

# Spring Boot Token based Authentication with Spring Security & JWT

Last modified: May 1, 2023 (<https://www.bezkoder.com/spring-boot-jwt-authentication/>)  bezkoder (<https://www.bezkoder.com/author/bezkoder/>)  
 Security (<https://www.bezkoder.com/category/security/>), Spring (<https://www.bezkoder.com/category/spring/>)

In this tutorial, we're gonna build a Spring Boot Application that supports Token based Authentication with JWT. You'll know:

- Appropriate Flow for User Signup & User Login with JWT Authentication
- Spring Boot Application Architecture with Spring Security
- How to configure Spring Security to work with JWT
- How to define Data Models and association for Authentication and Authorization
- Way to use Spring Data JPA to interact with PostgreSQL/MySQL Database

Lots of interesting things ahead, let's explore.

- Spring Boot Refresh Token with JW  
(<https://www.bezkoder.com/spring-boot-refresh-token-jwt/>)
- Spring Boot, Spring Data JPA – Buil  
(<https://www.bezkoder.com/spring-boot-rud-api-example/>)
- Spring Boot File upload example with Multipart File  
(<https://www.bezkoder.com/spring-boot-file-upload/>)
- @RestControllerAdvice example in Spring Boot  
(<https://www.bezkoder.com/spring-boot-restcontrolleradvice/>)
- Spring Boot @ControllerAdvice & @ExceptionHandler example  
(<https://www.bezkoder.com/spring-boot-controlleradvice-exceptionhandler/>)
- @DataJpaTest example for Spring Data Repository Unit Test  
(<https://www.bezkoder.com/spring-boot-unit-test-jpa-repositoryjpatest/>)
- Spring Boot Unit Test for Rest Controller  
(<https://www.bezkoder.com/spring-boot-webmvctest/>)
- Using MongoDB: Spring Boot JWT Auth with MongoDB  
(<https://www.bezkoder.com/spring-boot-jwt-auth-mongodb/>)
- The example that uses HttpOnly Cookies  
(<https://www.bezkoder.com/spring-boot-login-example-mysql/>) instead.
- Documentation: Spring Boot Swagger 3 example  
(<https://www.bezkoder.com/spring-boot-swagger-3/>)
- Caching: Spring Boot Redis Cache example  
(<https://www.bezkoder.com/spring-boot-redis-cache-example/>)
- Spring Boot custom Validation example

- Deploy Spring Boot App on AWS – Elastic Beanstalk  
(<https://www.bezkoder.com/deploy-spring-boot-aws-eb/>)
- Docker Compose: Spring Boot and MySQL example  
(<https://www.bezkoder.com/docker-compose-spring-boot-mysql/>)



## Contents [hide]

Overview of Spring Boot JWT Authentication example  
Spring Boot Signup & Login with JWT Authentication Flow  
Spring Boot Server Architecture with Spring Security  
Technology  
Project Structure  
Setup new Spring Boot project  
Configure Spring Datasource, JPA, App properties  
Create the models  
Implement Repositories  
Configure Spring Security  
Implement UserDetails & UserDetailsService  
Filter the Requests  
Create JWT Utility class  
Handle Authentication Exception  
Define payloads for Spring RestController  
Create Spring RestAPIs Controllers  
Run & Test  
Conclusion  
Source Code  
[Further Reading](#)



# Overview of Spring Boot JWT Authentication example

We will build a Spring Boot application in that:

- User can signup new account, or login with username & password.
- By User's role (admin, moderator, user), we authorize the User to access resources

These are APIs that we need to provide:



Methods	Urls	Actions
POST	/api/auth/signup	signup new account
POST	/api/auth/signin	login an account
GET	/api/test/all	retrieve public content
GET	/api/test/user	access User's content
GET	/api/test/mod	access Moderator's content
GET	/api/test/admin	access Admin's content

The database we will use could be PostgreSQL or MySQL depending on the way we configure project dependency & datasource.

This is our Spring Boot application demo running with MySQL

## Spring Boot + JWT + MySQL: Token based Authenti...



For Fullstack, please visit:

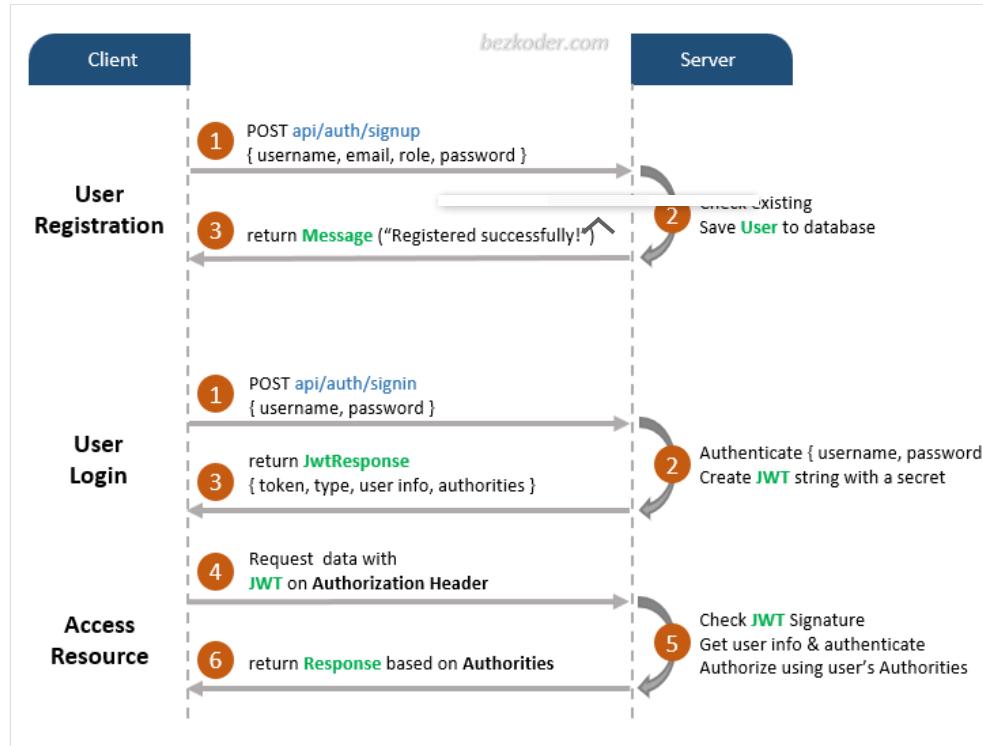
- Spring Boot + Vuejs: JWT Authentication Example  
(<https://www.bezkoder.com/spring-boot-vue-js-authentication-jwt-spring-security/>)
- Spring Boot + Angular 8: JWT Authentication Example



- Spring Boot + Angular 11: JWT Authentication Example  
(<https://www.bezkoder.com/angular-11-spring-boot-jwt-auth/>)
- Spring Boot + Angular 12: JWT Authentication example  
(<https://www.bezkoder.com/angular-12-spring-boot-jwt-auth/>)
- Spring Boot + Angular 13: JWT Authentication example  
(<https://www.bezkoder.com/angular-13-spring-boot-jwt-auth/>)
- Spring Boot + Angular 14: JWT Authentication example  
(<https://www.bezkoder.com/angular-14-spring-boot-jwt-auth/>)
- Spring Boot + React.js: JWT Authentication example  
(<https://www.bezkoder.com/spring-boot-react-jwt-auth/>)

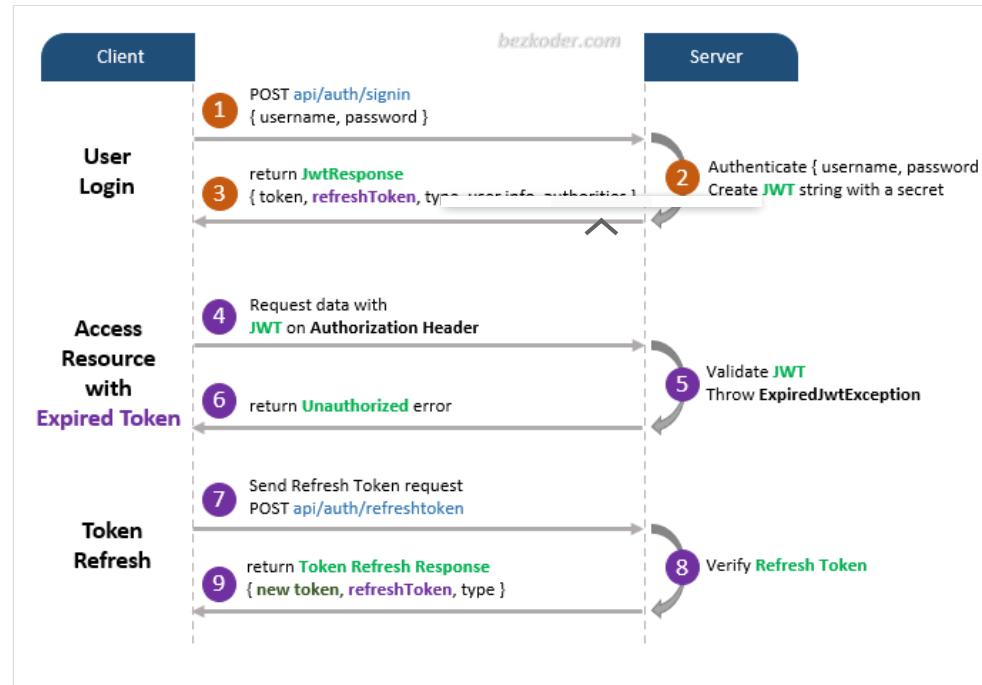
## Spring Boot Signup & Login with JWT Authentication Flow

The diagram shows flow of how we implement User Registration, User Login and Authorization process.



A legal JWT must be added to HTTP Authorization Header if Client accesses protected resources.

Ofertas pro Dia das M  es  
Americanas Express



More details at: Spring Boot Refresh Token with JWT example  
[\(https://bezkoder.com/spring-boot-refresh-token-jwt/\)](https://bezkoder.com/spring-boot-refresh-token-jwt/)

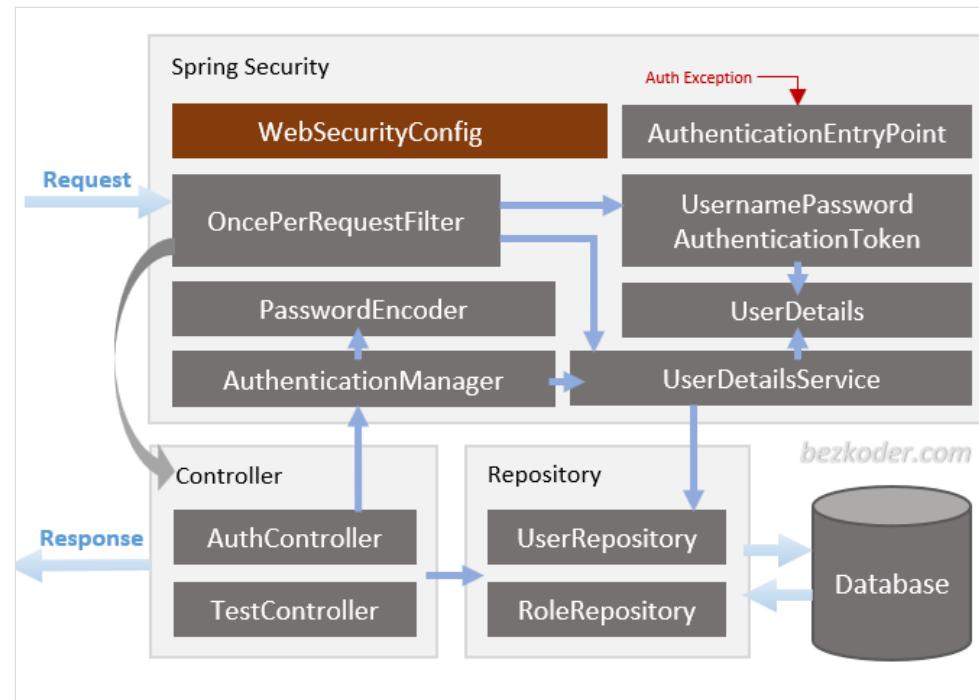
You can also visit The example that uses HttpOnly Cookies instead  
[\(https://www.bezkoder.com/spring-boot-login-example-mysql/\).](https://www.bezkoder.com/spring-boot-login-example-mysql/)

## Spring Boot Server Architecture with Spring Security

You can have an overview of our Spring Boot Server with the diagram below:

# Dia das MÃ©es Americanas

## Americanas Express



Now I will explain it briefly.

### Spring Security

that contains the elements below.

(`WebSecurityConfigurerAdapter` is deprecated from Spring 2.7.0, you can check the source code for update. More details at:

`WebSecurityConfigurerAdapter` Deprecated in Spring Boot  
(<https://www.bezkoder.com/websecurityconfigureradapter-deprecated-spring-boot/>)

- `UserDetailsService` (<https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/#tech-userdetailsservice>) interface has a method to load User by `username` and returns a `UserDetails` object that Spring Security can use for authentication and validation.
- `UserDetails` contains necessary information (such as: `username`, `password`, `authorities`) to build an `Authentication` object.
- `UsernamePasswordAuthenticationToken` (<https://docs.spring.io/spring-security/site/docs/current/api/org/springframework/security/authentication/UsernamePasswordAuthenticationToken.html>) gets `{username, password}` from login Request, `AuthenticationManager` will use it to authenticate a login account.
- `AuthenticationManager` (<https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/#core-services-authentication-manager>) has a `DaoAuthenticationProvider` (with help of `UserDetailsService` & `PasswordEncoder`) to validate `UsernamePasswordAuthenticationToken` object. If successful,

- OncePerRequestFilter (<https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/web/filter/OncePerRequestFilter.html>) makes a single execution for each request to our API. It provides a doFilterInternal() method that we will implement parsing & validating JWT, loading User details (using UserDetailsService ), checking Authorization (using UsernamePasswordAuthenticationToken ).
- AuthenticationEntryPoint (<https://docs.spring.io/spring-security/site/docs/current/api/org/springframework/security/web/AuthenticationEntryPoint.html>) will catch authentication error.

**Repository** contains UserRepository & RoleRepository to work with Database, will be imported into **Controller**.

**Controller** receives and handles request after it was filtered by OncePerRequestFilter .

- AuthController handles signup/login requests
- TestController has accessing protected resource methods with role based validations.

