ASPiS

Architecture for a Shibboleth-Protected iRODS System

Eric Liao, Mark Hedges, Tobias Blanke
King's College London

Andrea Weise, Adil Hasan, Jens Jensen
University of Reading, University of Liverpool, Science and Technology Facilities Council

Interoperability of Digital Repositories @
Queen Mary University of London, UK, 2009

Outline

1. Background
    - Access Management
    - Provenance Capture
2. Design
    - Access Management
    - Provenance Capture
3. Implementation
4. Demo

Project Overview

- Funded by JISC e-Infrastructure programme.
- Partners:
  - Centre for e-Research, King's College London
  - University of Liverpool
  - Science and Technology Facilities Council
  - (University of Reading - very helpful PhD student)
- Project Goals:
  1. enhanced access management for iRODS.
  2. enabling provenance capture in iRODS.
  3. engage a broader community with data grids.

Project Overview

- Funded by JISC e-Infrastructure programme.
- Partners:
  - Centre for e-Research, King's College London
  - University of Liverpool
  - Science and Technology Facilities Council
  - (University of Reading - very helpful PhD student)
- Project Goals:
  1. enhanced access management for iRODS.
  2. enabling provenance capture in iRODS.
  3. engage a broader community with data grids.

Project Overview

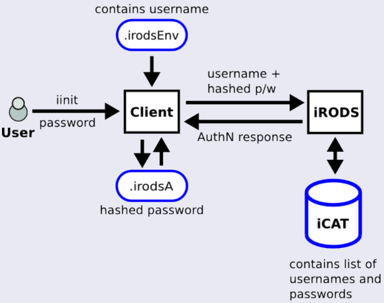
- Funded by JISC e-Infrastructure programme.
- Partners:
    - Centre for e-Research, King's College London
    - University of Liverpool
    - Science and Technology Facilities Council
    - (University of Reading - very helpful PhD student)
- Project Goals:
    1. enhanced access management for iRODS.
    2. enabling provenance capture in iRODS.
    3. engage a broader community with data grids.

Outline

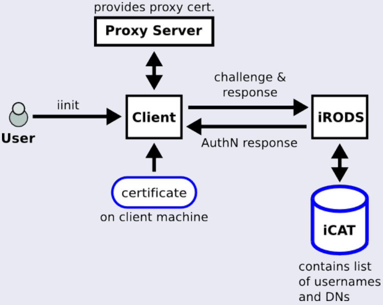
1. Background
    - Access Management
    - Provenance Capture
2. Design
    - Access Management
    - Provenance Capture
3. Implementation
4. Demo

iRODS Authentication

Username + Password



GSI



iRODS Authorization

- iCAT stores information on:
  - Users
  - Domains
  - Groups
  - Access Control Lists (ACLs)
- Access managed according to:
  - Mode of access (read / write / delete / annotate)
  - By user, domain, group
- Identity information held centrally

iRODS Authorization

- iCAT stores information on:
  - Users
  - Domains
  - Groups
  - Access Control Lists (ACLs)
- Access managed according to:
  - Mode of access (read / write / delete / annotate)
  - By user, domain, group
- Identity information held centrally

iRODS Authorization

- iCAT stores information on:
    - Users
    - Domains
    - Groups
    - Access Control Lists (ACLs)
- Access managed according to:
    - Mode of access (read / write / delete / annotate)
    - By user, domain, group
- Identity information held centrally

UK Federation

- **UK Access Management Federation for Education and Research**
  - Based on SAML (**S**ecurity **A**ssertion **M**arkup **L**anguage)
  - Provides a single access solution to online resources/services
  - Metadata based on the Internet2 eduPerson LDAP schema
- Core Federation eduPerson attributes
  - *ScopedAffiliation* → staff@kcl.ac.uk, visitor@stfc.ac.uk
  - *TargetedId* → idp.kcl.ac.uk!sp.stfc.uk!<opaque string>
  - *PrincipalName* → eric.liao@kcl.ac.uk
  - *Entitlement* → urn:mace:ac.uk:irods.stfc.ac.uk:visitor

