INFS 767 Fall 2001

RBAC Architectures and Mechanisms

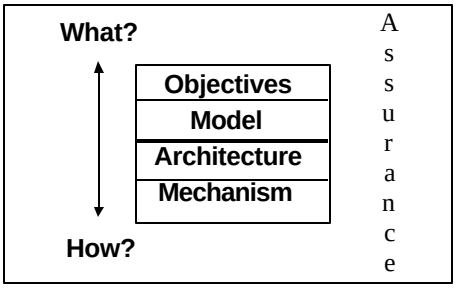
Prof. Ravi Sandhu

AUTHORIZATION, TRUST AND RISK

- **Information security is fundamentally about managing**
  - **authorization and**
  - **trust**

  so as to manage risk

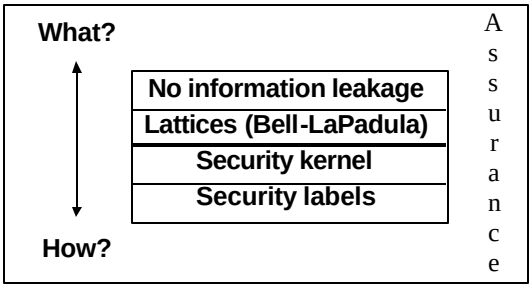
THE OM-AM WAY



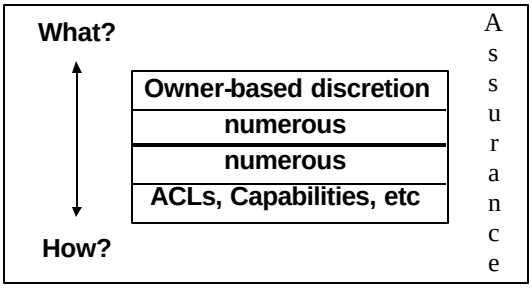
LAYERS AND LAYERS

- **Multics rings**
- **Layered abstractions**
- **Waterfall model**
- **Network protocol stacks**
- **Napolean layers**
- **RoFi layers**
- **OM-AM**
- **etcetera**

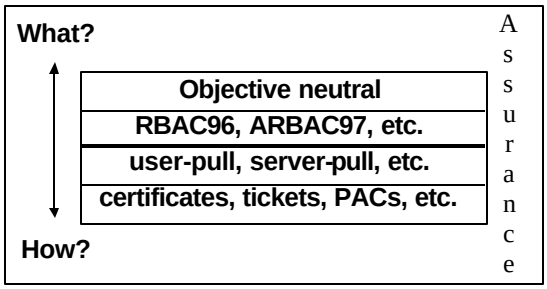
OM-AM AND MANDATORY ACCESS CONTROL (MAC)



OM-AM AND DISCRETIONARY ACCESS CONTROL (DAC)



OM-AM AND ROLE-BASED ACCESS CONTROL (RBAC)



DISTRIBUTED RBAC (DRBAC) CASE STUDY

- Approximately a dozen physical sites
- Approximately 2-3 simulation models/site
- Fewer than 100 roles structured in a very shallow hierarchy
  - A subset of roles is used in any single simulation model
- Fewer than 100 users
- A user uses only one role at a time
  - Convenient but not critical
- Moderate rate of change

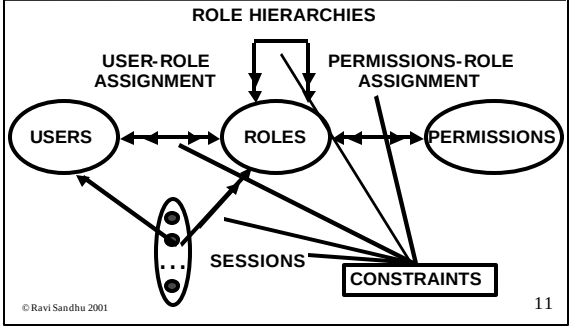
DISTRIBUTED RBAC (DRBAC) CASE STUDY

- Permission-role assignment
  - Locally determined at each simulation model
- User-role assignment
  - A user can be assigned to a role if and only if all simulation models using that role agree
  - A user is revoked from a role if and only if any simulation model using that role revokes the user

DISTRIBUTED RBAC (DRBAC) CASE STUDY

- Each simulation model has a security administrator role authorized to carry out these administrative tasks
- A simulation model can assign permissions to a role X at any time
  - even if X is previously unused in that simulation model
- Consequently any simulation model can revoke any user from any role!