Understand the architecture deeply and grasp the overview more easier:

Spring Boot Architecture for JWT with Spring Security  
(<https://bezkoder.com/spring-boot-jwt-mysql-spring-security-architecture/>)

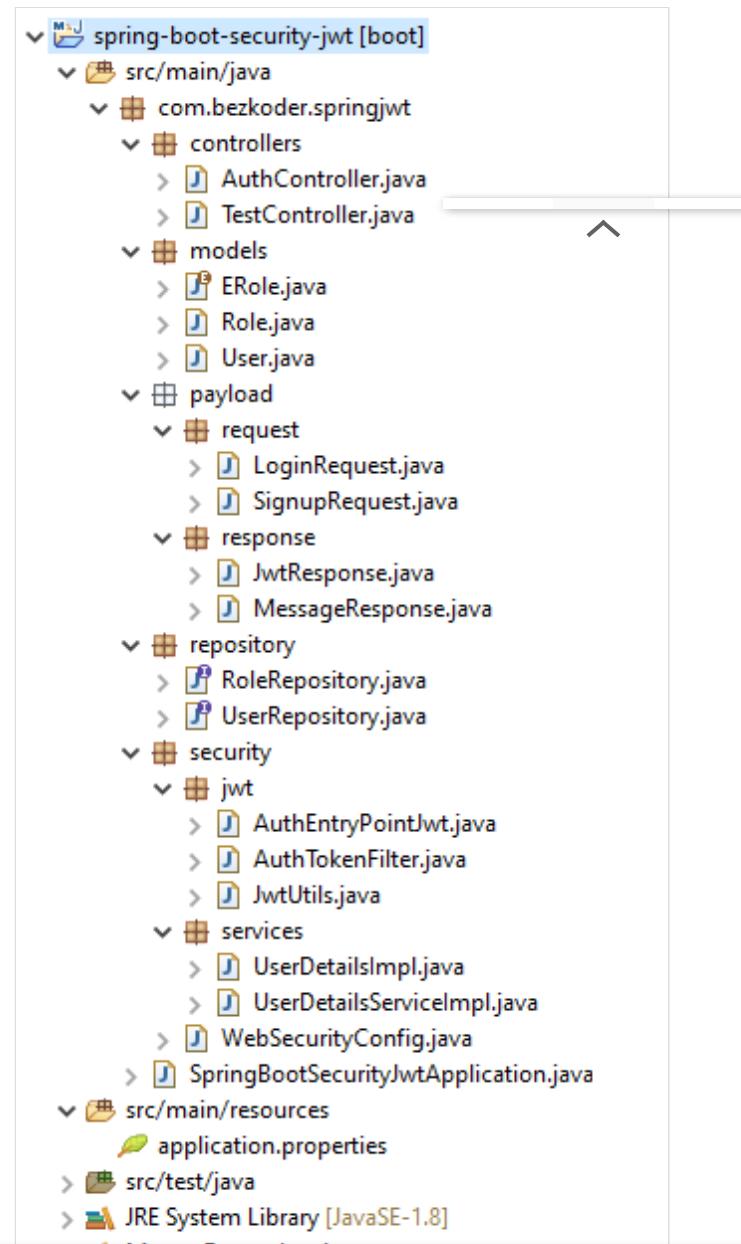
- Spring Boot 2 (with Spring Security, Spring Web, Spring Data JPA)
- jjwt 0.9.1
- PostgreSQL/MySQL
- Maven 3.6.1



## Project Structure

This is folders & files structure for our Spring Boot application:





**security:** we configure Spring Security & implement Security Objects here.

- WebSecurityConfig
  - ( ~~WebSecurityConfigurerAdapter~~ is deprecated from Spring 2.7.0, you can check the source code for update. More details at:  
WebSecurityConfigurerAdapter Deprecated in Spring Boot (<https://www.bezkoder.com/websecurityconfigureradapter-deprecated-spring-boot/>)
- UserDetailsServiceImpl implements UserDetailsService
- UserDetailsImpl implements UserDetails
- AuthEntryPointJwt implements AuthenticationEntryPoint
- AuthTokenFilter extends OncePerRequestFilter
- JwtUtils provides methods for generating, parsing, validating JWT

**controllers** handle signup/login requests & authorized requests.

- AuthController : @PostMapping('/signin'),  
@PostMapping('/signup')
- TestController : @GetMapping('/api/test/all'),  
@GetMapping('/api/test/[role]')

**repository** has interfaces that extend Spring Data JPA JpaRepository to interact with Database.

- UserRepository extends JpaRepository<User, Long>

- User : id, username, email, password, roles
- Role : id, name

**payload** defines classes for Request and Response objects

We also have **application.properties** for configuring Spring Datasource, Spring Data JPA and App properties (such as JWT Secret string or Token expiration time).

## Setup new Spring Boot project

Use Spring web tool (<http://start.spring.io/>) or your development tool (Spring Tool Suite (<https://spring.io/tools>), Eclipse, IntelliJ (<https://www.jetbrains.com/idea/download/>)) to create a Spring Boot project.

Then open **pom.xml** and add these dependencies:



```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>

^

<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-security</artifactId>
</dependency>

<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
</dependency>

<dependency>
    <groupId>io.jsonwebtoken</groupId>
    <artifactId>jjwt</artifactId>
    <version>0.9.1</version>
</dependency>
```

We also need to add one more dependency.

- If you want to use **PostgreSQL**:

```
<dependency>
    <groupId>org.postgresql</groupId>
    <artifactId>postgresql</artifactId>
    <scope>runtime</scope>
```

```
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <scope>runtime</scope>
</dependency>
```

## Configure Spring Datasource, JPA, App properties

Under `src/main/resources` folder, open `application.properties`, add some new lines.

### For PostgreSQL

```
spring.datasource.url= jdbc:postgresql://localhost:5432/testdb
spring.datasource.username= postgres
spring.datasource.password= 123

spring.jpa.properties.hibernate.jdbc.lob.non_contextual_creatio
spring.jpa.properties.hibernate.dialect= org.hibernate.dialect.

# Hibernate ddl auto (create, create-drop, validate, update)
spring.jpa.hibernate.ddl-auto= update

# App Properties
bezkoder.app.jwtSecret= bezKoderSecretKey
bezkoder.app.jwtExpirationMs= 86400000
```

```
spring.datasource.url= jdbc:mysql://localhost:3306/testdb?useSS
spring.datasource.username= root
spring.datasource.password= 123456

spring.jpa.properties.hibernate.dialect= org.hibernate.dialect.I
spring.jpa.hibernate.ddl-auto= update

# App Properties
bezkoder.app.jwtSecret= bezKoderSecretKey
bezkoder.app.jwtExpirationMs= 86400000
```



## Create the models

We're gonna have 3 tables in database: **users**, **roles** and **user\_roles** for many-to-many relationship.

Let's define these models.

In *models* package, create 3 files:

**ERole enum** in *ERole.java*.

In this example, we have 3 roles corresponding to 3 enum.

```
package com.bezkoder.springjwt.models;

public enum ERole {
    ROLE_USER,
    ROLE_MODERATOR,
    ROLE_ADMIN
```

```
package com.bezkoder.springjwt.models;

import javax.persistence.*;

@Entity
@Table(name = "roles")
public class Role {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;

    @Enumerated(EnumType.STRING)
    @Column(length = 20)
    private ERole name;

    public Role() {
    }

    public Role(ERole name) {
        this.name = name;
    }

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }
}
```

```
    }

    public void setName(ERole name) {
        this.name = name;
    }
}
```



**User model** in *User.java*.

It has 5 fields: id, username, email, password, roles.

```
package com.bezkoder.springjwt.models;

import java.util.HashSet;
import java.util.Set;
^
import javax.persistence.*;
import javax.validation.constraints.Email;
import javax.validation.constraints.NotBlank;
import javax.validation.constraints.Size;

@Entity
@Table( name = "users",
        uniqueConstraints = {
            @UniqueConstraint(columnNames = "username"),
            @UniqueConstraint(columnNames = "email")
        })
public class User {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    @NotBlank
    @Size(max = 20)
    private String username;

    @NotBlank
    @Size(max = 50)
    @Email
    private String email;
}
```

```
@ManyToMany(fetch = FetchType.LAZY)
@JoinTable( name = "user_roles",
            joinColumns = @JoinColumn(name = "user_id"),
            inverseJoinColumns = @JoinColumn(name = "role_id"))
private Set<Role> roles = new HashSet<>();

public User() {
}

public User(String username, String email, String password)
{
    this.username = username;
    this.email = email;
    this.password = password;
}

public Long getId() {
    return id;
}

public void setId(Long id) {
    this.id = id;
}

public String getUsername() {
    return username;
}

public void setUsername(String username) {
    this.username = username;
}
```

```
}

public void setEmail(String email) {
    this.email = email;
}

public String getPassword() {
    return password;
}

public void setPassword(String password) {
    this.password = password;
}

public Set<Role> getRoles() {
    return roles;
}

public void setRoles(Set<Role> roles) {
    this.roles = roles;
}
```

## Implement Repositories

Now, each model above needs a repository for persisting and accessing data. In *repository* package, let's create 2 repositories.

```
package com.bezkoder.springjwt.repository;

import java.util.Optional;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.bezkoder.springjwt.models.User;

@Repository
public interface UserRepository extends JpaRepository<User, Long> {
    Optional<User> findByUsername(String username);

    Boolean existsByUsername(String username);

    Boolean existsByEmail(String email);
}
```

## RoleRepository

This repository also extends `JpaRepository` and provides a finder method.

```
package com.bezkoder.springjwt.repository;

import java.util.Optional;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.bezkoder.springjwt.models.ERole;
import com.bezkoder.springjwt.models.Role;

@Repository
public interface RoleRepository extends JpaRepository<Role, Long> {
    Optional<Role> findByName(ERole name);
}
```

## Configure Spring Security

### Without WebSecurityConfigurerAdapter

WebSecurityConfigurerAdapter is deprecated from Spring 2.7.0. More details at:

WebSecurityConfigurerAdapter Deprecated in Spring Boot  
(<https://www.bezkoder.com/websecurityconfigureradapter-deprecated-spring-boot/>).

In `security` package, create `WebSecurityConfig` class.

WebSecurityConfig.java

```
package com.bezkoder.springjwt.security;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.security.authentication.AuthenticationManager;
import org.springframework.security.authentication.dao.DaoAuthenticationProvider;
import org.springframework.security.config.annotation.authentication.builders.AuthenticationManagerBuilder;
import org.springframework.security.config.annotation.web.builders.HttpSecurity;
import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;
import org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;
import org.springframework.security.config.http.SessionCreationPolicy;
import org.springframework.security.crypto.bcrypt.BCryptPasswordEncoder;
import org.springframework.security.crypto.password.PasswordEncoder;
import org.springframework.security.web.SecurityFilterChain;
import org.springframework.security.web.authentication.UsernamePasswordAuthenticationFilter;

import com.bezkoder.springjwt.security.jwt.AuthEntryPointJwt;
import com.bezkoder.springjwt.security.jwt.AuthTokenFilter;
import com.bezkoder.springjwt.security.services.UserDetailsService;

@Configuration
@EnableGlobalMethodSecurity(
    // securedEnabled = true,
    // jsr250Enabled = true,
    prePostEnabled = true)
public class WebSecurityConfig {
    @Autowired
    UserDetailsService userDetailsService;
```

```
public AuthTokenFilter authenticationJwtTokenFilter() {  
    return new AuthTokenFilter();  
}  
  
@Bean  
public DaoAuthenticationProvider authenticationProvider() {  
    DaoAuthenticationProvider authProvider = new DaoAuthenticati  
  
    authProvider.setUserDetailsService(userDetailsService);  
    authProvider.setPasswordEncoder(passwordEncoder());  
  
    return authProvider;  
}  
  
@Bean  
public AuthenticationManager authenticationManager(Authentica  
    return authConfig.getAuthenticationManager();  
}  
  
@Bean  
public PasswordEncoder passwordEncoder() {  
    return new BCryptPasswordEncoder();  
}  
  
@Bean  
public SecurityFilterChain filterChain(HttpSecurity http) throws  
    http.cors().and().csrf().disable()  
    .exceptionHandling().authenticationEntryPoint(unauthoriz  
    .sessionManagement().sessionCreationPolicy(SessionCreat  
    authorizeRequests().antMatchers("/api/auth/**").permitAll()
```

```
        http.addFilterBefore(authenticationJwtTokenFilter(), UsernamePasswordAuthenticationFilter.class);

    }

}
```

## With WebSecurityConfigurerAdapter

In `security` package, create `WebSecurityConfig` class that extends `WebSecurityConfigurerAdapter`.

`WebSecurityConfig.java`



```
@Bean
public AuthTokenFilter authenticationJwtTokenFilter() {
    return new AuthTokenFilter();
}

@Override
public void configure(AuthenticationManagerBuilder authenti
    authenticationManagerBuilder.userDetailsService(userDet
}

@Bean
@Override
public AuthenticationManager authenticationManagerBean() th
    return super.authenticationManagerBean();
}

@Bean
public PasswordEncoder passwordEncoder() {
    return new BCryptPasswordEncoder();
}

@Override
protected void configure(HttpSecurity http) throws Exceptio
    http.cors().and().csrf().disable()
        .exceptionHandling().authenticationEntryPoint(unaut
        .sessionManagement().sessionCreationPolicy(SessionC
        .authorizeRequests().antMatchers("/api/auth/**").pe
        .antMatchers("/api/test/**").permitAll()
        .anyRequest().authenticated();
```

```
    }  
}
```

Let me explain the code above.

- `@EnableWebSecurity` allows Spring to find and automatically apply the class to the global Web Security.
- `@EnableGlobalMethodSecurity` provides AOP security on methods. It enables `@PreAuthorize`, `@PostAuthorize`, it also supports JSR-250 ([https://en.wikipedia.org/wiki/JSR\\_250](https://en.wikipedia.org/wiki/JSR_250)). You can find more parameters in configuration in Method Security Expressions (<https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/#method-security-expressions>).
- We override the `configure(HttpSecurity http)` method from `WebSecurityConfigurerAdapter` interface. It tells Spring Security how we configure CORS and CSRF, when we want to require all users to be authenticated or not, which filter ( `AuthTokenFilter` ) and when we want it to work (filter before `UsernamePasswordAuthenticationFilter` ), which Exception Handler is chosen ( `AuthEntryPointJwt` ).
- Spring Security will load User details to perform authentication & authorization. So it has `UserDetailsService` interface that we need to implement.

- We also need a `PasswordEncoder` for the `DaoAuthenticationProvider`. If we don't specify, it will use plain text.

## Implement UserDetails & UserDetailsService

If the authentication process is successful, we can get User's information such as username, password, authorities from an `Authentication` object.

```
Authentication authentication =
    authenticationManager.authenticate(
        new UsernamePasswordAuthenticationToken(username, p
    );

UserDetails userDetails = (UserDetails) authentication.getPrinc
// userDetails.getUsername()
// userDetails.getPassword()
// userDetails.getAuthorities()
```

If we want to get more data (id, email...), we can create an implementation of this `UserDetails` interface.

*security/services/UserDetailsImpl.java*

```
package com.bezkoder.springjwt.security.services;

import java.util.Collection;
import java.util.List;
import java.util.Objects; ^_
import java.util.stream.Collectors;

import org.springframework.security.core.GrantedAuthority;
import org.springframework.security.core.authority.SimpleGrantedAuthority;
import org.springframework.security.core.userdetails.UserDetails;

import com.bezkoder.springjwt.models.User;
import com.fasterxml.jackson.annotation.JsonIgnore;

public class UserDetailsImpl implements UserDetails {
    private static final long serialVersionUID = 1L;

    private Long id;

    private String username;

    private String email;

    @JsonIgnore
    private String password;

    private Collection<? extends GrantedAuthority> authorities;

    public UserDetailsImpl(Long id, String username, String ema
```

```
this.password = password;
this.authorities = authorities;
}

public static UserDetailsImpl build(User user) {
    List<GrantedAuthority> authorities = user.getRoles().st
        .map(role -> new SimpleGrantedAuthority(role.ge
        .collect(Collectors.toList()));

    return new UserDetailsImpl(
        user.getId(),
        user.getUsername(),
        user.getEmail(),
        user.getPassword(),
        authorities);
}

@Override
public Collection<? extends GrantedAuthority> getAuthoritie
    return authorities;
}

public Long getId() {
    return id;
}

public String getEmail() {
    return email;
}
```



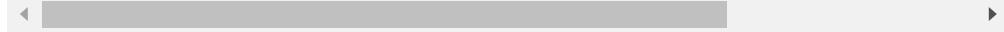
```
@Override  
public String getUsername() {  
    return username;  
}  
  
@Override  
public boolean isAccountNonExpired() {  
    return true;  
}  
  
@Override  
public boolean isAccountNonLocked() {  
    return true;  
}  
  
@Override  
public boolean isCredentialsNonExpired() {  
    return true;  
}  
  
@Override  
public boolean isEnabled() {  
    return true;  
}  
  
@Override  
public boolean equals(Object o) {  
    if (this == o)  
        return true;
```

```
    }  
}
```

Look at the code above, you can notice that we convert `Set<Role>` into `List<GrantedAuthority>`. It is important to work with Spring Security and Authentication object later.

As I have said before, we need `UserDetailsService` for getting `UserDetails` object. You can look at `UserDetailsService` interface that has only one method:

```
public interface UserDetailsService {  
    UserDetails loadUserByUsername(String username) throws User  
}
```



So we implement it and override `loadUserByUsername()` method.