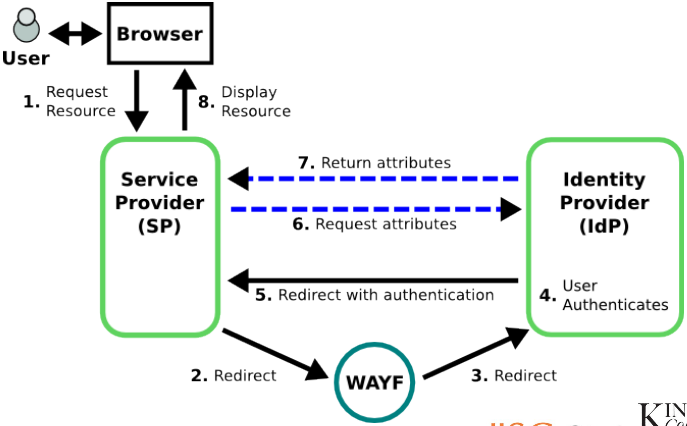
UK Federation

- **UK Access Management Federation for Education and Research**
  - Based on SAML (**S**ecurity **A**ssertion **M**arkup **L**anguage)
  - Provides a single access solution to online resources/services
  - Metadata based on the Internet2 eduPerson LDAP schema
- **Core Federation eduPerson attributes**
  - *ScopedAffiliation* → staff@kcl.ac.uk, visitor@stfc.ac.uk
  - *TargetedId* → idp.kcl.ac.uk!sp.stfc.uk!<opaque string>
  - *PrincipalName* → eric.liao@kcl.ac.uk
  - *Entitlement* → urn:mace:ac.uk:irods.stfc.ac.uk:visitor

Shibboleth

- SAML software for federated access to web based resources
- Based on circle of trust among organisations
- User identities managed locally to their institution
- Access to resources managed locally to the owning institution

Shibboleth Information Flow



Outline

1. Background
   - Access Management
   - Provenance Capture
2. Design
   - Access Management
   - Provenance Capture
3. Implementation
4. Demo

Overview

- Provenance → history of operation applied to a digital object
- Provenance is an important issue
  - Gives history of events
  - Allows to verify the authenticity of data
  - Determines quality of data
  - Supports researchers in many ways (e.g. re-executing experiments)
- Provenance in iRODS
  - iRODS does not capture changes made to data
  - iRODS's metadata is not sufficient to capture workflows

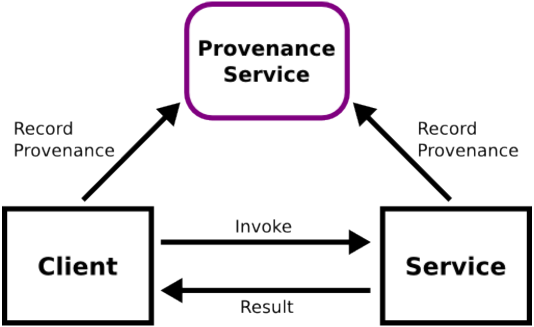
Overview

- Provenance → history of operation applied to a digital object
- Provenance is an important issue
  - Gives history of events
  - Allows to verify the authenticity of data
  - Determines quality of data
  - Supports researchers in many ways (e.g. re-executing experiments)
- Provenance in iRODS
  - iRODS does not capture changes made to data
  - iRODS's metadata is not sufficient to capture workflows

Overview

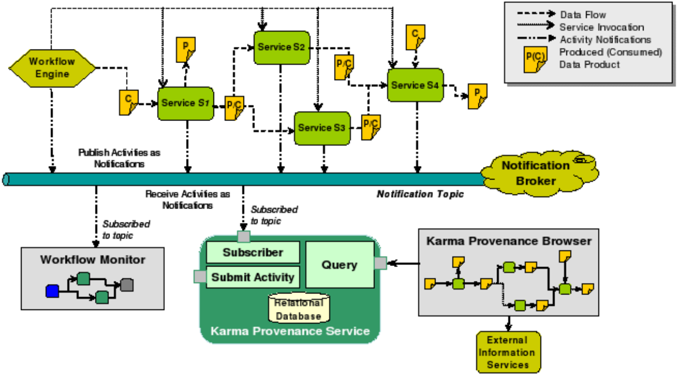
- Provenance → history of operation applied to a digital object
- Provenance is an important issue
  - Gives history of events
  - Allows to verify the authenticity of data
  - Determines quality of data
  - Supports researchers in many ways (e.g. re-executing experiments)
- Provenance in iRODS
  - iRODS does not capture changes made to data
  - iRODS's metadata is not sufficient to capture workflows

PASOA



- Independent protocols for recording and accessing provenance

Karma



processes

- Publish-subscribe notification protocol

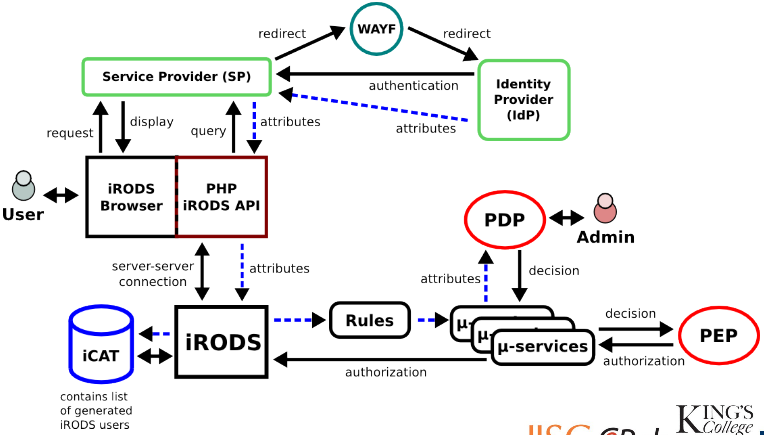
Outline

1. Background
   - Access Management
   - Provenance Capture
2. Design
   - Access Management
   - Provenance Capture
3. Implementation
4. Demo

