aiohttp-remotes Documentation

Ocean S.A.

Nov 05, 2020

CONTENTS:

The library is a set of useful tools for `aiohttp.web` server.

The full list of tools is:

- *AllowedHosts* – restrict a set of incoming connections to allowed hosts only.
- *BasicAuth* – protect web application by *basic auth* authorization.
- *Cloudflare* – make sure that web application is protected by CloudFlare.
- *ForwardedRelaxed* and *ForwardedStrict* – process `Forwarded` HTTP header and modify corresponding `scheme`, `host`, `remote` attributes in strong secured and relaxed modes.
- *Secure* – ensure that web application is handled by HTTPS (SSL/TLS) only, redirect plain HTTP to HTTPS automatically.
- *XForwardedRelaxed* and *XForwardedStrict* – the same as `ForwardedRelaxed` and `ForwardedStrict` but process old-fashion `X-Forwarded-*` headers instead of new standard `Forwarded`.

CHAPTER

ONE

API

1.1 Setup

coroutine aiohttp_remotes.**setup**(*app*, **tools*)

Setup tools provided by the module.

A tool is class instance from described below list, the functon registers provided tools into aiohttp application *app*, e.g.:

```
from aiohttp_remotes import BasicAuth, Secure, setup

app = web.Application()

await setup(app, Secure(), BasicAuth("user", "password", "realm"))
```

Order of tools is important: in the example redirect to HTTPS is performed *before* credentials check, thus login/password is sent via SSL encrypted connection.

1.2 AllowedHosts

class aiohttp_remotes.**AllowedHosts**(*allowed_hosts=('*',)*, *, *white_paths_()*)

Restrict a list of host/domain names that this aiohttp application can serve. This is a security measure to prevent *HTTP Host header attacks*, which are possible even under many seemingly-safe web server configurations.

Parameters

- **allowed_hosts** – an iterable of allowed client IP addresses. `'*'` is a wildcard for accepting any host.
- **white_paths** – an iterable of white paths, see *White paths* for details.

1.3 BasicAuth

class aiohttp_remotes.**BasicAuth**(*username*, *password*, *realm*, ** white_paths=()*)

Protect web application by basic auth authorization.

Parameters

- **username** (*str*) – user name
- **password** (*str*) – password

- **realm** (*str*) – realm
- **white_paths** – an iterable of white paths, see *White paths* for details.

1.4 CloudFlare

class aiohttp_remotes.**Cloudflare**(*client=None*)

Make sure that web application is protected by CloudFlare.

The tools should be used with *XForwardedStrict* or *XForwardedRelaxed* to setup HTTP *scheme*, *host* and *remote* properly.

Parameters **client** – aiohttp.ClientSession instance for performing HTTP requests to CloudFlare configuration resources.

The class creates a temporary client if None is provided.

1.5 Forwarded

class aiohttp_remotes.**ForwardedRelaxed**(*num=1*)

Modify scheme, host, remote attributes giving the values from *num* Forwarded HTTP header record (last one by default).

The tools is useful for getting real client IP, scheme (HTTPS or HTTP) and HOST if aiohttp application is deployed behind *Reverse Proxy* like NGINX.

The class does not perform any security check, use it with caution.

class aiohttp_remotes.**ForwardedStrict**(*trusted*, ***, *white_paths=()*)

Process Forwarded HTTP header and modify corresponding scheme, host, remote attributes in strong secure mode.

Restrict access (return *400 Bad Request*) if *reverse proxy* addresses are not match provided configuration.

Parameters

- **trusted** – a list of trusted reverse proxies, see *Trusted hosts* for details.
- **white_paths** – an iterable of white paths, see *White paths* for details.

1.6 Secure

class aiohttp_remotes.**Secure**(***, *redirect=True*, *redirect_url=None*, *white_paths=()*)

Ensure that web application is handled by HTTPS (SSL/TLS) only, redirect plain HTTP to HTTPS automatically.

Parameters

- **redirect** (*bool*) – do redirection instead of returning *400 Bad Request*.
- **redirect_url** – redirection URL, the same usr as requested non-secure HTTP if not specified.
- **white_paths** – an iterable of white paths, see *White paths* for details.

1.7 X-Forwarded

class aiohttp_remotes.**XForwardedRelaxed**(*num=1*)

Modify scheme, host, remote attributes giving the values from *num* X-Forwarded-* HTTP headers (last record by default).

The tools is useful for getting real client IP, scheme (HTTPS or HTTP) and HOST if aiohttp application is deployed behind *Reverse Proxy* like NGINX.

The class does not perform any security check, use it with caution.

class aiohttp_remotes.**XForwardedStrict**(*trusted*, *, *white_paths=()*)

Process X-Forwarded-* HTTP headers and modify corresponding scheme, host, remote attributes in strong secure mode.

Restrict access (return *400 Bad Request*) if *reverse proxy* addresses are not match provided configuration.

Parameters

- **trusted** – a list of trusted reverse proxies, see *Trusted hosts* for details.
- **white_paths** – an iterable of white paths, see *White paths* for details.

1.8 Trusted hosts

trusted parameter is a sequence of trusted hosts or networks.

The format is list of items, where every item describes a *reverse proxy*.

Item can be:

- A list of IP addresses or networks, every element is:
  - IP address is IPv4 or IPv6 in form accepted by ipaddress.ip_address().
  - Network is IPv4 or IPv6 network in form accepted by ipaddress.ip_network().
- Ellipsis ...

The check is performed against Forwarded or X-Forwarded-* HTTP headers.

The leftmost item in the list describes *reverse proxy* closest to web application's host.

IP address or network is specified by strict checking, ... is the placeholder for skip checking (should be rightmost element).

In practice ellipsis is secure if used with CloudFlare only. *Cloudflare* checks corresponding proxy against a list of CloudFlare proxy networks provided by the service at configuration stage.

1.9 White paths

Many classes from the library accepts *white_paths* parameter, an iterable of white paths.

If path is in the list all checks are skipped.

White list is useful for system routes like health checks and monitoring.

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX