MythX Domain Model Documentation

Release 2.2.0

Dominik Muhs

May 12, 2022

Contents:

1 The MythX Domain Models **1**
- 1.1 What is MythX? . . . 1
- 1.2 Installation . . . 1

2 Advanced Installation **3**
- 2.1 Stable release . . . 3
- 2.2 From sources . . . 3

3 mythx-models **5**
- 3.1 mythx-models package . . . 5

4 Contributing **13**
- 4.1 Types of Contributions . . . 13
- 4.2 Get Started! . . . 14
- 4.3 Pull Request Guidelines . . . 15
- 4.4 Deploying . . . 15

5 Credits **17**
- 5.1 Development Lead . . . 17

6 Indices and tables **19**

Python Module Index **21**

Index **23**

CHAPTER 1

The MythX Domain Models

This repository contains the domain models for the MythX smart contract security analysis platform.

Table of Contents

- *The MythX Domain Models*
  - *What is MythX?*
  - *Installation*

1.1 What is MythX?

MythX is a security analysis API that allows anyone to create purpose-built security tools for smart contract developers. Tools built on MythX integrate seamlessly into the development environments and continuous integration pipelines used throughout the Ethereum ecosystem.

1.2 Installation

The MythX domain models runs on Python 3.5+.

To get started, simply run

```
$ pip3 install mythx-models
```

Alternatively, clone the repository and run

```
$ pip3 install .
```

Or directly through Python's `setuptools`:

```
$ python3 setup.py install
```

CHAPTER 2

Advanced Installation

2.1 Stable release

To install the MythX domain models, run this command in your terminal:

```
$ pip3 install mythx-models
```

This is the preferred method to install the MythX domain models, as it will always install the most recent stable release.

If you don't have pip installed, this Python installation guide can guide you through the process.

2.2 From sources

The sources for the MythX domain models can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/dmuhs/mythx-models
```

Or download the tarball:

```
$ curl  -OL https://github.com/dmuhs/mythx-models/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```

CHAPTER 3

mythx-models

3.1 mythx-models package

3.1.1 Subpackages

mythx_models.request package

Submodules

mythx_models.request.analysis_list module

This module contains the AnalysisListRequest domain model.

class mythx_models.request.analysis_list.**AnalysisListRequest**

Bases: pydantic.main.BaseModel

class Config

Bases: object

allow_population_by_field_name = True

use_enum_values = True

endpoint

headers

method

parameters

payload

mythx_models.request.analysis_status module

This module contains the AnalysisStatusRequest domain model.

class mythx_models.request.analysis_status.**AnalysisStatusRequest**

Bases: pydantic.main.BaseModel

endpoint

headers

method

parameters

payload

mythx_models.request.analysis_submission module

This module contains the AnalysisSubmissionRequest domain model.

class mythx_models.request.analysis_submission.**AnalysisSubmissionRequest**

Bases: pydantic.main.BaseModel

class Config

Bases: object

allow_population_by_field_name = True

use_enum_values = True

endpoint

headers

method

parameters

payload

mythx_models.request.auth_login module

This module contains the AuthLoginRequest domain model.

class mythx_models.request.auth_login.**AuthLoginRequest**

Bases: pydantic.main.BaseModel

endpoint

headers

method

parameters

payload

mythx_models.request.auth_logout module

This module contains the AuthLogoutRequest domain model.

class mythx_models.request.auth_logout.**AuthLogoutRequest**

Bases: pydantic.main.BaseModel

endpoint

headers

method

parameters

payload

mythx_models.request.auth_refresh module

This module contains the AuthRefreshRequest domain model.

class mythx_models.request.auth_refresh.**AuthRefreshRequest**

Bases: pydantic.main.BaseModel

class Config

Bases: object

allow_population_by_field_name = True

use_enum_values = True

endpoint

headers

method

parameters

payload

mythx_models.request.base module

mythx_models.request.detected_issues module

This module contains the DetectedIssuesRequest domain model.

class mythx_models.request.detected_issues.**DetectedIssuesRequest**

Bases: pydantic.main.BaseModel

endpoint

headers

method

parameters

payload

mythx_models.request.oas module

mythx_models.request.version module

This module contains the VersionRequest domain model.

class mythx_models.request.version.**VersionRequest**

Bases: pydantic.main.BaseModel

endpoint

headers

method

parameters

payload

Module contents

This module contains the MythX request domain models.

mythx_models.response package

Submodules

mythx_models.response.analysis module

This module contains domain models regarding analysis jobs.

class mythx_models.response.analysis.**Analysis**

Bases: pydantic.main.BaseModel

class Config

Bases: object

allow_population_by_field_name = True

use_enum_values = True

class mythx_models.response.analysis.**AnalysisShort**

Bases: pydantic.main.BaseModel

class Config

Bases: object

allow_population_by_field_name = True

use_enum_values = True

class mythx_models.response.analysis.**AnalysisStatus**

Bases: str, enum.Enum

An Enum describing the status an analysis job can be in.

