invenio-s3 Documentation

Release 1.0.6

Esteban J. G. Gabancho

CONTENTS

S3 file storage support for Invenio.

The package offers integration with any S3 REST API compatible object storage.

Further documentation is available on https://invenio-s3.readthedocs.io/

USER'S GUIDE

This part of the documentation will show you how to get started in using Invenio-S3.

1.1 Installation

Invenio-S3 is on PyPI so all you need is:

```
$ pip install invenio-s3
```

1.2 Configuration

S3 file storage support for Invenio.

invenio_s3.config.**S3_ACCESS_KEY_ID = None**
    The access key to use when creating the client.

    This is entirely optional, and if not provided, the credentials configured for the session will automatically be used. See Configuring Credentials. for more information.

invenio_s3.config.**S3_DEFAULT_BLOCK_SIZE = 5242880**
    Default block size value used to send multi-part uploads to S3. Typically 5Mb is minimum allowed by the API.

invenio_s3.config.**S3_ENDPOINT_URL = None**
    S3 server URL endpoint.

    If using Amazon AWS S3 service this config variable can be set to None as the underlining library, boto3, will automatically construct the appropriate URL to use when communicating with a service.

    If set to a value (including the "http/https" scheme) it will be passed as endpoint_url to boto3 client.

invenio_s3.config.**S3_MAXIMUM_NUMBER_OF_PARTS = 10000**
    Maximum number of parts to be used. See AWS Multipart Upload Overview for more information.

invenio_s3.config.**S3_REGION_NAME = None**
    S3 region name

    This is entirely optional, and if not provided, the region name will be automatically set to 'us-east-1'.

    If set to a value it will be passed as region_name to boto3 client.

invenio_s3.config.**S3_SECRET_ACCESS_KEY = None**
    The secret key to use when creating the client.

    This is entirely optional, and if not provided, the credentials configured for the session will automatically be used. See Configuring Credentials. for more information.

invenio_s3.config.**S3_SIGNATURE_VERSION = 's3v4'**
>   Version of the S3 signature algorithm. Can be 's3' (v2) or 's3v4' (v4). See Amazon Boto3 documentation on configuration variables for more information.

invenio_s3.config.**S3_URL_EXPIRATION = 60**
>   Number of seconds the file serving URL will be valid.

>   See Amazon Boto3 documentation on presigned URLs for more information.

1.3 Usage

S3 file storage support for Invenio.

To use this module together with Invenio-Files-Rest there are a few things you need to keep in mind.

The storage factory configuration variable, `FILES_REST_STORAGE_FACTORY` needs to be set to `'invenio_s3.s3fs_storage_factory'` importable string.

We think the best way to use this module is to have one Localtion for each S3 bucket. This is just for simplicity, it can used however needed.

When creating a new location which will use the S3 API, the URI needs to start with `s3://`, for example `invenio files location s3_default s3://my-bucket --default` will create a new location, set it as default location for your instance and use the bucket `my-bucket`. For more information about this command check Invenio-Files-Rest documentation.

Then, there are a few configuration variables that need to be set on your instance, like the endpoint, the access key and the secret access key, see a more detailed description in *Configuration*.

Note: This module doesn't create S3 buckets automatically, so before starting they need to be created.

You might also want to set the correct CORS configuration so files can be used by your interface for things like previewing a PDF with some Javascript library.

API REFERENCE

If you are looking for information on a specific function, class or method, this part of the documentation is for you.

2.1 API Docs

S3 file storage support for Invenio.

class `invenio_s3.ext.`**`InvenioS3`**(*app=None*)
Invenio-S3 extension.

Extension initialization.

`init_app`(*app*)
Flask application initialization.

`init_config`(*app*)
Initialize configuration.

property `init_s3fs_info`
Gather all the information needed to start the S3FSFileSystem.

S3 file storage interface.

class `invenio_s3.storage.`**`S3FSFileStorage`**(*fileurl*, ***kwargs*)
File system storage using Amazon S3 API for accessing files.

