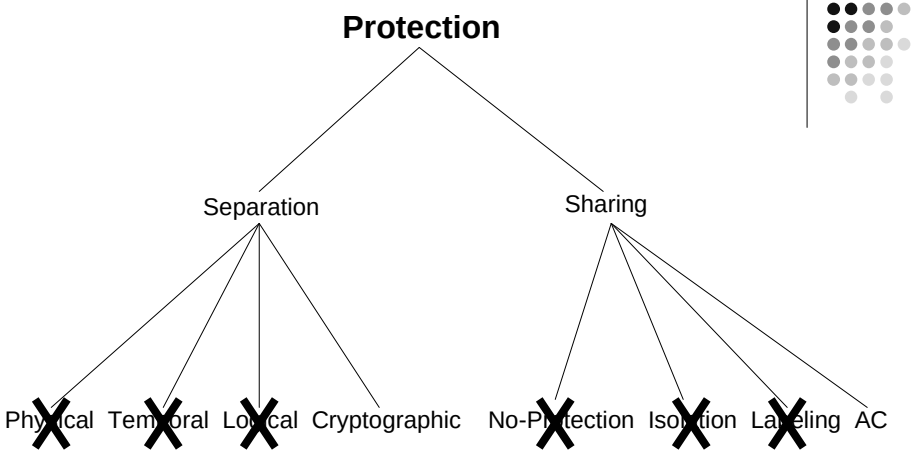
Early AC Work (Lampson, 1969)

- Formal notions of “subject”, “object”, access matrix

S_i \ O_j	General Ledger	Payroll	Accounts Receivable
Alice	R, W	-	R
Bob	-	R, W	R
Charles	R	-	R

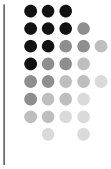
Present: The Birth of Connectivity

- Multiprogramming brought about the need for protection
- Networking has exacerbated this: it changed the picture in two major ways
  - Number of attackers (increased)
  - Number of protection mechanisms (decreased)



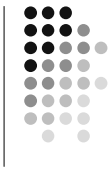
- Cryptography and Access Control remain
- Greater threat; fewer weapons
  - Increased importance of crypto and AC in networked envs.

When to use crypto and AC



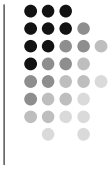
- AC mechanisms
  - Used in environments that can be trusted to run a program to check whether rules are being violated
- Separation mechanisms (crypto)
  - Used when the environment is not able to run a program to check rules (e.g., on a telephone wire), or is not trusted to enforce rules even if it can check them (e.g., PC running DOS)

Current Approaches: Crypto



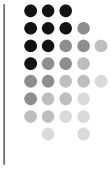
- Cryptographic mechanisms are reasonably available and well-understood
  - Symmetric, asymmetric, hash, signature, PRNG, …
- Not without problems, but generally pretty good
- Networked environments bring some difficulties (key management, in particular), but Kerberos, PKI, etc., offer some help

Current Approaches: AC

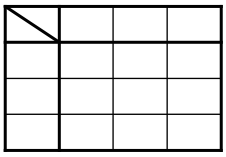


- Mechanisms are less well understood (or, at the very least, are less universally recognized and adopted)
  - Many, many (bewildering array of) tools and techniques in the literature and in the market
- However, a fairly comprehensive view has emerged over the past few years that has helped to put these tools/techniques into context and into perspective

AC in Networked Environments

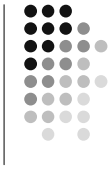


- Back to the access matrix:



- Networking potentially brings orders of magnitude increase in number of subjects, number of objects, and number of actions that can be performed (e.g., multinational corporation)
- Matrix is too large/sparse, too much of a bottleneck, and too limited in what it can express
  - Reduce size
  - Distribute data
  - Increase expressiveness

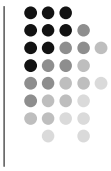
Size Reduction



- Role-Based Access Control (RBAC)
  - Ferraiolo, Kuhn (1992)
    - Lump subjects together in “bunches” (sets of entities with same job function) → fewer rows
- Generalized RBAC
  - Recognize different kinds of subject “bunches”
    - Role, Group, Clearance, etc.
  - Lump objects together in different “bunches”
    - Classification, Domain, etc.
  - Lump actions together in “bunches”
    - Level-of-Risk, etc.

