Recent developments in the UNICORE Portal

Maria Petrova-El Sayed, Martin Lischewski, Bernd Schuller, Krzysztof Benedyczak, et al.

June 2016
Outline

- Overview
- Workflow Templates
- Data Sharing
- Demo
UNICORE Portal - Overview

- Current features
  - Job creation, submission, monitoring and management
  - Simple workflow editor
  - Data management

- Flexible authentication and user registration
  - Supports Unity

- Configurable and extensible
Workflow Templates
Use case
Template

Workflow template parameters

Upload file

Editing workflow template file: template1.swf

☐ UC_SIZE: 

☐ UC_INPUT: 

June 2016
The Portal solution

- Masked as a normal job
- Users do not need to learn the workflow editor
- A template file can be uploaded from a favourable location
- The input parameter fields are automatically generated
Data Sharing
Use case 1

FZJ

Webapplication

JUST

Internet

file

file
Use case 2
The use case – overview

- HPC filesystems
  - Unix user IDs and groups
  - Many different ways of access (e.g. SSH)

- Data sharing
  - User A wants to grant read/write access to user B
  - Fine-grained: per file, per directory
  - User B does NOT need to have an account

- Unix permission model (incl. ACLs) is not enough!
Data sharing solution

- Portal
  - Front-end
  - UFTP authentication/authorization
  - Sharing logic and storage of access control entries

- Unity
  - Login and bridge to 3rd party login systems
  - Store user attributes (Unix login/groups)

- UFTPD
  - File access (always using the sharer's Unix login/groups)
Authentication with Unity

1. User provides credentials
2. Client or Webapp authenticates
3. Services are accessed
UFTP - high performance data transfer

**UFTPD server**
- Pseudo-FTP port (open in firewall) for clients
- Local command port (SSL protected) used by Auth server
- Run as root w/ setuid

**UFTP client**
- Authenticate
- Connect to UFTPD
- Send/receive data

**Authentication / Authorisation**
- Client authentication
- User ID mapping

1. Authenticate
2. Initiate client transfer. Pass secret and client IP
3. Transfer
Data sharing - current solution

- Web browser
- Portal server
  - Authenticate user via Unity
  - Connect to UFTPD to initiate transfer
  - Pipe data from UFTPD to client’s web browser
- Unity
  - Client authentication
  - Store user’s Unix ID and forward it to Portal as part of the login process (via SAML assertion)
Demo
Outlook

- Workflow template export feature in URC

- Portal / sharing logic
  - Better storage of ACL entries (relational DB)
  - Other options for storing UFTP Unix IDs? XUUDB?
  - Sharing with email address

- Deploy and realise use cases

- Further decouple sharing logic and ACL storage from UI
  - REST API?
  - Allow access with standalone UFTP client (full UFTP performance!)
Questions?