

---

# **mbtest Documentation**

***Release 2.3.0***

**Simon Brunning**

**May 05, 2020**



## CONTENTS:

<b>1</b>	<b>API Reference</b>	<b>3</b>
1.1	The <i>mbtest.server</i> module . . . . .	3
1.2	The <i>mbtest.imposters.imposters</i> module . . . . .	3
1.3	The <i>mbtest.imposters.stubs</i> module . . . . .	4
1.4	The <i>mbtest.imposters.predicates</i> module . . . . .	5
1.5	The <i>mbtest.imposters.responses</i> module . . . . .	7
1.6	The <i>mbtest.imposters.behaviors.copy</i> module . . . . .	8
1.7	The <i>mbtest.imposters.behaviors.lookup</i> module . . . . .	9
1.8	The <i>mbtest.imposters.behaviors.using</i> module . . . . .	10
1.9	The <i>mbtest.matchers</i> module . . . . .	11
1.10	The <i>mbtest.imposters.base</i> module . . . . .	11
<b>2</b>	<b>Indices and tables</b>	<b>13</b>
<b>3</b>	<b>Installation</b>	<b>15</b>
<b>4</b>	<b>Usage</b>	<b>17</b>
<b>5</b>	<b>Indices and tables</b>	<b>19</b>
	<b>Python Module Index</b>	<b>21</b>
	<b>Index</b>	<b>23</b>



Opinionated Python wrapper & utils for the [Mountebank](#) over the wire test double tool.  
Includes [pytest](#) fixture and [PyHamcrest](#) matchers.



## API REFERENCE

1.1 The *mbtest.server* module1.2 The *mbtest.imposters.imposters* module

**class** `mbtest.imposters.imposters.Imposter` (*stubs*, *port=None*, *protocol=<Protocol.HTTP: 'http'>*, *name=None*, *record\_requests=True*)

Represents a [Mountebank imposter](#). Think of an imposter as a mock website, running a protocol, on a specific port. Required behaviors are specified using stubs.

**Parameters**

- **stubs** (`Union[Stub, Proxy, Iterable[Union[Stub, Proxy]]]`) – One or more Stubs.
- **port** (`Optional[int]`) – Port.
- **protocol** (`Protocol`) – Protocol to run on.
- **name** (`Optional[str]`) – Impostor name - useful for interactive exploration of impostors on <http://localhost:2525/impostors>
- **record\_requests** (`bool`) – Record requests made against this impostor, so they can be asserted against later.

**class** `Protocol`

Imposter `Protocol`.

`HTTP = 'http'`

`HTTPS = 'https'`

`SMTP = 'smtp'`

`TCP = 'tcp'`

**property** `host`

Return type `str`

**property** `url`

Return type `furl`

**as\_structure** ()

Converted to a JSON serializable structure.

Return type `Any`

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Imposter*

**Returns** Converted object.

`mbtest.imposters.imposters.smtp_imposter` (*name='smtp', record\_requests=True*)

Canned SMTP server impostor.

**Return type** *Imposter*

## 1.3 The *mbtest.imposters.stubs* module

**class** `mbtest.imposters.stubs.Stub` (*predicates=None, responses=None*)

Represents a [Mountebank stub](#). Think of a stub as a behavior, triggered by a matching predicate.

**Parameters**

- **predicates** (*Union[Predicate, Iterable[Predicate], None]*) – Trigger this stub if one of these predicates matches the request
- **responses** (*Union[Response, Proxy, Iterable[Union[Response, Proxy]], None]*) – Use these response behaviors (in order)

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Stub*

**Returns** Converted object.

**class** `mbtest.imposters.stubs.Proxy` (*to, wait=None, inject\_headers=None, mode=<Mode.ALWAYS: 'proxyAlways'>*)

Represents a [Mountebank proxy](#).

**Parameters** **to** (*Union[furl, str]*) – The origin server, to which the request should proxy.

**class** `Mode`

Defines the replay behavior of the proxy.

