

"No provider for AuthGuard!" using CanActivate in Angular 2



EDIT : Obviously this is outdated, now you provide your guard at the `providers` array in an `NgModule`. Watch other answers or official documentation for more information.

27



- bootstrapping on a component is outdated
- `provideRouter()` is outdated as well



5

I'm trying to setup Authentication in my project, using a login and AuthGuard from the Angular2 guide : <https://angular.io/docs/ts/latest/guide/router.html>

I'm using the release : "@angular/router": "3.0.0-beta.1".

I'll try to explain as much as possible, feel free to tell me if you need more details.

I have my **main.ts** file which bootstraps the app with the following code :

```
bootstrap(MasterComponent, [  
  APP_ROUTER_PROVIDERS,  
  MenuService  
)  
.catch(err => console.error(err));
```

I load the MasterComponent, which loads a Header containing buttons that allow me to navigate through my app and it also contains my main for now.

I'm following the guide to make my app work the same way, with the following **app.routes.ts** :

```
export const routes: RouterConfig = [
```

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google



```
    AUTH_PROVIDERS  
  ];
```

And the **login.routes.ts** from the guide, which defines my AuthGuard :

```
export const LoginRoutes = [  
  { path: 'login', component: LoginComponent }  
];  
  
export const AUTH_PROVIDERS = [AuthGuard, AuthService];
```

my Master component has its own route definition, which also contains the guard I'm trying to setup. **master.routes.ts** :

```
export const MasterRoutes : RouterConfig = [  
  { path: '', redirectTo: '/accueil', pathMatch: 'full' },  
  
  {  
    path: 'accueil',  
    component: AccueilComponent  
  },  
  
  { path: 'dashboard', component: DashboardComponent, canActivate: [AuthGuard] },  
];
```

And I'm using the same files as the guide, which are **auth.guard.ts**, **auth.service.ts**, **login.component.ts** and **login.routes.ts**.

In my **header.component.ts** file, when I try to access any routes, it's working just fine, but when I try to access the guarded path (/dashboard), I get the **No provider for AuthGuard!** error.


I saw the recent post with the same issue as mine ([NoProviderError using CanActivate in Angular 2](#)), but to me the guard is bootstrapped correctly up to the **main.ts** file, so my router should know which routes should be provided with the AuthGuard right ?


Any help or advice would be greatly appreciated. Thanks !

[authentication](#) [angular](#)[angular2-routing](#)

Join **Stack Overflow** to learn, share knowledge, and build your career.

[Email Sign Up](#)[OR SIGN IN WITH](#) [Google](#)[Facebook](#)

Did you import AuthGuard into master.routes.ts and login.routes.ts ? – [Günter Zöchbauer](#) Jul 5 '16 at 12:40 

Yes, I did import them correctly. I did not mention it, but I manually navigate to the route because I wanted to link on a <button> using **this.router.navigate(['/dashboard'])**; – [Alex Beugnet](#) Jul 5 '16 at 12:42 

1 See stackoverflow.com/questions/38144547/... – [Günter Zöchbauer](#) Jul 5 '16 at 12:51

1 Weird thing is that my company's firewall is actually blocking the plunkr links, so I may need a bit of time... I'll edit this comment with a working plunkr hopefully – [Alex Beugnet](#) Jul 5 '16 at 13:13

1 You're welcome. Glad to hear you figured it out. – [Günter Zöchbauer](#) Jul 5 '16 at 14:14

11 Answers

▲ I had this same issue after going through the Route Guards section of Routing and Authorization tutorial on the Angular website <https://angular.io/docs/ts/latest/guide/router.html>, it is section 5.

58

▼ I am adding AuthGuard to one of my main routes and not to child routes like the tutorial shows.

I fixed it by added AuthGuard to my list of providers in my app.module.ts file, so that file now looks like this:

```
import { AppComponent } from './app.component';
import { AppRoutingModuleModule } from './app-routing.module';
import { AuthGuard } from './auth-guard.service';

import { AnotherPageComponent } from './another-page/another-page.component';
import { LoginPageComponent } from './login-page/login-page.component';

@NgModule({
  imports: [
    BrowserModule,
    FormsModule,
    JsonpModule,
    AppRoutingModuleModule,
    HttpClientModule
  ],
  declarations: [
    AppComponent,
    AnotherPageComponent,
    LoginPageComponent
  ],
  providers: [
    AuthGuard
  ],
  bootstrap: [AppComponent]
})
export class AppModule {}
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH

 Google

Facebook 

```
providers: [AuthGuard],
bootstrap: [AppComponent]
}))

export class AppModule { }
```

I have gone back through the tutorial and in their app.module.ts file, they do not add AuthGuard to the providers, not sure why.

answered Oct 21 '16 at 15:19



[Dan Stirling-Talbert](#)

818 7 9

You saved my day! – [Anand Vaidya](#) Apr 12 at 7:56



Also, don't fall into the trap of using a literal for the guard class inside your routing configuration, just because some blog articles do:

15

```
{ path: 'whatever', component: WhateverComponent, canActivate: ['WhateverGuard'] }
```



is not going to work (No provider for...), instead, use the class directly:

```
{ path: 'whatever', component: WhateverComponent, canActivate: [WhateverGuard] }
```

Another hint, when lazy loading components, the guard is applied in the routing configuration of the parent component, not in the routing configuration of the lazy loaded component.

answered Oct 22 '16 at 7:27



[Thorsten Westheider](#)

4,246 10 39 76

I put WhateverGuard into single quotes, and it showed me No provider .. that was so annoying – [Redplane](#) Jun 7 '17 at 15:50

... ..

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google



11

import this service before bootstrap runs.