RBAC3



MODEL CUSTOMIZATION

- Each session has a single role
- SM = {sm1, …, smk}, simulation models
- OP = {op1, …, opl}, operations
- P= SM X OP, permissions
- SMA = {sma1, …, smk}, administrative roles
- R Ç SMA = Æ
- Admin: SM « SMA

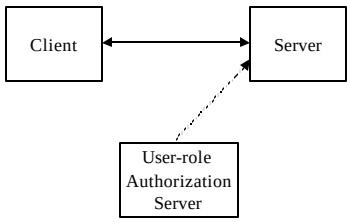
MODEL CUSTOMIZATION

- **Can formalize the administrative rules given earlier**
- **For each simulation model designate a unique user to be the chief security administrator who is authorized to assign and revoke users from the security administrator role for that model**

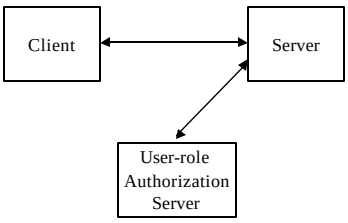
DRBAC ARCHITECTURES

- **Permission-role**
  - **Enforced locally at each simulation model**
- **Permission-role administration**
  - **Enforced locally at each simulation model**
  - **May need to communicate to other simulation models**
- **User-role**
  - **See following slides**
- **User-role administration**
  - **Centralized or decentralized**

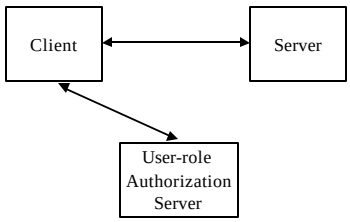
SERVER MIRROR



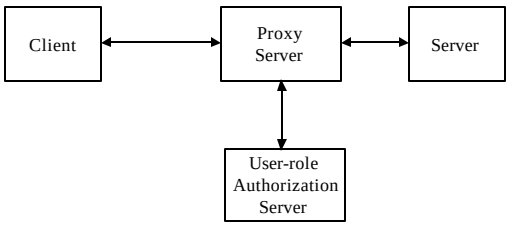
SERVER-PULL



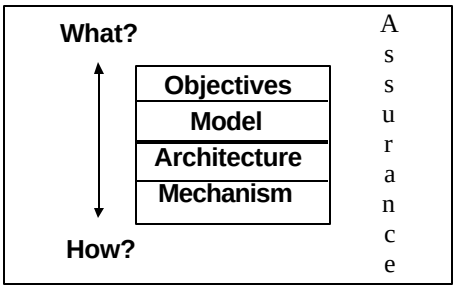
USER-PULL



PROXY-BASED



THE OM-AM WAY



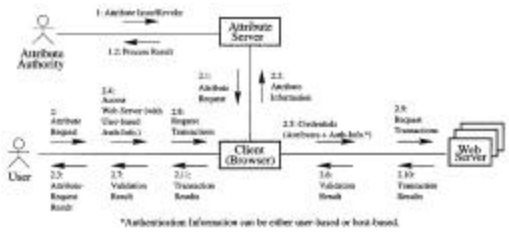
Secure Attribute Services on the Web

- **WWW (World Wide Web)**
  - **widely used for electronic commerce and business**
  - **supports synthesis of technologies**
  - **mostly, Web servers use identity-based access control**
    - **scalability problem**

Background

- **An attribute**
  - **a particular property of an entity**
    - **e.g., role, identity, SSN, clearance, etc.**
- **If attributes are provided securely,**
  - **Web servers can use those attributes**
    - **e.g., authentication, authorization, access control, electronic commerce, etc.**
- **A successful marriage of the Web and secure attribute services is required**

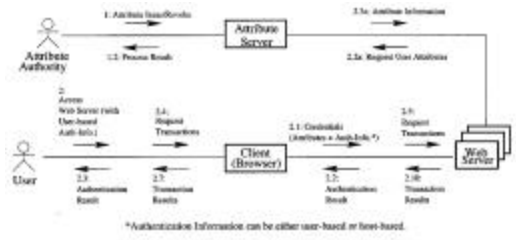
User-Pull Architecture



User-Pull Architecture

- **Each user**
  - **pulls appropriate attributes from the Attribute Server**
  - **presents attributes and authentication information to Web servers**
- **Each Web server**
  - **requires both identification and attributes from users**
- **High performance**
  - **No new connections for attributes**

