ge25519

Release 1.5.1

Nth Party, Ltd.

Aug 26, 2023

CONTENTS

1 Purpose **3**

2 Installation and Usage **5**

3 Development **7**
- 3.1 Documentation . . . 7
- 3.2 Testing and Conventions . . . 7
- 3.3 Contributions . . . 8
- 3.4 Versioning . . . 8
- 3.5 Publishing . . . 8
  - 3.5.1 ge25519 module . . . 8

Python Module Index **11**

Index **13**

Pure-Python data structure for working with Ed25519 (and Ristretto) group elements and operations.

CHAPTER

ONE

PURPOSE

This library provides a native Python implementation of Ed25519 group elements and a number of operations over them. The library makes it possible to fill gaps in application prototypes that may have specific limitations with respect to their operating environment or their ability to rely on non-Python dependencies.

The implementation is based upon and is compatible with the corresponding implementation of Ed25519 and Ristretto group elements used in libsodium. For more information and background about the underlying mathematical structures and primitives, consult materials about Curve25519, the Ristretto group, and the related Ed25519 system.

CHAPTER

TWO

INSTALLATION AND USAGE

This library is available as a package on PyPI:

```
python -m pip install ge25519
```

The library can be imported in the usual ways:

```
import ge25519
from ge25519 import *
```

CHAPTER

THREE

DEVELOPMENT

All installation and development dependencies are fully specified in `pyproject.toml`. The `project.optional-dependencies` object is used to specify optional requirements for various development tasks. This makes it possible to specify additional options (such as `docs`, `lint`, and so on) when performing installation using pip:

```
python -m pip install .[docs,lint]
```

3.1 Documentation

The documentation can be generated automatically from the source files using Sphinx:

```
python -m pip install .[docs]
cd docs
sphinx-apidoc -f -E --templatedir=_templates -o _source .. && make html
```

3.2 Testing and Conventions

All unit tests are executed and their coverage is measured when using pytest (see the `pyproject.toml` file for configuration details):

```
python -m pip install .[test]
python -m pytest
```

Concise unit tests are implemented with the help of fountains; new reference specifications for these tests can be generated by running the testing module directly:

```
python test/test_ge25519.py
```

Style conventions are enforced using Pylint:

```
python -m pip install .[lint]
python -m pylint src/ge25519 test/test_ge25519.py
```

3.3 Contributions

In order to contribute to the source code, open an issue or submit a pull request on the GitHub page for this library.

3.4 Versioning

Beginning with version 0.1.0, the version number format for this library and the changes to the library associated with version number increments conform with Semantic Versioning 2.0.0.

3.5 Publishing

This library can be published as a package on PyPI by a package maintainer. First, install the dependencies required for packaging and publishing:

```
python -m pip install .[publish]
```

Ensure that the correct version number appears in `pyproject.toml`, and that any links in this README document to the Read the Docs documentation of this package (or its dependencies) have appropriate version numbers. Also ensure that the Read the Docs project for this library has an automation rule that activates and sets as the default all tagged versions. Create and push a tag for this version (replacing `?.?.?` with the version number):

```
git tag ?.?.?
git push origin ?.?.?
```

Remove any old build/distribution files. Then, package the source into a distribution archive:

```
rm -rf build dist src/*.egg-info
python -m build --sdist --wheel .
```

Finally, upload the package distribution archive to PyPI:

```
python -m twine upload dist/*
```

3.5.1 ge25519 module

Pure-Python data structure for working with Ed25519 (and Ristretto) group elements and operations.

class ge25519.ge25519.**ge25519**

Bases: object

Base class for group elements representing elliptic curve points.