```
import { bootstrap } from '@angular/platform-browser-dynamic';

import { AppComponent } from './app.component';
import { AuthGuard } from './shared/auth.service';

bootstrap(AppComponent, [
  appRouterProviders,
  AuthGuard
]);
```

Angular 2 team did not mention this in main router docs, and it took couple of hours for me to figure it out.

answered Aug 5 '16 at 17:13



[Roman Gusiev](#)

168 10



7

The answer is further down in the tutorial. See the file listings in the "Add the LoginComponent" topic under the "Component-less route:..." section in "Milestone 5: Route Guards". It shows AuthGuard and AuthService being imported and added to the providers array in login-routing.module.ts, and then that module being imported into app.module.ts.



login-routing.module.ts

```
...
import { AuthGuard }          from './auth-guard.service';
import { AuthService }        from './auth.service';
...
@NgModule({
  ...
  providers: [
    AuthGuard,
    AuthService
  ]
})
export class LoginComponentModule {}
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



```
import { LoginRoutingModule } from './login-routing.module';

@NgModule({
  imports: [
    ...
    LoginRoutingModule,
    ...
  ],
  ...
  providers: [
    DialogService
  ],
  ...
})
```

edited Jul 27 '17 at 21:33

answered Jul 27 '17 at 21:18



unqualified

106 1 5

Thanks for this, I was stuck on the tutorial and this helped a lot. I thought that they should be in the appModule, but they aren't in the final version of the code and I couldn't figure out how it worked without them. – [wildrhombus](#) Feb 22 '18 at 3:17



Actually, it was only a typo in an import...

5

I was typing



```
import { AuthGuard } from '../Authentication/auth.guard';
```

instead of



```
import { AuthGuard } from '../authentication/auth.guard';
```

making it not working but at the same time not displaying me any error...

Join **Stack Overflow** to learn, share knowledge, and build your career.

[Email Sign Up](#)

OR SIGN IN WITH

[Google](#)[Facebook](#)

4 Thats why it is recommended to use typescript aware editor like Visual studio code (its the best). – [Abdul Mannan](#) Oct 12 '16 at 12:44

Yes, I've been using VS 2015 but clearly, it's not as good as Visual Code. – [Alex Beugnet](#) Dec 1 '16 at 18:16

I faced same kind of issue, finally found the issue is the import path is case sensitive – [Mahesh](#) Mar 9 '18 at 13:27 

I encountered this issue when I was following a tutorial. I tried most of the answer here but not getting any success. Then I tried the silly way like putting the AuthGuard before the other services in the provider and it works.

3

```
// app.module.ts
```

```
..
providers: [
  AuthGuard,
  UserService,
  ProjectService
]
```

answered Feb 27 '17 at 6:13



[christian crisologo](#)

31 1

Since you got the solution as it was due to syntax issue. I just wanted to share this info.

we need to provide the AuthGaudService as provider in only that module that correspond to respective route. No need to provide in main module or root module as main module will automatically load all the given sub module.This helps in keeping the code modular and encapsulated.

for example, suppose we have below scenario

```
1. we have module m1
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Below is just prototype, not the actual code for understanding purpose

```
//m1.ts
import {AuthGuard} from './auth.gaurd.service'
import {m1r} from './m1r'
@NgModule(
  imports: [m1r],
  providers: [AuthGuard]
)
export class m1{
}

//m1r.ts
import {AuthGuard} from './auth.gaurd.service'
const authRoute = [
  {path: '/r1', component: 'authComponent', canActivate: [AuthGuard]},
  {path: '/r2', component: 'other'}
]
export authRoute

//main.module.ts
import {m1} from ''
import {mainComponent} from ''
@NgModule({
  imports: [m1],
  bootstrap: [mainComponent]
})
export class MainModule{}
```

answered Sep 22 '16 at 7:54



Vikash

333 5 14



1

```
import { Injectable } from '@angular/core';
import { Router, CanActivate, ActivatedRouteSnapshot, RouterStateSnapshot } from
'@angular/router';
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google




```

    if (localStorage.getItem('currentUser')) {
      // logged in so return true
      return true;
    }

    // not logged in so redirect to login page with the return url
    this.router.navigate(['/login'], { queryParams: { returnUrl: state.url }});
    return false;
  }
}

```

answered Jan 28 '17 at 9:01

[Shailish kala](#)

706 9 10

▲ Importing both `HttpModule` and `HttpClientModule` helped me.

1

```

import { HttpClientModule } from '@angular/common/http';
import { HttpModule } from '@angular/http';

```

edited Dec 15 '17 at 16:07

[Liam](#)

17k 16 80 133

answered Dec 15 '17 at 6:47

[Ramya](#)

70 1 1 9

▲ you can try import AuthGuard in provider of that module and then import it in the routing component-routing.module.ts file also

0

```

@NgModule({
  providers: [
    AuthGuard
  ],})

```

answered Nov 8 '16 at 7:11

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook



Try to add

0



```
@Injectable({  
  providedIn: 'root'  
})
```

no need to add to module provider.

answered Dec 10 '18 at 15:37

[Ryan Huang](#)

250 2 6

Join **Stack Overflow** to learn, share knowledge, and build your career.

[Email Sign Up](#)

OR SIGN IN WITH



Google

[Facebook](#)