Angular TS – Security – Authentication – JWT Token

[**✅ Purpose:** 1](#_Toc205373086)

[**📁 Project Structure** 1](#_Toc205373087)

[**🧠 Prerequisites:** 2](#_Toc205373088)

[**🔐 1. auth.interceptor.ts** 2](#_Toc205373089)

[**🧰 2. auth.service.ts** 2](#_Toc205373090)

[**🧑‍💻 3. login.component.ts** 3](#_Toc205373091)

[**🏠 4. home.component.ts** 3](#_Toc205373092)

[**🌐 5. app.routes.ts** 4](#_Toc205373093)

[**📦 6. app.module.ts** 4](#_Toc205373094)

[**🔁 7. app.component.ts** 5](#_Toc205373095)

[**✅ To Run:** 5](#_Toc205373096)

**✅ Purpose:**

Implement an HttpInterceptor to attach a JWT token to each outgoing HTTP request, which simulates securing API calls from the Angular app.

**📁 Project Structure**

src/

│

├── app/

│ ├── interceptors/

│ │ └── auth.interceptor.ts ✅

│ ├── services/

│ │ └── auth.service.ts ✅

│ ├── components/

│ │ ├── login/

│ │ │ ├── login.component.ts ✅

│ │ │ ├── login.component.html

│ │ │ └── login.component.css

│ │ └── home/

│ │ ├── home.component.ts ✅

│ │ ├── home.component.html

│ │ └── home.component.css

│ ├── app.component.ts

│ ├── app.module.ts ✅

│ └── app.routes.ts ✅

└── main.ts

**🧠 Prerequisites:**

* Angular 18 CLI installed
* Bootstrap for styling (optional)
* A fake backend or real API returning a JWT token for login (we simulate this)

**🔐 1. auth.interceptor.ts**

// src/app/interceptors/auth.interceptor.ts

import { Injectable } from '@angular/core';

import { HttpInterceptor, HttpRequest, HttpHandler, HttpEvent } from '@angular/common/http';

import { Observable } from 'rxjs';

import { AuthService } from '../services/auth.service';

@Injectable()

export class AuthInterceptor implements HttpInterceptor {

constructor(private authService: AuthService) {}

intercept(req: HttpRequest<any>, next: HttpHandler): Observable<HttpEvent<any>> {

const authToken = this.authService.getToken();

if (authToken) {

const authReq = req.clone({

setHeaders: {

Authorization: `Bearer ${authToken}`

}

});

return next.handle(authReq);

}

return next.handle(req);

}

}

**🧰 2. auth.service.ts**

// src/app/services/auth.service.ts

import { Injectable } from '@angular/core';

@Injectable({

providedIn: 'root'

})

export class AuthService {

private tokenKey = 'jwtToken';

login(username: string, password: string): boolean {

// Simulate login

if (username === 'admin' && password === 'admin') {

localStorage.setItem(this.tokenKey, 'mock-jwt-token-123456');

return true;

}

return false;

}

logout(): void {

localStorage.removeItem(this.tokenKey);

}

getToken(): string | null {

return localStorage.getItem(this.tokenKey);

}

isLoggedIn(): boolean {

return !!this.getToken();

}

}

**🧑‍💻 3. login.component.ts**

// src/app/components/login/login.component.ts

import { Component } from '@angular/core';

import { Router } from '@angular/router';

import { AuthService } from '../../services/auth.service';

@Component({

selector: 'app-login',

template: `

<div class="container mt-5">

<h3>Login</h3>

<input [(ngModel)]="username" placeholder="Username" class="form-control" />

<input [(ngModel)]="password" type="password" placeholder="Password" class="form-control mt-2" />

<button class="btn btn-primary mt-2" (click)="login()">Login</button>

</div>

`

})

export class LoginComponent {

username = '';

password = '';

constructor(private authService: AuthService, private router: Router) {}

login() {

const success = this.authService.login(this.username, this.password);

if (success) {

this.router.navigate(['/home']);

} else {

alert('Invalid credentials');

}

}

}

**🏠 4. home.component.ts**

// src/app/components/home/home.component.ts

import { Component } from '@angular/core';

import { HttpClient } from '@angular/common/http';

@Component({

selector: 'app-home',

template: `

<div class="container mt-5">

<h2>Welcome to the Home Page</h2>

<button class="btn btn-info mt-3" (click)="getData()">Call Secured API</button>

<div \*ngIf="apiData" class="alert alert-success mt-3">

{{ apiData }}

</div>

</div>

`

})

export class HomeComponent {

apiData: string = '';

constructor(private http: HttpClient) {}

getData() {

this.http.get('https://jsonplaceholder.typicode.com/posts/1', { responseType: 'text' })

.subscribe(data => this.apiData = data);

}

}

**🌐 5. app.routes.ts**

// src/app/app.routes.ts

import { Routes } from '@angular/router';

import { LoginComponent } from './components/login/login.component';

import { HomeComponent } from './components/home/home.component';

export const appRoutes: Routes = [

{ path: '', component: LoginComponent },

{ path: 'home', component: HomeComponent }

];

**📦 6. app.module.ts**

// src/app/app.module.ts

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { AppComponent } from './app.component';

import { LoginComponent } from './components/login/login.component';

import { HomeComponent } from './components/home/home.component';

import { HttpClientModule, HTTP\_INTERCEPTORS } from '@angular/common/http';

import { FormsModule } from '@angular/forms';

import { RouterModule } from '@angular/router';

import { appRoutes } from './app.routes';

import { AuthInterceptor } from './interceptors/auth.interceptor';

@NgModule({

declarations: [AppComponent, LoginComponent, HomeComponent],

imports: [BrowserModule, FormsModule, HttpClientModule, RouterModule.forRoot(appRoutes)],

providers: [

{ provide: HTTP\_INTERCEPTORS, useClass: AuthInterceptor, multi: true }

],

bootstrap: [AppComponent]

})

export class AppModule {}

**🔁 7. app.component.ts**

// src/app/app.component.ts

import { Component } from '@angular/core';

@Component({

selector: 'app-root',

template: `<router-outlet></router-outlet>`

})

export class AppComponent {}

**✅ To Run:**

1. Install dependencies:

npm install

1. Run the app:

ng serve

1. Open in browser: http://localhost:4200

* Login with admin / admin
* You’ll be navigated to the home page.
* Click the button to make a simulated secured API call with the JWT token in the headers.