

Evolving the OCM test suite to ease implementations' compliance



OPENCLOUD**MESH**

Sandro Mesterheide

Agenda

- Recap CS3 from 2023
- Goal for the test suite
- E2E testing OCM and particularities
- Updates and milestones
- New Cypress test framework & Demo

Recap CS3 conference 2023

Open Cloud
Mesh



Test Suite Work



Who wants
NGI Funding
for this?



Funding secured!

- NGI0 Core fund by NLnet awarded 50k EUR for work on Open Cloud Mesh
- Allows for substantial work on the test suite, project plan contains a total of 8 milestones
- Work done by – currently:
 - Michiel de Jong, Giuseppe Lo Presti, Mahdi Baghbani, Sandro Mesterheide

Goal

- The test suite aims to test OCM compliance between implementations

	OC	NC	OCIS	Seafile
OC	Vendor tested	Often lacks compatibility	Often lacks compatibility	Often lacks compatibility
NC		Vendor tested	Often lacks compatibility	Often lacks compatibility
OCIS			Vendor tested	Often lacks compatibility
Seafile				Vendor tested

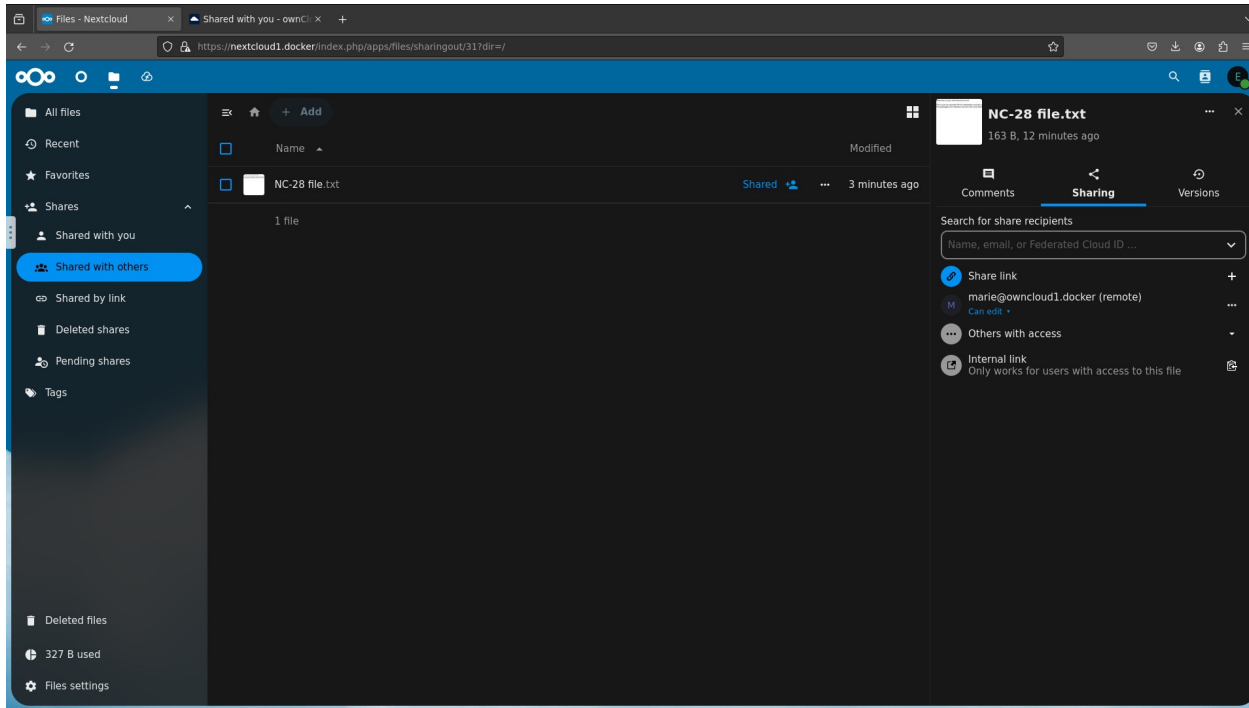


Vendor tested



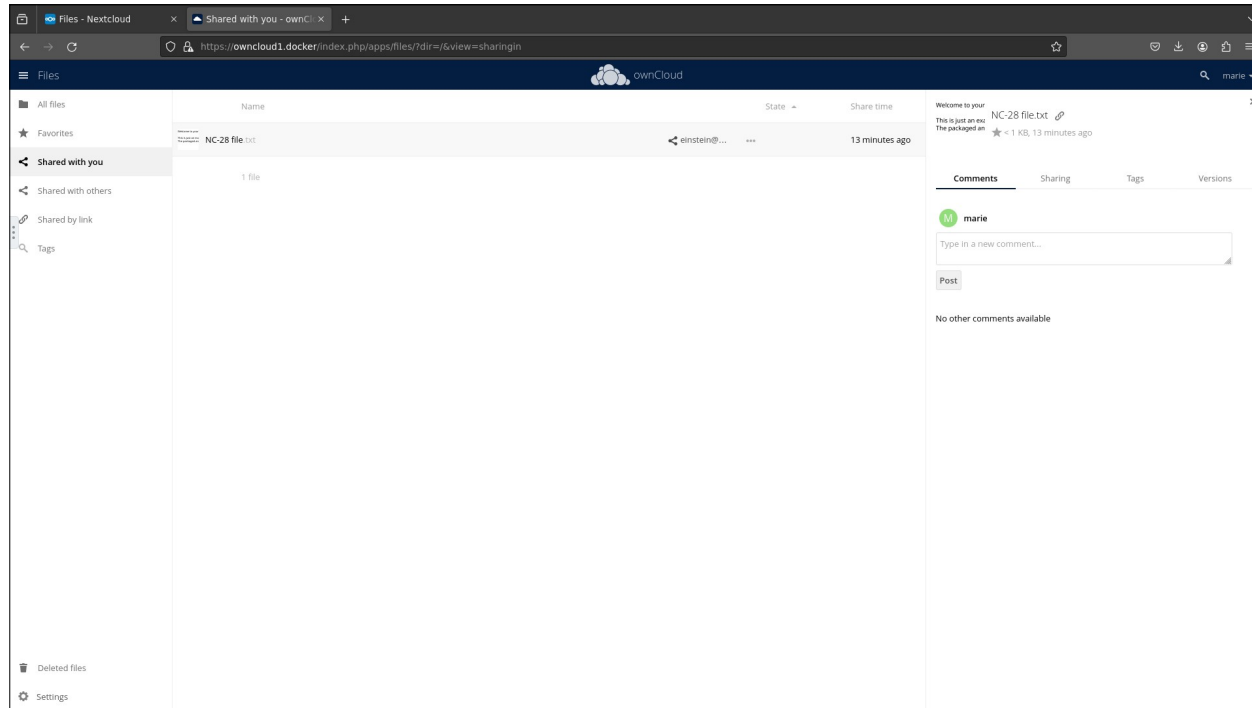
Often lacks compatibility

E2E testing OCM



- testing the web interface of different EFSS
- state observable in the GUI

E2E testing OCM



- testing the web interface of different EFSS
- state observable in the GUI

What can be tested?

- Share flow
 - OCM-to-user
 - OCM-to-group
 - Public link flow (= create share from public link)
 - Invite flow (ie. Scienemesh)
 - Federated groups flow (2..N instances)
- Capabilities (eg. MFA awareness)
- Notifications, Share updates (?)

Create a new share

After the provider created a local share, it sends a `share` object to the consumer containing the information which is needed to start synchronization between the two services.