*security/services/UserDetailsServiceImpl.java*

```
package com.bezkoder.springjwt.security.services;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.security.core.userdetails.UserDetail
import org.springframework.security.core.userdetails.UserDetail
import org.springframework.security.core.userdetails.UsernameNo
import org.springframework.stereotype.Service;
import org.springframework.transaction.annotation.Transactional

import com.bezkoder.springjwt.models.User;
import com.bezkoder.springjwt.repository.UserRepository;

@Service
public class UserDetailsServiceImpl implements UserDetailsServ
    @Autowired
    UserRepository userRepository;

    @Override
    @Transactional
    public UserDetails loadUserByUsername(String username) throws
        User user = userRepository.findByUsername(username)
            .orElseThrow(() -> new UsernameNotFoundException("User no
                return UserDetailsImpl.build(user);
            }

    }

```

## Filter the Requests

Let's define a filter that executes once per request. So we create `AuthTokenFilter` class that extends `OncePerRequestFilter` and override `doFilterInternal()` method.

*security/jwt/AuthTokenFilter.java*

```
package com.bezkoder.springjwt.security.jwt;

import java.io.IOException;

import javax.servlet.FilterChain; ^
import javax.servlet.ServletException;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.security.authentication.UsernamePasswordAuthenticationToken;
import org.springframework.security.core.context.SecurityContextHolder;
import org.springframework.security.core.userdetails.UserDetails;
import org.springframework.security.web.authentication.WebAuthenticationDetailsSource;
import org.springframework.util.StringUtils;
import org.springframework.web.filter.OncePerRequestFilter;

import com.bezkoder.springjwt.security.services.UserDetailsService;

public class AuthTokenFilter extends OncePerRequestFilter {
    @Autowired
    private JwtUtils jwtUtils;

    @Autowired
    private UserDetailsService userDetailsService;

    private static final Logger logger = LoggerFactory.getLogger(OncePerRequestFilter.class);
}
```

```
try {
    String jwt = parseJwt(request);
    if (jwt != null && jwtUtils.validateJwtToken(jwt))
        String username = jwtUtils.getUserNameFromJwtTo
        ...
        UserDetails userDetails = userDetailsService.lo
        UsernamePasswordAuthenticationToken authenticat
            userDetails, null, userDetails.getAutho
        authentication.setDetails(new WebAuthentication
        ...
        SecurityContextHolder.getContext().setAuthentic
    }
} catch (Exception e) {
    logger.error("Cannot set user authentication: {}", e)
}

filterChain.doFilter(request, response);
}

private String parseJwt(HttpServletRequest request) {
    String headerAuth = request.getHeader("Authorization");

    if (StringUtils.hasText(headerAuth) && headerAuth.start
        return headerAuth.substring(7, headerAuth.length())
    }

    return null;
}
}
```

What we do inside `doFilterInternal()` :

- get JWT from the Authorization header (by removing Bearer prefix)
- if the request has JWT , validate it, parse username from it
- from username , get UserDetails to create an Authentication object
- set the current UserDetails in SecurityContext  
<https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/#securitycontextholder-securitycontext-and-authentication-objects> using `setAuthentication(authentication)` method.

After this, everytime you want to get UserDetails , just use `SecurityContext` like this:

```
UserDetails userDetails =  
    (UserDetails) SecurityContextHolder.getContext().getAuthent  
  
    // userDetails.getUsername()  
    // userDetails.getPassword()  
    // userDetails.getAuthorities()
```

## Create JWT Utility class

This class has 3 funtions:

- generate a JWT from username, date, expiration, secret
- get username from JWT

```
package com.bezkoder.springjwt.security.jwt;

import java.util.Date;

import org.slf4j.Logger; ^
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.security.core.Authentication;
import org.springframework.stereotype.Component;

import com.bezkoder.springjwt.security.services.UserDetailsImpl
import io.jsonwebtoken.*;

@Component
public class JwtUtils {
    private static final Logger logger = LoggerFactory.getLogger(JwtUtils.class);

    @Value("${bezkoder.app.jwtSecret}")
    private String jwtSecret;

    @Value("${bezkoder.app.jwtExpirationMs}")
    private int jwtExpirationMs;

    public String generateJwtToken(Authentication authentication) {
        UserDetailsImpl userPrincipal = (UserDetailsImpl) authentication.getPrincipal();

        return Jwts.builder()
            .setSubject((userPrincipal.getUsername()))
            .setIssuedAt(new Date())
            .setExpiration(new Date(new Date().getTime() + jwtExpirationMs))
            .signWith(SignatureAlgorithm.HS256, jwtSecret)
            .compact();
    }
}
```

```
}

public String getUserNameFromJwtToken(String token) {
    return Jwts.parser().setSigningKey(jwtSecret).parseClaimsJ...
}

public boolean validateJwtToken(String authToken) {
    try {
        Jwts.parser().setSigningKey(jwtSecret).parseClaimsJ...
        return true;
    } catch (SignatureException e) {
        logger.error("Invalid JWT signature: {}", e.getMessage());
    } catch (MalformedJwtException e) {
        logger.error("Invalid JWT token: {}", e.getMessage());
    } catch (ExpiredJwtException e) {
        logger.error("JWT token is expired: {}", e.getMessage());
    } catch (UnsupportedJwtException e) {
        logger.error("JWT token is unsupported: {}", e.getMessage());
    } catch (IllegalArgumentException e) {
        logger.error("JWT claims string is empty: {}", e.getMessage());
    }
    return false;
}
}
```

Remember that we've added bezkoder.app.jwtSecret and  
bezkoder.app.jwtExpirationMs properties in application.properties

## Handle Authentication Exception

Now we create `AuthEntryPointJwt` class that implements `AuthenticationEntryPoint` interface. Then we override the `commence()` method. This method will be triggered anytime unauthenticated User requests a secured HTTP resource and an `AuthenticationException` is thrown.

*security/jwt/AuthEntryPointJwt.java*

```
package com.bezkoder.springjwt.security.jwt;

import java.io.IOException;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.security.core.AuthenticationException;
import org.springframework.security.web.AuthenticationEntryPoint;
import org.springframework.stereotype.Component;

@Component
public class AuthEntryPointJwt implements AuthenticationEntryPoint {

    private static final Logger logger = LoggerFactory.getLogger(AuthEntryPointJwt.class);

    @Override
    public void commence(HttpServletRequest request, HttpServletResponse response,
                         AuthenticationException authException) throws IOException {
        logger.error("Unauthorized error: {}", authException.getMessage());
        response.sendError(HttpServletResponse.SC_UNAUTHORIZED,
                           authException.getMessage());
    }
}
```

We've already built all things for Spring Security. The next sections of this tutorial will show you how to implement Controllers for our RestAPIs.

## Define payloads for Spring RestController

Let me summarize the payloads for our RestAPIs:

– Requests:

- LoginRequest: { username, password }
- SignupRequest: { username, email, password }

– Responses:

- JwtResponse: { token, type, id, username, email, roles }
- MessageResponse: { message }

To keep the tutorial not so long, I don't show these POJOs here.

You can find details for payload classes in source code of the project on Github (<https://github.com/bezkoder/spring-boot-spring-security-jwt-authentication>).

## Create Spring RestAPIs Controllers

### Controller for Authentication

This controller provides APIs for register and login actions.

- /api/auth/signup
  - check existing username / email

- authenticate { username, password }
- update SecurityContext using Authentication object
- generate JWT
- get UserDetails from Authentication object
- response contains JWT and UserDetails data

*controllers/AuthController.java*

```
package com.bezkoder.springjwt.controllers;

import java.util.HashSet;
import java.util.List;
import java.util.Set;
import java.util.stream.Collectors;

import javax.validation.Valid;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.ResponseEntity;
import org.springframework.security.authentication.AuthenticationManager;
import org.springframework.security.authentication.UsernamePasswordAuthenticationToken;
import org.springframework.security.core.Authentication;
import org.springframework.security.core.context.SecurityContextHolder;
import org.springframework.security.crypto.password.PasswordEncoder;
import org.springframework.web.bind.annotation.CrossOrigin;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import com.bezkoder.springjwt.models.ERole;
import com.bezkoder.springjwt.models.Role;
import com.bezkoder.springjwt.models.User;
import com.bezkoder.springjwt.payload.request.LoginRequest;
import com.bezkoder.springjwt.payload.request.SignupRequest;
import com.bezkoder.springjwt.payload.response.JwtResponse;
import com.bezkoder.springjwt.payload.response.MessageResponse;
```

```
@CrossOrigin(origins = "*", maxAge = 3600)
@RestController
@RequestMapping("/api/auth")
public class AuthController {
    @Autowired
    AuthenticationManager authenticationManager;

    @Autowired
    UserRepository userRepository;

    @Autowired
    RoleRepository roleRepository;

    @Autowired
    PasswordEncoder encoder;

    @Autowired
    JwtUtils jwtUtils;

    @PostMapping("/signin")
    public ResponseEntity<?> authenticateUser(@Valid @RequestBody
        Authentication authentication = authenticationManager.authenticate(
            new UsernamePasswordAuthenticationToken(loginRequest.getUsername(),
                loginRequest.getPassword()));

        SecurityContextHolder.getContext().setAuthentication(authentication);
        String jwt = jwtUtils.generateJwtToken(authentication);

        UserDetailSTmnl userDetailc = (UserDetailSTmnl) authentication.getDetails();
        return ResponseEntity.ok(new JwtAuthenticationResponse(jwt));
    }
}
```

```
        return ResponseEntity.ok(new JwtResponse(jwt,
                                                userDetails.getFirstName(),
                                                userDetails.getLastName(),
                                                userDetails.getAuthorities()));
    }

    @PostMapping("/signup")
    public ResponseEntity<?> registerUser(@Valid @RequestBody SignUpRequest signUpRequest) {
        if (userRepository.existsByUsername(signUpRequest.getUsername())) {
            return ResponseEntity
                .badRequest()
                .body(new MessageResponse("Error: Username is already taken!"));
        }

        if (userRepository.existsByEmail(signUpRequest.getEmail())) {
            return ResponseEntity
                .badRequest()
                .body(new MessageResponse("Error: Email is already in use!"));
        }

        // Create new user's account
        User user = new User(signUpRequest.getUsername(),
                             signUpRequest.getEmail(),
                             encoder.encode(signUpRequest.getPassword()));

        Set<String> strRoles = signUpRequest.getRole();
        Set<Role> roles = new HashSet<>();

        if (strRoles == null) {

```

```
strRoles.forEach(role -> {
    switch (role) {
        case "admin":
            Role adminRole = roleRepository.findByName(
                .orElseThrow(() -> new RuntimeException("Role not found"))
            );
            roles.add(adminRole);

            break;
        case "mod":
            Role modRole = roleRepository.findByName("MOD")
                .orElseThrow(() -> new RuntimeException("Role not found"));
            roles.add(modRole);

            break;
        default:
            Role userRole = roleRepository.findByName("USER")
                .orElseThrow(() -> new RuntimeException("Role not found"));
            roles.add(userRole);
    }
});

user.setRoles(roles);
userRepository.save(user);

return ResponseEntity.ok(new MessageResponse("User registered"));
}
```

There are 4 APIs:

- /api/test/all for public access
- /api/test/user for users has ROLE\_USER or ROLE\_MODERATOR or  
ROLE\_ADMIN
- /api/test/mod for users has ROLE\_MODERATOR
- /api/test/admin for users has ROLE\_ADMIN

Do you remember that we used

@EnableGlobalMethodSecurity(prePostEnabled = true) for  
WebSecurityConfig class?

```
@Configuration  
@EnableWebSecurity  
@EnableGlobalMethodSecurity(prePostEnabled = true)  
public class WebSecurityConfig { ... }
```

Now we can secure methods in our Apis with @PreAuthorize  
annotation easily.

*controllers/TestController.java*

```
package com.bezkoder.springjwt.controllers;

import org.springframework.security.access.prepost.PreAuthorize;
import org.springframework.web.bind.annotation.CrossOrigin;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@CrossOrigin(origins = "*", maxAge = 3600)
@RestController
@RequestMapping("/api/test")
public class TestController {
    @GetMapping("/all")
    public String allAccess() {
        return "Public Content.";
    }

    @GetMapping("/user")
    @PreAuthorize("hasRole('USER') or hasRole('MODERATOR') or hasRole('ADMIN')")
    public String userAccess() {
        return "User Content.";
    }

    @GetMapping("/mod")
    @PreAuthorize("hasRole('MODERATOR')")
    public String moderatorAccess() {
        return "Moderator Board.";
    }
}
```

```
    }  
}
```

## Run & Test



Run Spring Boot application with command: `mvn spring-boot:run`

Tables that we define in *models* package will be automatically generated in Database.

If you check MySQL database for example, you can see things like this:

```
mysql> describe users;
+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra
+-----+-----+-----+-----+
| id    | bigint(20) | NO   | PRI | NULL    | auto_increme
| email | varchar(50) | YES  | UNI | NULL    |
| password | varchar(120) | YES  |      | NULL    |
| username | varchar(20)  | YES  | UNI | NULL    |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> describe roles;
+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra
+-----+-----+-----+-----+
| id    | int(11)   | NO   | PRI | NULL    | auto_increment
| name  | varchar(20) | YES  |      | NULL    |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> describe user_roles;
+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra
+-----+-----+-----+-----+
| user_id | bigint(20) | NO   | PRI | NULL    | 
| role_id | int(11)    | NO   | PRI | NULL    | 
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

We also need to add some rows into **roles** table before assigning any role to User.

Run following SQL insert statements:

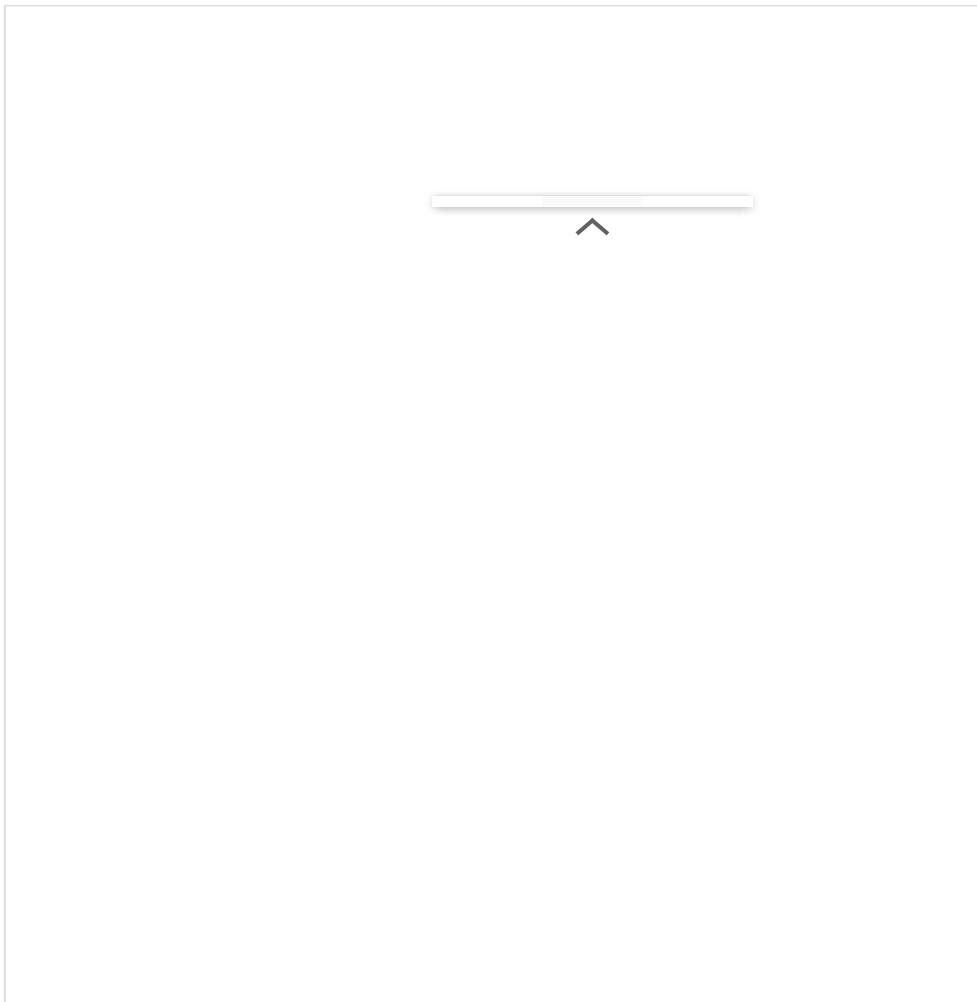
```
INSERT INTO roles(name) VALUES('ROLE_USER');
INSERT INTO roles(name) VALUES('ROLE_MODERATOR');
INSERT INTO roles(name) VALUES('ROLE_ADMIN');
```

Then check the tables:

```
mysql> select * from roles;
+----+-----+
| id | name      |
+----+-----+
| 1  | ROLE_USER |
| 2  | ROLE_MODERATOR |
| 3  | ROLE_ADMIN |
+----+-----+
3 rows in set (0.00 sec)
```

Register some users with /signup API:

- **admin** with ROLE\_ADMIN
- **mod** with ROLE\_MODERATOR and ROLE\_USER
- **zkoder** with ROLE\_USER



Our tables after signup could look like this.



