“Standards for Cloud Federation”

Robert Bohn, PhD
Advanced Network Technologies Division
ITL/NIST

Martial Michel, PhD
Data Machines Corp

IEEE COMPUTER SOCIETY
STANDARDS ACTIVITIES BOARD
WEBINAR SERIES
16 March 2021
Today’s talk

• Introduction
• The Promise of Federation
• Description of Federation
• NIST/IEEE Collaboration
• NIST Cloud Reference Architecture (NIST SP 500-332)
• IEEE/P2320: Standards for Intercloud Interoperability and Federation
• Current Status
• Applications & APIs
Introduction

• NIST defines a *Community Cloud* as supporting organizations that have a common set of interests, e.g., mission, security, policy

• When that community cloud cannot be implemented in one public or private cloud, there is a need to clearly define and implement mechanisms to support the governance and processes *which enable federation and interoperability between different cloud service provider environments* to form a general or mission-specific federated Community Cloud.

• This is the core of *Requirement 5: Frameworks to Support Federated Community Clouds* in the NIST US Government Cloud Computing Technology Roadmap, *NIST SP 500-293 Volume I*
The Promise of Federation

- We need to connect real-world collaboration needs (data and resource sharing) with practical collaboration methods and tools.
- *Federation* addresses how to securely manage collaborations when the resources being shared are inherently distributed.
- There is a spectrum of collaboration needs and approaches:
  - Bare-bones federation: Small-scale, manually managed
  - Industrial federation: Large-scale, highly distributed, automated, accounting, auditing, legal
The Essence of Federation

How can UserA find (discover) SPB?
How can SPB manage its discoverability?
How can SPB validate UserA’s credentials and make access decision?

AuthN – Authentication is establishing your identity.
AuthZ – Authorization is establishing your privilege
NIST/IEEE Collaboration

• NIST recognized the importance of “Frameworks to support seamless implementation of federated community cloud environments” in USG Cloud Computing Roadmap (NIST SP 500-293).

• Collaboration between NIST and IEEE P2302 will help build consensus on creating an Intercloud - an open, transparent infrastructure amongst cloud providers to support evolving technological and business models and the growing demand for standards that address Intercloud interoperability.

• From the IEEE P2302 Project Authorization Request –
  • **Purpose:** This standard creates an economy amongst cloud providers that is transparent to users and applications, which provides for a dynamic infrastructure that can support evolving business models.
  • **Scope:** This standard defines topology, functions, and governance for cloud-to-cloud interoperability and federation.
NIST PWGFC/IEEE P2302 Intercloud Goals & Outputs

• The NIST PWGFC will develop a **cloud federation vocabulary and conceptual model** based on the Scope and Purpose.
  • The PWGFC interim outputs will be contributed to the IEEE P2302 Working Group in real-time.
  • The PWGFC ultimate output will be a NIST Special Publication.

• The IEEE P2302 Intercloud Working Group will develop a **cloud federation standard** based on the Scope and Purpose.
  • The PWGFC interim contributions will serve as input.
  • Feedback on PWGFC vocabulary and conceptual architecture contributions will be provided to the PWGFC in real-time.
  • The P2302 initial output will be an IEEE Standard.
  • Plan to contribute the P2302 Standard to ISO/JTC1 to create an International Standard.
3-Plane Model of Cloud Federation
Essential Characteristics of a Cloud Federation

- A federation is a virtual security and collaboration context that is not necessarily “owned” by any one user or organization.

- Since only specific users, sites, and organizations collaborate for common goals, these participating entities have membership in the federation and identity credentials that are linked to each member.

- Users, sites, and organizations can participate in a federation by choosing to share some of their resources and metadata and making them discoverable and accessible to other federation members.