REQUEST BODY SCHEMA: `application/json`

The JSON object to create a new share at the consumer side.

```
shareWith
  required
  string
  Consumer specific identifier of the user, group or
  federation the provider wants to share the
  resource with. This is known in advance. Please
  note that the consumer service endpoint is
  known in advance as well, so this is no part of the
  request body.
name
  required
  string
  Name of the resource (file or folder).
description
  string
  Optional description of the resource (file or
  folder).
providerId
  required
  string
  Identifier to identify the shared resource at the
  provider side. This is unique per provider such
  that if the same resource is shared twice, this
  providerId will not be repeated.
owner
  required
  string
  Provider specific identifier of the user who owns
  ..
```


Updates from last year

- Runs in CI pipeline (GitHub for the moment)
 - Recently reduced runtime
 - using self-signed SSL certificates
- Replaced test framework: Puppeteer with Cypress
- Systems under test updated
 - Nextcloud 28, OC 10, more coming...
- exemplary Share-With flow in Cypress
- Videos and Screenshots as CI-Job artifacts

Coverage and milestones

Bredth (# platforms) and depth (OCM features)

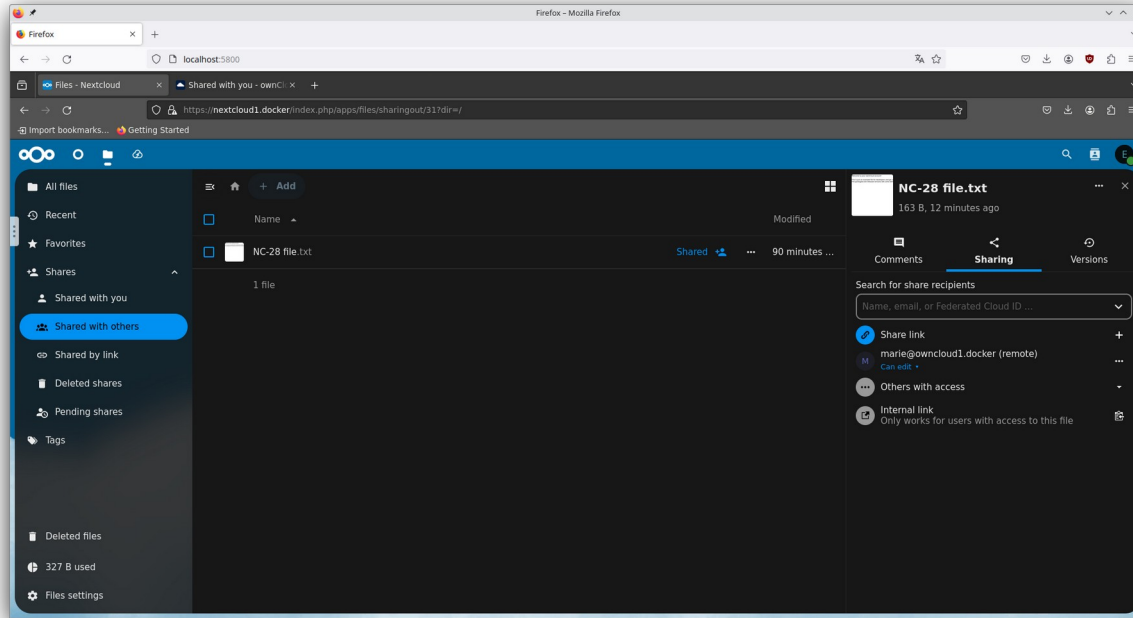
- Share flow
 - OCM-to-user → MS2
 - OCM-to-group → MS2
 - Public link flow → MS3
 - Invite flow → MS4
 - Federated groups flow → MS7
- MFA awareness → MS7
- Notifications, Share updates (no plans)
- Nextcloud
- OwnCloud
- OCIS
- Seafile

