Integrated Identity and Access Management with I2MI Tools

http://arch.doit.wisc.edu/keith/i2/
Integ-tb-kh-01.ppt
Tom Barton, U Chicago
Keith Hazelton, U Wisconsin
Internet2 Fall Member Meeting
Sept. 21, 2005, Philadelphia
Session overview

- Identity and Access Management (IAM) service-based model & I2MI tools
- Service integration across I2MI and with I2MI: illustrative examples
- I2MI suite and integration techniques
- Gaps in the tools and in the model
- Harmonization objectives for I2MI tools
From Construction to Integration

- Construction
  - Raw materials into systems
- Integration
  - Subsystems into whole systems
  - Multiple systems into ecosystems
- We’re all moving from construction to integration
- The integration story across IAM services and with IAM services
### IAM: Generic Services

<table>
<thead>
<tr>
<th>Verb</th>
<th>Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reflect</td>
<td>Data of interest from systems of record into registry, directory</td>
</tr>
<tr>
<td>Join</td>
<td>Identity information across systems</td>
</tr>
<tr>
<td>Manage</td>
<td>Credentials, group memberships, affiliations, privileges, services, policies</td>
</tr>
<tr>
<td>Provide</td>
<td>IAM info via - relay thru run-time request/response - provisioning into App/Service stores</td>
</tr>
<tr>
<td>Authenticate (AuthN)</td>
<td>Claimed identities</td>
</tr>
<tr>
<td>Authorize (AuthZ)</td>
<td>Access or denial of access</td>
</tr>
<tr>
<td>Log</td>
<td>Usage for audit</td>
</tr>
</tbody>
</table>
Reflect, Join, and Manage Credentials: One mapping to I2MI

Systems of Record
- Stdnt
- HR
- Other

Enterprise Directory
- Registry
- LDAP

Reflect
Join
Manage Credentials
• Collect bits of identity information in all the relevant IT systems
• Use business logic to
  • Establish which records correspond to the same person
  • Maintain that identity join in the face of changes to data in collected systems
• Assign a unique identifier for cross-system link
Manage Credentials

• When to assign, activate credentials
  • (as early as possible)
• Who gets them? Applicants? Prospects?
• “Guest” NetIDs (temporary, identity-less)
• Reassignment (never; except…)
• Please send me a feed…
  • Argument for WebISO
Manage IAM Info and Provide it via run-time calls or provisioning

Enterprise Directory

Apps / Resources

AuthZ

Log

PROVIDE Provision

PROVIDE Relay

Manage Groups, Privilges,...

Join Manage Creds

Reflect

Systems of Record
IAM Services mapped to I2MI Tools

Systems of Record

- Central AuthN/WebISO
- Grouper
- Signet
- Grouper
- Reflect
- Manage Groups, Privs,.....
- Enterprise Directory
- Provide
- Provide
- Relay
- AuthZ Log
- Apps / Resources
- AuthZ Log
- AuthZ Log
Other I2MI artifacts

- Object classes (info models, data structures, protocol bindings)
- Documentation of practices
- Implementation roadmaps
• "New hires should get into system as soon as they accept the offer of employment instead of after they show up for work."

• They should get NetIDs, email, calendar, portal, but not Health Services or Rec. Sports facility access."
New services: integration issues

• “Undergrad applicants should get into the system once they've accepted offer of admittance and not when they show up to register for classes.”

• They should get NetIDs, email, calendar, portal, but not Health Services or Rec. Sports facility access."
Weaving in the new service

- Reflect
  - Employee/Applicant lifecycle events (accepted offer, start date,...)
- Join
  - Many new employees are students; Some employees apply to become students.
- Credential / AuthN
  - Give them NetIDs with lower level of assurance because Identity proofing is weak until they show up in person.
Weaving in the new service

- Manage affiliations
  - Define lifecycle of employee/applicant role, automate state changes based on events in Systems of Record / Authoritative Systems (HR, SIS)

- Manage privileges
  - Map employee/applicant-lifecycle affiliations to appropriate set of privileges
Weaving in the new service

• Provide/Provision
  • Push new hire info out to systems that need to create accounts/user records

• Provide/Relay
  • Make Affil/Priv info available at App/Service run time when user requests it

• AuthZ
  • Make sure target apps act in line with the person/affil/privilege info
Attribute Management & Delivery: Affiliation, Privilege, & Privacy

- **SIS**
- **HR**

Core Business Systems

Distributed Authorities

Library ERMs/ Self

**Loader**

**Group Registry**

**Subject API**

**Signet**

**Privilege Registry**

**LDAP**

Shibboleth/GridShib

- **uid: jdoe**
- **eduPersonAffiliation: ...**
- **isMemberOf: ...**
- **eduCourseMember: ...**
- **eduPersonEntitlement: ...**

