OCM
Towards IETF Standardization of Trusted Resource Sharing

Micke Nordin - Giuseppe Lo Presti
Michiel de Jong - Mahdi Baghbani
(Presenter: Richard Freitag)

What is OCM?

Open Cloud Mesh (OCM) is a server federation protocol that is used to notify a Receiving Party that they have been granted access to some Resource.

Alice on System A wishes to share a resource (e.g., a file) with another user (e.g., Bob on System B) without transferring the resource itself or requiring Bob to log in to System A.

OCM - Establishing Contact

- Direct Contact
  - Use of OCM Address
  - AKA Federated Cloud ID
- Address Books
  - Address book of Sending Party
  - e.g., Contacts app

- Public Link Flow
  - Anonymous link for viewing a resource
  - Add to local system
- Public Invite Flow
  - Share resources via e.g., mailing lists
- Invite Flow
  - Requires receiving party to accept invite
  - Establish bidirectional trust

Who uses OCM?

- Current support for OCM
  - Nextcloud
  - OpenCloud
  - OwnCloud
  - Reva/CernBox
  - OpenGeoMesh
  - SeaFile
- Future support? (Realistic)
  - Filesender
  - rclone
- Future support? (Wishlist…)
  - Dropbox
  - Google Drive
  - OneDrive
  - ...

OCM Security

New security features in 1.2.0 and 1.3.0 are:

- **Request signing**
- **MFA signaling**
- **New /token endpoint**
- **Deprecation of less secure access methods**
- **Where are you from**
- **Providers and protocols**

IETF Internet Draft

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 21 August 2026

G. Lo Presti
CERN
M. B. de Jong
M. Baghbani
Ponder Source
M. Nordin
SUNET
17 February 2026

Open Cloud Mesh
draft-ietf-ocm-open-cloud-mesh-03

Abstract

Open Cloud Mesh (OCM) is a server federation protocol that is used to notify a Receiving Party that they have been granted access to some Resource. It has similarities with authorization flows such as OAuth, as well as with social internet protocols such as ActivityPub and email.

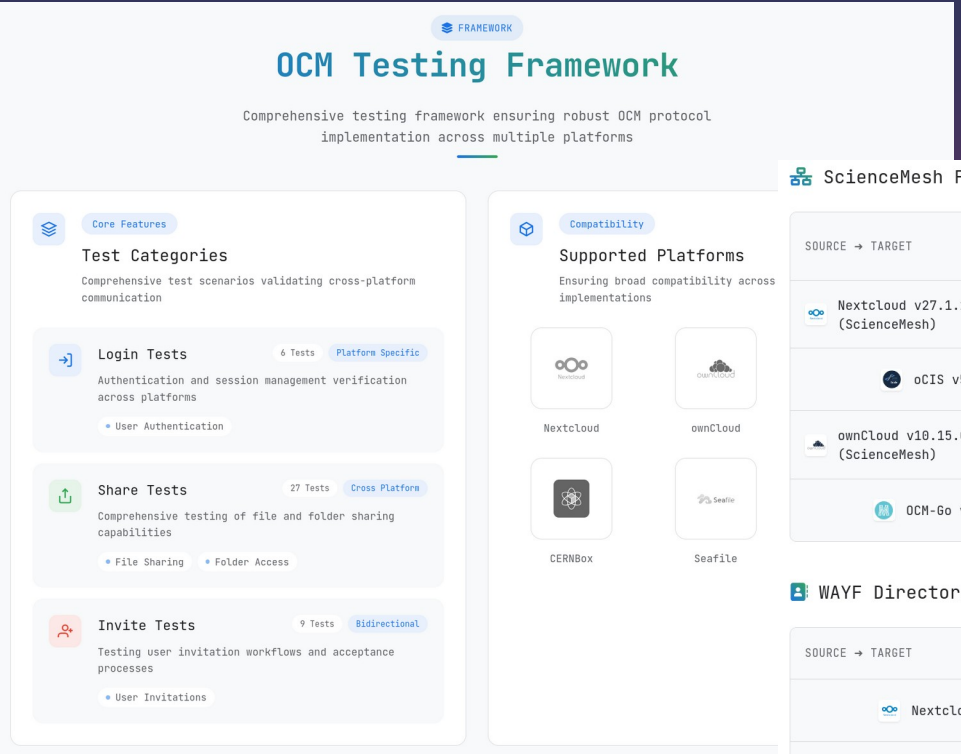
A core use case of OCM is when a user (e.g., Alice on System A) wishes to share a resource (e.g., a file) with another user (e.g., Bob on System B) without transferring the resource itself or requiring Bob to log in to System A.

While this scenario is illustrative, OCM is designed to support a broader range of interactions, including but not limited to file transfers.