Dockerized setup

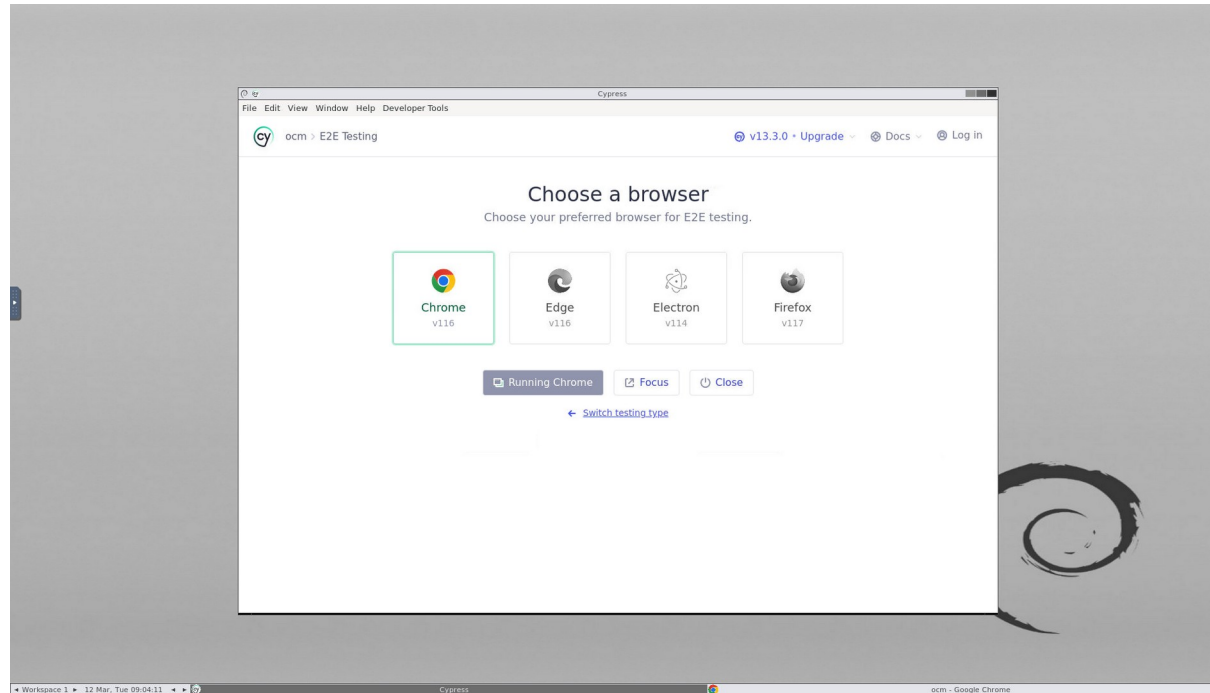
CONTAINER ID	IMAGE	PORTS	NAMES
e8bd919e3998	theasp/novnc:latest	0.0.0.0:5700->8080/tcp	vnc-server
d994e3d12e0d	jlesage/firefox:latest	0.0.0.0:5800->5800/tcp	firefox
bd7becdb9a4e	pondersource/dev-stock-ocmstub		meshdir.docker
678e41e47b27	pondersource/dev-stock-revad	0.0.0.0:4504->80/tcp	revanextcloud2.docker
6e587b71aac4	pondersource/dev-stock-revad	0.0.0.0:4503->80/tcp	revanextcloud1.docker
2efcf64a073d	pondersource/dev-stock-revad	0.0.0.0:4502->80/tcp	revaowncloud2.docker
ee2e21d7c7e2	pondersource/dev-stock-revad	0.0.0.0:4501->80/tcp	revaowncloud1.docker
2b974bded893	pondersource/dev-stock-nextcloud-sciencemesh	443/tcp	nextcloud2.docker
41f767682498	mariadb	3306/tcp	marianextcloud2.docker
1a6183c0db46	pondersource/dev-stock-nextcloud-sciencemesh	443/tcp	nextcloud1.docker
8abd3cde7256	mariadb	3306/tcp	marianextcloud1.docker
950cee52dda1	pondersource/dev-stock-owncloud-ocm-test-suite	443/tcp	owncloud2.docker
5fe33dc1ed96	mariadb	3306/tcp	mariaowncloud2.docker
275ba0482595	pondersource/dev-stock-owncloud-ocm-test-suite	443/tcp	owncloud1.docker
28622d4140c5	mariadb	3306/tcp	mariaowncloud1.docker
131d5fe02492	cs3org/wopiserver:latest	0.0.0.0:8880->8880/tcp	wopi.docker
0e337114aefc	collabora/code:latest	0.0.0.0:9980->9980/tcp	collabora.docker

