pinn Documentation

Release 0.1.2+1.gd7d3f2d.dirty

Pinn Technologies, Inc

Apr 01, 2019

CONTENTS:

Fig. 1: Python library bindings for the Pinn REST API

The easiest way to install is by using PyPi, either directly or within your requirements.txt:

```
pip install --upgrade pinn
```

You can also install directly from source:

```
python setup.py install
```

CHAPTER

ONE

CONFIGURATION

Now that you have the package properly installed, you can configure the library with your Pinn secret key. If you don't have access, first reach out to sales@pinn.ai.

You may set the `secret_key` value directly (convenient for shell testing), but typically you should set `PINN_SECRET_KEY` as an environmental variable for the package to pick up automatically.

Directly set the key like so:

```
>>> import pinn
>>> pinn.secret_key = 'sk_H5dqd648Ix9vJ7J70NjRlhXyP6AouUx8'
```

Or, set your environment variable for the library to autoload:

```
$ export PINN_SECRET_KEY=sk_H5dqd648Ix9vJ7J70NjRlhXyP6AouUx8
```

Now you're all set to start making API calls.

CHAPTER

TWO

BASICS

The library is very straightforward to use. Each resource has at minumum one of the following CRUDL methods:

- `create()`
- `retrieve()`
- `update()`
- `delete()`
- `list()`

For example to create a new Pinn user:

```
>>> user = pinn.User.create()
>>> user.user_id
'usr_zsWljQAN7p6hBbbi15FXXUJp'
```

Some time later we can get retrieve the user like so:

```
>>> user = pinn.User.retrieve(user_id='usr_zsWljQAN7p6hBbbi15FXXUJp')
>>> user.created_at
1553984207
```

Some resources like a User can be updated:

```
# You can update the user without retrieving
>>> user = pinn.User.update(user_id='usr_zsWljQAN7p6hBbbi15FXXUJp',
                            metadata={'external_id': 'ypf3JZpTB5DfByiZt'})
>>>
# Or if you already have a User object on hand, you can update the property directly␣
↪and save
>>> user.metadata = {'external_id': 'ypf3JZpTB5DfByiZt'}
>>> user.save()
```

All resources created under your environment can be deleted, use with extreme caution:

```
# Delete the resource when you have an ID on hand
>>> pinn.User.delete(user_id='usr_zsWljQAN7p6hBbbi15FXXUJp')
True
# or delete the resource if you already have it on hand
>>> user.delete()
True
```

Finally listing resources in reverse chronological order:

```
>>> pinn.User.list()
```

CHAPTER

THREE

PRODUCTION CHECKLIST

Once everything is working, take a glance at our checklist here to make sure your implementation is up to snuff.

- Pinn `secret_key` value is stored securely on a server and never in version control
- Pinn ID tokens are always verified before claims are trusted
- Enrollment keys are only issued once user has sufficient privledge within your system
- Pinn errors are all handled for
- The Pinn User ID is mapped to my User resource in a persistent datastore

CHAPTER

FOUR

API REFERENCE

pinn.**secret_key = None**
The Pinn secret key value, used to authenticate API requests.

Type str

pinn.**api_host = 'https://pinnapis.com'**
Pinn API host, defaults to production Pinn-hosted service.

Type str

pinn.**api_version = None**
Pinn API verision string, set this to use a specific API version.

Type str

4.1 Resources

4.1.1 User

class pinn.**User**(*response*)
User resource and interface.

response
Underlying dictionary response

Type dict

object
Identifier for the resource

Type str

user_id
Unique ID for the user

Type str

created_at
Unix timestamp in seconds for when the user was created

Type int

updated_at
Unix timestamp in seconds for when the user was last updated

Type int

authenticated_at
Unix timestamp in seconds for when the user last authenticated

Type int

device_enrolled
True if the user has at least 1 enrolled device

Type bool

left_palm_enrolled
True if the user has enrolled their left palm

Type bool

right_palm_enrolled
True if the user has enrolled their right palm

Type bool

metadata
Arbitrary key/value data attached to the user

Type dict

classmethod create(*metadata=None*)
Create a new Pinn user.

Parameters **metadata** (*dict, optional*) – The ID of the Pinn user being authorized to enroll.

Returns The newly created user

Return type *User*

Raises

- ***pinn.errors.RequestFailedError*** – Invalid metadata dict provided
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

classmethod delete(*user_id*)
Update a Pinn user.

Parameters **user_id** (*str*) – The ID of the Pinn user to delete.

Returns A Deleted response

Return type bool

Raises

- ***pinn.errors.RequestFailedError*** – User not found, Invalid metadata or status
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

classmethod retrieve(*user_id*)
Retrieve a Pinn user.

