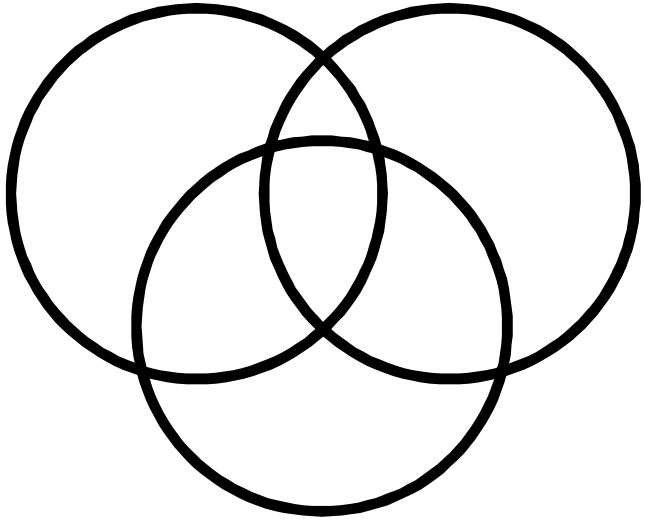
Application-Centric Security: How to Get There

Prof. Ravi Sandhu
Executive Director and Endowed Chair
Institute for Cyber Security (ICS)
University of Texas at San Antonio
September 2009