```
mysql> select * from users;
+----+-----+-----+
| id | email           | password          |
+----+-----+-----+
| 1  | admin@bezkoder.com | $2a$10$mr4MU5esBbUd6JWuwWKTA.tRy.jo
| 2  | mod@bezkoder.com  | $2a$10$VcdzH8Q.o4KEo6df.Xesd0mXdXQw
| 3  | user@bezkoder.com | $2a$10$c/cAdrKfiLLCDcnXvdI6MumFMthI
+----+-----+
3 rows in set (0.00 sec)
```

```
mysql> select * from roles;
+----+-----+
| id | name            |
+----+-----+
| 1  | ROLE_USER        |
| 2  | ROLE_MODERATOR   |
| 3  | ROLE_ADMIN        |
+----+-----+
3 rows in set (0.00 sec)
```

```
mysql> select * from user_roles;
+-----+-----+
| user_id | role_id |
+-----+-----+
|      2  |      1  |
|      3  |      1  |
|      2  |      2  |
+-----+-----+
```

```
+-----+-----+
4 rows in set (0.00 sec)
```

**Access public resource: GET /api/test/all**

The screenshot shows the Postman application interface. At the top, there is a header bar with tabs for 'GET' (selected), 'http://localhost:8080/api/test/all', and a 'Send' button. Below the header are tabs for 'Params', 'Authorization', 'Headers (7)' (which is underlined in red), 'Body', 'Pre-request Script', 'Tests', and 'Settings'. Under the 'Headers' tab, there is a section titled 'Query Params' with a table:

KEY	VALUE	DESCRIPTION
Key	Value	Description

Below the table, there are tabs for 'Body', 'Cookies', 'Headers (9)', and 'Test Results'. The 'Test Results' tab is selected, showing the response status: 'Status: 200 OK', 'Time: 325ms', and 'Size: 313 B'. Underneath these tabs are buttons for 'Pretty', 'Raw', 'Preview', 'Visualize BETA', 'Text' (with a dropdown arrow), and a copy icon. The main body area displays the response content: '1 Public Content.'

**Access protected resource: GET /api/test/user**

The screenshot shows a POSTMAN interface with the following details:

- Method:** GET
- URL:** <http://localhost:8080/api/test/user>
- Headers:** (7)
- Body:** (Raw JSON response)
- Status:** 401 Unauthorized
- Time:** 218ms
- Size:** 458 B

**Query Params:**

KEY	VALUE	DESCRIPTION
Key	Value	Description

**Body (Pretty JSON):**

```
1 {  
2     "timestamp": "2019-10-15T15:01:26.948+0000",  
3     "status": 401,  
4     "error": "Unauthorized",  
5     "message": "Error: Unauthorized",  
6     "path": "/api/test/user"  
7 }
```

**Login an account:** POST /api/auth/signin

The screenshot shows a Postman request to `http://localhost:8080/api/auth/signin`. The request method is `POST`. The `Body` tab is selected, showing a JSON payload:

```
1 {  
2   "username": "mod",  
3   "password": "12345678"  
4 }
```

The response status is `200 OK`, time `1559ms`, and size `612 B`. The response body is displayed in `JSON` format:

```
1 {  
2   "id": 2,  
3   "username": "mod",  
4   "email": "mod@bezkoder.com",  
5   "roles": [  
6     "ROLE_USER",  
7     "ROLE_MODERATOR"  
8   ],  
9   "accessToken":  
10    "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJtb2QiLCJpYXQiOjE1NzExNTE4NDIsImV4cCI6MTU3MTIzMwremuN19pOvxfEiuH7AU2GIEdoQ9Ejs98cARunxfCyKbbG_oZWl0NfOkUZ5ONjfTxXK-yoOxc9mKn  
11 }  
12 
```

**Access ROLE\_USER resource: GET /api/test/user**

The screenshot shows a Postman request configuration for a GET request to `http://localhost:8080/api/test/user`. The 'Headers' tab is selected, displaying one header: `Authorization` with the value `Bearer eyJhbGciOiJIUzUxMiJ9.eyJzdWlOIjtb2QiLCJpYXQiOjE1N...`. The response status is `200 OK`, time is `50ms`, and size is `311 B`. The response body contains the message `User Content.`

### Access ROLE\_MODERATOR resource: GET /api/test/mod

The screenshot shows a Postman request configuration for a GET request to `http://localhost:8080/api/test/mod`. The 'Headers' tab is selected, displaying one header: `Authorization` with the value `Bearer eyJhbGciOiJIUzUxMiJ9.eyJzdWlOIjtb2QiLCJpYXQiOjE1N...`. The response status is `200 OK`, time is `46ms`, and size is `314 B`.

**Access ROLE\_ADMIN resource: GET /api/test/admin**

The screenshot shows a POSTMAN interface with the following details:

- Method: GET
- URL: <http://localhost:8080/api/test/admin>
- Headers tab selected, showing:
  - Authorization: Bearer eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJtb2QiLCJpYXQiOjE1N...
  - Key Value Description
- Temporary Headers tab (7 items) is collapsed.
- Body tab selected, showing:
  - Pretty
  - Raw
  - Preview
  - Visualize BETA
  - JSON ▾
  - Copy icon

```
1 {
2   "timestamp": "2019-10-15T15:22:51.341+0000",
3   "status": 403,
4   "error": "Forbidden",
5   "message": "Forbidden",
6   "path": "/api/test/admin"
7 }
```
- Status: 403 Forbidden
- Time: 49ms
- Size: 443 B

## Solve Problem: javax.validation cannot be resolved

For Spring Boot **2.3** and later, you can see the compile error:

The import javax.validation cannot be resolved

It is because Validation Starter no longer included in web starters. So

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-validation</artifactId>
</dependency>
```



- For Gradle:

```
dependencies {
    ...
    implementation 'org.springframework.boot:spring-boot-starter-'
}
```

## Solve Problem with JDK 14

If you run this Spring Boot App with JDK 14 and get following error when trying to authenticate:

```
FilterChain java.lang.NoClassDefFoundError: javax/xml/bind/Data
```

Just add following dependency to *pom.xml*:

```
<dependency>
    <groupId>jakarta.xml.bind</groupId>
    <artifactId>jakarta.xml.bind-api</artifactId>
    <version>2.3.2</version>
```

# Conclusion

Congratulation!

Today we've learned so many interesting things about Spring Security and JWT Token based Authentication in just a Spring Boot example.

Despite we wrote a lot of code, I hope you will understand the overall architecture of the application, and apply it in your project at ease.

For understanding the architecture deeply and grasp the overview more easier:

Spring Boot Architecture for JWT with Spring Security

(<https://bezkoder.com/spring-boot-jwt-mysql-spring-security-architecture/>)

Validate the signup request (password, confirm password):

Spring Boot custom Validation example

(<https://www.bezkoder.com/spring-boot-custom-validation/>)

You should continue to know how to implement Refresh Token:

Spring Boot Refresh Token with JWT example

(<https://bezkoder.com/spring-boot-refresh-token-jwt/>)

Or visit The example that uses HttpOnly Cookies instead

(<https://www.bezkoder.com/spring-boot-login-example-mysql/>).

You can also know how to deploy Spring Boot App on AWS (for free) with this tutorial (<https://bezkoder.com/deploy-spring-boot-aws-eb/>).

Happy learning! See you again.

## Further Reading

- Spring Security Reference (<https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/>)
- In-depth Introduction to ~~JWT JSON WEB TOKEN~~ (<https://bezkoder.com/jwt-json-web-token/>)
- Spring Boot Pagination & Filter example | Spring JPA, Pageable (<https://bezkoder.com/spring-boot-pagination-filter-jpa-pageable/>)

### Related Posts:

- Spring Boot, Spring Data JPA – Building Rest CRUD API example (<https://bezkoder.com/spring-boot-jpa-crud-rest-api/>)
- CRUD GraphQL APIs example with Spring Boot, MySQL & Spring JPA (<https://bezkoder.com/spring-boot-graphql-mysql-jpa/>)
- Spring Boot Rest XML example – Web service with XML Response (<https://bezkoder.com/spring-boot-rest-xml/>)
- Spring Boot File upload example with Multipart File (<https://bezkoder.com/spring-boot-file-upload/>)
- @RestControllerAdvice example in Spring Boot (<https://bezkoder.com/spring-boot-restcontrolleradvice/>)
- Spring Boot @ControllerAdvice & @ExceptionHandler example (<https://bezkoder.com/spring-boot-controlleradvice-exceptionhandler/>)
- @DataJpaTest example for Spring Data Repository Unit Test (<https://bezkoder.com/spring-boot-unit-test-jpa-repo-dataintest/>)

- Deploy Spring Boot App on AWS – Elastic Beanstalk  
(<https://www.bezkoder.com/deploy-spring-boot-aws-eb/>)
- Docker Compose: Spring Boot and MySQL example  
(<https://www.bezkoder.com/docker-compose-spring-boot-mysql/>)



#### Fullstack CRUD App:

- Spring Boot + Vue.js example (<https://www.bezkoder.com/spring-boot-vue-js-crud-example/>)
- Angular 8 + Spring Boot example  
(<https://www.bezkoder.com/angular-spring-boot-crud/>)
- Angular 10 + Spring Boot example  
(<https://www.bezkoder.com/angular-10-spring-boot-crud/>)
- Angular 11 + Spring Boot example  
(<https://www.bezkoder.com/angular-11-spring-boot-crud/>)
- Angular 12 + Spring Boot example  
(<https://www.bezkoder.com/angular-12-spring-boot-crud/>)
- Angular 13 + Spring Boot example  
(<https://www.bezkoder.com/spring-boot-angular-13-crud/>)
- Angular 14 + Spring Boot example  
(<https://www.bezkoder.com/spring-boot-angular-14-crud/>)
- React + Spring Boot example (<https://www.bezkoder.com/react-spring-boot-crud/>)

If you need a working front-end for this back-end, you can find Client App in the posts:

- Vue.js JWT Authentication with Vuex and Vue Router

(<https://www.bezkoder.com/angular-10-jwt-auth/>)

– Angular 11 JWT Authentication example with Web Api

(<https://www.bezkoder.com/angular-11-jwt-auth/>)

– Angular 12 JWT Authentication example with Web Api

(<https://www.bezkoder.com/angular-12-jwt-auth/>)

– Angular 13 JWT Authentication example with Web Api

(<https://www.bezkoder.com/angular-13-jwt-auth/>)

– React JWT Authentication (without Redux) example

(<https://www.bezkoder.com/react-jwt-auth/>)

– React Hooks: JWT Authentication (without Redux) example

(<https://www.bezkoder.com/react-hooks-jwt-auth/>)

– React Redux: JWT Authentication example

(<https://www.bezkoder.com/react-redux-jwt-auth/>)

#### Associations:

– JPA One To One example with Hibernate in Spring Boot

(<https://www.bezkoder.com/jpa-one-to-one/>)

– JPA One To Many example with Hibernate and Spring Boot

(<https://www.bezkoder.com/jpa-one-to-many/>)

– JPA Many to Many example with Hibernate in Spring Boot

(<https://www.bezkoder.com/jpa-many-to-many/>)

Documentation: Spring Boot + Swagger 3 example (with OpenAPI 3)

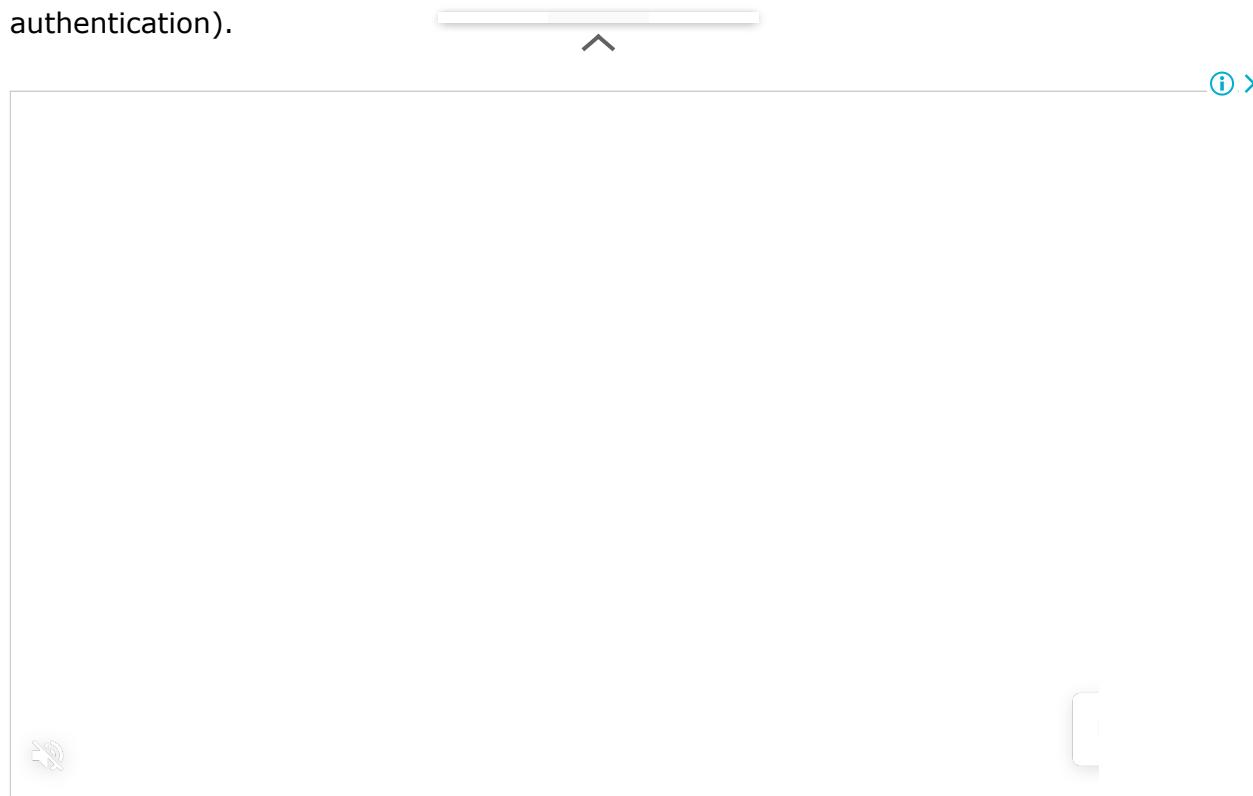
(<https://www.bezkoder.com/spring-boot-swagger-3/>)

Caching: Spring Boot Redis Cache example

(<https://www.bezkoder.com/spring-boot-redis-cache-example/>)

## Source Code

You can find the complete source code for this tutorial on Github (<https://github.com/bezkoder/spring-boot-spring-security-jwt-authentication>).



authentication (<https://www.bezkoder.com/tag/authentication/>)

authorization (<https://www.bezkoder.com/tag/authorization/>)

jwt (<https://www.bezkoder.com/tag/jwt/>)

spring boot (<https://www.bezkoder.com/tag/spring-boot/>)

spring security (<https://www.bezkoder.com/tag/spring-security/>)

token based authentication (<https://www.bezkoder.com/tag/token-based-authentication/>)



## 402 thoughts to “Spring Boot Token based Authentication with Spring Security & JWT”

**Yassine**

May 15, 2022 at 4:11 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-22218>)

Thanks BezKoder for the tutorial, it was very helpful.

