Dirac – Rucio update

Janusz Martyniak, IC

Adapting Belle II Dirac-Rucio integration solution for a Multi-VO environment

- The software consists of 3 files:

1. RucioFileCatalog plugin – this is the bit which is automatically used by Dirac for VOs willing to use Rucio FT en lieu of the DFC

2. Dirac SE-> RSE sync agent – reads/updates RSE and user database from Dirac CS

3. RSS sync agent - sets RSE status based on SE status. Requires Dirac v7r3 or later.

- All files originally contained many BELLE II specific parts. Care has been taken to retain backward compatibility.

Agents

- Eliminate a need of a client config file,
- Use X509 creds, specify VO:

```
client = Client(account=self.clientConfig[vo]['privilegedAccount'],     <- VO root
                rucio_host=self.clientConfig[vo]['rucioHost'],
                auth_host=self.clientConfig[vo]['authHost'],
                ca_cert=self.caCertPath,
                auth_type='x509',
                creds={'client_cert': hostcert, 'client_key': hostkey},
                user_agent='rucio-clients', vo=vo)
```

All parameters above are taken from Dirac env (e.g. CA cert path) or from the CS

Both agents are deployed on our test Dirac instance.

RucioFileCatalog Client plugin

- Belle developed a dirac specific Rucio client which delegates file registration to a server in a *transactional* way.

- This client has been deployed for adding files only – this is the only case where Dirac upload a file and Rucio registers it.

- Belle use their own *scope resolution algorithm* which is embedded in Rucio code along with the default one.

- The algorithm name is defined in a config file (on the server and the client) in a form key:value which does not scale for multiple VOs if they wish to use different algorithms.

- Dirac scope algorithm has been added to Rucio which puts the scope name as a second element of the LFN.

RucioFileCatalogClient plugin –contd.

Server adjustments:

- Enable long VO name resolution (aka Simon's fix)
- Set the scope algorithm to Dirac: *extract_scope = dirac* ([common])
- Set a variable: *SCOPE_NAME_REGEX = '/([^/]*)(/.*)'*
- Create a top container for the VO (i.e. /gridpp)

On Dirac:

Deploy SE→RSE agent – it will also create a user database with their scopes.

Basic tests:

Upload:

dirac-dms-add-file -ddd LFN:/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile test_200Mfile UKI-LT2-IC-HEP-disk

dirac_dms_add_file/DIRAC.Resources.Catalog.RucioFileCatalogClient [140124828706624] DEBUG: {'Successful': {'/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile': True}, 'Failed': {}}

Is it there?

gfal-ls -l srm://gfe02.grid.hep.ph.ic.ac.uk:8443/pnfs/hep.ph.ic.ac.uk/data/gridpp/gridpp/user.janusz.martyniak/dirac_test

-rw-r--r--   1 47   48   209715200 Dec 16 16:07 test_200Mfile

Replicas:

dirac-dms-lfn-replicas /gridpp/user.janusz.martyniak/dirac_test/test_200Mfile

LFN   StorageElement   URL

===

/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile UKI-LT2-IC-HEP-disk srm://gfe02.grid.hep.ph.ic.ac.uk:8443/srm/managerv2?SFN=/pnfs/hep.ph.ic.ac.uk/data/gridpp/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile

Basic tests: Containers and Datasets.

```
(rucio) -bash-4.2$ rucio list-dids user.janusz.martyniak:*

+-----------------------------+--------------+
| SCOPE:NAME                  | [DID TYPE]   |
|-----------------------------+--------------|
| user.janusz.martyniak:/gridpp | CONTAINER    |
+-----------------------------+--------------+
```

After uploading a file:

```
(rucio_ral) -bash-4.2$ rucio list-dids user.janusz.martyniak:*

+-----------------------------------------------------------+--------------+
| SCOPE:NAME                                                | [DID TYPE]   |
|-----------------------------------------------------------+--------------|
| user.janusz.martyniak:/gridpp                            | CONTAINER    |
| user.janusz.martyniak:/gridpp/user.janusz.martyniak      | CONTAINER    |
| user.janusz.martyniak:/gridpp/user.janusz.martyniak/dirac_test | DATASET |
+-----------------------------------------------------------+——————+
```

Other commands:

Download:

dirac-dms-get-file /gridpp/user.janusz.martyniak/dirac_test/test_200Mfile

{'Failed': {},

 'Successful': {'/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile': '/home/hep/martynia/Dirac/test_200Mfile'}}

Access URL:

dirac-dms-lfn-accessURL /gridpp/user.janusz.martyniak/dirac_test/test_200Mfile UKI-LT2-IC-HEP-disk

{'Failed': {},

 'Successful': {'/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile':
'srm://gfe02.grid.hep.ph.ic.ac.uk:8443/srm/managerv2?SFN=/pnfs/hep.ph.ic.ac.uk/data/gridpp/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile'}}

Metadata:

dirac-dms-lfn-metadata /gridpp/user.janusz.martyniak/dirac_test/test_200Mfile

{'Failed': {},

 'Successful': {'/gridpp/user.janusz.martyniak/dirac_test/test_200Mfile': {'Checksum': 'bb800001',

'ChecksumType': 'AD',

'CreationDate': datetime.datetime(2021, 12, 16, 16, 7, 10),

'GUID': 'f1c9645d-bc14-efdd-c7d8-a322685f26eb',

'Mode': 436,

'ModificationDate': datetime.datetime(2021, 12, 16, 16, 7, 10),

'NumberOfLinks': 1,

'Size': 209715200,

'Status': '-'}}}

Remove? Need admin permissions

(rucio_ral) -bash-4.2$ rucio-admin account add-attribute --key admin --value True janusz.martyniak

(rucio_ral) -bash-4.2$ rucio-admin account list-attributes janusz.martyniak

```
+-------+---------+
| Key   | Value   |
|-------+---------|
| admin | True    |
+-------+————+
```

Now remove a sole replica:

-bash-4.2$ dirac-dms-remove-files /gridpp/user.janusz.martyniak/dirac_test/test_200Mfile

Successfully removed 1 files

Conclusions

- Successfully adapted BELLE2 Dirac-Rucio FC implementation

- Accepted by Rucio (merged in Feb 2022, in rucio 1.29 LTS)

- Requires Dirac v7r3 or later.

- Tested at IC with our Rucio server.