Parameters **user_id** (*str*) – The ID of the Pinn user to query.

Returns A User resource

Return type *User*

Raises

- ***pinn.errors.RequestFailedError*** – User not found
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

classmethod update(*user_id*, *metadata*, *status*)

Update a Pinn user.

Parameters

- **user_id** (*str*) – The ID of the Pinn user to update.
- **metadata** (*dict, optional*) – Metadata to update
- **status** (*str, optional*) – User status to update

Returns A User resource

Return type *User*

Raises

- ***pinn.errors.RequestFailedError*** – User not found, Invalid metadata or status
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

4.1.2 App

class pinn.**App**(*response*)

App resource and interface.

response

Underlying dictionary response

Type dict

object

Identifier for the resource

Type str

app_id

Unique ID for the app

Type str

created_at

Unix timestamp in seconds for when the app was created

Type int

updated_at

Unix timestamp in seconds for when the app was last updated

Type int

publishable_key

Non-secret key that is embedded to configure the SDK

Type str

app_type
Either *ios*, *android* or *web*

Type str

name
Name of the app

Type str

description
Optional app description text

Type str

classmethod create(*app_type*, *name*, *description=None*)
Create a new Pinn iOS/Android/Web app.

Parameters

- **app_type** (*str*) – Either *ios*, *android* or *web*
- **name** (*str*) – Human readable name for the app
- **description** (*str, optional*) – Descriptive text for the app

Returns Newly created app resource

Return type *App*

Raises

- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

classmethod delete(*app_id*)
Delete a Pinn iOS/Android/Web app.

Note: This may break your integration if app is currently in use live.

Parameters **app_id** (*str*) – ID of the app to delete.

Returns A deleted response

Return type Deleted

Raises

- ***pinn.errors.RequestFailedError*** – App not found
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

classmethod retrieve(*app_id*)
Retrieve a Pinn iOS/Android/Web app.

Parameters **app_id** (*str*) – ID of the app to query.

Returns The queried app resource

Return type *App*

Raises

- ***pinn.errors.RequestFailedError*** – App not found
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

4.1.3 Device

class pinn.**Device**(*response*)

Device resource and interface.

response

Underlying dictionary response

Type dict

object

Identifier for the resource

Type str

device_id

Unique ID for the device

Type str

user_id

ID of the user the device is registered to

Type str

created_at

Unix timestamp in seconds for when the app was created

Type int

updated_at

Unix timestamp in seconds for when the app was last updated

Type int

name

Human readable name of the device

Type str

make

Manufacturer make of the device

Type str

model

Model identifier of the device

Type str

platform

Either *ios* or *android*

Type str

platform_version

Last recorded platform version of the device

Type str

framework_version
Last recorded framework version for the device

Type str

push_notification_token
Device APNS or FCM push notification token

Type str, optional

classmethod retrieve(*device_id*)
Retrieve a specific key for a given user.

4.1.4 Event

class pinn.**Event**(*response*)
Event resource and interface.

response
Underlying dictionary response

Type dict

object
Identifier for the resource

Type str

event_id
Unique ID for the event

Type str

created_at
Unix timestamp in seconds for when the app was created

Type int

data
Underlying data for the event

Type dict

request
ID of the request this event corresponds to

Type str

event_type
The type of the event that occured

Type str

livemode
True if the event occured outside of sandbox environment

Type bool

4.1.5 Log

class pinn.**Log**(*response*)
Log resource and interface.

response
Underlying dictionary response

Type dict

object
Identifier for the resource

Type str

log_id
Unique ID for the log

Type str

user_id
ID of the user that authenticated

Type str

device_id
ID of the device that authenticated

Type str

app_id
ID of the mobile app that was involved

Type str

web_app_id
ID of web app that was involved

Type str, optional

created_at
Unix timestamp in seconds for when the app was created

Type int

success
True if authentication was successful

Type bool

score
Combined score for all scorable biometrics

Type float

factors
Breakdown of all factors used an their score if applicable

Type dict

auth_type
Either *web* or *mobile*

Type str

4.1.6 WebhookEndpoint

class pinn.**WebhookEndpoint**(*response*)
Webhook endpoint resource and interface.

response
Underlying dictionary response

Type dict

object
Identifier for the resource

Type str

webhook_endpoint_id
Unique ID for the webhook endpoint

Type str

created_at
Unix timestamp in seconds for when the key was created

Type int

created_at
Unix timestamp in seconds for when the key was updated

Type int

status
The status of the webhook endpoint

Type str

url
Fully qualified endpoint URL where webhook events are delivered to

Type str

events
The list of events this webhook is subscribed to

Type list

livemode
True if webhook endpoint is in live environment

Type bool

secret
A secret to authenticate the enroll request from Pinn mobile SDK

Type str

4.1.7 EnrollmentKey

class pinn.**EnrollmentKey**(*response*)
Enrollment Key resource and interface.

response
Underlying dictionary response

Type dict

object
Identifier for the resource

Type str

enrollment_key_id
Unique ID for this enrollment key

Type str

user_id
ID of the user who the key corresponds to

Type str

created_at
Unix timestamp in seconds for when the key was created

Type int

expires_at
Unix timestamp in seconds for when the key will expire

Type int

livemode
True if enrollment key is in live environment

Type bool

secret
A secret to authenticate the enroll request from Pinn mobile SDK

Type str

classmethod create(*user_id*)
Create a new enrollment key for a given user.