And for every one who has an error with getting some data, getting  
“Unauthorized” error, just change :

`@PreAuthorize("hasRole('Admin')")`

to :

`@PreAuthorize("hasRole('ROLE_ADMIN')")`

This is how i resolved the issue, and it may depend on how roles are  
saved in your database 😊



hi bezkoder i really enjoy your content on spring jwt authentication.  
bless you bro.

**M.Nizar**

March 9, 2022 at 8:49 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-20085>)

Hi, thanks for this helpful tutorial, my question is about disabling csrf  
(....csrf().disable ).

By using spring boot security with JWT, do we need to disable csrf? isn't  
my application vulnerable to csrf attacks by doing so? If not is it  
vulnerable to XSS attacks and how can I protect it ?

**hunter**

March 1, 2022 at 1:21 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19817>)

hello and thx for this tuto.

After signing up and signing in (as admin) I've got this token  
"eyJhbGciOiJIUzUxMiJ9.eyJzdWliOjhZG1pbilsImIhdCI6MTY0NjEzOTg2Ny  
wiZXhwIjoxNjQ2MjI2MjY3fQ.Yc14R2gDZWIPQmxmRQ4TnPpj73el8pv-  
QTovuHZVEFf\_-otD8Wkeso6OVn6xYm5LgihiqtNp\_RTExCZmCp\_UA"  
when I tried to access to "localhost:8080/api/test/admin" with value

QTovuHZVEFF\_-otD8Wkeso6OVn6xYm5LgIhiqtNp\_RTExCZmCp\_UA" i've  
got 403 Forbidden error  
Any help please

**bezkoder**

March 2, 2022 at 2:19 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19841>)

Hi, you need to add “Bearer [token]” on your HTTP request  
Authorization header.

**Kai**

February 21, 2022 at 3:21 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19659>)

(My English is not very well.)  
I got the token = “undefined” issue too.  
And I compared all the code between my code and this example code.  
  
I found this.  
I use lombok in my project.  
In the “public class JwtResponse”.  
It use different name of the  
“private String token”

```
String email, List roles) {  
    this.token = accessToken;  
    this.id = id;  
    this.username = username;  
    this.email = email;  
    this.roles = roles;  
}  
///
```

I remove the lombok and use the same code with the example.  
or I keep my lombok and change my field name from "private String token;" to "private String accessToken;".  
Both can resolve my problem.

Hope it will help.

**Witcher**

February 16, 2022 at 2:13 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19523>)

Hi!

Thank you for your tutorial it helped me a lot!  
Would it be possible from your side to guide me with some additional case related to this tutorial? I want to create put request – editing username. I was trying to do something like this:

```
@PutMapping("/usernameupdate/{id}")
public ResponseEntity updateUsername(@PathVariable("id"
    Optional user = userRepository.findById(id);
    if (userRepository.existsByUsername(usernameUpdateRequest.getUsername()))
        return ResponseEntity
            .badRequest()
            .body(new MessageResponse("Error: User with name " + username + " already exists"));
    else{
        user.setUsername(usernameUpdateRequest.getUsername());
        userRepository.save(user);
        return ResponseEntity
            .body(new MessageResponse("Username is updated successfully"));
    }
}
```

but i have issues. I added usernameUpdateRequest class and added findByld in userrepo. Problem is, that findByld returns User instead of user and it complicates the whole thing. Best regards!

**avinash**

February 13, 2022 at 5:53 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19422>)

thank you so much so nicely instructed.

**Dhazki**

February 8, 2022 at 10:56 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19239>)

Thank you for great example, but I can't find, where the configuration for "Forbidden" error handling class?

**Bogo**

February 14, 2022 at 9:54 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19454>)

I have the same question!

**Plibr**

February 2, 2022 at 1:50 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19042>)

Hi, bezkoder, useful tut!

What about adding next entities to this REST API?

For example, adding that <https://www.bezkoder.com/spring-boot-file-upload/> (<https://www.bezkoder.com/spring-boot-file-upload/>) to this auth app



2. What kind of checking process we should follow for token confirmation in Controller before any endpoint getting handled?

Thanks a lot?

**Bhumika**

January 22, 2022 at 7:53 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-18617>)

Hi,

I have implemented the above for my project . I need to write test cases but am getting "Error :unauthorised" ,which I think is coming security/jwt/AuthEntryPointJwt.java.

I have annotated other end points with @PreAuthorize("hasRole('XYZ')")

.

What should I mock to avoid this error ?

**del bechir**

December 14, 2021 at 5:18 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-16601>)

Hello Guys, Hope that y're fine!!

Firstly, I want to inform that my english isn't good, but Plz I will try to explain my problem like that:

```
{  
    "id": 1,  
    "username": "mod",  
    "email": "[email protected] (/cdn-cgi/l/email-protection)",  
    "roles": [  
        "ROLE_MODERATOR",  
        "ROLE_USER"  
    ],  
    "accessToken": "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJtb2QiLCJp  
YXQiOjE2Mzk0NTg3MDYsImV4cCI6MTYzOTU0NTEwNn0.uue5gH2HX9R8Wdxp7  
xP2cNMXPi3uRVsFgmdGVzG77QqsxL9AUFfwT3W_RTyWaoKHCTyfPKr8KuvNao  
JQNz_-nA",  
    "tokenType": "Bearer"  
}
```

But when I want to access to the mod or user content, I'm not able to see it!!

Response of access:

```
{  
    "path": "/api/test/mod",  
    "error": "Unauthorized",  
    "message": "Full authentication is required to access thi  
s resource",  
    "status": 401  
}
```

I also note that on my logger console, i have this message  
"Unauthorized error: Full authentication is required"

**bezkoder**

December 14, 2021 at 9:44 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-16612>)

Hi, you can check your HTTP Authorization Headers with  
Bearer Token. It should be:

Bearer eyJhbGciOiJ...

**del bechir**

December 16, 2021 at 5:48 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-16678>)

Thanks for your answer!  
It works now..

**chetanpail**

March 21, 2022 at 12:59 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-20479>)

where exactly must we see this http authorization header  
please

**Eduardo**

December 4, 2021 at 3:09 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-16189>)

Hi I'd like to know where ~~dispatcherservlet~~ fits in in that picture

**pras**

November 16, 2021 at 1:33 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-15288>)

Hi, very good article.

Can the roles get from database?

@PreAuthorize("hasRole('USER') or hasRole('MODERATOR') or hasRole('ADMIN')")

Thank you

**Ritu**

November 15, 2021 at 6:06 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-15270>)

what is "Bearer token" in header

when i call "localhost:8080/api/auth/signin" i am getting error;



```
"path": "/api/auth/signin"  
  
AuthTokenFilter->request: org.springframework.[email protected]  
(/cdn-cgi/l/email-protection)  
AuthTokenFilter->headerAuth: null  
AuthTokenFilter->jwt: null
```

**rahul**

November 9, 2021 at 9:52 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-15042>)

can we use email instead of username to login using current method?

**Jeff**

October 19, 2021 at 8:37 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-14138>)

Great tutorial(s)!

Question around best practices. For the signup, could you also return a token or is it better to keep them separate and have the client app (angular in this case) call login after a successful signup?

Thanks!

**Paolo**

October 12, 2021 at 9:05 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-13536>)

Hi, thank you for the excellent tutorial!

I have a problem, when I sign up/sign in with Postman all work well, but when I do a GET request api/test/..., I receive UNEXPECTED ERROR, access denied, the error is on the line filterChain.doFilter(request, response); in AuthTokenFilter class, can you help me?

thank you

**bezkoder**

October 13, 2021 at 3:28 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-13574>)

Hi, you can check the Authorization header with Bearer token.

**Krutz**

October 7, 2021 at 1:01 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-13231>)

Thank you.



**Rafael Lima**

October 2, 2021 at 4:06 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-12924>)

Thanks for the tutorial. I had no problem at all!!

**Suresh**

August 14, 2021 at 9:41 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10828>)

Hello bezkoder,

Thanks for the tutorial ! I have followed it, and application starts well, test data entered. When I try to sing up, I am getting a 401. Not sure what's wrong in my request, should be something small, someone reading this could also help to solve the issue, TIA !

URL - <http://localhost:8080/api/auth/signup>  
[\(http://localhost:8080/api/auth/signup\)](http://localhost:8080/api/auth/signup)

Req - POST

Payload

```
{  
  "username": "testadmin",  
  "email": "[email protected] (/cdn-cgi/l/email-protection)",  
  "password": "*****",  
  "role": ["admin"]}
```



**Asam Alabdalah**

August 28, 2021 at 12:28 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-11321>)

Hello, please how ~~aid~~ you resolve this. Thanks

**Sean (<http://sean.com>)**

August 9, 2021 at 8:16 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10634>)

I have a classCastException when I use the UserDetailsImpl class for sign in request.

It is saying can not cast String to UserDetails interface.

**symphorien**

August 9, 2021 at 1:42 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10625>)

hi i have a problem!

all the requests that I make after my authentication do not go through  
in fact I do not have access to resources. all my API requests other  
than those from the Auth package do not pass authorization default.



**bezkoder**

August 10, 2021 at 2:03 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10644>)

Hi, check your Authorization Header with Bearer prefix first.

**Noobie**

August 2, 2021 at 1:37 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10335>)

I solved it and to all my fellows peers looking for Full authentication is required to access this resource, i change the url to "/signup". You have to add this "url" to WebSecurityConfig class as well. Cheers

**Noobie**

August 2, 2021 at 1:26 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10334>)

Hi bezkoder ,

Can you please help ? I received : Unauthorized error: Full authentication is required to access this resource for the signup API call to spring boot. This is the first initial sign up call. There should be no token available for the bearer during sign up right? Can please

.security.AuthEntryPointJwt

Unauthorized error: Full authentication is required to access this resource



**brian**

July 27, 2021 at 8:06 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10133>)

Awesome tut. Extremely comprehensive and easy to understand.  
Thanks!!!



**Hanifa**

July 11, 2021 at 8:11 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9538>)

Hello, thank you for your tutorial, it helps me a lot.  
I want to ask, why you use Set instead of List for Roles mapping on user object? meanwhile, you convert it to list too...



**Jafor Iqbal**

**Rajeev**

June 27, 2021 at 3:09 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9312>)

```
{  
    "username":"user",  
    "password":"%&67#&46"  
}
```

I am sending encoded password how can we decode and validate and successfully login?

**Ramil Yabbarov**

June 25, 2021 at 12:53 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9286>)

Hell, my IDEA ia angry about this: Field Injection is not Recommended  
Can i ignore this, or please give example of Constructor injection

@Autowired

```
UserDetailsServiceImpl userDetailsService;
```

@Autowired

```
@rivate AuthEntryPointJwt unauthorizedHandler;
```



**Matt**

June 17, 2021 at 4:18 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9139>)

Hey fellow tutorial followers,



I was also encountering this error:

`o.s.s.c.bcrypt.BCryptPasswordEncoder : Encoded password does not look like BCrypt`

I fixed it by changing the `passwordEncoder` method in the `WebSecurityConfig` from this:

```
@Bean  
public PasswordEncoder passwordEncoder() {  
    return new BCryptPasswordEncoder();  
}
```

to this:

```
@Bean  
public PasswordEncoder passwordEncoder() {  
    return PasswordEncoderFactories.createDelegatingPasswordEncoder();  
}
```

Hope this helps some people.

**Norwyn**

August 18, 2021 at 3:40 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10967>)

I am getting this error after changing like you suggested  
java.lang.IllegalArgumentException: There is no  
PasswordEncoder mapped for the id "null"

**Than**

February 22, 2022 at 8:27 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19678>)

That is probably because you directly inserted your user data into the database. Therefore, the password is not encrypted. Use the signup API to create your user data.

**JohnSnowbell**

June 14, 2021 at 2:20 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9070>)

Hi Bezkoder,

Thank you for your tutorial!

the following error : "query did not return a unique result: 2; nested exception is javax.persistence.NonUniqueResultException: query did not return a unique result: 2".

I can't figure how to solve this problem...

Thank you!

**Victor**

June 12, 2021 at 4:10 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9038>)

Hi bezkoder, I have implemented the application and everything works fine in oracle DB 12c but there is a problem. I can successfully register user from postman and if I check my DB I can see the data saved including hashed password, when I try to login with the same credentials I get the error below:

o.s.s.c.bcrypt.BCryptPasswordEncoder : Empty encoded password  
v.i.crm.security.jwt.AuthEntryPointJwt : Unauthorized error: Bad credentials.

Please help thanks.

**Matt**

June 21, 2021 at 7:20 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9039>)

**Vexees**

May 31, 2021 at 3:58 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8789>)

Hi Bezkoder,



First of all, thank you for your tutorial, it's very clear !

Do you plan to do a tutorial on the back and/or front side which present how to manage the users table ?

From a production perspective, to my mind, it would complete this series of tutorials (The superuser would be admin which can manage moderators and users)

Thank you!

**Huginn**

May 25, 2021 at 11:22 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8709>)

hello man why are the data not saved on my db? I use the oracle 19c! after registration, no data is saved



Hi, the source code configuration works with MySQL database,  
not Oracle. So you need to use Oracle dependency in *pom.xml*/  
and configure data source and hibernate in  
*application.properties* file 😊

**Mine**

May 24, 2021 at 8:07 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8684>)

Hello, when I create a constructor in the sign in section return  
`JwtResponse` , what will the String correspond to jwt?

```
@PostMapping("/signin")
    public ResponseEntity authenticateUser(@Valid @Request
    ...
    ...
    return ResponseEntity.ok(new JwtResponse(jwt,
        userDetails.getUsername(),
        userDetails.getPassword(),
        roles));
}
```



```
public class JwtResponse {  
  
    private String token;  
    private String type = "Basic";  
    private String username;  
    private String password;  
    private List roles;  
  
    public JwtResponse(String jwt, String username, String  
.....=jwt  
        this.username = username;  
        this.password = password;  
        this.roles = roles;  
    }  
}
```

**Syed**

May 16, 2021 at 7:37 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8574>)

Hello...,

For sign in purpose....,

Instead of UserName can we use email ID for sign in.....

**Karol**

May 13, 2021 at 6:56 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8520>)

It has been best guide I have ever seen in my entire life! Great staff.  
Clearly explained.

**Loic-tech**

April 30, 2021 at 2:27 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8329>)

very Brilliant !!!!!

**Alex**

April 26, 2021 at 2:03 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8285>)

Hello,

I always get 401 Full Authentication required when trying to get any protected resource. I have the Authorization Header and the Token on all requests, also the application-type JSON headers I still always get the error, both from Angular and from Postman.



**bezkoder**

April 27, 2021 at 3:16 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8289>)

Hi, please make sure that you have 'Bearer ' string as prefix of the token.

**Alex**

April 27, 2021 at 8:23 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8291>)

I have it, still have the same error.

I think it's a problem with creating the authentication, maybe there's something that needs to be changed with the security config. The authentication seems to never be created and sending the token does nothing unfortunately.

**Alex**

April 27, 2021 at 8:56 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8292>)

UPDATE:

Make sure you DON't add  
@Order(Ordered.HIGHEST\_PRECEDENCE) to the  
security config or any of the jwt classes. I added it to  
avoid some error from csrf, but it ended up ruining  
the authentication.

**Dominik**

April 19, 2021 at 8:55 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8178>)

I got that error after sending

```
{  
    "username": "mod",  
    "email": "[email protected] (/cdn-cgi/l/email-protection)",  
    "password": "12345",  
    "role": ["mod", "user"]  
}
```

to POST `http://localhost:8080/api/auth/signup`  
(`http://localhost:8080/api/auth/signup`) by Postman  
and im getting this kind of error

```
{  
    "timestamp": "2021-04-19T08:47:32.067+00:00",  
    "status": 415,  
    "error": "Unsupported Media Type".
```

**bezkoder**

April 19, 2021 at 11:10 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8180>)

Hi, you should set `Content-Type: application/json` in your HTTP request Header.

**Ali s**

April 18, 2021 at 7:17 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8166>)

Hi M8!

Great tutorial!

just one thing my spring runs fine and finds my database and all.

It just does not add any tables to it at all.

Any idea why?

I get no error messages.