**ONCE** = 'proxyOnce'

**ALWAYS** = 'proxyAlways'

**TRANSPARENT** = 'proxyTransparent'

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*



**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Proxy*

**Returns** Converted object.

## 1.4 The *mbtest.imposters.predicates* module

**class** *mbtest.imposters.predicates.BasePredicate*

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *BasePredicate*

**Returns** Converted object.

**class** *mbtest.imposters.predicates.Predicate* (*path=None, method=None, query=None, body=None, headers=None, xpath=None, operator=<Operator.EQUALS: 'equals'>, case\_sensitive=True*)

Represents a [Mountebank predicate](#). A predicate can be thought of as a trigger, which may or may not match a request.

### Parameters

- **path** (*Union[str, furl, None]*) – URL path.
- **method** (*Optional[Method]*) – HTTP method.
- **query** (*Optional[Mapping[str, Union[str, int, bool]]]*) – Query arguments, keys and values.
- **body** (*Optional[str]*) – Body text. Can be a string, or a JSON serialisable data structure.
- **headers** (*Optional[Mapping[str, str]]*) – Headers, keys and values.
- **xpath** (*Optional[str]*) – xpath query
- **operator** (*Operator*) –
- **case\_sensitive** (*bool*) –

**exception** *InvalidPredicateOperator*

**class** *Method*

Predicate HTTP method.

**DELETE** = 'DELETE'

**GET** = 'GET'

**HEAD** = 'HEAD'

**POST** = 'POST'

```
    PUT = 'PUT'

class Operator
    Predicate operator.

    EQUALS = 'equals'
    DEEP_EQUALS = 'deepEquals'
    CONTAINS = 'contains'
    STARTS_WITH = 'startsWith'
    ENDS_WITH = 'endsWith'
    MATCHES = 'matches'
    EXISTS = 'exists'

    classmethod has_value(name)
        Return type bool

as_structure()
    Converted to a JSON serializable structure.

    Return type Any
    Returns Structure suitable for JSON serialisation.

static from_structure(structure)
    Converted from a JSON serializable structure.

    Parameters structure (Any) – JSON structure to be converted.
    Return type Predicate
    Returns Converted object.

class mbtest.imposters.predicates.AndPredicate(left, right)

    as_structure()
        Converted to a JSON serializable structure.

        Return type Any
        Returns Structure suitable for JSON serialisation.

    static from_structure(structure)
        Converted from a JSON serializable structure.

        Parameters structure (Any) – JSON structure to be converted.
        Return type AndPredicate
        Returns Converted object.

class mbtest.imposters.predicates.OrPredicate(left, right)

    as_structure()
        Converted to a JSON serializable structure.

        Return type Any
        Returns Structure suitable for JSON serialisation.
```

**static** `from_structure (structure)`

Converted from a JSON serializable structure.

**Parameters** `structure` (*Any*) – JSON structure to be converted.

**Return type** *OrPredicate*

**Returns** Converted object.

**class** `mbtest.imposters.predicates.TcpPredicate (data)`

**as\_structure ()**

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static** `from_structure (structure)`

Converted from a JSON serializable structure.

**Parameters** `structure` (*Any*) – JSON structure to be converted.

**Return type** *TcpPredicate*

**Returns** Converted object.

## 1.5 The *mbtest.imposters.responses* module

**class** `mbtest.imposters.responses.Response (body="", status_code=200, wait=None, repeat=None, headers=None, mode=None, copy=None, decorate=None, lookup=None, shell_transform=None)`

Represents a Mountebank 'is' response behavior.

### Parameters

- **body** (*str*) – Body text for response. Can be a string, or a JSON serialisable data structure.
- **status\_code** (*Union[int, str]*) – HTTP status code
- **wait** (*Union[int, str, None]*) – Add latency, in ms.
- **repeat** (*Optional[int]*) – Repeat this many times before moving on to next response.
- **headers** (*Optional[Mapping[str, str]]*) – Response HTTP headers
- **mode** (*Optional[Mode]*) – Mode - text or binary
- **copy** (*Optional[Copy]*) – Copy behavior
- **decorate** (*Optional[str]*) – Decorate behavior.
- **lookup** (*Optional[Lookup]*) – Lookup behavior
- **shell\_transform** (*Union[str, Iterable[str], None]*) – shellTransform behavior

**class** `Mode`

An enumeration.

**TEXT** = 'text'

**BINARY** = 'binary'

**property** `body`

**Return type** `str`

**as\_structure**()

Converted to a JSON serializable structure.

**Return type** `Any`

**Returns** Structure suitable for JSON serialisation.

**static from\_structure**(*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (`Any`) – JSON structure to be converted.

**Return type** `Response`

**Returns** Converted object.

**class** `mbtest.imposters.responses.TcpResponse`(*data*)

**as\_structure**()

Converted to a JSON serializable structure.

**Return type** `Any`

**Returns** Structure suitable for JSON serialisation.

**static from\_structure**(*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (`Any`) – JSON structure to be converted.

**Return type** `TcpResponse`

**Returns** Converted object.

## 1.6 The `mbtest.imposters.behaviors.copy` module

**class** `mbtest.imposters.behaviors.copy.Copy`(*from\_*, *into*, *using*)

Represents a `copy` behavior.

**Parameters**

- **from** – The name of the request field to copy from, or, if the request field is an object, then an object specifying the path to the request field.
- **into** (`str`) – The token to replace in the response with the selected request value.
- **using** (`Using`) – The configuration needed to select values from the response.

**as\_structure**()

Converted to a JSON serializable structure.

**Return type** `Any`

**Returns** Structure suitable for JSON serialisation.

**static from\_structure**(*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (`Any`) – JSON structure to be converted.

**Return type** *Copy*

**Returns** Converted object.

## 1.7 The *mbtest.imposters.behaviors.lookup* module

**class** `mbtest.imposters.behaviors.lookup.Lookup` (*key*, *datasource\_path*, *data-source\_key\_column*, *into*)

Represents a *lookup* behavior.

### Parameters

- **key** (*Key*) – How to select the key from the request.
- **datasource\_path** (*Union[str, Path]*) – The path to the data source.
- **datasource\_key\_column** (*str*) – The header of the column to match against the key.
- **into** (*str*) – The token to replace in the response with the selected request value.

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Lookup*

**Returns** Converted object.

**class** `mbtest.imposters.behaviors.lookup.Key` (*from\_*, *using*, *index=0*)

The information on how to select the key from the request.

### Parameters

- **from** – The name of the request field to copy from, or, if the request field is an object, then an object specifying the path to the request field.
- **using** (*Using*) – The configuration needed to select values from the response
- **index** (*int*) – Index of the item from the result array to be selected.

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Key*

**Returns** Converted object.

## 1.8 The *mbtest.imposters.behaviors.using* module

**class** `mbtest.imposters.behaviors.using.Using(method, selector)`

How to select values from the response.

**Parameters**

- **method** (*Method*) – The method used to select the value(s) from the request.
- **selector** (*str*) – The selector used to select the value(s) from the request.

**class** `Method`

An enumeration.

**REGEX** = 'regex'

**XPATH** = 'xpath'

**JSONPATH** = 'jsonpath'

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Using*

**Returns** Converted object.

**class** `mbtest.imposters.behaviors.using.UsingRegex(selector, ignore_case=False, multiline=False)`

Select values from the response using a regular expression.

**Parameters**

- **selector** (*str*) – The selector used to select the value(s) from the request.
- **ignore\_case** (*bool*) – Uses a case-insensitive regular expression
- **multiline** (*bool*) – Uses a multiline regular expression

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *UsingRegex*

**Returns** Converted object.

**class** `mbtest.imposters.behaviors.using.UsingXpath(selector, ns=None)`

Select values from the response using an xpath expression.