Storage initialization.

`copy`(*src*, **args*, ***kwargs*)
Copy data from another file instance.

If the source is an S3 stored object the copy process happens on the S3 server side, otherwise we use the normal `FileStorage` copy method.

`delete`()
Delete a file.

`initialize`(*size=0*)
Initialize file on storage and truncate to given size.

`save`(**args*, ***kwargs*)
Save incoming stream to storage.

Just overwrite parent method to allow set the correct block size.

`send_file`(*filename*, *mimetype=None*, *restricted=True*, *checksum=None*, *trusted=False*, *chunk_size=None*, *as_attachment=False*)
Send the file to the client.

    update(*incoming_stream*, *seek=0*, *size=None*, *chunk_size=None*, *progress_callback=None*)

        Update a file in the file system.

invenio_s3.storage.**s3fs_storage_factory**(***kwargs*)

    File storage factory for S3.

invenio_s3.storage.**set_blocksize**(*f*)

    Decorator to set the correct block size according to file size.

ADDITIONAL NOTES

Notes on how to contribute, legal information and changes are here for the interested.

3.1 Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

3.1.1 Types of Contributions

Report Bugs

Report bugs at https://github.com/inveniosoftware/invenio-s3/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with "bug" is open to whoever wants to implement it.

Implement Features

Look through the GitHub issues for features. Anything tagged with "feature" is open to whoever wants to implement it.

Write Documentation

Invenio-S3 could always use more documentation, whether as part of the official Invenio-S3 docs, in docstrings, or even on the web in blog posts, articles, and such.

Submit Feedback

The best way to send feedback is to file an issue at https://github.com/inveniosoftware/invenio-s3/issues.

If you are proposing a feature:

- Explain in detail how it would work.

- Keep the scope as narrow as possible, to make it easier to implement.

- Remember that this is a volunteer-driven project, and that contributions are welcome :)

3.1.2 Get Started!

Ready to contribute? Here's how to set up *invenio-s3* for local development.

1. Fork the *inveniosoftware/invenio-s3* repo on GitHub.

2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/invenio-s3.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv invenio-s3
$ cd invenio-s3/
$ pip install -e .[all]
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass tests:

```
$ ./run-tests.sh
```

The tests will provide you with test coverage and also check PEP8 (code style), PEP257 (documentation), flake8 as well as build the Sphinx documentation and run doctests.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -s
    -m "component: title without verbs"
    -m "* NEW Adds your new feature."
    -m "* FIX Fixes an existing issue."
    -m "* BETTER Improves and existing feature."
    -m "* Changes something that should not be visible in release notes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

3.1.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests and must not decrease test coverage.

2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring.

3. The pull request should work for Python 3.6, 3.7 and 3.8. Check https://github.com/inveniosoftware/invenio-s3/actions?query=event%3Apull_request and make sure that the tests pass for all supported Python versions.

3.2 Changes

Version 1.0.6 (released 2021-10-21)

- Unpin boto3 and s3fs

Version 1.0.5 (released 2021-10-20)

- Bump versions to support Flask v2.

Version 1.0.4 (released 2021-07-30)

- Fix number of parts calculations.
- Removed Python 2.7 support.

Version 1.0.3 (released 2020-04-25)

- Allow for dynamic part size for multipart uploads.
- Adds new configuration variables to define default part size and maximum number of parts.

Version 1.0.2 (released 2020-02-17)

- Fixes typos on configuration variables and cached properties.
- Adds AWS region name and signature version to configuration.

Version 1.0.1 (released 2019-01-23)

- New configuration variable for URL expiration.
- Enhances file serving.
- Unpins Boto3 library.
- Fixes test suit configuration.

Version 1.0.0 (released 2018-09-19)

- Initial public release.

3.3 License

MIT License

Copyright (C) 2018 Esteban J. G. Gabancho.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

3.4 Authors

S3 file storage support for Invenio.

- Esteban J. G. Gabancho <egabancho@gmail.com>

PYTHON MODULE INDEX

i