ravi.sandhu@utsa.edu
www.profsandhu.com

Security Objectives

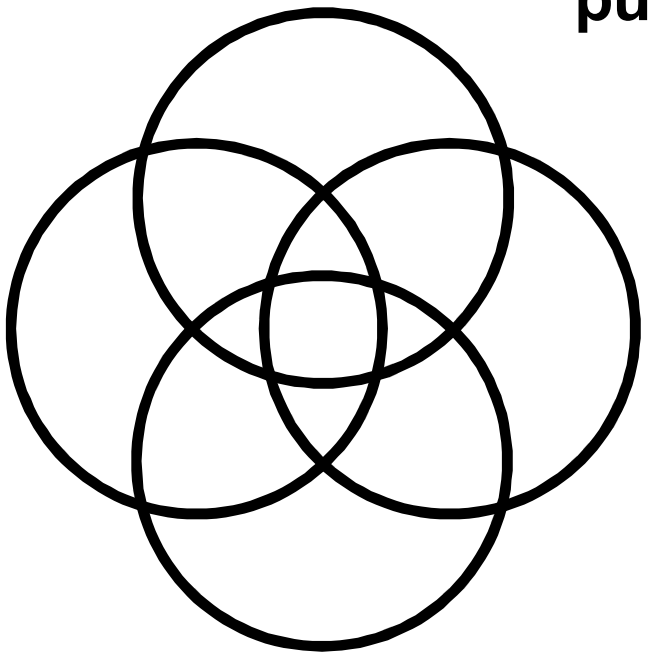


INTEGRITY
modification

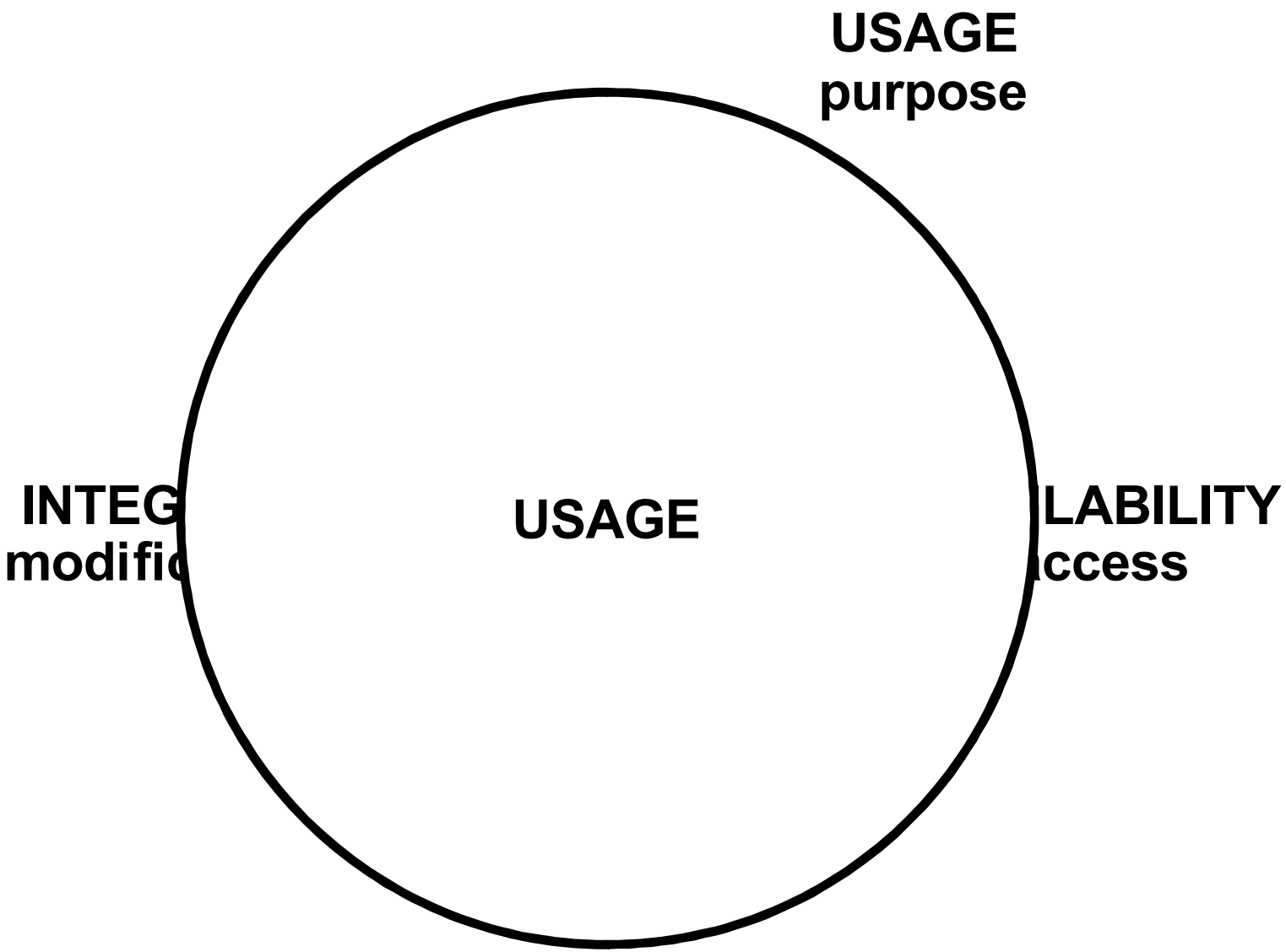
AVAILABILITY
access

CONFIDENTIALITY
disclosure

Security Objectives



Security Objectives



Security Trends and Change Drivers

Stand-alone computers	→	**Internet**
Vandals	→	**Criminals, Nation states, Terrorists**
Enterprise security	→	**Mutually suspicious yet mutually dependent security**
Few standard services	→	**Many and new innovative services**

