

◆ Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



Securing Django REST APIs with JWT Authentication using Simple-JWT: A Step-by-Step Guide with Project



Mehedi Khan · [Follow](#)

Published in [Django Unleashed](#) · 4 min read · Feb 19, 2024

👏 95

🗨 1



...



Django REST APIs with JWT Authentication using Simple-JWT

JWT (JSON Web Tokens) authentication in Django Rest Framework (DRF) with Simple-JWT is a popular choice for securing APIs. Simple-JWT is a library that provides straightforward JWT authentication for Django applications. Let's walk through setting up JWT authentication in Django Rest Framework with Simple-JWT, including a real project example.

Step 1: Install Required Packages

Install Django, Django Rest Framework and Simple-JWT:

```
pip install django djangorestframework djangorestframework_simplejwt
```

Step 2: Setting Up a Django Project

If you haven't already, create a new Django project:

```
django-admin startproject <myproject>
cd myproject
```

Replace `<myproject>` with the desired name of your project. This command will generate a new directory with the specified project name containing the necessary files and directories for your project. Here's how our project might look:

```
myproject/
└── myproject/
    ├── __init__.py
    ├── settings.py
    ├── urls.py
    ├── wsgi.py
    └── asgi.py
└── manage.py
```

Step 3: Setting Up a Django App

Create a new Django app within your project:

```
python manage.py startapp <myapp>
```

Replace `<myapp>` with the desired name of your app. Here's how our project might look after create app:

```
myproject/
└── myproject/
    ├── __init__.py
    ├── settings.py
    ├── urls.py
    ├── wsgi.py
    └── asgi.py
└── myapp/
    ├── __init__.py
    ├── admin.py
    ├── apps.py
    ├── migrations/
    │   └── __init__.py
    ├── models.py
    ├── tests.py
    ├── urls.py
    └── views.py
└── manage.py
```

Step 4: Configuring Django Rest Framework

Add ‘rest_framework’ and ‘rest_framework_simplejwt’ to `INSTALLED_APPS` in your project’s `settings.py` file:

```
INSTALLED_APPS = [
    # All default apps go here

    'rest_framework',
    'rest_framework_simplejwt',

    'myapp', # Replace 'myapp' with the name of your app
]
```

Step 5: Configuring JWT Settings

In your settings.py file, configure Simple-JWT settings. Here's a basic configuration:

```
# settings.py
from datetime import timedelta # import this library top of the settings.py file

# put on your settings.py file below INSTALLED_APPS
REST_FRAMEWORK = {
    'DEFAULT_PERMISSION_CLASSES': (
        'rest_framework.permissions.IsAuthenticated',
    ),
    'DEFAULT_AUTHENTICATION_CLASSES': (
        'rest_framework_simplejwt.authentication.JWTAuthentication',
    ),
}

SIMPLE_JWT = {
```

[Open in app ↗](#)

Medium



Search



Write



}



Step 6: Token Endpoints

Include Simple-JWT token endpoints in your project's urls.py:

```
from rest_framework_simplejwt.views import TokenObtainPairView, TokenRefreshView
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('api/token/', TokenObtainPairView.as_view(), name='token_obtain_pair'),
    path('api/token/refresh/', TokenRefreshView.as_view(), name='token_refresh')
    path('', include('myapp.urls')), # include your app urls.py here
]
```

Step 7: Create API Views

Create your API views in `myapp/views.py` within your app. These are endpoints that will require authentication.

```
from rest_framework.views import APIView
from rest_framework.response import Response
from rest_framework.permissions import IsAuthenticated
from rest_framework_simplejwt.authentication import JWTAuthentication

class Home(APIView):
    authentication_classes = [JWTAuthentication]
    permission_classes = [IsAuthenticated]

    def get(self, request):
        content = {'message': 'Hello, World!'}
        return Response(content)
```

Step 8: URL Configuration

First create myapp/urls.py file in your app. Then configure your URLs in myapp/urls.py within your app:

```
touch myapp/urls.py
```

```
from django.urls import path
from .views import Home

urlpatterns = [
    path('', Home.as_view()),
]
```

Step 9: Running Migrations

Finally, run the following command to apply any database migrations for your app:

```
python manage.py migrate
```

Step 10: Testing

For Creating superuser for testing and run surver following commands below:

```
python manage.py createsuperuser
python manage.py runserver
```

Now, you can test your API endpoints using tools like Postman. Obtain an access token by making a POST request to (<http://127.0.0.1:8000/api/token/>) with valid user credentials like below screenshot.

