

ZABBIX

zabbix_utils – the official
Python library for Zabbix
API

Andrey Biba
Integration Team Lead



ZABBIX

What should you know about the library?



Official support

The Zabbix Integration team officially supports a Python library.



Zabbix Get

The library offers the functionality of the 'zabbix_get' utility.

ZABBIX

What should you know about the library?



Zabbix Get

The library offers the functionality of the 'zabbix_get' utility.



Zabbix Sender

The library provides the same functionality as the 'Zabbix_sender' utility.

ZABBIX

What should you know about the library?



Zabbix Sender

The library provides the same functionality as the 'Zabbix_sender' utility.



Zabbix API

The library includes all the necessary classes to interact with the Zabbix API.

ZABBIX

What should you know about the library?



Zabbix API

The library includes all the necessary classes to interact with the Zabbix API.



Own Tools

The library is being used internally within the team and the company.

ZABBIX

What should you know about the library?



Own Tools

The library is being used internally within the team and the company.



Compatibility

The library was developed by incorporating the most effective practices from other community libraries and ensuring maximum compatibility.

ZABBIX

What should you know about the library?



Compatibility

The library was developed by incorporating the most effective practices from other community libraries and ensuring maximum compatibility.

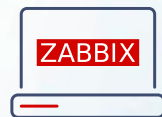


Zabbix 5.0+

The minimum tested and supported version of Zabbix is 5.0, however, it is designed to function on previous versions due to its universal code style.

ZABBIX

What should you know about the library?



Zabbix 5.0+

The minimum tested and supported version of Zabbix is 5.0, however, it is designed to function on previous versions due to its universal code style.



Python 3.8+

The minimum required version of Python is 3.8.

ZABBIX

What should you know about the library?



Python 3.8+

The minimum required version of Python is 3.8.

The background of the image is a dark, blurred screenshot of a code editor. It shows various lines of code with different colors (yellow, blue, purple) used for syntax highlighting. The text is out of focus, but some words like 'return', 'if', and 'else' are faintly visible. The overall aesthetic is that of a developer's workspace.

async

ZABBIX

async

Improved Performance

Zabbix async python library allows tasks to run concurrently, significantly improving performance by utilizing system resources more efficiently

Resource Utilization

Asynchronous programming with Zabbix async python library allows applications to make better use of available resources.

Concurrency and Parallelism

Zabbix async python library facilitates both concurrency and parallelism, enabling developers to design applications that fully utilize multi-core processors

What is

ZABBIX

API

?

The Zabbix API is an HTTP-based API(Application Programming Interface), and it is shipped as a part of the web frontend.

ZABBIX API

The Zabbix API allows you to programmatically retrieve and modify configuration of Zabbix and provides access to historical data.

ZABBIX API

ZABBIX

API

To make request to API send HTTP **POST** requests to the `api_jsonrpc.php`

```
curl --request POST \  
  --url 'https://example.com/zabbix/api_jsonrpc.php' \  
  --header 'Content-Type: application/json-rpc' \  
  --data '{"jsonrpc":"2.0","method":"apiinfo.version","params":{},"id":1}'
```

```
{  
  "jsonrpc": "2.0",  
  "result": "6.4.0",  
  "id": 1  
}
```

hostprototype.delete

host.update

problem.get

ZABBIX

configuration.import

API

history.get

drule.create

graph.get

service.create

ZABBIX

How to use Zabbix python
library in your projects?

ZABBIX

How to use Zabbix python library in your projects?

External integrations

If the default Zabbix webhook functionality is insufficient or if your service employs a custom protocol, you can utilize **zabbix_utils** to develop a middleware daemon.



ZABBIX

How to use Zabbix python
library in your projects?

ZABBIX

How to use Zabbix python
library in your projects?



Custom monitoring solutions

Leveraging the Zabbix python library in scripts can significantly broaden Zabbix monitoring capabilities, offering nearly limitless possibilities for customization and integration.

ZABBIX

How to use Zabbix python library in your projects?

ZABBIX

How to use Zabbix python
library in your projects?

ZABBIX

How to use Zabbix python
library in your projects?

ZABBIX

How to use Zabbix python library in your projects?

Zabbix statistic gathering

With `zabbix_utils`, you have the capability to collect various Zabbix statistics data such as new values per second (NVPS), the number of hosts, the quantity of templates, and other relevant metrics.



ZABBIX

**How to use Zabbix python
library in your projects?**

ZABBIX

**How to use Zabbix python
library in your projects?**



ZABBIX

Inventory export

Python library can be used to export Zabbix inventory data, such as historical item records, trends, existing host configurations, and other valuable information.

How to use Zabbix python library in your projects?

ZABBIX

How to use Zabbix python
library in your projects?

ZABBIX

How to use Zabbix python
library in your projects?

ZABBIX

How to use Zabbix python library in your projects?

Zabbix automation

The Zabbix API can be employed to manage different facets of Zabbix operations, including maintenance periods, user and user role management, template exporting/importing, and additional functionalities.



Library statistics



Total downloads

6904

Last month

2346

Last week

788

Last day

37

Library statistics



Published

5 months ago

Releases since published

6

Project Forks

3





How to start using zabbix_utils?

ZABBIX



How to start using zabbix_utils?