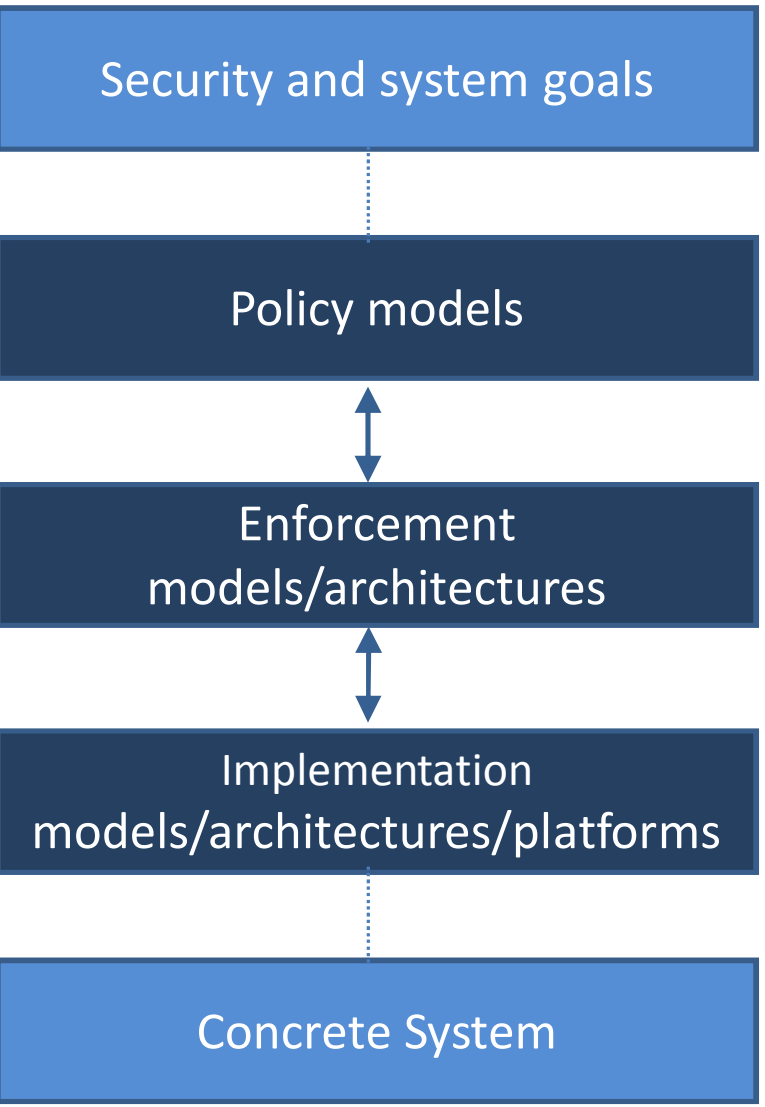
We are at an inflection point

Application-Centric Security Models

So how do we customize an application-centric security model?

- Meaningfully combine the essential insights of
  - Discretionary Access Control (DAC)
  - Mandatory Access Control (MAC)
    - Aka LBAC (Lattice-Based Access Control), BLP (Bell-LaPadula), MLS (Multi-level Security)
  - Role-Based Access Control (RBAC)
  - Attribute-Based Access Control (ABAC)
  - Usage Control (UCON)
  - Many others
- Directly address the application-specific trade-offs
  - Within the security objectives of confidentiality, integrity, availability and usage
  - Across security, performance, cost and usability objectives
- Divide and conquer by separating
  - Real-world concerns of practical distributed systems and ensuing staleness and approximations (enforcement layer)
  - Policy concerns in a idealized environment (policy layer)

PEI Layers World-View



- Necessarily informal

- Specified in terms of users, subjects, objects, administrators, labels, roles, groups, etc. in an idealized setting.
- Security analysis (e.g. security objectives, security properties, etc.)