Kind regards

Tags

You should set hibernate ddl to create, or update. See the code below  
spring.jpa.hibernate.ddl-auto = update

**Kerim**

April 17, 2021 at 5:27 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8141>)

Hi man thanks , it was realy a nice tutorial ,but i need help i tested the /signup post request it did worked and i checked my data Base it was implemented with a user but when i try /signin with the same user that i just did i got {

```
"timestamp": "2021-04-17T05:12:02.331+00:00",
"status": 401,
"error": "Unauthorized",
"message": "",
"path": "/pcd/auth/signin"
}
```

ps: i changed the path to ("/pcd/auth/\*\*").permitAll() in the webConfig the Jwt token is not generating its directly rendering into Authentication Entry point and displaying the message pleaase help

hi i resolved it thanks :DDD

**Tega**

June 18, 2021 at 11:04 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9155>)

Hello, please how did you resolve this. Thanks

**Asam Alabdalah**

August 28, 2021 at 12:24 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-11320>)

Hello, please how did you resolve this. Thanks

**bennie**

April 15, 2021 at 11:53 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8109>)

Howdy! This post could not be written much better!

Reading through this tutorial helps me alot about Spring Security with JWT!

**deneme**

April 13, 2021 at 11:58 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8075>)

I did everything the same but I get ~~this~~ error when I try to login

2021-04-13 14:56:47.174 WARN 105872 — [nio-8080-exec-2]

o.s.s.c.bcrypt.BCryptPasswordEncoder : Encoded password does not look like BCrypt

2021-04-13 14:56:47.187 ERROR 105872 — [nio-8080-exec-2]

c.a.A.security.jwt.AuthEntryPointJwt : Unauthorized error: Bad credentials

**Anil**

April 13, 2021 at 8:35 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8074>)

Hi Sir,

I am getting below issue while signup user:

```
{  
  "timestamp": "2021-04-13T08:33:24.035+00:00",  
  "status": 401,  
  "error": "Unauthorized",  
  "path": "/app/auth/signup"  
}
```

**Duy**

April 12, 2021 at 11:55 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8053>)

My user not have enabled column ~~and~~ when i try to login, server said that can not authorization cause no enabled column, so how to get over it

**korn**

March 26, 2021 at 9:57 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7766>)

Hello, I have a questioned why you picked relatively old lib jjwt from Jul, 2018 ?

Any reason behind that?

I see as I assume continuation of this lib but nobody uses it:

<https://mvnrepository.com/artifact/io.jsonwebtoken/jjwt-root>  
<https://mvnrepository.com/artifact/io.jsonwebtoken/jjwt-root>

Any thoughts?

**Sofia**

March 26, 2021 at 3:01 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7766>)

I have an issue with the <http://localhost:8080/api/auth/signin> (<http://localhost:8080/api/auth/signin>). I have created the roles in the role table and I was able to create some users from <http://localhost:8080/api/auth/signup> (<http://localhost:8080/api/auth/signup>). The problem comes when I want to sign in.

I create an user with the following parameter -> {"username": "admin", "email": "[email protected] (/cdn-cgi/l/email-protection)", "password": "admin", "role": ["admin", "mod", "user"]} and then when I tried to sign in with "username": "admin" and "password": "admin" I have a 415 Unsupported Media Type.

I'm sure I am using the POST method, and I tried to change from application/json to application/x-www-form-urlencoded with no success.

If you can help me I will be very thankful!

Thanks anyway for this great tutorial. Your GitHub was very helpful

**Mayur Kandalkar**

March 23, 2021 at 2:24 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7727>)

Thank you so much. I have got so much to learn.

March 22, 2021 at 7:31 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7713>)

Hi.

I'm having this issue like when I try to fetch details roles are not coming. I'm getting an empty array. kindly help if someone knows how to resolve this issue.



Thanks in advance

**bezkoder**

March 23, 2021 at 8:05 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7718>)

Hi, you should use SQL to create 3 records in `roles` table first.

**sathiyaraj**

March 21, 2021 at 7:17 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7697>)

Resolved

[`org.springframework.web.HttpMediaTypeNotAcceptableException`:  
Could not find acceptable representation]



Hallo Bezkoder ...

It greats tutorial, thanks a lot for your sharing.

I have a silly questions but it make me confuse. User have 1 role,  
Moderator have 2 roles and Admin have only 1 roles. Where I must  
set it? How do I set Admin with 3 roles ?

Thank you in advance

**khouloud yengui**

March 3, 2021 at 2:44 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7363>)

hello , thank you for these great tutorial .

I changed in application.properties file :

```
spring.datasource.url=jdbc:mysql://localhost:3306/testdb?  
serverTimezone=UTC  
spring.datasource.username=root  
spring.datasource.password=  
spring.datasource.driver-class-name= com.mysql.cj.jdbc.Driver  
  
spring.jpa.database= MYSQL  
spring.jpa.show-sql=true  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.properties.hibernate.dialect=  
org.hibernate.dialect.MySQL55Dialect
```

```
# App Properties
bezkoder.app.jwtSecret= bezKoderSecretKey
bezkoder.app.jwtExpirationMs= 86400000
so the tables were generated automaticly and i can open my database
with phpMyAdmin ,but when i triedto send this request
http://localhost:8080/api/auth/signup
(http://localhost:8080/api/auth/signup) in Postman, i have this error :
POST http://localhost:8080/api/auth/signup
(http://localhost:8080/api/auth/signup)
Error: connect ECONNREFUSED 127.0.0.1:8080
Request Headers
Content-Type: application/json
User-Agent: PostmanRuntime/7.26.8
Accept: */*
Cache-Control: no-cache
Postman-Token: bd8240e4-e192-41bb-9b10-477b324d7e0e
Host: localhost:8080
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Request Body
{
  "username": "mod",
  "email": "[email protected] (/cdn-cgi/l/email-protection)",
  "password": "1234",
  "role": ["mod", "user"]
}
```

**xcalib**

February 28, 2021 at 11:17 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7306>)

Great tutorial, thanks so much! 

Wondering what's the difference between the following types of exports:

`export default new AuthService()` vs `export default App?`

**Vazgen**

February 26, 2021 at 7:44 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7270>)

what about payload package. i am beginner and dont know what will request and what response. please provide me

**bezkoder**

February 27, 2021 at 10:08 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7289>)

Hi, just think payload as the content (or data) of http request/response 😊

**kapil**

January 30, 2021 at 9:18 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6697>)

Signup: <http://localhost:8080/signup> (<http://localhost:8080/signup>)

```
body:- {  
    "username": "kapil",  
    "email": "[email protected] (/cdn-cgi/l/email-protection)",  
    "password": "12345",  
    "role": [  
        "ROLE_ADMIN"  
    ]  
}
```

Response:-

```
{  
    "timestamp": "2021-01-30T21:15:37.537+00:00",  
    "status": 401,  
    "error": "Unauthorized",  
    "message": "Error: Unauthorized",  
    "path": "/signup"  
}
```

bezkoder

**Eddie**

January 24, 2021 at 9:35 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6507>)

Hi, I have the following error when trying POST signup request:

```
2021-01-24 16:32:42.688 ERROR 1136 — [nio-9095-exec-9]
c.e.s.security.jwt.AuthEntryPointJwt : Unauthorized error: Full
authentication is required to access this resource
2021-01-24 16:33:24.495 ERROR 1136 — [nio-9095-exec-2] o.a.c.c.C.[.[].
[/.][dispatcherServlet] : Servlet.service() for servlet [dispatcherServlet]
in context with path [] threw exception

org.springframework.security.web.firewall.RequestRejectedException:
The request was rejected because the URL contained a potentially
malicious String "//"
at
org.springframework.security.web.firewall.StrictHttpFirewall.rejectedBl
ocklistedUrls(StrictHttpFirewall.java:456) ~[spring-security-web-
5.4.2.jar:5.4.2]
at
org.springframework.security.web.firewall.StrictHttpFirewall.getFirewal
ledRequest(StrictHttpFirewall.java:429) ~[spring-security-web-
5.4.2.jar:5.4.2]
at
```

```
Proxy.java:183) ~[spring-security-web-5.4.2.jar:5.4.2]
at
org.springframework.web.filter.DelegatingFilterProxy.invokeDelegate(
DelegatingFilterProxy.java:358) ~[spring-web-5.3.3.jar:5.3.3]
at
org.springframework.web.filter.DelegatingFilterProxy.doFilter(Delegati
ngFilterProxy.java:271) ~[spring-web-5.3.3.jar:5.3.3]
at
org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(Applic
ationFilterChain.java:193) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilt
erChain.java:166) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.springframework.web.filter.RequestContextFilter.doFilterInternal(R
equestContextFilter.java:100) ~[spring-web-5.3.3.jar:5.3.3]
at
org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePer
RequestFilter.java:119) ~[spring-web-5.3.3.jar:5.3.3]
at
org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(Applic
ationFilterChain.java:193) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilt
erChain.java:166) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
```

```
RequestFilter.java:119) ~[spring-web-5.3.3.jar:5.3.3]
at
org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(Applic
ationFilterChain.java:193) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilt
erChain.java:166) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.springframework.web.filter.CharacterEncodingFilter.doFilterIntern
al(CharacterEncodingFilter.java:201) ~[spring-web-5.3.3.jar:5.3.3] at
org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePer
RequestFilter.java:119) ~[spring-web-5.3.3.jar:5.3.3]
at
org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(Applic
ationFilterChain.java:193) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilt
erChain.java:166) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrap
perValve.java:202) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.apache.catalina.core.StandardContextValve.invoke(StandardConte
xtValve.java:97) ~[tomcat-embed-core-9.0.41.jar:9.0.41]
at
org.apache.catalina.authenticator.AuthenticatorBase.invoke(Authentic
```

```
at  
org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.j  
ava:92) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
org.apache.catalina.core.StandardEngineValve.invoke(StandardEngine  
Valve.java:78) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.j  
ava:343) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
org.apache.coyote.http11.Http11Processor.service(Http11Processor.j  
ava:374) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
org.apache.coyote.AbstractProcessorLight.process(AbstractProcessorLi  
ght.java:65) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
org.apache.coyote.AbstractProtocol$ConnectionHandler.process(Abstr  
actProtocol.java:888) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
org.apache.tomcat.util.net.NioEndpoint$SocketProcessor.doRun(NioEn  
dpoint.java:1597) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
org.apache.tomcat.util.net.SocketProcessorBase.run(SocketProcessorB  
ase.java:49) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at  
java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadP
```

at

```
org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(Ta  
skThread.java:61) ~[tomcat-embed-core-9.0.41.jar:9.0.41]  
at java.base/java.lang.Thread.run(Thread.java:834) ~[na:na]
```

2021-01-24 16:33:24.547 ERROR 1136 — [nio-9095-exec-2]

c.e.s.security.jwt.AuthEntryPointJwt : Unauthorized error: Full  
authentication is required to access this resource

Would you please tell me what am I missing?

Thanks!

**Eddie**

January 24, 2021 at 9:46 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6509>)

No worry, I figure it out. The issue is I am having // (instead of /) from my Post end-point url. After correcting, everything is working.

**Peter**

January 11, 2021 at 10:36 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6178>)

**JOSHUA**

January 9, 2021 at 5:14 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6132>)

Thanks a lot for the detailed explanation.

Please if I want to use this as an auth server micro service, how do I go about that? Thanks

**hung**

January 5, 2021 at 6:56 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6091>)

Just desire to say your Spring Boot tutorial is amazing!

**Shiro**

January 4, 2021 at 7:15 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6076>)

Thank you for this tutorial. It was really easy to understand it.

Can you upload a video or code on forgetting password in this code?

Hi, This tutorial was very helpful to me. Thank you!

**Yoesoff**

December 30, 2020 at 4:35 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5990>)

Hi, I am wondering why your signin response is different compared mine, here is my signin response result:

```
{  
    "token": "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJ1c2VyMSIsImlt  
    "type": null,  
    "id": 5,  
    "username": "user1",  
    "email": "[email protected]",  
    "roles": [  
        "ROLE_USER"  
    ]  
}
```

My type is always returns null "type": null, (where did you define tokenType and accessToken?)

Hi, you can find type = "Bearer" in  
payload/response/JwtResponse.java :

```
public class JwtResponse {  
    private String token;  
    private String type = "Bearer";  
    private Long id;  
    ...
```

**nawfel bougrine**

December 28, 2020 at 7:24 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5959>)

good evening, i did exactly the same as you but when i run my application i get the following error:

```
sun.misc.Unsafe.park(Native Method)  
java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:21  
5)  
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje  
ct.awaitNanos(AbstractQueuedSynchronizer.java:2078)  
java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueu  
e.take(ScheduledThreadPoolExecutor.java:1093)  
java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueu  
e.take(ScheduledThreadPoolExecutor.java:809)
```

```
or.java:1127)
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecu
tor.java:617)
java.lang.Thread.run(Thread.java:745)
2020-12-28 20:20:04.631 WARN 44708 — [ main]
o.a.c.loader.WebappClassLoaderBase : The web application [ROOT]
appears to have started a thread named [HikariPool-1 connection
adder] but has failed to stop it. This is very likely to create a memory
leak. Stack trace of thread:
sun.misc.Unsafe.park(Native Method)
java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:21
5)
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.awaitNanos(AbstractQueuedSynchronizer.java:2078)
java.util.concurrent.LinkedBlockingQueue.poll(LinkedBlockingQueue.ja
va:467)
java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.j
ava:1066)
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecut
or.java:1127)
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecu
tor.java:617)
java.lang.Thread.run(Thread.java:745)
2020-12-28 20:20:04.633 WARN 44708 — [ main]
ConfigServletWebServerApplicationContext : Exception encountered
during context initialization – cancelling refresh attempt:
```

```
2020-12-28 20:20:04.634 INFO 44708 — [ main]
com.zaxxer.hikari.HikariDataSource : HikariPool-1 – Shutdown
initiated...
2020-12-28 20:20:04.642 INFO 44708 — [ main]
com.zaxxer.hikari.HikariDataSource : HikariPool-1 – Shutdown
completed.
2020-12-28 20:20:04.666 INFO 44708 — [ main]
ConditionEvaluationReportLoggingListener :
Error starting ApplicationContext. To display the conditions report re-
run your application with 'debug' enabled.
2020-12-28 20:20:04.694 ERROR 44708 — [ main]
o.s.b.d.LoggingFailureAnalysisReporter :
```

\*\*\*\*\*

APPLICATION FAILED TO START

\*\*\*\*\*

#### Description:

An attempt was made to call a method that does not exist. The attempt was made from the following location:

org.hibernate.validator.internal.engine.ValidatorFactoryImpl.  
(ValidatorFactoryImpl.java:153)

The following method did not exist:

javax.validation.spi.ConfigurationState.getParameterNameProvider()Lj
avax/validation/ParameterNameProvider;



```
jar:file:/C:/Users/hp/.m2/repository/javax/validation/validation-
api/1.0.0.GA/validation-api-
1.0.0.GA.jar!/javax/validation/spi/ConfigurationState.class
jar:file:/C:/Users/hp/.m2/repository/jakarta/validation/jakarta.validatio
n-api/2.0.2/jakarta.validation-api- ^_
2.0.2.jar!/javax/validation/spi/ConfigurationState.class
```

The class hierarchy was loaded from the following locations:

```
javax.validation.spi.ConfigurationState:
file:/C:/Users/hp/.m2/repository/javax/validation/validation-
api/1.0.0.GA/validation-api-1.0.0.GA.jar
```

Action:

Correct the classpath of your application so that it contains a single, compatible version of javax.validation.spi.ConfigurationState

can someone help me please ?

**bezkoder**

December 29, 2020 at 1:59 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5965>)

Hi, first remove in your repository maven the directory hibernate and then make a `maven clean install` to reimport correctly your dependencies.

**PedroGuacamole**

December 27, 2020 at 8:52 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5940>)

You can't even imagine how grateful I am to you! Great tutorial!

**carmen**

December 21, 2020 at 10:01 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5875>)

I'm thoroughly enjoying your tutorial. I'm still new to the whole thing about Spring Boot and you made my day.

Thanks a lot!

**Flukii**

December 16, 2020 at 12:58 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5814>)

Thanks for this amazing tutorial. It was really easy to understand it.