ERROR = 'Error'

FINISHED = 'Finished'

IN_PROGRESS = 'In Progress'

QUEUED = 'Queued'

mythx_models.response.analysis_list module

This module contains the AnalysisListResponse domain model.

class mythx_models.response.analysis_list.**AnalysisListResponse**
Bases: pydantic.main.BaseModel

mythx_models.response.analysis_status module

This module contains the AnalysisStatusResponse domain model.

class mythx_models.response.analysis_status.**AnalysisStatusResponse**
Bases: *mythx_models.response.analysis.AnalysisShort*

The API response domain model for the status of a single analysis.

mythx_models.response.analysis_submission module

This module contains the AnalysisSubmissionResponse domain model.

class mythx_models.response.analysis_submission.**AnalysisSubmissionResponse**
Bases: *mythx_models.response.analysis.AnalysisShort*

mythx_models.response.auth_login module

This module contains the AuthLoginResponse domain model.

class mythx_models.response.auth_login.**AuthLoginResponse**
Bases: pydantic.main.BaseModel

class Config
Bases: object

allow_population_by_field_name = True

use_enum_values = True

mythx_models.response.auth_logout module

This module contains the AuthLogoutResponse domain model.

class mythx_models.response.auth_logout.**AuthLogoutResponse**
Bases: pydantic.main.BaseModel

mythx_models.response.auth_refresh module

This module contains the AuthRefreshResponse domain model.

class mythx_models.response.auth_refresh.**AuthRefreshResponse**
Bases: pydantic.main.BaseModel

class Config
Bases: object

allow_population_by_field_name = True

use_enum_values = True

mythx_models.response.base module

mythx_models.response.detected_issues module

This module contains the response models for the detected issues endpoint and a report helper.

class mythx_models.response.detected_issues.**DetectedIssuesResponse**
Bases: pydantic.main.BaseModel

class mythx_models.response.detected_issues.**IssueReport**
Bases: pydantic.main.BaseModel

mythx_models.response.issue module

This module contains domain models regrading found issues.

class mythx_models.response.issue.**Issue**
Bases: pydantic.main.BaseModel

class Config
Bases: object

allow_population_by_field_name = True

use_enum_values = True

class mythx_models.response.issue.**IssueDescription**
Bases: pydantic.main.BaseModel

class mythx_models.response.issue.**LineLocation**
Bases: pydantic.main.BaseModel

class mythx_models.response.issue.**SEVERITY**
Bases: str, enum.Enum

An Enum holding the possible severities an issue can have.

HIGH = 'High'

LOW = 'Low'

MEDIUM = 'Medium'

NONE = 'None'

UNKNOWN = 'Unknown'

class mythx_models.response.issue.**SourceLocation**
Bases: pydantic.main.BaseModel

class Config
Bases: object

allow_population_by_field_name = True

class mythx_models.response.issue.**SourceMap**
Bases: pydantic.main.BaseModel

components

static decompress(*source_map: str*) → List[mythx_models.response.issue.SourceMapLocation]

static sourcemap_reducer(*accumulator: Tuple[int, int, int, str], component: str*) → List[str]

class mythx_models.response.issue.**SourceMapLocation**
Bases: pydantic.main.BaseModel

mythx_models.response.oas module

mythx_models.response.version module

This module contains the VersionResponse domain model.

class mythx_models.response.version.**VersionResponse**
Bases: pydantic.main.BaseModel

Module contents

This module contains the MythX response domain models.

mythx_models.exceptions package

Module contents

This module contains exceptions raised by MythX.

exception mythx_models.exceptions.**MythXAPIError**
Bases: Exception

An exception denoting an API-related error.

This is usually raised when the API takes too long to respond, or a response contains an HTTP status code that is not 200 OK. In this case, the exception is passed on to the developer. This should give them early warnings about malformed data on their side, or recover in case the API is not available or experiences some kind of error we cannot handle.

mythx_models.util package

Module contents

This module contains various utility functions for MythX domain models.

mythx_models.util.**deserialize_api_timestamp**(*timestamp_str: str*) → datetime.datetime
Deserialize a JavaScript API timestamp into Python datetime format.