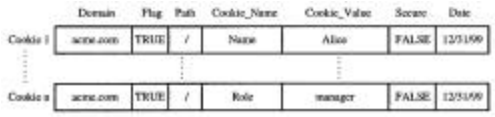
Server-Pull Architecture



Related Technologies

- **Cookies**
  - **in widespread current use for maintaining state of HTTP**
  - **becoming standard**
  - **not secure**
- **Public-Key Certificates (X.509)**
  - **support security on the Web based on PKI**
  - **standard**
  - **simply, bind users to keys**
  - **have the ability to be extended**

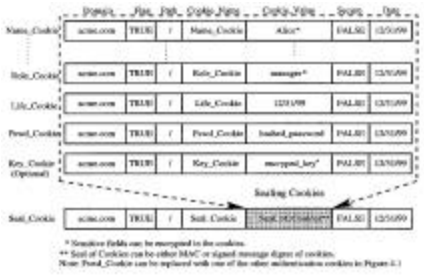
Cookies



Security Threats to Cookies

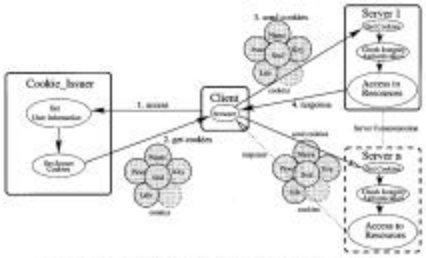
- **Cookies are not secure**
  - **No authentication**
  - **No integrity**
  - **No confidentiality**
- **can be easily attacked by**
  - **Network Security Threats**
  - **End-System Threats**
  - **Cookie Harvesting Threats**

Secure Cookies on the Web



A Set of Secure Cookies

How to Use Secure Cookies



Pswd_Cookie can be replaced with one of the other authentication cookies in Figure 4.1

Applications of Secure Cookies

- **User Authentication**
- **Electronic Transaction**
- **Eliminating Single-Point Failure**
- **Pay-per-Access**
- **Attribute-based Access Control**

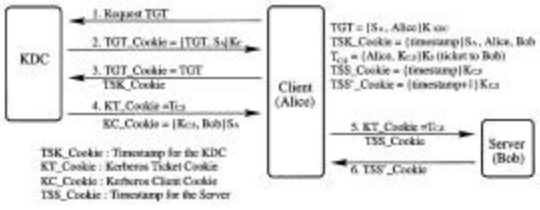
Authentication Cookies

Server-Pull Architecture

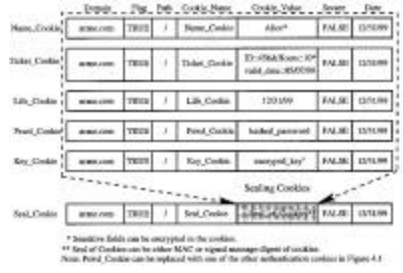
- **Each user**
  - **presents only authentication information to Web servers**
- **Each Web server**
  - **pulls users' attributes from the Attribute Server**
- **Authentication information and attribute do not go together**
- **More convenient for users**
- **Less convenient for Web servers**

Secure Cookies for Electronic Transactions

Kerberos-Based Authentication by Secure Cookies

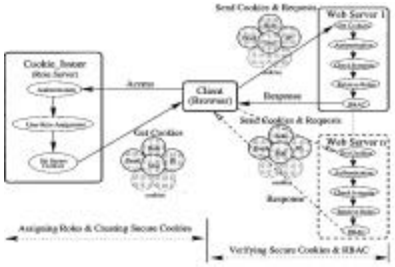


Secure Cookies for Pay-Per-Access



Secure Cookies for RBAC

RBAC on the Web by Secure Cookies



X.509 Certificate

- **Digitally signed by a certificate authority**
  - **to confirm the information in the certificate belongs to the holder of the corresponding private key**
- **Contents**
  - **version, serial number, subject, validity period, issuer, optional fields (v2)**
  - **subject's public key and algorithm info.**
  - **extension fields (v3)**
  - **digital signature of CA**
- **Binding users to keys**
- **Certificate Revocation List (CRL)**

X.509 Certificate

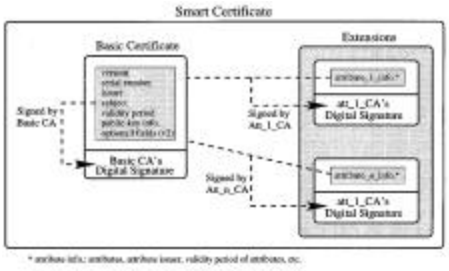
Smart Certificates

- **Short-Lived Lifetime**
  - **More secure**
    - **typical validity period for X.509 is months (years)**
    - **users may leave copies of the corresponding keys behind**
    - **the longer-lived certificates have a higher probability of being attacked**
  - **No Certificate Revocation List (CRL)**
    - **simple and less expensive PKI**