**Parameters**

- **selector** (*str*) – The selector used to select the value(s) from the request.
- **ns** (*Optional*[*Mapping*[*str*, *str*]]) – The ns object maps namespace aliases to URLs

**as\_structure()**

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure(structure)**

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *UsingXpath*

**Returns** Converted object.

**class** mbtest.imposters.behaviors.using.UsingJsonpath(*selector*)

Select values from the response using a jsonpath expression.

**Parameters** **selector** (*str*) – The selector used to select the value(s) from the request.

**static from\_structure(structure)**

Converted from a JSON serializable structure.

**Parameters** **structure** – JSON structure to be converted.

**Return type** *UsingJsonpath*

**Returns** Converted object.

## 1.9 The *mbtest.matchers* module

### 1.10 The *mbtest.imposters.base* module

**class** mbtest.imposters.base.JsonSerializable

Object capable of being converted to a JSON serializable structure (using *as\_structure()*) or from such a structure ((using *from\_structure()*).

**abstract as\_structure()**

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**abstract static from\_structure(structure)**

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *JsonSerializable*

**Returns** Converted object.





## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`



## INSTALLATION

Install from [Pypi](#) as usual, using `pip`, `tox`, or `setup.py`.

Also requires [Mountebank](#) to have been installed:

```
$ npm install mountebank@2.2 --production
```



**USAGE**

A basic example:

```
import requests
from hamcrest import assert_that
from brunns.matchers.response import is_response
from mbtest.matchers import had_request
from mbtest.imposters import Imposter, Predicate, Response, Stub

def test_request_to_mock_server(mock_server):
    # Set up mock server with required behavior
    imposter = Imposter(Stub(Predicate(path="/test"),
                             Response(body="sausages")))

    with mock_server(imposter) as server:
        # Make request to mock server - exercise code under test here
        response = requests.get("{} /test".format(imposter.url))

        assert_that("We got the expected response",
                     response, is_response().with_status_code(200).and_body("sausages
↪"))
        assert_that("The mock server recorded the request",
                     server, had_request().with_path("/test").and_method("GET"))
```

Needs a `pytest` fixture, most easily defined in `conftest.py`:

```
import pytest
from mbtest import server

@pytest.fixture(scope="session")
def mock_server(request):
    return server.mock_server(request)
```



## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`





## PYTHON MODULE INDEX

### m

- `mbtest.imposters.base`, [11](#)
- `mbtest.imposters.behaviors.copy`, [8](#)
- `mbtest.imposters.behaviors.lookup`, [9](#)
- `mbtest.imposters.behaviors.using`, [10](#)
- `mbtest.imposters.imposters`, [3](#)
- `mbtest.imposters.predicates`, [5](#)
- `mbtest.imposters.responses`, [7](#)
- `mbtest.imposters.stubs`, [4](#)



## INDEX

### A

ALWAYS (*mbtest.imposters.stubs.Proxy.Mode* attribute),  
4  
AndPredicate (class in *mbtest.imposters.predicates*),  
6

as\_structure() (*mbtest.imposters.base.JsonSerializable*  
method), 11  
as\_structure() (*mbtest.imposters.behaviors.copy.Copy*  
method), 8  
as\_structure() (*mbtest.imposters.behaviors.lookup.Key*  
method), 9

as\_structure() (*mbtest.imposters.behaviors.lookup.Lookup*  
method), 9  
as\_structure() (*mbtest.imposters.behaviors.using.Using*  
method), 10  
as\_structure() (*mbtest.imposters.behaviors.using.UsingRegex*  
method), 10

as\_structure() (*mbtest.imposters.behaviors.using.UsingXpath*  
method), 11  
as\_structure() (*mbtest.imposters.imposters.Imposter*  
method), 3  
as\_structure() (*mbtest.imposters.predicates.AndPredicate*  
method), 6