Parameters **timestamp_str** – The JS timestamp, e.g. `2019-01-10T01:29:38.410Z`

Returns A Python datetime object

mythx_models.util.**dict_delete_none_fields**(*d: Dict[KT, VT]*) → Dict[KT, VT]
Remove all keys that have "None" values in a dict.

Parameters **d** – The dictionary to sanitize

Returns The dict instance with all "None keys" removed

`mythx_models.util.`**`serialize_api_timestamp`**(*ts_obj: datetime.datetime*) → Optional[str]

Serialize a Python datetime object to its JS equivalent.

Parameters **ts_obj** – A Python datetime object

Returns The JS timestamp, e.g. `2019-01-10T01:29:38.410Z`

3.1.2 Module contents

Top-level package for mythx-models.

CHAPTER 4

Contributing

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

You can contribute in many ways:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at https://github.com/dmuhs/mythx-models/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.

4.1.2 Fix Bugs

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

4.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with "enhancement" and "help wanted" is open to whoever wants to implement it.

4.1.4 Write Documentation

The MythX domain models could always use more documentation, whether as part of the official MythX docs, in docstrings, or even on the web in blog posts, articles, and such.

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/dmuhs/mythx-models/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 :)

4.2 Get Started!

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

1. Fork the *mythx-models* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/mythx-models.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 mythx-models
$ cd mythx-models/
$ python setup.py develop
```

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 flake8 and the tests:

```
$ flake8 mythx-models tests or make lint
$ python3 setup.py test or make test
```

   To get flake8, just pip install them into your virtualenv.

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

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

4.3 Pull Request Guidelines

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

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python, 3.5 and 3.6, and 3.7. Check https://travis-ci.org/dmuhs/mythx-models/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bumpversion patch # possible: major / minor / patch
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.

CHAPTER 5

Credits

5.1 Development Lead

- Dominik Muhs <dominik.muhs@consensys.net>

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

m

mythx_models, 12
mythx_models.exceptions, 11
mythx_models.request, 8
mythx_models.request.analysis_list, 5
mythx_models.request.analysis_status, 6
mythx_models.request.analysis_submission, 6
mythx_models.request.auth_login, 6
mythx_models.request.auth_logout, 7
mythx_models.request.auth_refresh, 7
mythx_models.request.detected_issues, 7
mythx_models.request.version, 8
mythx_models.response, 11
mythx_models.response.analysis, 8
mythx_models.response.analysis_list, 9
mythx_models.response.analysis_status, 9
mythx_models.response.analysis_submission, 9
mythx_models.response.auth_login, 9
mythx_models.response.auth_logout, 9
mythx_models.response.auth_refresh, 9
mythx_models.response.detected_issues, 10
mythx_models.response.issue, 10
mythx_models.response.version, 11
mythx_models.util, 11

Index

A

allow_population_by_field_name (*mythx_models.request.analysis_list.AnalysisListRequest.Config attribute*), 5
allow_population_by_field_name (*mythx_models.request.analysis_submission.AnalysisSubmissionRequest.Config attribute*), 6
allow_population_by_field_name (*mythx_models.request.auth_refresh.AuthRefreshRequest.Config attribute*), 7
allow_population_by_field_name (*mythx_models.response.analysis.Analysis.Config attribute*), 8
allow_population_by_field_name (*mythx_models.response.analysis.AnalysisShort.Config attribute*), 8
allow_population_by_field_name (*mythx_models.response.auth_login.AuthLoginResponse.Config attribute*), 9
allow_population_by_field_name (*mythx_models.response.auth_refresh.AuthRefreshResponse.Config attribute*), 10
allow_population_by_field_name (*mythx_models.response.issue.Issue.Config attribute*), 10
allow_population_by_field_name (*mythx_models.response.issue.SourceLocation.Config attribute*), 10
Analysis (*class in mythx_models.response.analysis*), 8
Analysis.Config (*class in mythx_models.response.analysis*), 8
AnalysisListRequest (*class in mythx_models.request.analysis_list*), 5
AnalysisListRequest.Config (*class in mythx_models.request.analysis_list*), 5
AnalysisListResponse (*class in mythx_models.response.analysis_list*), 9
AnalysisShort (*class in mythx_models.response.analysis*), 8
AnalysisShort.Config (*class in mythx_models.response.analysis*), 8
AnalysisStatus (*class in mythx_models.response.analysis*), 8
AnalysisStatusRequest (*class in mythx_models.request.analysis_status*), 6
AnalysisStatusResponse (*class in mythx_models.response.analysis_status*), 9
AnalysisSubmissionRequest (*class in mythx_models.request.analysis_submission*), 6
AnalysisSubmissionRequest.Config (*class in mythx_models.request.analysis_submission*), 6
AnalysisSubmissionResponse (*class in mythx_models.response.analysis_submission*), 9
AuthLoginRequest (*class in mythx_models.request.auth_login*), 6
AuthLoginResponse (*class in mythx_models.response.auth_login*), 9
AuthLoginResponse.Config (*class in mythx_models.response.auth_login*), 9
AuthLogoutRequest (*class in mythx_models.request.auth_logout*), 7
AuthLogoutResponse (*class in mythx_models.response.auth_logout*), 9
AuthRefreshRequest (*class in mythx_models.request.auth_refresh*), 7
AuthRefreshRequest.Config (*class in mythx_models.request.auth_refresh*), 7
AuthRefreshResponse (*class in mythx_models.response.auth_refresh*), 9
AuthRefreshResponse.Config (*class in mythx_models.response.auth_refresh*), 9

C

components (*mythx_models.response.issue.SourceMap attribute*), 11

D

decompress() (*mythx_models.response.issue.SourceMap static method*), 11
deserialize_api_timestamp() (*in module mythx_models.util*), 11
DetectedIssuesRequest (*class in mythx_models.request.detected_issues*), 7
DetectedIssuesResponse (*class in mythx_models.response.detected_issues*), 10
dict_delete_none_fields() (*in module mythx_models.util*), 11

E

endpoint (*mythx_models.request.analysis_list.AnalysisListRequest attribute*), 5
endpoint (*mythx_models.request.analysis_status.AnalysisStatusRequest attribute*), 6
endpoint (*mythx_models.request.analysis_submission.AnalysisSubmissionRequest attribute*), 6
endpoint (*mythx_models.request.auth_login.AuthLoginRequest attribute*), 6
endpoint (*mythx_models.request.auth_logout.AuthLogoutRequest attribute*), 7
endpoint (*mythx_models.request.auth_refresh.AuthRefreshRequest attribute*), 7
endpoint (*mythx_models.request.detected_issues.DetectedIssuesRequest attribute*), 7
endpoint (*mythx_models.request.version.VersionRequest attribute*), 8
ERROR (*mythx_models.response.analysis.AnalysisStatus attribute*), 8

F

FINISHED (*mythx_models.response.analysis.AnalysisStatus attribute*), 8

H

headers (*mythx_models.request.analysis_list.AnalysisListRequest attribute*), 5
headers (*mythx_models.request.analysis_status.AnalysisStatusRequest attribute*), 6