The screenshot shows a web browser window displaying the GitHub repository page for `python-zabbix-utils`. The browser's address bar shows the URL `https://github.com/zabbix/python-zabbix-utils`. The repository page includes the following elements:

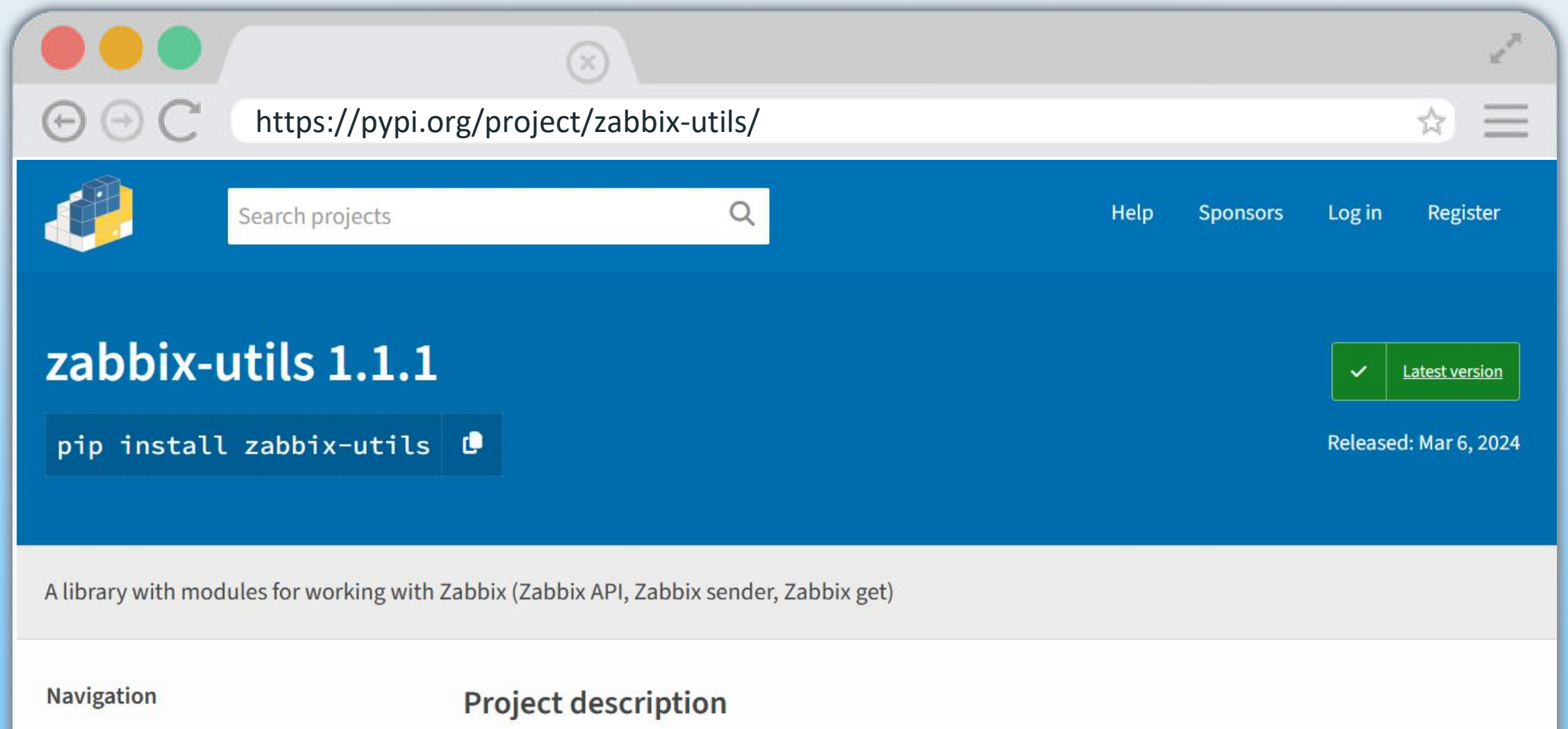
- Repository Header:** The repository name `python-zabbix-utils` is displayed as a public repository. It has 7 watchers, 3 forks, and 38 stars.
- Navigation:** The `main` branch is selected. There are buttons for "Go to file", a plus sign, and a green "Code" button.
- Recent Activity:** A table of recent commits is shown:

Author	Commit Message	Time Ago
aiantsen	Release v1.1.1	2 weeks ago
	removed external requirements	2 weeks ago
	changed the format of Sender respons...	2 months ago
- About Section:** The repository is described as a "Python library 'zabbix-utils'". It includes links for "Readme", "MIT license", and "Activity".

ZABBIX



How to start using zabbix_utils?



The screenshot shows a web browser window with the URL `https://pypi.org/project/zabbix-utils/`. The page features a blue header with a search bar, navigation links for Help, Sponsors, Log in, and Register, and a main content area with the project name `zabbix-utils 1.1.1`, a code block for installation, and a description of the library.

Search projects

Help Sponsors Log in Register

zabbix-utils 1.1.1

✓ Latest version

```
pip install zabbix-utils
```

Released: Mar 6, 2024

A library with modules for working with Zabbix (Zabbix API, Zabbix sender, Zabbix get)

Navigation Project description

ZABBIX



How to start using zabbix_utils?

The screenshot shows a web browser window with the following elements:

- Address Bar:** <https://blog.zabbix.com/python-zabbix-utils/27056/>
- Page Header:** **ZABBIX** BLOG, a search icon, and a language dropdown menu set to "English".
- Main Content:** A large banner for the article "Introducing zabbix_utils - The official...". The banner features the ZABBIX logo, the word "BLOG", and a Python logo icon on a blue background with a network diagram.

```
1  from zabbix_utils import ZabbixAPI
2
3  api = ZabbixAPI(url="127.0.0.1")
4  api.login(user="User", password="zabbix")
5
6  users = api.user.get(
7  |     output=['userid', 'name']
8  )
9
10 for user in users:
11 |     print(user['name'])
12
13 api.logout()
```

**Thank you for
your attention**