Smart Certificates

- **Containing Attributes Securely**
  - **Web servers can use secure attributes for their purposes**
  - **Each authority has independent control on the corresponding information**
    - **basic certificate (containing identity information)**
    - **each attribute can be added, changed, revoked, or re-issued by the appropriate authority**
      - e.g., role, credit card number, clearance, etc.
  - **Short-lived certificate can remove CRLs**

Separate CAs in a Certificate



Smart Certificates

- **Postdated Certificates**
  - **The certificate becomes valid at some time in the future**
  - **possible to make a smart certificate valid for a set of duration**
  - **supports convenience**
- **Confidentiality**
  - **Sensitive information can be**
    - **encrypted in smart certificates**
      - e.g. passwords, credit card numbers, etc.

A Smart Certificate

Applications of Smart Certificates

- **On-Duty Control**
- **Compatible with X.509**
- **User Authentication**
- **Electronic Transaction**
- **Eliminating Single-Point Failure**
- **Pay-per-Access**
- **Attribute-based Access Control**

Injecting RBAC to Secure a Web-based Workflow System

Gail-Joon Ahn and Ravi Sandhu
George Mason University

Myong Kang and Joon Park
Naval Research Laboratory

WORKFLOW MANAGEMENT SYSTEMS

- **Control and coordinate processes that may be processed by different processing entities**
- **Received much attention**
- **Marriage with Web technology**
- **Minimal security services**

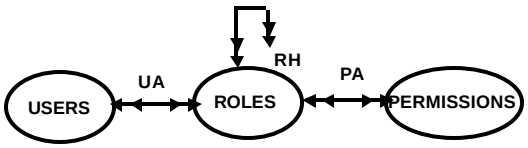
OBJECTIVE

- **Inject role-based access control (RBAC) into an existing web-based workflow system**

WHY RBAC?

- **A mechanism which allows and promotes an organization-specific access control policy based on roles**
- **Has become widely accepted as the proven technology**

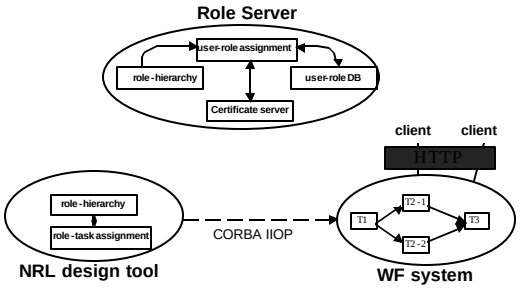
SIMPLIFIED RBAC MODEL



ROLE-BASED SECURE WORKFLOW SYSTEM

- **Workflow Design Tool**
- **Workflow (WF) System**
- **Role Server**

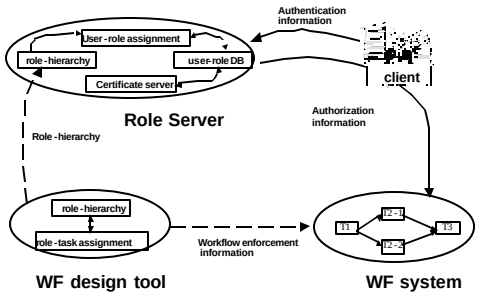
BASIC COMPONENTS



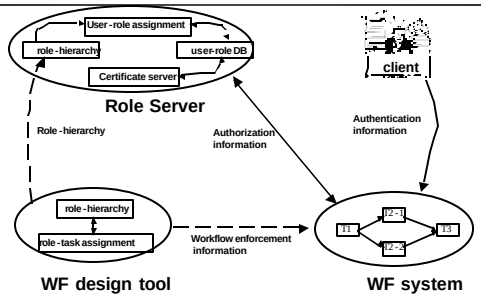
ARCHITECTURES

- **USER-PULL STYLE**
- **SERVER-PULL STYLE**

USER-PULL STYLE



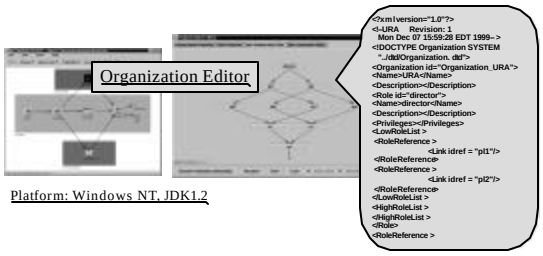
SERVER-PULL STYLE



NRL (Naval Research Lab.) DESIGN TOOL

- **design workflow model**
- **create role and role hierarchies**
- **assign role to task**
- **exporting role hierarchies to role server**