I have a question, If I have a case that some users (from the database) require password from the database or some user required password from Ldap. Can you give me a hint?



**Aliza**

December 10, 2020 at 10:39 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5713>)

Thank you for this excellent tutorial! I was getting lost with other more complex tutorials and having trouble understanding how everything connected, but when I followed this one my authentication worked.

**Andres**

December 15, 2020 at 7:57 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5805>)

Hi, Aliza, did you tested

`http://localhost:8080/api/auth/signup  
(http://localhost:8080/api/auth/signup)`

with

```
{  
  "username": "mod",  
  "email": "[email protected] (/cdn-cgi/l/email-protection)",  
  "password": "12345678",  
  "role": ["mod", "user"]  
}
```

with postman I am getting 401 unauthorized, this happen to you

**Timati**

December 28, 2020 at 2:18 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5953>)

Hi, thank you for the very detailed tutorial!

Have a question:

Why you user 'parseClaimsJws' in function  
validateJwtToken of JwtUtils class  
instead of 'parseClaimsJwt'?

**andres**

December 9, 2020 at 8:45 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5704>)

Hi, I followed the tutorial, it runs but the tables are not being created.

I already tested the connection to postgres, check in the browser for the login. Everything looks good but the tables are not being created, I changed the spring.jpa.hibernate.ddl-auto to create but still nothing.

I appreciate any guidance, thanks.

**ahmad**

I cloned your code and the frontend too

it works fine

I added a user made him an admin and mod and user  
and was added successfully but when I sign in and click on user or  
admin board they both say UNAUTHORIZED  
although im signed in  
why is that?

**bezkoder**

December 3, 2020 at 2:01 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5631>)

Hi, please make sure that you used Bearer token on  
Authorization header.

**fero**

November 27, 2020 at 8:31 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5587>)

Hello BezKoder, how can i update (put) user\_role table in postman?

hello bezcoder ,thank you for this amazing tutorial !! but if i want to add new class (Employee) and make this class as a User .. what i have to do ..should i use inheritance or relationship between them ?

**Brian**

November 26, 2020 at 12:36 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5566>)

Looks like to fix the issue I posted previously:

Description: A component required a bean named 'entityManagerFactory' that could not be found. Action: Consider defining a bean named 'entityManagerFactory' in your configuration.

[INFO]

I needed to delete the org.hibernate directory from my local maven repository (.m2)

<https://stackoverflow.com/questions/54825926/spring-boot-2-1-entitymanagerfactory-bean-could-not-be-found>  
[\(https://stackoverflow.com/questions/54825926/spring-boot-2-1-entitymanagerfactory-bean-could-not-be-found\)](https://stackoverflow.com/questions/54825926/spring-boot-2-1-entitymanagerfactory-bean-could-not-be-found)

**Brian**

November 26, 2020 at 12:26 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/>)

I'm getting the following error when running mvn spring-boot:run - debug

```
WARN 8120 — [ main] ConfigServletWebServerApplicationContext : Exception encountered during context initialization – cancelling refresh attempt: org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'userRepository' defined in com.bezkoder.springjwt.repository.UserRepository defined in @EnableJpaRepositories declared on JpaRepositoriesRegistrar.EnableJpaRepositoriesConfiguration: Cannot create inner bean '(inner bean)#6b2e46af' of type [org.springframework.orm.jpa.SharedEntityManagerCreator] while setting bean property 'entityManager'; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name '(inner bean)#6b2e46af': Cannot resolve reference to bean 'entityManagerFactory' while setting constructor argument; nested exception is org.springframework.beans.factory.NoSuchBeanDefinitionException: No bean named 'entityManagerFactory' available
```

2020-11-26 07:14:03.393 INFO 8120 — [ main]

```
o.s.s.concurrent.ThreadPoolTaskExecutor : Shutting down ExecutorService 'applicationTaskExecutor'
```

2020-11-26 07:14:03.527 INFO 8120 — [ main]

```
o.apache.catalina.core.StandardService : Stopping service [Tomcat]
```

2020-11-26 07:14:03.580 INFO 8120 — [ main]

```
ConditionEvaluationReportLoggingListener :
```

Error starting ApplicationContext. To display the conditions report re-

\*\*\*\*\*

APPLICATION FAILED TO START

\*\*\*\*\*

Description:



A component required a bean named 'entityManagerFactory' that could not be found.

Action:

Consider defining a bean named 'entityManagerFactory' in your configuration.

[INFO] -----

[INFO] BUILD FAILURE

**Robert**

November 25, 2020 at 3:26 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5554>)

Hi, first of all I would like to say its great article, thank you for your effort.

I have one question: how to change login field to another field from User entity? E.g. I want to use email and password for login. How to force UsernamePasswordAuthenticationToken to get as principal email instead of username ?



**Donatas**

November 23, 2020 at 6:11 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5527>)

Hi, I have registered a new user with role "mod". When I sign in with it and calling a GET `http://localhost:8080/api/test/mod` (`http://localhost:8080/api/test/mod`) URL, I am getting an error in "postman"

```
{  
    "timestamp": "2020-11-23T17:58:29.065+0000",  
    "status": 401,  
    "error": "Unauthorized",  
    "message": "Error: Unauthorized",  
    "path": "/api/test/mod"  
}
```

Can you help me? I have taken this tutorial with git clone and I was doing everything step by step. Some people got into the same trouble above in comments but didn't get the answer.

Also, thanks for this tutorial. 10/10!

**bezkoder**

November 24, 2020 at 1:53 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5534>)

**Andres**

December 15, 2020 at 11:00 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5808>)

Hi Bezkoder,

So that means that I need to login, but because there is no user created yet, I need to create the first one to login and to be able to get a token ?

If I do that, by curl it would be something like this ? (I think the confusion is because following the tutorial some of us assume that we can use the signup with no need for credentials...that make sense?, thank you.)

```
TOKEN=$(curl -s -X POST -H 'Accept: application/json' -H  
'Content-Type: application/json' -d  
'{"username":"{username}","password":"{password}","reme  
mberMe":false}' https://(https://)  
{hostname}/api/authenticate | jq -r '.id_token')  
  
curl -H 'Accept: application/json' -H "Authorization: Bearer  
$TOKEN" -d 'name=the_name' -d 'em[email protected]  
(/cdn-cgi/l/email-  
protection)'https://{hostname}/api/myresource
```

Hello, thank you for your great tutorial.

I have one question. This part of code shouldn't return error code when validation is false?

This code does nothing even if validation fails.

```
try {  
    String jwt = parseJwt(request);  
    if (jwt != null && jwtUtils.validateJwtToken(jwt)) {  
        String username = jwtUtils.getUserNameFromJwtToken(jwt);  
  
        UserDetails userDetails = userDetailsService.loadUserByUsername(username);  
        UsernamePasswordAuthenticationToken authentication = new UsernamePasswordAuthenticationToken(userDetails, null, userDetails.getAuthorities());  
        authentication.setDetails(new WebAuthenticationDetailsSource(request));  
  
        SecurityContextHolder.getContext().setAuthentication(authentication);  
    }  
} catch (Exception e) {  
    logger.error("Cannot set user authentication: " + e.getMessage());  
}
```

**bezkoder**

November 29, 2020 at 4:00 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5599>)

Spring Security filter chain will validate and return error code automatically.

**Vikram Kumar**

November 21, 2020 at 4:09 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5498>)

Hello bezkoder, i dont want signup page only i need is signin page so what should i change in the source code

**Enrique Gordon**

November 19, 2020 at 8:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5472>)

Hola bezcoder , amazing tutorial. Applying your tutorial i did try to implement the process in my app , but keep, but trying to access the user data i got a anonymous user, thus i dont know what to do!  
Any suggestion?

**bezkoder**

**Enrique Gordon**

November 23, 2020 at 9:47 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5521>)

the point is the user get logged and all works fine .. if i check my login method service implementation all data is there .

When the user logs in the app works .

But when trying to access that logged user from other method in other class through SecurityContextHolder i got an anonymous user

Please help....a better idea i set here:

<https://stackoverflow.com/questions/64927997/how-to-receive-the-authenticated-user-from-springsecurity-in-other-service-impl>

(<https://stackoverflow.com/questions/64927997/how-to-receive-the-authenticated-user-from-springsecurity-in-other-service-impl>)

Thanks in advance

**Enrique Gordon**

And by the way really thanks for your  
tutorials.Amazing !!!!

**Jasso jassinto**

November 19, 2020 at 5:47 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5470>)

Man you are those people with god level explanation, I was struggling  
with implementation in React at the auth states, i love you.

**Arturo Mayorga**

November 16, 2020 at 9:21 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5425>)

With this guide, there is no need of CSRF protection?

**Jasso jassinto**

November 19, 2020 at 5:56 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5471>)

I suggest you all noobs to avoid ask low level questions since this

I am sure that this quite polite gentleman developer want to help all of us, but it is annoying explaining all questions, he did a lot just sharing his code <3

**Manjunath**

April 30, 2021 at 2:15 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-8328>)

I do have same questions. Do I need to have CSRF protection enabled? Any additional checks required for CSRF?

**Linh Tran**

November 16, 2020 at 3:29 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5421>)

Hi,

I'm a beginner on spring security and JWT. I am so glad that I found your tutorial and I am following it now. It's really well explained with great structure. However, I am having a small problem with the post method for /signup.

When I ran the application in local and made a post request on endpoint /api/auth/signup, it returned a successful message. But when I checked the database, it shows empty cells on other columns of the

id | email | password | username

—+—+—+—

6 | | |

7 | | |

(2 rows)



Do you think what could be the possible causes for this problem?

Many thanks in advance.

**bezkoder**

November 17, 2020 at 2:06 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5429>)

Hi, you can check the body data with JSON format is correct or not.

**Linh Tran**

November 17, 2020 at 9:00 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5433>)

Here is a sample of my body data with JSON format.

Should be fine I think ?

```
{  
  "username": "eduard"
```

```
"password": "12356547567",
"role": ["user", "mod"]
}
```

**David**

November 13, 2020 at 1:28 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5386>)

Excellent guide! Bravo!

**Rohit**

November 5, 2020 at 6:15 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5293>)

I have got the problem at the signup. All things seems good but not working. Please help.

```
{
  "timestamp": "2020-11-05T06:04:19.643+00:00",
  "status": 404,
  "error": "Not Found",
  "message": "No message available",
  "path": "/api/auth/signup"
}
```

**Rohit**

November 7, 2020 at 1:35 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5316>)

Doing silly mistake that I cannot keep  
@RequestMapping("/api/auth").

Thank you for the great tutorial.

**Black widow**

November 3, 2020 at 2:55 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5273>)

Hi bezkoder !

Thanks a lot for this very good tutorial ! I am a beginner and this helped me a lot !

Now, I would like to add swagger-ui to my project and I have this error when I want to access to url localhost:8080/swagger-ui.html :  
c.m.d.a.m.s.jwt.AuthEntryPointJwt : Unauthorized error: Full authentication is required to access this resource

I tried to add .antMatchers("/swagger-ui.html").permitAll() to WebSecurityConfig but it's the same... I know it's outside of this tutorial but maybe you can help me and tell me what do I need to edit to make it work ?

Thanks !

**Edgard**

October 30, 2020 at 10:05 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5251>)

Hi , very good tutorial! But I'm having trouble when I call "signup". Inside "doFilterInternal", after verify if "jwt != null && jwtUtils.validateJwtToken(jwt)", this line is called:  
"filterChain.doFilter(request, response)" and bring me back an Unauthorized error.

I don't understand what exactly do this line, can any one help me?

I think is important to say that I'm not passing any value on my header(besides the postman defaults). I see in the example above that the header has 9 fields, but mine has 8. Do I need to send something in my header when I'm signingUp?

**Edgard**

October 30, 2020 at 10:22 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5252>)

here I have a better error  
description:InsufficientAuthenticationException,  
messageDetails:"Full authentication is required to access this resource"

October 30, 2020 at 1:48 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5244>)

Hello there,

First of all thank you so much for your tutorial, it is very clear and it helped me a lot !

The only question that remains for me is where are stored the tokens in this configuration ? Are they stored inMemory and if so, how could we change this aspect of your code ?

**egg**

November 7, 2020 at 9:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5327>)

+1

Does someone has the answer to that question ? I am trying to had an api logout

Wonderful tutorial indeed

**dev**

October 26, 2020 at 8:53 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5202>)

Thank you

**Ana**

October 21, 2020 at 6:31 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5143>)

Hello, thank you very much for sharing, excellent tutorial.

I did my project guiding me around here and it works fine for me except when I use the @PreAuthorize notation, it throws me the error:

Postman

```
{  
    "date": "2020-10-21T18:22:06.284+0000",  
    "code": 500,  
    "message": "Internal server error",  
    "description": "Access is denied",  
    "details": []  
}
```

Console

2020-10-21 15:22:06.273 WARN 19507 — [nio-9091-exec-9]

n.u.e.r.AEresponseEntityExceptionHandler : Exception.class ->  
(uri=/api/v1/users/4): Access is denied.

2020-10-21 15:22:06.372 WARN 19507 — [nio-9091-exec-9]

.m.m.a.ExceptionHandlerExceptionResolver : Resolved  
[org.springframework.security.access.AccessDeniedException: Access  
is denied]

Any ideas? Thanks

October 23, 2020 at 6:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5169>)

Hi, it didn't work for me with @PreAuthorize ("hasRole ('ADMIN')")  
but it work for me with @PreAuthorize ("hasAuthority ('ADMIN')")



**Jacky**

October 26, 2020 at 4:22 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5206>)

Would you mind showing your @PreAuthorize code?



**pap Malik**

October 17, 2020 at 5:20 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5095>)

Hello, thanks for this tutorial it was very helpful !  
but i wanted to do a Reset password with sending email using ur code  
but i had many problems  
Can u take a look thanks  
<https://stackoverflow.com/questions/64403483/how-can-i-change-modelandview-to-responseentity-in-controller-spring-security>  
(<https://stackoverflow.com/questions/64403483/how-can-i-change-modelandview-to-responseentity-in-controller-spring-security>)



**ravindar**

October 16, 2020 at 8:08 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5060>)

When i invoke the api(protected by jwt) through browser can i have a login form if yes please help on how?

**Jacky**

October 16, 2020 at 1:36 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5057>)

If Admin has to access database, should I create in TestController several 'access' methods?

For example, if Admin has to access users and orders, should I create GetMapping for users and GetMapping for orders by two different methods? Thank you.

**OswaldoMM**

October 14, 2020 at 1:43 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5034>)

Hi, thanks for this tutorial.

I'm in trouble, java throws an exception of 401:



I'm working with a simple login form with Thymeleaf and Spring Boot  
What could it be?  
Thank you!

**bezkoder**

October 15, 2020 at 2:40 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5040>)

Hi, please check if the HTTP request header has the correct  
Bearer Token.

**Jacky**

October 10, 2020 at 9:55 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4985>)

Hello. Why I can't reach localhost:8080/register nor any other link? It says: log shows: "Unauthorized error: Full authentication is required to access this resource". Hope you could help me. Thanks.

**bezkoder**

**Mihir**

October 5, 2020 at 8:09 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4929>)

Hi,

Excellent tutorial, easy to understand the concept.

I tried to implement the same, except that I do not want the signup page/utility. i.e. assume the user table will be populated by some other means. So I implemented everything as the tutorial except signup utility. When I try to login (signin) it fails even though record is fetched correctly.

2020-10-05 15:51:54.346 INFO 17544 — [nio-8071-exec-3]

com.example.controller.AuthController : Login....

2020-10-05 15:51:54.346 INFO 17544 — [nio-8071-exec-3]

com.example.controller.AuthController : User: mihir

..hibernate logs from user table fetched the row correctly.

..hibernate logs from role table to get the role correctly.

2020-10-05 15:51:54.750 WARN 17544 — [nio-8071-exec-3]

o.s.s.c.bcrypt.BCryptPasswordEncoder : Encoded password does not look like BCrypt

2020-10-05 15:51:54.759 ERROR 17544 — [nio-8071-exec-3]

c.b.iris.security.jwt.AuthEntryPointJwt : Unauthorized error: Bad credentials

```
{  
    "username": "mihir",  
    "password": "test"  
}
```

"test" is also saved (and returned in logs) from database.