- Participating members agree upon the common goals and governance of their federation, based on well-known roles, attributes and policies.
Relevant Existing Tools and Standards

- **Securing the communication:**
  - SSL/TLS
  - HMAC

- **Identity, Authorization, Policy**
  - Account name and password
  - Public Key Infrastructure (PKI) and PKI Proxy Certs
  - Kerberos
  - Shibboleth
  - Grid Security Infrastructure (GSI)
  - SAML and XACML
  - OpenID, OAuth, and OpenID Connect
  - UMA

- **Catalogs and Discovery:**
  - Lightweight Directory Access Protocol (LDAP)
  - Active Directory and Active Directory Federation Services
  - Web Service API Gateways
  - DNS/DNSSEC
  - OWL-S

- **Trust and Governance:**
  - Blockchain
  - Consensus Algorithms, e.g., Proof-of-Work, Raft, PAXOS
Areas of Possible/Needed Federation-Specific Standards

How FMs interact with Users, Sites, Admins, and other FMs

- FM Admin API
- FM Federation Admin API
- FM-Site Admin API
- FM-User API
- FM-FM API

Possible to define a standard format for describing or defining a specific type of federation instance

- Resources to be shared and their metadata
- Roles & Attributes
- Resource Discovery
- Federation Membership
- Federation Member Identity Credentials
- Authorization to grant or revoke federation membership
- Authorization to grant or revoke member roles or attributes
- Governance, policies, SLAs
- Security considerations
IEEE SIIF Objectives

• **Purpose:** This standard creates an economy amongst cloud providers that is transparent to users and applications, which provides for a dynamic infrastructure that can support evolving business models.

• **Scope:** To define topology, functions, and governance for cloud-to-cloud interoperability and federation.

  ➢ Support Transparent Infrastructure
    ➢ Like the Internet
    ➢ Like the Phone Network

  ➢ Cloud Implementation Independent
    ➢ Like the Internet Router
    ➢ Like the Phone Network CO Switch
    ➢ Based on Standards

  ➢ Simple Protocol Set, Easy to Join
    ➢ Like an ISP, simple IP based protocols enough to get started
    ➢ Supports Regional Governance

  ➢ Support for Generalized Resource Federation
    ➢ Not Just VM’s – IaaS, PaaS, *aaS
    ➢ Extensibility to Any Describable Resource Type
    ➢ Communities can Add Resource Types

  ➢ Support for Multiple (Open or Proprietary) Federation Topologies
    ➢ Network Abstraction

  ➢ Global Scale Capable
Possible Deployments

- **3rd Party Centralized Trust (3PCT)**
  - Site
  - FM
  - Site
  - Site

- **Peer-to-Peer**
  - Site
  - FM
  - FM
  - Site

**External**

**Internal**
Peer to Peer

**Federation Hosting Service A**
- Member
- FHS Endp
- FHS API Server
- FHS-FHSAPI Server
- FHS-FHS Endp

**Federation Hosting Service B**
- FHS-FHSAPI Server
- FHS-FHS Endp

**FHS Operator**
- FHSOp Endp
- FHSOp Server
Federation Matters: Cloud Bursting

Cloud to Cloud Federation is not yet commonplace

- **Identity:**
  - LDAP on primary
  - Matches/adapted/ignored on burst clouds

- **Data:**
  - copied (as needed)
  - through Virtual Private Network/Cloud tunnels

- **Compute:**
  - Different compute paradigm per cloud
  - migrated/pulled/transformed container images
OpenSource Technology Stacks

**Kubernetes Cluster Federation** [https://github.com/kubernetes-sigs/kubefed](https://github.com/kubernetes-sigs/kubefed)

“coordinate the configuration of multiple Kubernetes clusters from a single set of APIs in a hosting cluster [...] intentionally low-level, and intended to be foundational for more complex multicluster use cases”

part of the “Multicluster Special Interest Group” “focused on solving common challenges related to the management of multiple Kubernetes clusters”

**Federated Keystone** [https://docs.openstack.org/security-guide/identity/federated-keystone.html](https://docs.openstack.org/security-guide/identity/federated-keystone.html)

Keystone supports multiple form of Single Sign Ons, and provide Virtual Organization mapping

- Keystone as an Identity Provider (IdP): enables federation between multiple OpenStack
- Keystone as a Service Provider (SP): consuming identity properties issued by external Identity Provider