Fewer rows, fewer columns, less data in each cell

Data Distribution



- Store pieces of the matrix in different places
  - Access Control List (ACL)
    - Single column, stored with object
  - Capability List
    - Single row, stored with subject

(Each has advantages & disadvantages.)

  - Attribute Certificate (AC)
    - Single cell, stored anywhere (object, subject, Attribute Authority, centralized database, distributed database)

(Advantages in flexibility and termination; disadvantages in performance.)

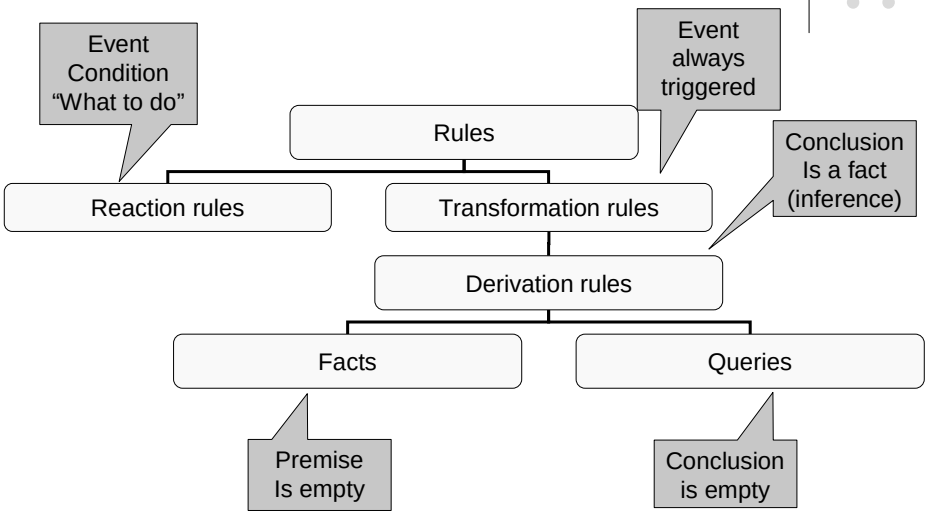
Expressiveness Improvement

- Access matrix works well for True/False decision on a small number of actions (R,W,X)
- When there is a larger number of actions, and decisions are based on sets of arbitrarily-complex conditions, need something more powerful
  - General language for expressing rules
    - E.g., RuleML

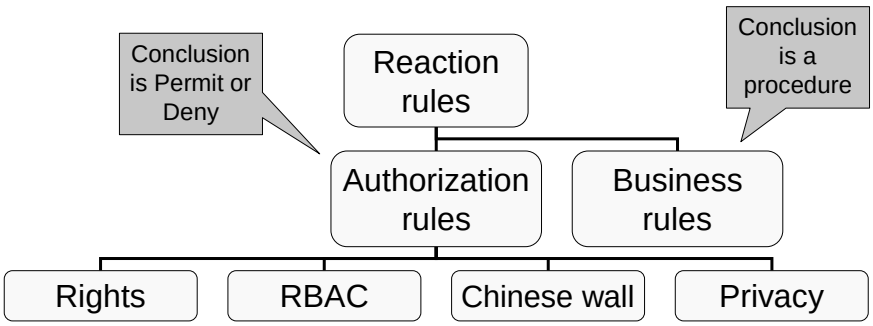
RuleML

- Rule = Premise + Conclusion
  = (Event + Condition) + Conclusion
- "Reaction Rule": Event not always triggered
- "Transformation Rule": Event always triggered
- "Authorization Rule": Conclusion is Permit / Deny
- "Business Rule": Conclusion is a procedure
- "Derivation Rule": Conclusion is a fact