as\_structure() (*mbtest.imposters.predicates.OrPredicate*  
method), 6

as\_structure() (*mbtest.imposters.predicates.Predicate*  
method), 6

as\_structure() (*mbtest.imposters.predicates.TcpPredicate*  
method), 7

as\_structure() (*mbtest.imposters.responses.Response*  
method), 8

as\_structure() (*mbtest.imposters.responses.TcpResponse*  
method), 8

as\_structure() (*mbtest.imposters.stubs.Proxy*  
method), 4

as\_structure() (*mbtest.imposters.stubs.Stub*  
method), 4

### B

BasePredicate (class  
*mbtest.imposters.predicates*), 5

BINARY (*mbtest.imposters.responses.Response.Mode* at-  
tribute), 7

body() (*mbtest.imposters.responses.Response* prop-  
erty), 7

### C

CONTAINS (*mbtest.imposters.predicates.Predicate.Operator*  
attribute), 6

Copy (class in *mbtest.imposters.behaviors.copy*), 8

### D

DNE\_EQUALS (*mbtest.imposters.predicates.Predicate.Operator*  
attribute), 6

DELETE (*mbtest.imposters.predicates.Predicate.Method*  
attribute), 5

### E

ENDS\_WITH (*mbtest.imposters.predicates.Predicate.Operator*  
attribute), 6

EQUALS (*mbtest.imposters.predicates.Predicate.Operator*  
attribute), 6

EXISTS (*mbtest.imposters.predicates.Predicate.Operator*  
attribute), 6

### F

from\_structure() (*mbtest.imposters.base.JsonSerializable*  
static method), 11

from\_structure() (*mbtest.imposters.behaviors.copy.Copy*  
static method), 8

from\_structure() (*mbtest.imposters.behaviors.lookup.Key*  
static method), 9

from\_structure() (*mbtest.imposters.behaviors.lookup.Lookup*  
static method), 9

from\_structure() (*mbtest.imposters.behaviors.using.Using*  
static method), 10

from\_structure() (*mbtest.imposters.behaviors.using.UsingJsonpath*  
static method), 11

from\_structure() (*mbtest.imposters.behaviors.using.UsingRegex*  
static method), 10

from\_structure() (*mbtest.imposters.behaviors.using.UsingXpath*  
static method), 11

[from\\_structure\(\)](#) (*mbtest.imposters.imposters.Imposter* static method), 4  
[from\\_structure\(\)](#) (*mbtest.imposters.predicates.AndPredicate* static method), 6  
[from\\_structure\(\)](#) (*mbtest.imposters.predicates.BasePredicate* static method), 5  
[from\\_structure\(\)](#) (*mbtest.imposters.predicates.OrPredicate* static method), 6  
[from\\_structure\(\)](#) (*mbtest.imposters.predicates.Predicate* static method), 6  
[from\\_structure\(\)](#) (*mbtest.imposters.predicates.TcpPredicate* static method), 7  
[from\\_structure\(\)](#) (*mbtest.imposters.responses.Response* static method), 8  
[from\\_structure\(\)](#) (*mbtest.imposters.responses.TcpResponse* static method), 8  
[from\\_structure\(\)](#) (*mbtest.imposters.stubs.Proxy* static method), 5  
[from\\_structure\(\)](#) (*mbtest.imposters.stubs.Stub* static method), 4

**G**  
[GET](#) (*mbtest.imposters.predicates.Predicate.Method* attribute), 5

**H**  
[has\\_value\(\)](#) (*mbtest.imposters.predicates.Predicate.Operator* class method), 6  
[HEAD](#) (*mbtest.imposters.predicates.Predicate.Method* attribute), 5  
[host\(\)](#) (*mbtest.imposters.imposters.Imposter* property), 3  
[HTTP](#) (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3  
[HTTPS](#) (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3