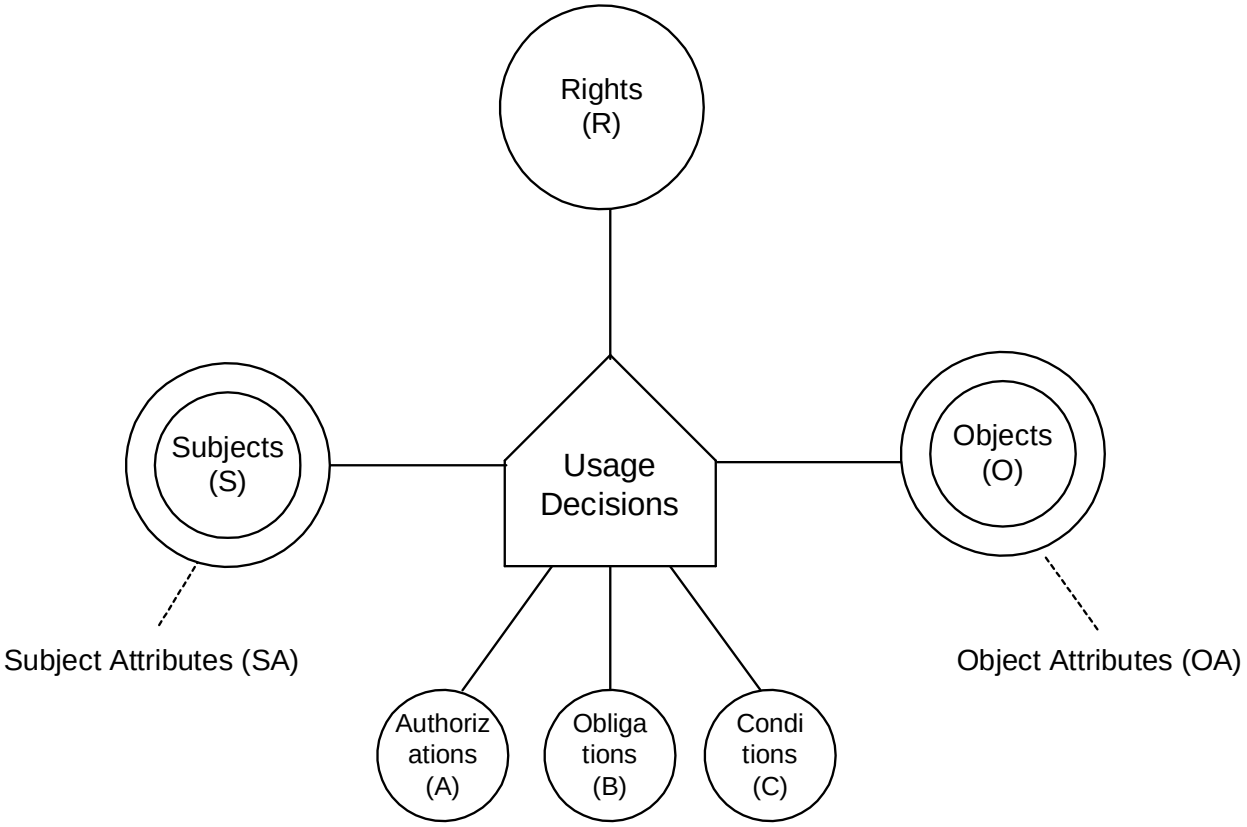
- Approximated policy realized using system architecture with trusted servers, secure protocols, etc. in a real-world setting
- Enforcement level security analysis (e.g. safe approximations with respect to network latency, protocol proofs, security properties, etc.)

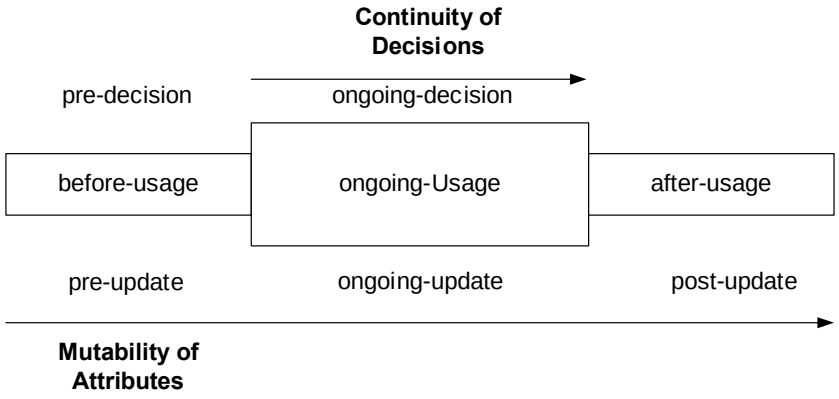
- Technologies and standards such as SOA, Cloud, SaaS, TCG/TPM, MILS, X.509, SAML, XACML, Oath, Oauth, etc.
- Implementation level security analysis (e.g. vulnerability analysis, penetration testing, protocol proofs, security properties, etc.)

- Layered software stacks executing on hardware

Usage Control Model (UCON)

- unified model integrating
  - authorization
  - obligation
  - conditions
- and incorporating
  - continuity of decisions
  - mutability of attributes





UCON is Attribute-Based Access Control on Steroids

Usage Control Model (UCON)

- Inspired by
  - DAC
  - LBAC
  - RBAC
  - ABAC
  - é and many, many others
- UCON
  - ABAC on steroids
  - Simple, familiar, usable and effective use cases demonstrate the need for UCON
    - Automatic Teller Machines
    - CAPTCHAs at Public web sites
    - End User Licencse Agreements
    - Terms of Usage for WiFi in Hotels, Airports
    - Rate limits on call center workers

Butler Lampson Paraphrased (I think)

- Computer scientists could never have designed the web because they would have tried to make it work.
  But the Web does ñwork.ò
  What does it mean for the Web to ñworkò?
- Security geeks could never have designed the ATM network because they would have tried to make it secure.
  But the ATM network is ñsecure.
  What does it mean for the ATM network to be ñsecureò?