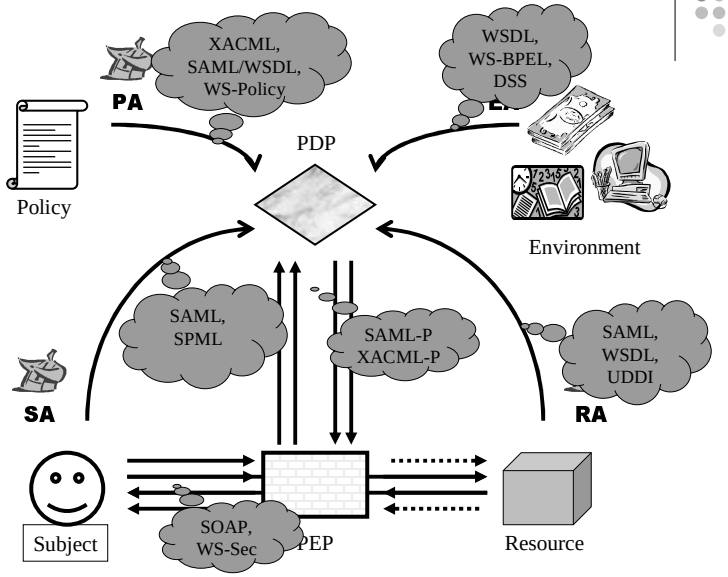
Rule Taxonomy



Rule Taxonomy (cont'd)



Overall Picture



Roadmap

- Past
  - The birth of a problem and early solutions
- Present
  - The birth of connectivity and current approaches
- Future
  - The birth of ubiquity and next steps
- Conclusions

Future: The Birth of Ubiquity

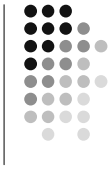
- Move from connectivity to <u>total connectivity</u>
  - Anytime, anywhere, with anyone
    - Mobile users, using mobile devices, to access mobile resources
    - Mobile data, going to unknown places for processing
    - Mobile agents, traveling to stationary data
    - The rise of ad hoc networks…

Is there room for cowards in our "brave new world"?

The Birth of Ubiquity (cont'd)

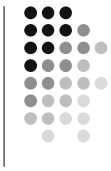
- What are the new challenges in a world of total connectivity?
  - Traditional W-5 of investigative inquiry:
    - Who?
    - What?
    - When?
    - Why?
    - Where?

Who?



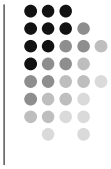
- User has multiple devices (desktop, laptop, phone, PDA, …) that s/he uses to connect to the network
  - Multiple identities; multiple personas
- *Access control across **identity***
  - Tying together various personas when desired in order to give seamless experience to user
  - Respecting privacy when necessary (AC while preserving anonymity / pseudonymity [Brands])

What?



- Protected resources such as XML documents are often hierarchically structured (each part has many sub-parts) and logical in nature (not actually stored in any single physical place)
- *Access control across **ephemeral objects***
  - Properly protecting all relevant pieces, wherever they (and their sub-pieces) may physically reside

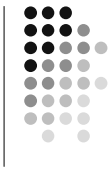
When?



- For audit purposes, as well as for various legal reasons, it may be necessary to prove that a previous PERMIT / DENY decision was valid (i.e., the correct decision to have made)

- *Access control across **time***
  - Creation, storage, and management of evidence with respect to access decisions over a period of months, years, or decades (even after all devices involved at time of access no longer exist)

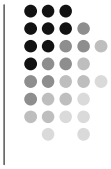
Why?



- The reason that an access is requested can be a relevant and important part of the query

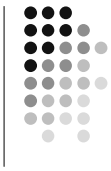
- *Access control across **intention***
  - Purpose for a request needs to be transmitted with the request and evaluated by PDP
  - system must ensure that eventual behaviour is consistent with stated purpose (policy enforcement, both at decision time and at time of use)

Where?



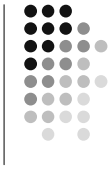
- Protected objects may be requested and retrieved from many places around the globe; such accesses will cross multiple boundaries
- *Access control across **domains***
  - Geographic / National
  - Corporate / Organizational
  - Jurisdictional / Legal
  - Technological / Medium

Roadmap



- Past
  - The birth of a problem and early solutions
- Present
  - The birth of connectivity and current approaches
- Future
  - The birth of ubiquity and next steps
- Conclusions

Conclusions



- Much progress has been made in access control since the early days of computing
- Connectivity brought challenges and required an evolution of existing mechanisms
- Ubiquity now brings even more challenges
  - Will **more evolution** be sufficient, or will we need **revolutionary change** (radical new approaches for access control) to deal with the difficulties raised by W-5 requirements?