**Laravel Sanctum**

What is Laravel Sanctum ? Laravel Sanctum provides a featherweight authentication system for SPAs (single page applications), mobile applications, and simple, token based APIs. Sanctum allows each user of your application to generate multiple API tokens for their account. These tokens may be granted abilities / scopes which specify which actions the tokens are allowed to perform..

**You have to just follow a few steps to get following web services**

**Login API**

**Details API**

**Getting Started**

**Step 1: setup database in .env file**

DB\_DATABASE=youtube

DB\_USERNAME=root

DB\_PASSWORD= redhat@123

**Step 2:Install Laravel Sanctum.**

composer require laravel/sanctum

**Step 3:Publish the Sanctum configuration and migration files .**

php artisan vendor:publish --provider="Laravel\Sanctum\SanctumServiceProvider"

**Step 4:Run your database migrations.**

php artisan migrate

**Step 5:Add the Sanctum's middleware.**

../app/Http/Kernel.php

use Laravel\Sanctum\Http\Middleware\EnsureFrontendRequestsAreStateful;

...

protected $middlewareGroups = [

...

'api' => [

EnsureFrontendRequestsAreStateful::class,

'throttle:60,1',

\Illuminate\Routing\Middleware\SubstituteBindings::class,

],

];

...

],

**Step 6:To use tokens for users.**

use Laravel\Sanctum\HasApiTokens;

class User extends Authenticatable

{

use HasApiTokens, Notifiable;

}

**Step 7:Let's create the seeder for the User model**

php artisan make:seeder UsersTableSeeder

**Step 8:Now let's insert as record**

use Illuminate\Support\Facades\DB;

use Illuminate\Support\Facades\Hash;

...

...

DB::table('users')->insert([

'name' => 'John Doe',

'email' => 'john@doe.com',

'password' => Hash::make('password')

]);

**Step 9:To seed users table with user**

php artisan db:seed --class=UsersTableSeeder

**Step 10: create a controller nad /login route in the routes/api.php file:**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\User;

use Illuminate\Support\Facades\Hash;

class UserController extends Controller

{

//

function index(Request $request)

{

$user= User::where('email', $request->email)->first();

// print\_r($data);

if (!$user || !Hash::check($request->password, $user->password)) {

return response([

'message' => ['These credentials do not match our records.']

], 404);

}

$token = $user->createToken('my-app-token')->plainTextToken;

$response = [

'user' => $user,

'token' => $token

];

return response($response, 201);

}

}

**Step 11: Test with postman, Result will be below**

{

"user": {

"id": 1,

"name": "John Doe",

"email": "john@doe.com",

"email\_verified\_at": null,

"created\_at": null,

"updated\_at": null

},

"token": "AbQzDgXa..."

}

**Step 11: Make Details API or any other with secure route**

Route::group(['middleware' => 'auth:sanctum'], function(){

//All secure URL's

});

Route::post("login",[UserController::class,'index']);