Open Cloud Mesh handles interactions only up to the point where the Receiving Party is informed of their access to the Resource. Actual Resource access is subsequently managed by other protocols, such as WebDAV.

https://datatracker.ietf.org/doc/draft-ietf-ocm-open-cloud-mesh/

OCM Test Suite – Verification and Validation



ScienceMesh Federation Tests

SOURCE → TARGET	Nextcloud v27.1.11 (ScienceMesh)	oCIS v5.0.9	ownCloud v10.15.0 (ScienceMesh)	OCM-Go v1.0.0
Nextcloud v27.1.11 (ScienceMesh)	✗	✗	✗	?
oCIS v5.0.9	✗	✓	✗	?
ownCloud v10.15.0 (ScienceMesh)	✗	✗	✗	?
OCM-Go v1.0.0	?	?	?	✓

WAYF Directory Service Tests

SOURCE → TARGET	Nextcloud v33	CERNBox v2	OCM-Go v1.0.0
Nextcloud v33	✗	✗	?
CERNBox v2	✗	✓	?
OCM-Go v1.0.0	?	?	✓

https://cs3org.github.io/ocm-test-suite/

OCM and EOSC – Practical Demo

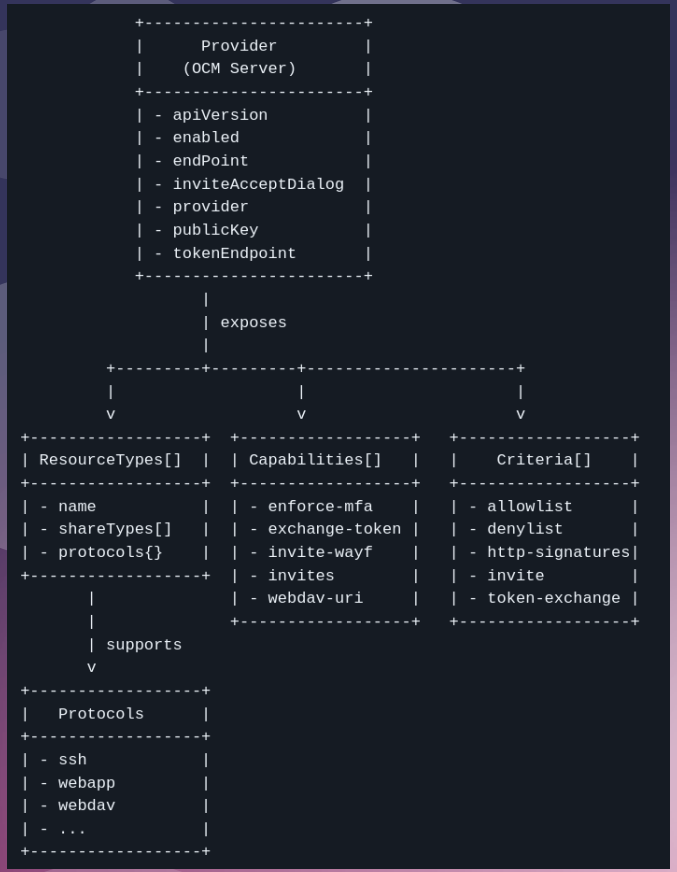
https://sunet.drive.sunet.se/s/Enjq67fGtLKwqbQ

Security – MFA and OCM – Trust still needs Governance

If a Receiving Server exposes the capability enforce-mfa, it indicates that it will try and comply with a MFA requirement set on a Share. If the Sending Server trusts the Receiving Server, the Sending Server MAY set the requirement mfa-enforced on a Share, which the Receiving Server MUST honor. A compliant Receiving Server that signals that it is MFA-capable MUST NOT allow access to a Resource protected with the mfa-enforced requirement, if the Receiving Party has not provided a second factor to establish their identity with greater confidence.

Since there is no way to guarantee that the Receiving Server will actually enforce the MFA requirement, it is up to the Sending Server to establish a trust with the Receiving Server such that it is reasonable to assume that the Receiving Server will honor the MFA requirement. This establishment of trust will inevitably be implementation dependent, and can be done for example using a pre approved allow list of trusted Receiving Servers. The procedure of establishing trust is out of scope for this specification: a mechanism similar to the ScienceMesh (https://sciencemesh.io) integration for the Invite (Section 4.4) capability may be envisaged.

OCM – Providers and Protocols



Relevance to the grid computing community:

- **Traditional sync and share can be used**
- **Protocols have been proposed and can be implemented: ssh, webapp, webdav**
- **Data movement and data mobility can be simplified**

OCM – Get Involved!

- **https://github.com/cs3org/OCM-API**
- **https://matrix.to/#/#cs3org_OCM:gitter.im**
- **https://datatracker.ietf.org/doc/draft-lopresti-open-cloud-mesh/**
- **https://mailman3.ietf.org/mailman3/lists/ocm.ietf.org/**