QuanTM Architecture

Oleg Sokolsky
Adam Aviv, Andrew West, Jian Chang
Insup Lee, Jonathan Smith, Sampath Kannan,
Matt Blaze
University of Pennsylvania
Architecture overview

[Diagram of the QuanTM Architecture]
Trust manager
Trust manager

• Check credentials, compute compliance value
  – Qualitative value representing compliance with the policy
• Construct trust dependency graph (TDG)
• Current implementation based on KeyNote
  – Other access control logics can be used
Trust Dependency Graph (TDG)

• An encoding of a request and its dependencies:
  – Action requesters
  – Delegation chains
  – Credentials
• Edges represent trust dependencies
  – Place-holders for reputation (shown later)
• Can be easily derived from PTM (KeyNote) local policy and credentials
TDG example

- Reputations are assigned to TDG elements
- Reflect trust in principals, delegations, credentials
Reputation manager
Reputation manager

- Calculate trust value
  - Assign reputations to TDG edges using the reputation database
  - “push” reputation values up the graph
- Reputations is updated based on feedback
  - Outcomes from previous requests
- Current implementation based on TNA-SL
  - Trust network analysis – Subjective logic
  - Consensus and discount operators
Decision manager
Decision manager

- Uses an application-specific meta-policy
  - Context monitors
  - Cost-benefit analysis
  - Game-theoretic formalization
- Simple example: Threshold policy
  - If CV='maybe' and TV>0.5 -> Fulfill request
  - If CV='true' always fulfill request
  - In general, thresholds can be adaptive
Summary

• QuanTM architecture is a prototype implementation of QTM
• Outcomes of admitted requests produce measurable effect
  – Used as feedback to reputation database
• Design for modularity
  – Different PTM and RTM systems can be plugged in