Technology Stack to Technology Stack Federation is not yet commonplace

- Expected: different abstractions to the OS
- Federation goes beyond Identity

**P2302** WG aiming to define a set of APIs that are implementation agnostic
(Draft) Federation Hosting Service Operator (FHSOperator) API

**FHSOperator Core** Calls for an FHSOperator to manage a FHS

- **POST** `/FHSOperator` Add a user authorized to create federations
- **GET** `/FHSOperator` List federation owners
- **GET** `/FHSOperator?param1=value1&param2=value2` Query for any kind of server data.
- **GET** `/FHSOperator/{member_id}` Show federation owner details including federations owned.
- **PUT** `/FHSOperator/{member_id}` Change a federation owner's password.
- **DELETE** `/FHSOperator/{member_id}` Delete a federation owner.
- **DELETE** `/FHSOperator/{fed_id}` Delete a federation.

**FHSOperator FHS-FHS** Calls to Enable/Manage FHS-FHS Communication.

- **POST** `/FHSOperator/AllowConnection` Allow a connection from another FHS
- **POST** `/FHSOperator/Connect` Request connection to another FHS
- **GET** `/FHSOperator/ListConnections` List FHS Connections
- **GET** `/FHSOperator/ListConnection/{connection_id}` Specific Connection Info
- **DELETE** `/FHSOperator/Disconnect/{connection_id}` Disconnect from another FHS

⚠️ Work in Progress – Final version may differ
## (Draft) FHS to FHS API

<table>
<thead>
<tr>
<th><strong>Connection</strong></th>
<th>Calls to connect and disconnect FHS.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>POST</strong> /Connect</td>
<td>Connect to another FHS.</td>
</tr>
<tr>
<td><strong>DELETE</strong> /Disconnect</td>
<td>Disconnect from another FHS.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Federation</strong></th>
<th>Calls to join, manage, and leave a specific Federation Instance.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>POST</strong> /JoinFederation</td>
<td>Join a Federation.</td>
</tr>
<tr>
<td><strong>PUT</strong> /UpdateFederation</td>
<td>Update a Federation.</td>
</tr>
<tr>
<td><strong>GET</strong> /ValidateMember</td>
<td>Validate a Member from another FHS.</td>
</tr>
<tr>
<td><strong>DELETE</strong> /LeaveFederation</td>
<td>Leave a Federation.</td>
</tr>
</tbody>
</table>

### Federation FHS-FHS

| **GET** /Federation/Query | Get a list of federations known to this FHS at other connected FHSs |
| **POST** /Federation/Join/{fed_id} | Request to join a federation known to this FHS but at another connected FHSs |
| **POST** /Federation/JoinGrant/{request_id} | Grant a federation join request |
| **DELETE** /Federation/Leave/{fed_id} | Leave a federation that originated at another connected FHSs |

⚠️ **Work in Progress – Final version may differ**
Members to FHS API (2/2)

### Attributes
Calls to manage federation roles and attributes.

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td><code>/Attribute/{fed_id}</code></td>
<td>Create an attribute in a federation</td>
</tr>
<tr>
<td>GET</td>
<td><code>/Attribute/{fed_id}</code></td>
<td>List the attributes in this federation</td>
</tr>
<tr>
<td>DELETE</td>
<td><code>/Attribute/{fed_id}/[attr_id]</code></td>
<td>Delete an attribute from a federation.</td>
</tr>
</tbody>
</table>

### Membership
Calls to manage federation membership.

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td><code>/Membership/{fed_id}</code></td>
<td>Grant membership to a federation</td>
</tr>
<tr>
<td>GET</td>
<td><code>/Membership/{fed_id}</code></td>
<td>List members of a federation</td>
</tr>
<tr>
<td>GET</td>
<td><code>/Membership/{fed_id}/[member_id]</code></td>
<td>Get specific member info</td>
</tr>
<tr>
<td>DELETE</td>
<td><code>/Membership/{fed_id}/[member_id]</code></td>
<td>Revoke a member's federation membership.</td>
</tr>
</tbody>
</table>