**ShARPe**

**Attribute Release Policies**

**Attribute Authority**

**Core Business Systems**

**Distributed Authorities**

**Library ERMs/ Self**
• Manages claims expressed about organization’s managed identities
• Under development by the Meta-Access Management System (MAMS) project in Australia
• Initially targeted at Library Enterprise Resource Managers
  • Possibly expand to all users for self-management of per-user ARPs
  • Enables ARP assignment to groups too
• Organizational counterpart to MS’s “Identity Metasystem” concept?
• **Subject** - a person, group, application, or other type of object whose identity is managed by your IAM system
• Abstract the underlying technology and data model from a relying application
• Enable identifier namespaces to be selected to match application needs
  • Username vs. opaque registryID vs. …
• **Scenarios**
  • Map authenticated user to internal security principal
  • Search for or select subjects within application
Subject API: Integration with Site’s IAM

A subject:
subjectID, sourceID, typeID
attribute1 = <single-value>
attribute2 = <multi-value>
...

Subject Types

person
person, application
group

1 - search/select
2 - retrieve
3 - present

Source adapter 1
Source adapter 2
Source adapter 3

LDAP
RDBMS
RDBMS

JDBC, JNDI, or user-supplied

Site IAM Infrastructure

A java application

Subject interface
Source interface
More Subject API Info

- Subject and Source interface specs are at v0.1 – they may yet change
  - Searching
  - Some per-subjectType methods?
- Grouper includes a GroupSourceAdapter that is a provider of ‘group’ subjectTypes from the Group Registry
- Subject API will not support the Join function
- JDBC source adapter is included now, JNDI source adapter will be provided in a subsequent release
Empty spaces in the I2MI toolbox

- Those pesky lines between the boxes--left to the reader
- The lines are where service integration happens
  - Metadirectory functions
  - Provisioning (in the general sense)
  - ??
- Should I2MI try to help with integration tasks?
Modeling the lines: Initial thoughts

- **Data flows?**
  - Event publication on a service bus
  - Content-based routing (à la OM)

- **Service invocations?**
  - SAML request/responses
  - WS-* wide world of web services

- **Is it a particle or a wave?**
  - Document-oriented transformation services
Revive MOM & SIS?

- Keith’s early, prescient ideas
  - MOM = Message Oriented Middleware
  - SIS = Smart Information Switch
- Roland’s deep design (mantra: OM)
  - Content based information routing
  - Governed by embedded policy engine (SPOCP’s brain)
- Walter’s Nth generation provisioner
  - Nexus: consumer-specific mappings & transforms configured transparently
Standard interfaces and constructs are needed to develop, evaluate, or integrate off-the-shelf "identity information integration tools" into the IAM picture

- Interface between SIS and provisioner
- Structure of a stateful lifecycle policy
- Presentation, management & persistence of policy configuration
Role of Grouper, Signet (& ShARPe?)

- Are they source systems?
- Are they registries?
- Until we’ve got a SIS, need to provision directly from each, treated as registries
- Probably should ultimately be treated as source systems
  - Avoid duplicating lifecycle management logic in multiple provisioners
Refine Model of Stateful Lifecycle Management?

- Service-oriented epicycles on some state(s)?
- Standard means of specifying it to SIS?
• ShARPe manages Attribute Release Policy
  • Currently siloed with the Shibboleth Attribute Authority
  • Should it somehow expand to constrain the Delivery service more generally?
• Signet is also a policy execution tool
• We lack a unifying framework and an understanding of the requirements it should serve
Harmonizing I2MI Tools:
Objectives

• Deployment of second and subsequent I2MI tools should be harmonious with the first
  • Common UI look and feel
  • Common 3rd party libraries
  • Common customization & configuration technique
  • Common & complementary build layout & build process
Harmonizing I2MI Tools: Objectives

- We should eat our own dogfood
  - Common technique for integration with site IAM infrastructure
  - Capable of integration with external privilege and/or group management
- Common or coordinated web presence, documentation, product placement info
  - Just starting to address this
  - Steering group formed & documentation resource assigned
Further I2MI Integration Needs

- Cookbook of ways to deploy I2MI tools to address various attribute and access management scenarios
- Mechanism for sustained viability of I2MI tools
  - On-going support
  - Post-working-group model for continued development & QA
The service model could be implemented with any number of toolsets

- I2MI
- Home-grown (perl scripts, SQL tables & procedures, OpenLdap directories,…)
- Vendor offerings
  - Novell, Sun, Oracle, Microsoft, IBM,…

The latter is how many of the non-I2 institutions will proceed. Are their needs out of scope for us?