The screenshot shows a Postman interface with the following details:

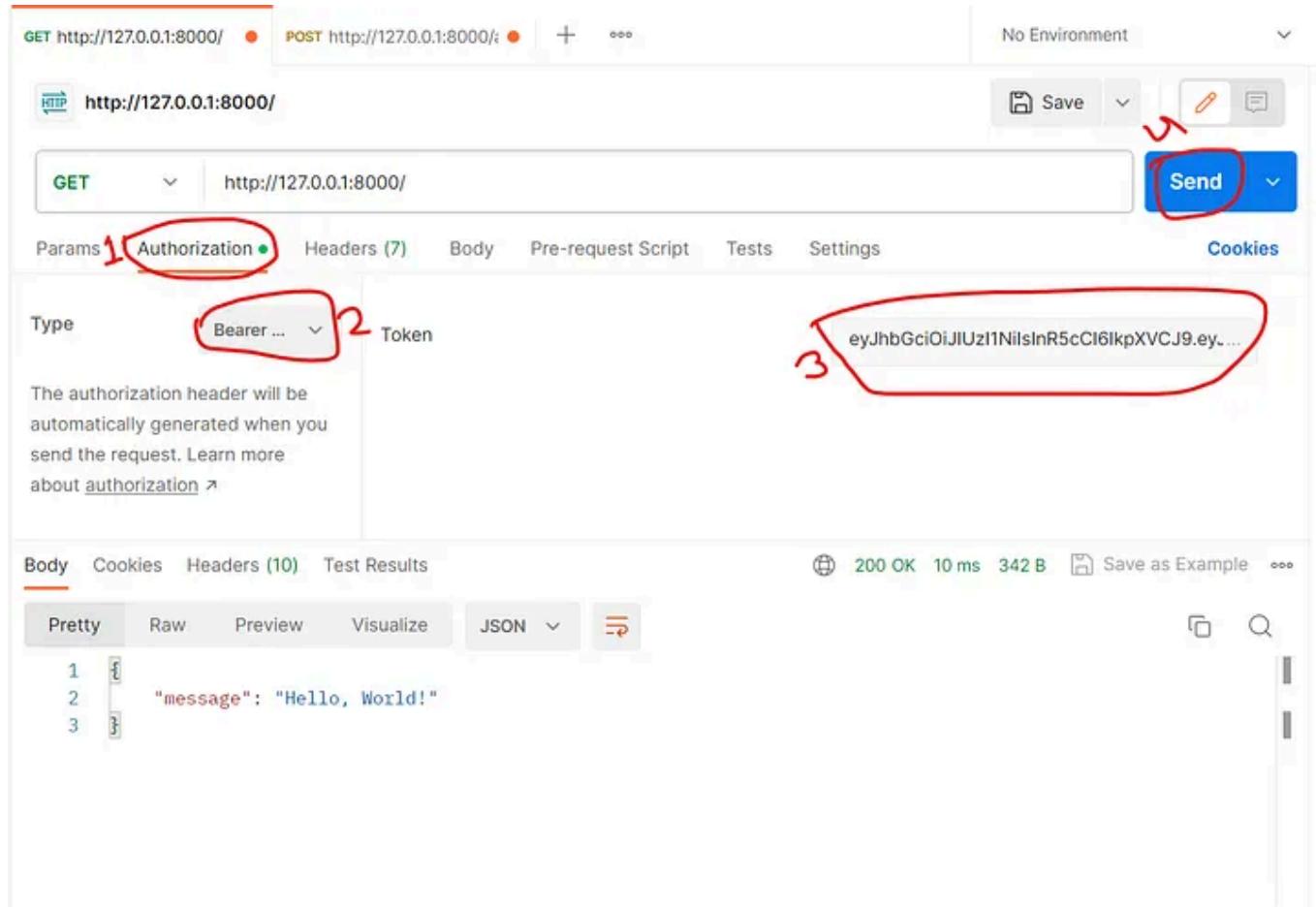
- Request Method:** POST
- URL:** http://127.0.0.1:8000/api/token/
- Body (JSON):**

```
1 {
2   "username": "admin",
3   "password": "admin123"
4 }
```
- Response Headers:**
 - 200 OK
 - 992 ms
 - 794 B
 - Save as Example
- Response Body (Pretty JSON):**

```
1 {
2   "refresh": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
            eyJ0b2tlbl90eXB1IjoicmVmcmVzaiCisImV4cCI6MTcwODQ3NDA3NSwiWF0IjoxNzA4Mzg3Njc1LCJqdGkiOiIyN2ZlNWZlYz
            k00TU0Mzk40Dg0YmE30TA4NDNhYjgwMyIsInVzZXJfaWQiOjF9.3Ifj4fVsiHf0fUj8R6WjfNM3WcjY8M_p75mihKona0",
3   "access": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
            eyJ0b2tlbl90eXB1IjoicmVmcmVzaiCisImV4cCI6MTcwODQ3NDA3NSwiWF0IjoxNzA4Mzg3Njc1LCJqdGkiOiIyN2ZlNWZlYz
            RmZTQzZjhNNWFiN2M5NmVhMDQ2NGUyIiwidXNlc19pZCI6MX0.5cYvxxw1YGMzFN9oNcXh5AZFFB4woV2oaKSGgtFIqM5A"
4 }
```