headers (*mythx_models.request.analysis_submission.AnalysisSubmissionRequest attribute*), 6
headers (*mythx_models.request.auth_login.AuthLoginRequest attribute*), 6
headers (*mythx_models.request.auth_logout.AuthLogoutRequest attribute*), 7
headers (*mythx_models.request.auth_refresh.AuthRefreshRequest attribute*), 7
headers (*mythx_models.request.detected_issues.DetectedIssuesRequest attribute*), 7
headers (*mythx_models.request.version.VersionRequest attribute*), 8
HIGH (*mythx_models.response.issue.SEVERITY attribute*), 10

I

IN_PROGRESS (*mythx_models.response.analysis.AnalysisStatus attribute*), 8
Issue (*class in mythx_models.response.issue*), 10
Issue.Config (*class in mythx_models.response.issue*), 10
IssueDescription (*class in mythx_models.response.issue*), 10
IssueReport (*class in mythx_models.response.detected_issues*), 10

L

SourceLocation (*class in mythx_models.response.issue*), 10
LOW (*mythx_models.response.issue.SEVERITY attribute*), 10

M

MEDIUM (*mythx_models.response.issue.SEVERITY attribute*), 10
method (*mythx_models.request.analysis_list.AnalysisListRequest attribute*), 5
method (*mythx_models.request.analysis_status.AnalysisStatusRequest attribute*), 6
method (*mythx_models.request.analysis_submission.AnalysisSubmissionRequest attribute*), 6
method (*mythx_models.request.auth_login.AuthLoginRequest attribute*), 6
method (*mythx_models.request.auth_logout.AuthLogoutRequest attribute*), 7
method (*mythx_models.request.auth_refresh.AuthRefreshRequest attribute*), 7
method (*mythx_models.request.detected_issues.DetectedIssuesRequest attribute*), 7
method (*mythx_models.request.version.VersionRequest attribute*), 8
mythx_models (*module*), 12
mythx_models.exceptions (*module*), 11
mythx_models.response (*module*), 8
mythx_models.request.analysis_list (*module*), 5
mythx_models.request.analysis_status (*module*), 6
mythx_models.request.analysis_submission (*module*), 6
mythx_models.request.auth_login (*module*), 6
mythx_models.request.auth_logout (*module*), 7

mythx_models.request.auth_refresh (*module*), 7
mythx_models.request.detected_issues (*module*), 7
mythx_models.request.version (*module*), 8
mythx_models.response (*module*), 11
mythx_models.response.analysis (*module*), 8
mythx_models.response.analysis_list (*module*), 9
mythx_models.response.analysis_status (*module*), 9
mythx_models.response.analysis_submission (*module*), 9
mythx_models.response.auth_login (*module*), 9
mythx_models.response.auth_logout (*module*), 9
mythx_models.response.auth_refresh (*module*), 9
mythx_models.response.detected_issues (*module*), 10
mythx_models.response.issue (*module*), 10
mythx_models.response.version (*module*), 11
mythx_models.util (*module*), 11
MythXAPIError, 11

N

NONE (*mythx_models.response.issue.SEVERITY attribute*), 10

P

parameters (*mythx_models.request.analysis_list.AnalysisListRequest attribute*), 5
parameters (*mythx_models.request.analysis_status.AnalysisStatusRequest attribute*), 6
parameters (*mythx_models.request.analysis_submission.AnalysisSubmissionRequest attribute*), 6
parameters (*mythx_models.request.auth_login.AuthLoginRequest attribute*), 6
parameters (*mythx_models.request.auth_logout.AuthLogoutRequest attribute*), 7
parameters (*mythx_models.request.auth_refresh.AuthRefreshRequest attribute*), 7
parameters (*mythx_models.request.detected_issues.DetectedIssuesRequest attribute*), 7
parameters (*mythx_models.request.version.VersionRequest attribute*), 8
payload (*mythx_models.request.analysis_list.AnalysisListRequest attribute*), 5
payload (*mythx_models.request.analysis_status.AnalysisStatusRequest attribute*), 6
payload (*mythx_models.request.analysis_submission.AnalysisSubmissionRequest attribute*), 6
payload (*mythx_models.request.auth_login.AuthLoginRequest attribute*), 6
payload (*mythx_models.request.auth_logout.AuthLogoutRequest attribute*), 7
payload (*mythx_models.request.auth_refresh.AuthRefreshRequest attribute*), 7
payload (*mythx_models.request.detected_issues.DetectedIssuesRequest attribute*), 7
payload (*mythx_models.request.version.VersionRequest attribute*), 8

Q

QUEUED (*mythx_models.response.analysis.AnalysisStatus attribute*), 9

S

serialize_api_timestamp() (*in module mythx_models.util*), 12
SEVERITY (*class in mythx_models.response.issue*), 10
SourceLocation (*class in mythx_models.response.issue*), 10
SourceLocation.Config (*class in mythx_models.response.issue*), 10
SourceMap (*class in mythx_models.response.issue*), 10
sourcemap_reducer() (*mythx_models.response.issue.SourceMap static method*), 11
SourceMapLocation (*class in mythx_models.response.issue*), 11

U

UNKNOWN (*mythx_models.response.issue.SEVERITY attribute*), 10
use_enum_values (*mythx_models.request.analysis_list.AnalysisListReq attribute*), 5
use_enum_values (*mythx_models.request.analysis_submission.Analysi attribute*), 6
use_enum_values (*mythx_models.request.auth_refresh.AuthRefreshReq attribute*), 7
use_enum_values (*mythx_models.response.analysis.Analysis.Config attribute*), 8
use_enum_values (*mythx_models.response.analysis.AnalysisShort.Con attribute*), 8
use_enum_values (*mythx_models.response.auth_login.AuthLoginRespo attribute*), 9
use_enum_values (*mythx_models.response.auth_refresh.AuthRefreshRe attribute*), 10
use_enum_values (*mythx_models.response.issue.Issue.Config attribute*), 10

V

VersionRequest (*class in mythx_models.request.version*), 8

`VersionResponse` (*class in mythx_models.response.version*), 11