### Authorization
Calls to grant/revoking authorization attributes.

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUT</td>
<td><code>/Authorization/{fed_id}/[member_id]/[attr_id]</code></td>
<td>Grant an authorization attribute to a federation member.</td>
</tr>
<tr>
<td>DELETE</td>
<td><code>/Authorization/{fed_id}/[member_id]/[attr_id]</code></td>
<td>Revoke an authorization attribute from a federation member.</td>
</tr>
</tbody>
</table>

### Services
Calls to manage a federation's services.

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td><code>/Services/{fed_id}</code></td>
<td>Register a service with a federation</td>
</tr>
<tr>
<td>GET</td>
<td><code>/Services/{fed_id}/[svc_id]</code></td>
<td>Get info for a specific service in this federation</td>
</tr>
<tr>
<td>DELETE</td>
<td><code>/Services/{fed_id}/[svc_id]</code></td>
<td>Delete a specific service in this federation</td>
</tr>
<tr>
<td>PUT</td>
<td><code>/Services/{fed_id}/[svc_owner_id]/[svc_id]</code></td>
<td>Update the metadata, discovery policy or access policy for a service in this federation</td>
</tr>
</tbody>
</table>

### Discovery
Query catalog for available services in a federation.

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><code>/Discovery/{fed_id}/[member_id]?param1=value1&amp;param2=value2</code></td>
<td>Look-up services.</td>
</tr>
</tbody>
</table>

### Invocation
Calls to invoke services available in a federation. Any HTTP verb might be possible.

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPTIONS</td>
<td><code>/Invocation/{fed_id}/[svc_id]</code></td>
<td>Call a service I am authorized to use.</td>
</tr>
<tr>
<td>HEAD</td>
<td><code>/Invocation/{fed_id}/[svc_id]</code></td>
<td>Call a service I am authorized to use.</td>
</tr>
<tr>
<td>GET</td>
<td><code>/Invocation/{fed_id}/[svc_id]</code></td>
<td>Call a service I am authorized to use.</td>
</tr>
<tr>
<td>POST</td>
<td><code>/Invocation/{fed_id}/[svc_id]</code></td>
<td>Call a service I am authorized to use.</td>
</tr>
<tr>
<td>PUT</td>
<td><code>/Invocation/{fed_id}/[svc_id]</code></td>
<td>Call a service I am authorized to use.</td>
</tr>
<tr>
<td>PATCH</td>
<td><code>/Invocation/{fed_id}/[svc_id]</code></td>
<td>Call a service I am authorized to use.</td>
</tr>
<tr>
<td>DELETE</td>
<td><code>/Invocation/{fed_id}/[svc_id]</code></td>
<td>Call a service I am authorized to use.</td>
</tr>
</tbody>
</table>

⚠️ Work in Progress – Final version may differ
Open Research Cloud Alliance

A global research community partnership to enable access to data, compute and application services across multiple scientific research clouds

Enable federation between and among:
- Academic research clusters
- Scientific research clouds
- Domain specific research and data federation organizations
- Commercial public cloud providers supporting research

Federation Services
Centralized or Distributed
- Access, security, audit policies
- Identity and Authorization
- Data/Resource Discovery Services
- Publication of Data and Resources

Researcher completes analytics and retrieves results

POC: Khalil Yazdi <khalilyazdi@outlook.com>
How to Find Information

• IEEE P2302 Intercloud Working Group URL
  • [http://sites.ieee.org/sagroups-2302/](http://sites.ieee.org/sagroups-2302/)
• Request to be on IEEE P2302 Intercloud Working Group List
  • [STDS-P2302@ieee.org](mailto:STDS-P2302@ieee.org)
• Robert Bohn – Chair IEEE/P2302  [Robert.Bohn@nist.gov](mailto:Robert.Bohn@nist.gov)
• Martial Michel – Co-Chair IEEE/P2302, Data Machines Corp, [martialmichel@datamachines.io](mailto:martialmichel@datamachines.io)
Thank You