**I**  
[Imposter](#) (class in *mbtest.imposters.imposters*), 3  
[Imposter.Protocol](#) (class in *mbtest.imposters.imposters*), 3

**J**  
[JSONPATH](#) (*mbtest.imposters.behaviors.using.Using.Method* attribute), 10  
[JsonSerializable](#) (class in *mbtest.imposters.base*), 11

**K**  
[Key](#) (class in *mbtest.imposters.behaviors.lookup*), 9

**L**  
[Lookup](#) (class in *mbtest.imposters.behaviors.lookup*), 9

**M**  
[MATCHES](#) (*mbtest.imposters.predicates.Predicate.Operator* attribute), 6  
[mbtest.imposters.base](#) module, 11  
[mbtest.imposters.behaviors.copy](#) module, 8  
[mbtest.imposters.behaviors.lookup](#) module, 9  
[mbtest.imposters.behaviors.using](#) module, 10  
[mbtest.imposters.imposters](#) module, 3  
[mbtest.imposters.predicates](#) module, 5  
[mbtest.imposters.responses](#) module, 7  
[mbtest.imposters.stubs](#) module, 4  
[module](#)  
[mbtest.imposters.base](#), 11  
[mbtest.imposters.behaviors.copy](#), 8  
[mbtest.imposters.behaviors.lookup](#), 9  
[mbtest.imposters.behaviors.using](#), 10  
[mbtest.imposters.imposters](#), 3  
[mbtest.imposters.predicates](#), 5  
[mbtest.imposters.responses](#), 7  
[mbtest.imposters.stubs](#), 4

**O**  
[ONCE](#) (*mbtest.imposters.stubs.Proxy.Mode* attribute), 4  
[OrPredicate](#) (class in *mbtest.imposters.predicates*), 6

**P**  
[POST](#) (*mbtest.imposters.predicates.Predicate.Method* attribute), 5  
[Predicate](#) (class in *mbtest.imposters.predicates*), 5  
[Predicate.InvalidPredicateOperator](#), 5  
[Predicate.Method](#) (class in *mbtest.imposters.predicates*), 5  
[Predicate.Operator](#) (class in *mbtest.imposters.predicates*), 6  
[Proxy](#) (class in *mbtest.imposters.stubs*), 4  
[Proxy.Mode](#) (class in *mbtest.imposters.stubs*), 4  
[PUT](#) (*mbtest.imposters.predicates.Predicate.Method* attribute), 5

**R**  
[REGEX](#) (*mbtest.imposters.behaviors.using.Using.Method* attribute), 10  
[Response](#) (class in *mbtest.imposters.responses*), 7  
[Response.Mode](#) (class in *mbtest.imposters.responses*), 7

## S

SMTP (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3  
 smtp\_imposter() (in module *mbtest.imposters.imposters*), 4  
 STARTS\_WITH (*mbtest.imposters.predicates.Predicate.Operator* attribute), 6  
 Stub (class in *mbtest.imposters.stubs*), 4

## T

TCP (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3  
 TcpPredicate (class in *mbtest.imposters.predicates*), 7  
 TcpResponse (class in *mbtest.imposters.responses*), 8  
 TEXT (*mbtest.imposters.responses.Response.Mode* attribute), 7  
 TRANSPARENT (*mbtest.imposters.stubs.Proxy.Mode* attribute), 4

## U

url() (*mbtest.imposters.imposters.Imposter* property), 3  
 Using (class in *mbtest.imposters.behaviors.using*), 10  
 Using.Method (class in *mbtest.imposters.behaviors.using*), 10  
 UsingJsonpath (class in *mbtest.imposters.behaviors.using*), 11  
 UsingRegex (class in *mbtest.imposters.behaviors.using*), 10  
 UsingXPath (class in *mbtest.imposters.behaviors.using*), 10

## X

XPATH (*mbtest.imposters.behaviors.using.Using.Method* attribute), 10