The public interface of this class and those of derived classes are defined primarily to support the representation of elliptic curve points and the implementation of common operations over those points (*e.g.*, as in the oblivious library).

static is_canonical(*s: bytes*) → int

Determine whether a binary representation of an element is in canonical form.

static has_small_order(*s: bytes*) → int

class ge25519.ge25519.**ge25519_p2**(*X: fe25519.fe25519.fe25519*, *Y: fe25519.fe25519.fe25519*, *Z: fe25519.fe25519.fe25519*)

Bases: *ge25519.ge25519.ge25519*

Specialized class for group elements representing elliptic curve points.

static from_p3(*p:* ge25519.ge25519.ge25519_p3) → *ge25519.ge25519.ge25519_p2*

static from_p1p1(*p:* ge25519.ge25519.ge25519_p1p1) → *ge25519.ge25519.ge25519_p2*

dbl() → *ge25519.ge25519.ge25519_p1p1*

class ge25519.ge25519.**ge25519_p3**(*X: fe25519.fe25519.fe25519 = None*, *Y: fe25519.fe25519.fe25519 = None*, *Z: fe25519.fe25519.fe25519 = None*, *T: fe25519.fe25519.fe25519 = None*, *root_check: bool = None*)

Bases: *ge25519.ge25519.ge25519*

Specialized class for group elements representing elliptic curve points.

static zero() → *ge25519.ge25519.ge25519_p3*
Constant corresponding to the zero element.

static from_bytes(*bs: bytes*) → *ge25519.ge25519.ge25519_p3*
Construct an element from its binary representation.

static from_bytes_ristretto255(*bs: bytes*) → *ge25519.ge25519.ge25519_p3*
Construct a Ristretto point from its binary representation.

static from_hash_ristretto255(*h: bytes*) → bytes
Construct a Ristretto point from a hash value.

static from_uniform(*r: bytes*) → *ge25519.ge25519.ge25519_p3*

static from_p1p1(*p:* ge25519.ge25519.ge25519_p1p1) → *ge25519.ge25519.ge25519_p3*

is_on_curve() → int

is_on_main_subgroup() → int

dbl() → *ge25519.ge25519.ge25519_p1p1*

mul_l() → *ge25519.ge25519.ge25519_p3*

static scalar_mult_base(*a: bytes*) → *ge25519.ge25519.ge25519_p3*

scalar_mult(*a: bytes*) → *ge25519.ge25519.ge25519_p3*
Method that supports the implementation of a scalar multiplication operation for elliptic curve points.

static elligator_ristretto255(*t: fe25519.fe25519.fe25519*) → *ge25519.ge25519.ge25519_p3*

static elligator2(*r: fe25519.fe25519.fe25519*, *x_sign: int*) → *ge25519.ge25519.ge25519_p3*

to_bytes() → bytes
Emit binary representation of this element.

to_bytes_ristretto255() → bytes
Emit binary representation of the Ristretto point that this element represents.

class ge25519.ge25519.**ge25519_p1p1**(*X: fe25519.fe25519.fe25519 = None*, *Y: fe25519.fe25519.fe25519 = None*, *Z: fe25519.fe25519.fe25519 = None*, *T: fe25519.fe25519.fe25519 = None*)

Bases: *ge25519.ge25519.ge25519*

Specialized class for group elements representing elliptic curve points.

static dbl(*p:* ge25519.ge25519.ge25519_p3) → *ge25519.ge25519.ge25519_p1p1*

static madd(*p:* ge25519.ge25519.ge25519_p3, *q:* ge25519.ge25519.ge25519_precomp) → *ge25519.ge25519.ge25519_p1p1*

Method that supports scalar multiplication of a base element.

static add(*p:* ge25519.ge25519.ge25519_p3, *q:* ge25519.ge25519.ge25519_cached) → *ge25519.ge25519.ge25519_p1p1*

Method that supports the implementation of an addition operation for elliptic curve points.

static sub(*p:* ge25519.ge25519.ge25519_p3, *q:* ge25519.ge25519.ge25519_cached) → *ge25519.ge25519.ge25519_p1p1*

Method that supports the implementation of a subtraction operation for elliptic curve points.