Requirements

- Devolve authentication service to user's home institution
- Common interface layer to decouple authorization services
- Allowing fine-grained access rights to be defined for roles, not just user identities
- No interference to iRODS core system

Architecture



iRODS Rules

<from the iRODS core.irb file>

```
...
acPreprocForDataObjOpen|$objPath1 like
/$rodsZoneProxy/home/$userNameClient/*|msiSortDataObj(random)|nop
acPreprocForDataObjOpen||acGetShibAuthorization(acPreprocForDataObjOpen,
$userNameClient)##msiSortDataObj(random)|nop##nop
...
acGetShibAuthorization(*rule, *user)||acGetAuthorizationInfo(*rule, *user)|nop
acGetAuthorizationInfo(*rule, *user)||msiGetShibAttributes(*user,
*attributes)##msiGetObjectPermissions(*rule, $objPath1, *readPerm, *updatePerm,
*deletePerm)##acCheckPermissions(*rule, *attributes, *readPerm, *updatePerm,
*deletePerm)|nop##nop##nop acCheckPermissions(*rule, *attributes, *readPerm, *updatePerm,
*deletePerm)||msiCheckPermissions(*attributes, *readPerm, *updatePerm, *deletePerm, *rule,
*decision)##msiEnforceAuthorizationDecision($userNameClient, $objPath1, *rule, *decision,
log_file)|nop##nop #acEnforceAuthorizationDecision(*rule,
*decision)||msiEnforceAuthorizationDecision($userNameClient, $objPath1, *rule, *decision,
log_file)|nop
...
```

iRODS Microservices

```
1 acGetShibAuthorization
2  + acGetAuthorizationInfo
3      + msiGetShibAttributes
4      + msiGetObjectPermissions
5      + acCheckPermissions
6          + msiCheckPermissions
7          + msiEnforceAuthorizationDecision
```

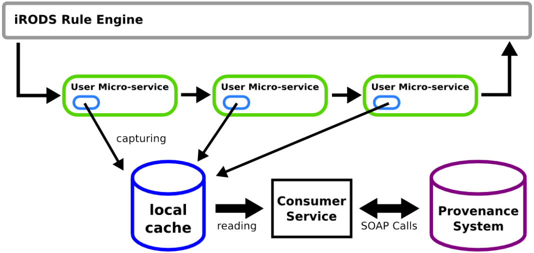
Outline

1. Background
    - Access Management
    - Provenance Capture
2. Design
    - Access Management
    - Provenance Capture
3. Implementation
4. Demo

Requirements

- Key points:
    - Manage data throughout its lifecycle
    - Capture and record information about the data analysis
    - Enforce ownership of data thoughout its lifetime
    - Ensure data access is auditable
    - Ensure infrastructure is robust and scalable
- No interference with iRODS core system
- Provenance system should be applicable for any other system
- Easy to use

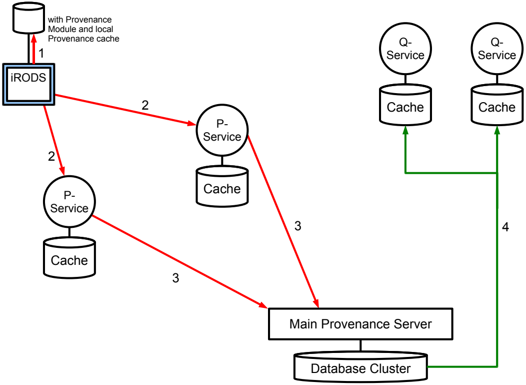
Architecture



Microservice Chain

- Embed provenance microservice in user microservice
- User deals with capturing specific data
- Decouples capturing and reading

Distributed Framework



Access Management

- **User Interface**
    - Highly modified iRODS Browser supporting Shibboleth
- **Middleware**
    - Extended PHP-iRODS interface
    - PHP authentication module
- **iRODS Integration**
    - Custom rules and microservices

Provenance Capture

- **Provenance Framework**
  - Java interface with distributed framework
- **Middleware**
  - Java interface with local provenance cache
- **iRODS Integration**
  - Custom rules and microservices

Live Demo!

Work so far & Future plans

Completed Work

- Developed prototypes for iRODS-Shibboleth integration
- Developed prototypes for iRODS-Provenance integration

Future Work

- Integration of access control and provenance systems
- Testing with use cases

Work so far & Future plans

Completed Work

- Developed prototypes for iRODS-Shibboleth integration
- Developed prototypes for iRODS-Provenance integration

Future Work

- Integration of access control and provenance systems
- Testing with use cases

Thank you

Contacts

mark.hedges at kcl.ac.uk

eric.liao at kcl.ac.uk

tobias.blanke at kcl.ac.uk

a.hasan at rl.ac.uk

j.jensen at stfc.ac.uk