NRL DESIGN TOOL (Cont'd)

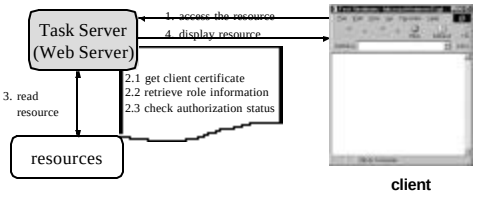


Platform: Windows NT, JDK1.2

WORKFLOW SYSTEM

- **each task server is web server**
- **user should present client authentication certificate**
- **user's privilege is authorized by content of certificate (specially client's role information)**

ROLE AUTHORIZATION ON WORKFLOW SYSTEM



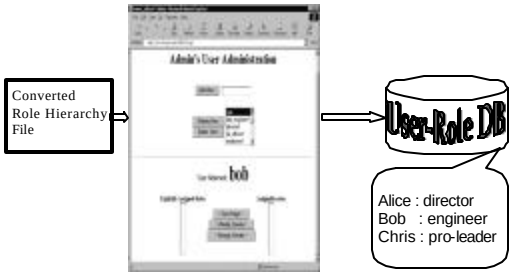
ROLE SERVER

- **User Role Assignment**
- **Certificate Server**

USER ROLE ASSIGNMENT

- **maintain role hierarchies and user database**
- **assign users to roles**
- **generate user-role database**

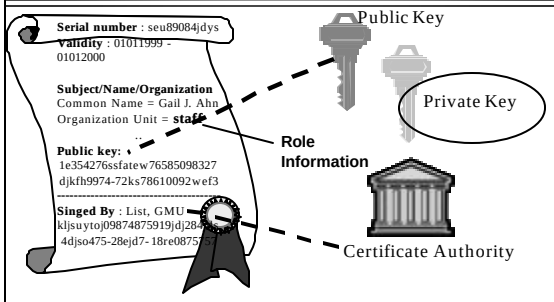
USER ROLE ASSIGNMENT (Cont'd)



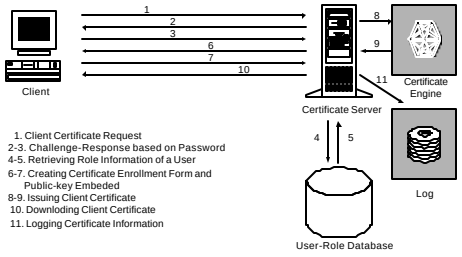
CERTIFICATE SERVER

- **authenticate client**
- **retrieve client's role information from user-role database**
- **issue certificate with client's role information**

X.509 CERTIFICATE

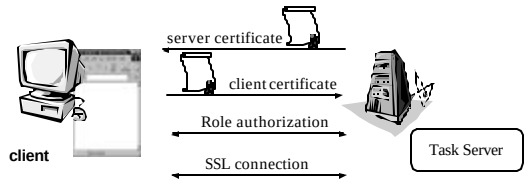


CERTIFICATE ISSUE

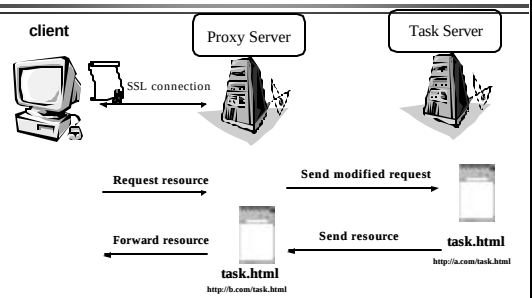


1. Client Certificate Request
2-3. Challenge-Response based on Password
4-5. Retrieving Role Information of a User
6-7. Creating Certificate Enrollment Form and Public-key Embeded
8-9. Issuing Client Certificate
10. Downloding Client Certificate
11. Logging Certificate Information

CERTIFICATE AUTHORIZATION OVER SSL



REVERSE PROXYING
(MINIMAL CHANGES IN SERVER SIDE)



FINAL SCENARIO