class ge25519.ge25519.**ge25519_precomp**(*yplusx: fe25519.fe25519.fe25519 = None*, *yminusx: fe25519.fe25519.fe25519 = None*, *xy2d: fe25519.fe25519.fe25519 = None*)

Bases: *ge25519.ge25519.ge25519*

Specialized class for group elements corresponding to entries found in the table of precomputed points.

static zero() → *ge25519.ge25519.ge25519_precomp*

Constant corresponding to the zero element.

class ge25519.ge25519.**ge25519_cached**(*YplusX: fe25519.fe25519.fe25519 = None*, *YminusX: fe25519.fe25519.fe25519 = None*, *Z: fe25519.fe25519.fe25519 = None*, *T2d: fe25519.fe25519.fe25519 = None*)

Bases: *ge25519.ge25519.ge25519*

Specialized class for group elements representing elliptic curve points.

static zero() → *ge25519.ge25519.ge25519_cached*

Constant corresponding to the zero element.

static from_p3(*p:* ge25519.ge25519.ge25519_p3) → *ge25519.ge25519.ge25519_cached*

PYTHON MODULE INDEX

g
ge25519.ge25519, 8

INDEX

A

add() (*ge25519.ge25519.ge25519_p1p1 static method*), 10

D

dbl() (*ge25519.ge25519.ge25519_p1p1 static method*), 9
dbl() (*ge25519.ge25519.ge25519_p2 method*), 9
dbl() (*ge25519.ge25519.ge25519_p3 method*), 9

E

elligator2() (*ge25519.ge25519.ge25519_p3 static method*), 9
elligator_ristretto255() (*ge25519.ge25519.ge25519_p3 static method*), 9

F

from_bytes() (*ge25519.ge25519.ge25519_p3 static method*), 9
from_bytes_ristretto255() (*ge25519.ge25519.ge25519_p3 static method*), 9
from_hash_ristretto255() (*ge25519.ge25519.ge25519_p3 static method*), 9
from_p1p1() (*ge25519.ge25519.ge25519_p2 static method*), 9
from_p1p1() (*ge25519.ge25519.ge25519_p3 static method*), 9
from_p3() (*ge25519.ge25519.ge25519_cached static method*), 10
from_p3() (*ge25519.ge25519.ge25519_p2 static method*), 9
from_uniform() (*ge25519.ge25519.ge25519_p3 static method*), 9

G

ge25519 (*class in ge25519.ge25519*), 8
ge25519.ge25519
    module, 8
ge25519_cached (*class in ge25519.ge25519*), 10
ge25519_p1p1 (*class in ge25519.ge25519*), 9
ge25519_p2 (*class in ge25519.ge25519*), 8
ge25519_p3 (*class in ge25519.ge25519*), 9
ge25519_precomp (*class in ge25519.ge25519*), 10

H

has_small_order() (*ge25519.ge25519.ge25519 static method*), 8

I

is_canonical() (*ge25519.ge25519.ge25519 static method*), 8
is_on_curve() (*ge25519.ge25519.ge25519_p3 method*), 9
is_on_main_subgroup() (*ge25519.ge25519.ge25519_p3 method*), 9

M

madd() (*ge25519.ge25519.ge25519_p1p1 static method*), 9
module
    ge25519.ge25519, 8
mul_l() (*ge25519.ge25519.ge25519_p3 method*), 9

S

scalar_mult() (*ge25519.ge25519.ge25519_p3 method*), 9
scalar_mult_base() (*ge25519.ge25519.ge25519_p3 static method*), 9
sub() (*ge25519.ge25519.ge25519_p1p1 static method*), 10

T

to_bytes() (*ge25519.ge25519.ge25519_p3 method*), 9
to_bytes_ristretto255() (*ge25519.ge25519.ge25519_p3 method*), 9

Z

zero() (*ge25519.ge25519.ge25519_cached static method*), 10

zero() (*ge25519.ge25519.ge25519_p3 static method*), 9
zero() (*ge25519.ge25519.ge25519_precomp static method*), 10