Django JWT Authentication using Simple-JWT

Then, use this **access token** to make requests to authenticated endpoints like(<http://127.0.0.1:8000/>) Type need to change as **Bearer Token**.



Step 11: Conclusion

You've now set up JWT authentication in Django Rest Framework using Simple-JWT. This provides a secure way to authenticate users and access protected resources in your Django API. Make sure to handle token expiration and refreshing appropriately in your client application.

For More details you may follow this documentation [Django simple-JWT](#).

[GitHub - MehediMK/Django-REST-APIs-with-JWT-Authentication: Enhance the security of your Django...](#)

Enhance the security of your Django REST APIs with JSON Web Token (JWT) authentication using the Simple-JWT library...

github.com

Thank you for reading. If you find something wrong or better ways to do it, let me know in the comments below.

If you like the post, hit the  button below so that others may find it useful. You can [follow me on GitHub](#), [daily.dev community](#) and [connect with me on LinkedIn](#).

More Libraries:



Mehedi Khan

Django

[View list](#) 27 stories





Mehedi Khan

Python

[View list](#) 12 stories



Django Rest Framework

Django Rest Api

Django Simple Jwt

API

Django Jwt



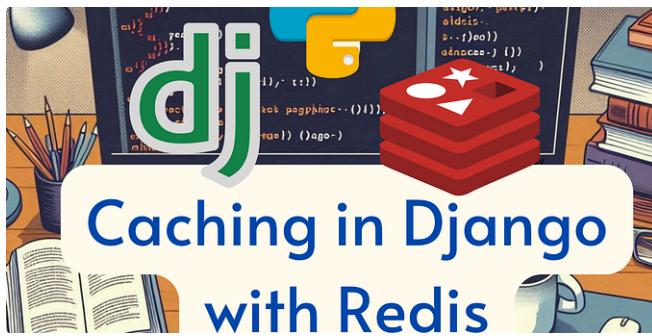
Written by Mehedi Khan

208 Followers · Writer for Django Unleashed

[Follow](#)