- EFSS not exposed to host
- Usage via embedded Firefox (in interactive mode) and Cypress

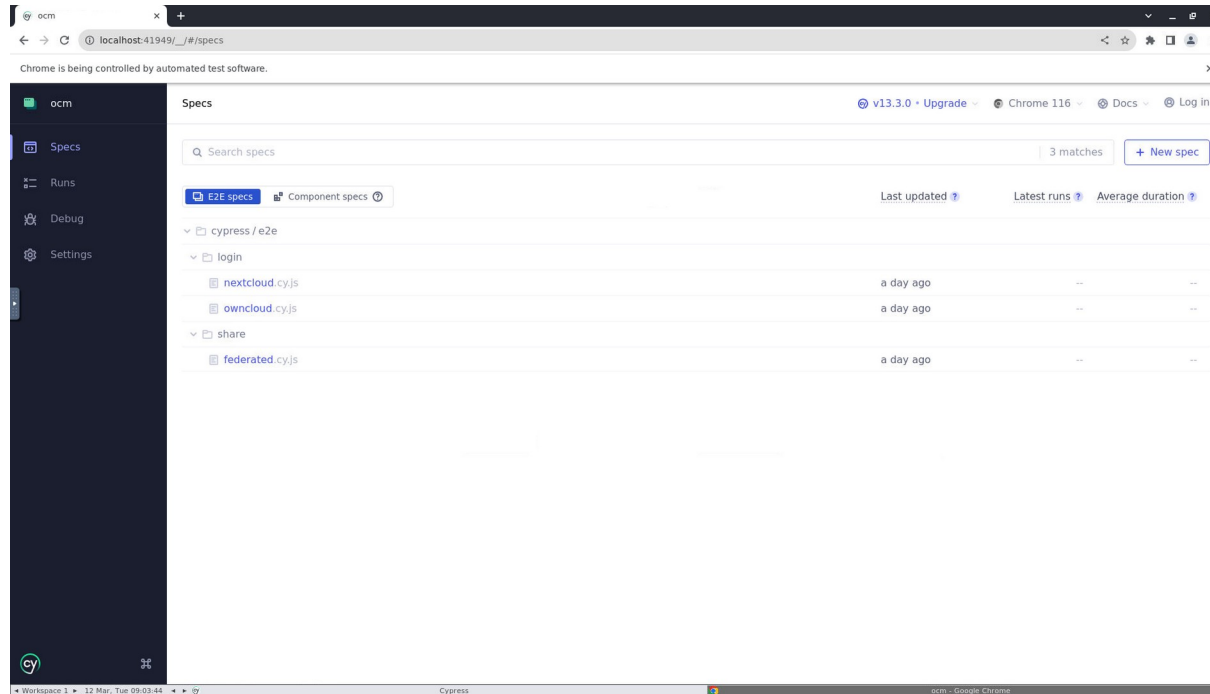
Demo embedded Firefox



Demo „Headed“ Cypress



Demo „Headed“ Cypress



Demo Artifacts

The screenshot shows a GitHub Actions workflow run for the repository 'pondersource / dev-stock'. The workflow is named 'Enhance OCM Test Suite #78' and is in a 'Success' state. It was triggered by a pull request from MahdiBaghbani. The run details show that 4 jobs were completed, including e2e tests for chrome, edge, firefox, and electron. An artifact named 'cypress-videos' was produced during the runtime, with a size of 200 KB.

Summary

Jobs

- e2e (chrome)
- e2e (edge)
- e2e (firefox)
- e2e (electron)

Run details

- Usage
- Workflow file

Triggered via pull request 5 days ago

Author	Status	Total duration	Artifacts
MahdiBaghbani synchronize #88	Success	4m 17s	1

oem-test-suite.yml

on: pull_request

Matrix: e2e

4 jobs completed

Show all jobs

Artifacts

Produced during runtime

Name	Size
cypress-videos	200 KB

<https://github.com/pondersource/dev-stock/actions/runs/8177348239>