This key is used as proof that your system is authorizing a given Pinn user to enroll. You MUST ensure proper trust is established between the user and your service before performing this operation.

Parameters **user_id** (*str*) – The ID of the Pinn user being authorized to enroll.

Returns A special purpose key to authorize enrollment

Return type *EnrollmentKey*

Raises

- ***pinn.errors.RequestFailedError*** – User with ID supplied does not exist
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

4.1.8 RecoveryKey

class pinn.**RecoveryKey**(*response*)
Recovery key resource and interface.

response
Underlying dictionary response

Type dict

object
Identifier for the resource

Type str

recovery_key_id
Unique ID for this recovery key

Type str

user_id
ID of the user who the key corresponds to

Type str

created_at
Unix timestamp in seconds for when the key was created

Type int

expires_at
Unix timestamp in seconds for when the key will expire

Type int

livemode
True if recovery key is in live environment

Type bool

secret
A secret to authenticate the recovery request from Pinn mobile SDK

Type str

flow
The auth flow required for recovery

Type list

classmethod create(*user_id*, *flow*)
Create a new recovery key for a given user.

Parameters

- **user_id** (*str*) – ID of user to create the recovery key for.
- **flow** (*list, optional*) – List of remote factors to check prior to re-enrollment

Returns The newly created recovery key

Return type *RecoveryKey*

Raises

- ***pinn.errors.RequestFailedError*** – User not found, invalid *flow* provided
- ***pinn.errors.APIError*** – Internal server error
- ***pinn.errors.APIConnectionError*** – API is unreachable [501-503]

4.2 Errors

exception pinn.errors.**APIConnectionError**(*response*)
The Pinn host is unreachable, you can and should retry this request

exception pinn.errors.**APIError**(*response*)
An internal server error caused the request to fail.

Pinn is automatically notified if issues are encountered that are internal to our service. You should retry this request in case the error was intermittent.

exception pinn.errors.**AuthenticationError**(*response*)
The secret key set is invalid and cannot be used to authenticate your request.

exception pinn.errors.**ConfigurationError**
This error is thrown fast if the library is misconfigured.

For example, if you attempt to call the library without a *secret_key* value set a ConfigurationError will be thrown.

exception pinn.errors.**ForbiddenError**(*response*)
This error is returned if you attempt to access a resource you don't own.

Do not attempt to retry this error, and if you receieve this error it likely means you have an incorrect *secret_key* set on the library.

exception pinn.errors.**IDTokenVerificationError**
This exception can be raised during an attempt to verify an ID token.

Potentially this can be from:

- An invalid JWT signature
- Missing required claims
- *amr* claim mismatch from expectations

exception pinn.errors.**InvalidRequestError**(*response*)
The request was malformed and could not be understood by the server.

This could mean that an incorrect type was sent for a parameter, an invalid HTTP method was used or that the resource you are requesting does not exist. Do not attempt to retry the request.

exception pinn.errors.**PinnError**(*response*)
Base exception class for a Pinn API Error.

static from_response(*response*)
Create an error of the right PinnError subclass from an API response.

exception pinn.errors.**RateLimitError**(*response*)
The API rate limit has been exceeded.

You should attempt to retry this request after some time has passed.

exception pinn.errors.**RequestFailedError**(*response*)
The request was properly formed with valid syntax, but could not be performed.

This occurs in scenarios where circumstances prevent an operation from being executed, typically this can be situations like invalid resource state to perform the op.

4.3 Utilities

class pinn.utils.**IDToken**
Provides functionality for verifying an incoming Pinn ID token.

static verify(*id_token*, *amr*)
Verify a new incoming Pinn ID token value.

Parameters

- **id_token** (*str*) – The ID Token received from a Pinn SDK

- **amr** (*list*) – List of required authentication methods required

Returns A dictionary of the verified claims

Return type dict

4.4 Other

pinn.**healthy**()

Perform a health check against the configured Pinn host.

Returns True if host is available, False otherwise

Return type bool

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

p

INDEX

O

P

R

S

U

V

W