I'm a Software engineer. I'm comfortable with Python, Django, and Full-stack Web Development. Follow To Support Me On Medium

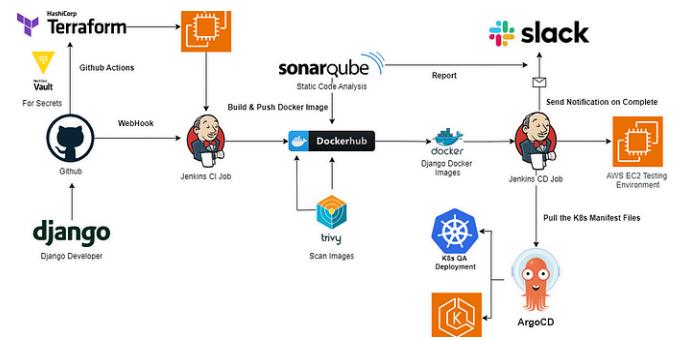
More from Mehedi Khan and Django Unleashed



 Mehedi Khan in Django Unleashed

Caching in Django with Redis: A Step-by-Step Guide

Caching in Django is an essential technique for improving the performance of web...



 Joel Wembo in Django Unleashed

Technical Guide: End-to-End CI/CD DevOps with Jenkins, Docker,...

Building an end-to-end CI/CD pipeline for Django applications using Jenkins, Docker,...

Jun 10

88



...



Apr 12

746

16



...



Julio Cesar Siles Perez in Django Unleashed

Django Project Structure: A Comprehensive Guide

In web development, crafting a seamless digital experience goes beyond writing code...

Aug 6, 2023

601

9



...

Mehedi Khan in Django Unleashed

Complete Tutorial: Set-up PostgreSQL Database with Djang...

Set-up PostgreSQL database with Django application involves several steps. Here are...

Apr 22

103



...

[See all from Mehedi Khan](#)
[See all from Django Unleashed](#)

Recommended from Medium



Sai Krupa Reddy Surarapu in The Pythoneers

Building a Django REST API with File Upload and User...

In this article, we'll walk through the process of building a Django REST API that supports...

★ Aug 19 · 97 saves · 1 comment



...

Engr Muhammad Tanveer sultan

When and How to Use Django AbstractUser and...

When working with Django, managing custom user models is a common requirement,...

★ Aug 17 · 53 saves



...

Lists



Coding & Development

11 stories · 818 saves



data science and AI

40 stories · 247 saves



Company Offsite Reading List

8 stories · 150 saves

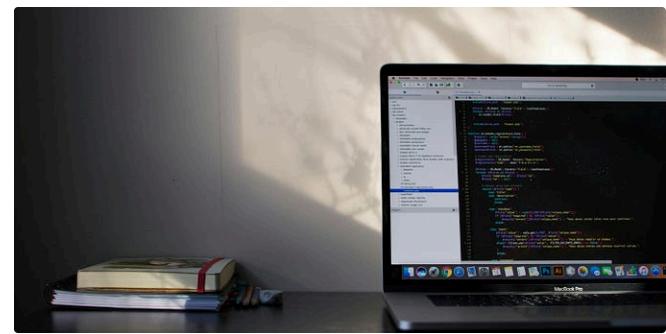


Natural Language Processing

1716 stories · 1288 saves



Gajanan Rajput



Michal Drózdż

Top 7 Django Commands Every Developer Needs to Know

Dive into the essential Django commands every developer should master. From...

Apr 2 509 7



...



 Mathur Danduprolu

How to Customize Django Admin Interface ?

The Django admin interface is a powerful tool that allows developers to manage their...

Jun 7 23 1



...

Django REST APIs with JWT Authentication using dj-rest-auth

How to enable JWT Authentication in your Django REST API?

Jul 4 5



...



 Andrés Álvarez Iglesias

Django AllAuth Chapter 2—How to install and configure Django...

In this chapter we'll explore the basics of the AllAuth extension: from the installation to a...

Jul 10 34



...

See more recommendations