Security

Prof. Steven A. Demurjian
Computer Science & Engineering Department
Director, CS&E Graduate Program
Co-Director, Research Informatics, Biomedical Informatics Division, CICATS (http://cicats.uconn.edu/cores/bmi.html)
Computer Science & Engineering Department
The University of Connecticut
371 Fairfield Road, Box U-2155
Storrs, CT 06269-2155

steve@engr.uconn.edu
http://www.engr.uconn.edu/~steve
(860) 486 - 3719
Our Five-Pronged Security Emphasis

- Collaborative Extension to NIST Role-Based Access Control: Applied to Medical Domain
- Secure Information Exchange via XML with MAC/RBAC
- Assurance, Consistency, Integrity, Access Control, RBAC, DAC, MAC
- Security Enforcement, Application and Software Security
- Secure Software Design To Design and Write Secure Software Programs
- Security for Services-Based (Prior) and Cloud (Future) Computing
NIST RBAC w/Collaboration & Workflow

Ph.D. Research Topic of S. Berhe

Issues:

- Collaboration Increasing in Health Care (PCMH)
- Security Models focus on Limiting Access
  - RBAC has Separation of Duty and Mutual Exclusion
  - Denies Sharing
- Extend NIST RBAC with
  - Obligated Collaboration Primitives and Constraints
  - Workflow Links Collaboration Steps for Enforcement

Papers

Security for XML Documents

- Can We Customize the Displayed XML Instance Based on Role?
- How Can we Incorporate RBAC, MAC, etc.?
- Set of XML Security Schemas for Roles, Users, and Constraints
  - Capture RBAC and MAC
  - Apply Security Schemas to XML Documents
    - An XML Document Appears Differently Based on Role, MAC, Time, Value
    - Security Schema Filters Document
- Different Schemas for Roles, Users, MAC, DAC
Attaining Security in XML

Given an XML Application of Schemas and Associated Instances, can we:

- Define Schemas/Instances for Clearances, Roles, Users, User-Role Authorizations, and Delegation
- Augment Application’s Schemas/Instances with MAC Security Classifications (if Needed)

Then, as XML Instances are Dynamically Identified to Suit a User’s Needs for an Application, can we:

- Retrieve and Filter those XML Instance(s) Based on User’s Role, MAC, and/or Delegation
- Deliver Filtered Instances(s) to User

Role of XACML? [http://www.oasis-open.org/committees/xacml/]

Work is Ongoing at this Point …
Security for Services-Based (Prior) and Cloud (Future) Computing

- Work of C. Phillips
- Premise: Artifacts - set of
  - DB, Legacy, COTS, GOTS, Each w/API
  - Middleware, Services, Web Services, etc.
- Premise: Users
  - New and Existing
  - Utilize Artifact APIs
- Can we Control User Access to APIs by …
  - Role (who)
  - Classification (MAC)
  - Time (when)
  - Data (what)
- Future: Extend to Cloud
Security Enforcement Framework Architecture

Both JINI and CORBA Implementations

- Database Client
- COTS Client
- Wrapped Resource for COTS Application
- Wrapped Resource for Database Application
- Wrapped Resource for Legacy Application
- General Resource
- Legacy Client
- Java Client
- Software Agent
- Lookup Service
- Unified Security Resource (USR)
  - Security Policy Services
  - Security Authorization Services
  - Security Registration Services
  - Security Analysis and Tracking (SAT)

Comcast-8
Secure Software Design: Extend UML

Extending UML for the Design and Definition of Security Requirements

Address Security in Use-Case Diagrams, Class Diagrams, Sequence Diagrams, etc.

Bi-Directional Translation – From Security Definitions (Extensions) to Underlying formal Functional Model

Iterate, Revise

Formal Security Policy Definition that Track Design State via a Functional Representation

Must Prove Generation Captures all Security Requirements

Many Alternative Security Model Solutions

Security Model Generation

RBAC99 GMU/NIST

RBAC/MAC

Oracle Security

Security Enforcement
Survey Management Example

Prototype Implemented in Borland Together Architect
Secure Software Design: New UML Diagrams

- Work of Jaime Pavlich-Mariscal
- To Transition from Design to Development …
  - Define new UML Diagrams Role-Slice, User, and Delegation Diagrams
  - Automatically Generate Enforcement Code
- Secure Subsystem: Classes on Which Privileges will be Defined

```
Public_Survey_Results
+Get_General_Statistics()
+Get_Questions()
```

```
Survey_List
+Add_Survey_Header()
+Survey_Title_Search()
+Update_Survey_List()
+Delete_Survey_Header()
```

```
Survey_Header
+Create_Survey_Header()
+Add_Question()
+Add_Special_Question()
```
Sample New Role-Slice Diagram

- A Role Slice is a Specialized Class Diagram that Represents Permissions for the RBAC Model

<<RoleSlice>>
Staff
{abstract}

+<<pos>> Survey_Header
+<<pos>> Add_Survey_Header()

Survey_Header

<<RoleSliceComposition>>

<<RoleSlice>>
Senior Staff

+<<pos>> Add_Survey_Header()

Survey_Header

+<<pos>> Create_Survey_Header()
+<<pos>> Add_Special_Question()

<<RoleSlice>>
Junior Staff

+<<neg>> Update_Survey_List()
Concluding Remarks

- Security is Part of an Overall Security Strategy
  - Definition of Security Requirements
  - Realization of Security at Application Level
  - Integration of Security from User to OS to DB
  - Rigorous Definition of Security Policy
  - Dynamic Nature of Security Privileges
  - Enforcement of Defined Privileges at Application and DB Levels

- Overall, Security in Today’s World Integral Part of Everyday Life - Some Key Concerns
  - Confidentiality of an Individual’s Data
  - Identity Theft
  - Protecting National Infrastructure
  - HIT and Health Information Exchange