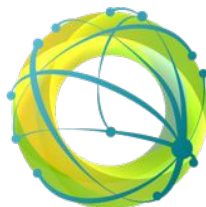




FTS & XRootD Workshop



Jozef Stefan Institute, Ljubljana, 27-31st March 2023



FTS

File Transfer Service

FTS: Cloud Storage Transfers

Mihai Patrascoiu
on behalf of the FTS team

Key Aspects

- Cloud Storages don't support HTTP-TPC
 - must always be TPC passive party
- Cloud Storages don't support dirlist, checksum and mkdir calls
 - FTS must avoid calling these actions
- Pre-signed URLs will not help you
 - no guarantee FTS schedules within time window
 - URLs get posted everywhere (Web Monitoring, ActiveMQ, etc)
- Need to configure FTS with signing secrets (...and more)
 - requires FTS admin intervention

How does it all work?



Walkthrough

Configure S3 Cloud access on FTS3-Pilot:

user: /DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ftssuite/CN=123456/CN=Robot: fts3 testsuite

voms attributes: /dteam/Role=NULL/Capability=NULL

cloud storage: fts-s3test-cloud.cern.ch

FTS instance: fts3-pilot.cern.ch

access key: my_access_key

secret_key: my_secret_key

Walkthrough (cont'd)

I. FTS admin configures FTS-REST Cloud Storage:

1. ~~Via admin webpage /admin/cloud_storage~~ (front-end issues) OR
2. Via POST /admin/cloud_storage --data=@json.config OR
3. Via database

```
> INSERT INTO t_cloudStorage (cloudStorage_name)
      VALUES ("S3:fts-s3test-cloud.cern.ch");

> INSERT INTO t_cloudStorageUser (cloudStorage_name, user_dn, vo_name, access_token,
access_token_secret) VALUES ("S3:fts-s3test-cloud.cern.ch", "/DN..", "/dteam/..",
"my_access_key", "my_secret_key");
```

Walkthrough (cont'd)

II. FTS admin configures S3 Alternate (and S3 Region) fields

- Region field needed for S3v4 signatures
- Alternate puts bucket name in URL instead of subhost
(fts-s3test-cloud.cern.ch/mybucket/ **VS** mybucket.fts-s3test-cloud.cern.ch)
- Step requires configuration of Gfal2 file: /etc/gfal2.d/s3_custom.conf
 - Any file ending in *.conf will work
 - Instructs Gfal2 with StorageEndpoint-specific configuration

```
$ cat <<EOT > s3_custom.conf
[S3:fts-s3test-cloud.cern.ch]
ALTERNATE=true
REGION=myregion
EOT
```

Walkthrough (cont'd)

III. User instructs FTS which part is active or not during submission:

- Done by appending “?copy_mode=<push|pull>” to one of the URLs
- FTS/Gfal2 reads the `copy_mode` opaque flag

```
$ fts-rest-transfer-submit -s https://fts3-pilot.cern.ch:8446/  
https://eospublic.cern.ch/eos/opstest/dteam/file.test  
s3s://fts-s3test-cloud.cern.ch/mybucket/file.test?copy_mode=push
```

Walkthrough (cont'd)

IV. User instructs FTS to skip all other validation steps (PROPFIND, checksum, etc)

- Done by adding the `--strict-copy` / `{ "strict_copy": true }` flag
- FTS/Gfal2 will now execute only the `HTTP COPY` command

```
$ fts-rest-transfer-submit --strict-copy -s https://fts3-pilot.cern.ch:8446/  
https://eospublic.cern.ch/eos/opstest/dteam/file.test  
s3s://fts-s3test-cloud.cern.ch/mybucket/file.test?copy_mode=push
```


Debugging

- FTS & S3 Cloud Storage almost never works on the first try

Does the server use S3v2 or S3v4 signature?

Does the server use alternate mode or not?

Are the access/secret pairs even correct?

Debugging (cont'd)

- Before attempting any FTS configuration, must verify all works via **Davix**

```
$ davix-http -X GET --trace header,s3,sensitive [--s3alternate] [--s3region]
    --s3accesskey my_access_key
    --s3secretkey my_secret_key
    s3s://fts-s3test-cloud.cern.ch/mybucket/test.file
```

- And via **Gfal2**

```
$ gfal-copy -vvv --log-file=gfal2.log --just-copy
    [-D"S3:FTS-S3TEST-CLOUD.CERN.CH:ALTERNATE=TRUE"]
    [-D"S3:FTS-S3TEST-CLOUD.CERN.CH:REGION=myregion"]
    -D"S3:FTS-S3TEST-CLOUD.CERN.CH:ACCESS_KEY=my_access_key"
    -D"S3:FTS-S3TEST-CLOUD.CERN.CH:SECRET_KEY=my_secret_key"
    s3s://fts-s3test-cloud.cern.ch/mybucket/test.file /tmp/test.file
```

The redesigned way

I. FTS admin or privileged user configures FTS-REST Cloud Storage:

1. Via admin webpage `/admin/cloud_storage` OR
2. Via POST `/admin/cloud_storage --data=@json.config`

Add new cloud storage: Cloud Storage type

Storage Name	Region	
<input type="text" value="fts-s3test-cloud.cern.ch"/>	<input type="text" value="myregion"/>	<input checked="" type="checkbox"/> Alternate

The redesigned way (cont'd)

I. FTS admin or privileged user configures FTS-REST Cloud Storage:

1. Via admin webpage `/admin/cloud_storage` OR
2. Via POST `/admin/cloud_storage --data=@json.config`

fts-s3test-cloud.cern.chAlternate: [x] Region: myregion

User	VO roles	Access key	Secret key
/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ftssuite/CN=123456/CN=Robot: fts3 testsuite	/dteam/Role=NULL/Capability=NULL	<input type="text" value="my_access_key"/>	<input type="text" value="my_secret_key"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

The redesigned way (cont'd)

- All configuration done in just one place
- Intuitive configuration webpage
- FTS will identify S3 transfers and designate it automatically as the passive party
- [Not decided yet]: *Allow strict copy to be configured via the webpage*

- Similarly, GCloud and Swift Cloud configuration is improved
- Thanks to Eraldo for his work on this area!

Thank you!