What am I missing here?

**Micko**

September 26, 2020 at 6:19 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4863>)

Excellent tutorial!

**Savan Java**

September 19, 2020 at 8:51 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4793>)

\*\*\*\*\*

APPLICATION FAILED TO START

\*\*\*\*\*

Description:

'com.bezkoder.springjwt.repository.UserRepository' that could not be found.

The injection point has the following annotations:

-

@org.springframework.beans.factory.annotation.Autowired(required=true)

Action:

Consider defining a bean of type 'com.bezkoder.springjwt.repository.UserRepository' in your configuration.

Jliv

September 19, 2020 at 7:17 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4792>)

Hi,

```
private String parseJwt(HttpServletRequest request) {  
    String headerAuth = request.getHeader("Authorization");  
    System.out.println(request.getHeaderNames());  
  
    if (StringUtils.hasText(headerAuth) && headerAuth.startsWith("Bearer  
    ")) {  
        return headerAuth.substring(7, headerAuth.length());  
    }  
}
```

headerAuth is null, that is my problem.. Any idea?

**Minh**

September 17, 2020 at 9:50 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4765>)

Hi bezkoker

yeah, i got it, thanks for your reply, wish you all the best and success

**Minh**

September 17, 2020 at 7:33 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4761>)

hi bezkoder

- this is great tutorial, it's worked for me,. But can i ask:  
+ Why do you have to create and use "SignupRequest" and  
"LoginRequest" for Signing Up and Login request, is it OK if use User  
entity for this purpose?

Many thanks for your help!

Hi, User entity is highly related to database model, while "SignupRequest" and "LoginRequest" stand for Http request data. That means, you can add or change data of the request body by changing "SignupRequest" and "LoginRequest" class, but the persistence of table in database (corresponding to User entity) should not be changed regularly.

**Haba**

September 16, 2020 at 11:54 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4740>)

Unauthorized error: Full authentication is required to access this resource . I got this error when try to accès to this url : <http://localhost:8080/api/test/user> (<http://localhost:8080/api/test/user>)

**bezkoder**

September 16, 2020 at 12:04 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4741>)

Hi, please check the HTTP request Authorization Header. Also don't forget to login a user account to get the Token.

December 27, 2020 at 3:27 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5936>)

Hi, did you succeed?



### LPT

September 14, 2020 at 1:37 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4703>)

Hi, Thank you a lot for this tutorial. But I have a problems when I run your project and it still ok. But I found on my eclipse log that:

io.jsonwebtoken.SignatureException: JWT signature does not match locally computed signature. JWT validity cannot be asserted and should not be trusted.

It has this error. Can you help me fix this



### Raj

September 12, 2020 at 4:58 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4691>)

the actual JWT implementation code, thanks a lot.



Hi Bezkoder, Your codes really helped me alot .It is such a awesome code. I just loved it.

It was working fine earlier but recently i got this error while trying to login and i am stuck for a while :

"The signing key's size is 104 bits which is not secure enough for the HS512 algorithm. The JWT JWA Specification (RFC 7518, Section 3.2) states that keys used with HS512 MUST have a size  $\geq$  512 bits (the key size must be greater than or equal to the hash output size). Consider using the io.jsonwebtoken.security.Keys class's 'secretKeyFor(SignatureAlgorithm.HS512)' method to create a key guaranteed to be secure enough for HS512. See <https://tools.ietf.org/html/rfc7518#section-3.2> (<https://tools.ietf.org/html/rfc7518#section-3.2>) for more information."

What may be the solution for this? Please help!

**Yogen Bhattarai**

September 11, 2020 at 9:11 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4685>)

The secret key i was giving was short length , i changed the secret key and it worked.Thanks!!!!

Good Morning

The tutorial is very cool, but I have a problem registering

Give me error 500

I downloaded your code, changed the database and still give me the same error

follow a code

I hope I can help

```
http://localhost:8080/api/auth/signup  
(http://localhost:8080/api/auth/signup)  
{  
    "username": "mod",  
    "email": "[email protected] (/cdn-cgi/l/email-protection)",  
    "password": "123456789",  
    "role": ["mod", "user"]  
}  
  
{  
    "timestamp": "2020-09-04T10:20:52.561+0000",  
    "status": 500,  
    "error": "Internal Server Error",  
    "message": "Error: Role is not found.",  
    "path": "/api/auth/signup"  
}
```

September 4, 2020 at 1:28 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4603>)

Hi, you should insert 3 rows into **roles** table first.



**Ricardo Miguel Barros Ferreira**

September 7, 2020 at 11:01 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4625>)

thanks



**Myo Myint**

August 31, 2020 at 5:39 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4546>)

Thank you for your tutorial, bez.



**Manish**

August 18, 2020 at 1:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4361>)

Such Helpful article explained very well and it is saying "it took lot of effort to make it understandable".



authentication mechanism with difference of session-based authentication.

After that i able to create custom project with no difficulties and that because of this article [reference](#)



Thank You

God Bless bezkoder 😊

**bezkoder**

August 18, 2020 at 3:07 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4364>)

Hi, thank you for your comment. I am so happy to know that my effort helps you! 😊

**Brenda**

August 15, 2020 at 10:45 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4332>)

There is a problem in many to many part.

Values cannot be inserted to USER\_ROLES table.

How can I fix it?

**bezkoder**

August 16, 2020 at 3:08 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4336>)

Hi, you can drop all the tables and follow step by step in the video demo from the beginning:

<https://youtu.be/o8DEk4XGcZw> (<https://youtu.be/o8DEk4XGcZw>)

**Barry**

August 15, 2020 at 10:26 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4330>)

It's an amazing tutorial for me. I wanted to try it but there is a problem.

In signup method of AuthController, roles cannot be defined. It assigns a message "Error: Role is not found."

Here is my ROLES\_JWT Table

ID ROLE

1 ROLE\_USER

2 ROLE\_ADMIN

3 ROLE\_MODERATOR

When I wrote a json object for signing up, it throws 505 internal error message because of this reason.

```
"password" : "user123456789",  
"email" : "[email protected] (/cdn-cgi/l/email-protection)",  
"role" : ["user","mod"]  
}
```



Where is the problem ? How can I fix it?

I use Oracle 11g Database

I hope you can reply my question

Thank you

**bezkoder**

August 16, 2020 at 3:07 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4334>)

Hi, you can follow the step in video demo:

<https://youtu.be/o8DEk4XGcZw> (<https://youtu.be/o8DEk4XGcZw>)

**Sercan**

August 15, 2020 at 9:02 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4326>)



**aladin**

August 21, 2020 at 11:22 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4419>)

any help ?

**vikram**

August 15, 2020 at 6:41 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4325>)

Hi bezkoder,

Thanks for tutorial , I am able to call all APIs through Postman, But in browser i am unable to access this application. I am getting below error page. i used the # App Properties  
bezkoder.app.jwtSecret= bezKoderSecretKey  
bezkoder.app.jwtExpirationMs= 86400000

Can you please help on this. did i missed any thing.

Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Sat Aug 15 23:40:56 IST 2020

**Aris**

August 14, 2020 at 11:03 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4310>)

Amazing tutorial bezkoder! My only issue is the fact that it appears too low in Google Search.

I tried DZone's tutorial for Bootiful App but yours is much more understandable and well-written.

Thanks for the effort!

**bezkoder**

August 14, 2020 at 12:39 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4312>)

Thank you, I'm so happy to know that my work helps other people like you 😊

**rmm**

August 12, 2020 at 2:32 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4285>)

```
protected void doFilterInternal(HttpServletRequest request,  
HttpServletResponse response, FilterChain filterChain)  
throws ServletException, IOException {  
// ...  
UserDetails userDetails =   
userDetailsService.loadUserByUsername(username);  
// ...  
}
```

Well, it will hit the database everytime when a request comes in, basically every click on the UI, every incoming api call, etc triggers a select query against the DB in order to fetch the user data again and again and again...

I know, I know security is important, but don't you think it is a little resource consuming way to maintain security? I imagine, the user logs in, the backend checks that everything is OK, returns the token, and that's it. No more DB harassment for finding out that the user and its roles are still existing and intact... in the same session... in every couple of seconds. If the user holds the token (and it is not expired) the backend should accept that token as a valid one – without checking the user data in the db.

What are the good ways to keep it still secured but skip the unnecessary resource consuming calls?

Storing the Authentication object in the session? Why not? Spring must have some tools to manage this.

Storing every user detail (password included) in the encoded jwt? Ok, it

AuthProvider with caching functionality? And with cache invalidation ofc.  
Other solutions?

Cheers!

**Serge**

October 21, 2020 at 1:02 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5141>)

I have the same question. It seems to me that the logic behind JWT would be to store the user authorisation in the token (hence the name "claims"). They are as secure (or as insecure) as the token itself. Obviously, it makes changing user rights somehow difficult, but you could use the expiration date to set a limit.

**Luiz Carlos**

August 9, 2020 at 6:40 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4249>)

Dude, great tutorial.. I've tested it out and worked perfectly on postman... However, if I create an account using the interface on the browser and I log in with user role, when I try accessing "User content"

**hani**

August 9, 2020 at 3:51 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4248>)

hey , great job but am sorry where did you specify the roles that can access a specific url for exp (the user can access the /user and cannot access the /admin )

**Pawel**

August 9, 2020 at 3:39 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4247>)

How to test such a configuration, namely the controller methods from @PreAuthorize ?

**javacoder**

August 7, 2020 at 5:25 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4233>)

I got this error: 'Can't find the request for  
<http://localhost:9000/api/auth/signup's>

Thanks in advance

**Kevin**

August 5, 2020 at 10:32 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4208>)

Hello!!Very nice tutorial you open my mind!!

I have 1 question for you if i want to add more than one restcontroller just to get some data from database , i added @PreAuthorize annotation and configured security config properly but i am getting 401 error???

My question is do i need every time that i make a post request to sent header so i can get authorized ??? or when i log in automaticly from sesion storage server knows that i am authenticated...????

**moon**

August 5, 2020 at 9:26 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4207>)

Thank you... great help me.. now.... thank you

Hi Thanks for this tutorial, I have just one easy query on how to get current user object here in backend

**foxi.chen**

July 29, 2020 at 12:52 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4053>)

Hello,

thanks for your tutorial. I have a question: why does the statement "filterChain.doFilter(HttpServletRequest, HttpServletResponse)" call the authentication process again? Actually, in the method "doFilterInternal", it has already authenticated the token.

In my log, there are SQL of authentication process called twice, so I debug the programme, and find the doFilter method call the UserDetailsImpl once again. How do I avoid this? or is it normal?

I hope my question is clear.

thanks

ps: my log like this:

Hibernate: select \* from user where username=?

Hibernate: select \* from user where username=?

**cvril**

for logout it is really usefull to create it or just remove cookies from front is enough ?

because i did not see a great implementation of logout....  
is there ip verification in your code (i did not analyzed all code)

thank you!

**Somach**

July 22, 2020 at 12:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3889>)

Hi,

Can i use h2 database for this?

If it's possible, can someone tell me what do i need to change for h2?

I tried to change it but i can't open h2 and insert roles.

**cyril**

July 30, 2020 at 5:21 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4089>)

you just have to change the driver in pom.xml and in application.properties, then change jdbc url in application properties



**Alex**

July 20, 2020 at 9:13 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3861>)

IT workkkks !! Thank U so MUCH !! 

**CrypticDelight**

July 20, 2020 at 6:38 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3856>)

Great tutorial, just having issue with one bit...AuthEntryPointJwt class.

I'm building a multi-module project using Gradle. When I build the application and attempt to run it it fails saying "WebSecurity required a bean of type "AuthEntryPointJwt" that could not be found."

Wondering if you had seen such an issue before?

**Alan Naicson**

July 19, 2020 at 10:58 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3840>)

Works perfectly, Thank you! That was one of the best tutorials I have seen about this subject

**Driger Khanh**

July 18, 2020 at 6:00 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3808>)

Hi thank you for your post, it really helps me.

I'm stuck at creating a new user, The user table is created but user\_role is null.

Can you help me with it?

**Kunal Pandey**

July 17, 2020 at 7:49 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3797>)

Thank a lot man. That's the cleanest example.

**Rafat**

July 15, 2020 at 1:38 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3758>)

Hi, thank you for this great tutorial;

I have one question about UsernamePasswordAuthenticationFilter, Do we only use it in the http.addFilterBefore()

I thought that we will use it for each request (Endpoint of the Test

**Kai K**

July 1, 2020 at 3:17 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3560>)

Hello and thank you for this great tutorial!

I tried to get it running with openjdk 14 and get following error when trying to authenticate:

FilterChain java.lang.NoClassDefFoundError:  
javax/xml/bind/DatatypeConverter

Could found help here:

<https://stackoverflow.com/questions/43574426/how-to-resolve-java-lang-noclassdeffounderror-javax-xml-bind-jaxbexception-in-j>  
(<https://stackoverflow.com/questions/43574426/how-to-resolve-java-lang-noclassdeffounderror-javax-xml-bind-jaxbexception-in-j>)

So by adding following dependency to pom.xml:

jakarta.xml.bind  
jakarta.xml.bind-api  
2.3.2

everything works fine.

bezkoder

**Nikola**

June 30, 2020 at 1:12 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3542>)

Hi,

when i try to run the app i get this error Consider defining a bean of type

'org.springframework.security.authentication.AuthenticationManager'  
in your configuration.

This happens in authController

I know you defined a bean and wired it in the controller, but i dont get it.

Can you explain please?

**Pavan Patidar**

June 29, 2020 at 5:29 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3534>)

Hi,

I want to host some static content from the application build, but i am not able to allow publicly accessible content from static directory.

can you suggest something.

**mercy**

June 28, 2020 at 11:02 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3502>)

Thank you so much, have been trying to create tokens for so long.

**Edyta**

June 22, 2020 at 5:06 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3402>)

What can be the reason why I'm not receiving ANY response from server?

I've made everything as in tutorial step by step...It's very weird.

**Jhonatan**

December 30, 2020 at 3:13 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5989>)

Hello, I have the same problem that you, ¿you got any solution for that?

Hey again,

I want to sign up and sign in using also the email address. Any help  
please?

(Spring boot / Angular)



**Sachin**

June 17, 2020 at 12:58 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3308>)

Sir How To Add The Data i DataBase using Postman Will You Please  
Explain The Example

**Gabor**

June 15, 2020 at 12:56 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3282>)

one of the best tutorials i've ever seen!  
especially according to spring security



**vikas ghodke**

**azzurptkl**

June 13, 2020 at 4:15 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3260>)

Hi, I've downloaded this repo from github and upgraded it to use spring boot 2.3.0 but now whenever i try to login using wrong credentials the message property on the response is empty.

This is what I get, it happens with every unauthorized response:

```
{  
    "timestamp": "2020-06-13T16:12:23.219+00:00",  
    "status": 401,  
    "error": "Unauthorized",  
    "message": "",  
    "path": "/api/auth/signin"  
}
```

The logger by the way display the correct message in the console ("Unauthorized error: Bad credentials").

Do you happen to know the reason?

**Ingo K**

June 10, 2020 at 2:47 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3208>)

Field jwtUtils in com.ingokoeppe.xsamples.security.jwt.AuthTokenFilter required a bean of type 'com.ingokoeppe.xsamples.security.jwt.JwtUtils' that could not be found.

**Ingo K**

June 10, 2020 at 3:08 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3209>)

Nevermind.. i forgot to add @Component to the JwtUtils.java class 😊

**hemanth**

June 10, 2020 at 9:24 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3200>)

Hey thanks for this tutorial. But I added course model(id,name,description) and controller doing crud operations. But when i authorize and then add values to the course using postman , its getting added but the name and description fields are displaying null.  
May i know whats the problem

Hi, I have problem with application, because app throw "Unauthorized error: Bad credentials". And above is "Empty encoded password".

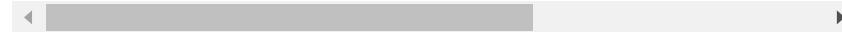
o.s.s.c.bcrypt.BCryptPasswordEncoder : Empty encoded password  
p.z.l.security.jwt.AuthEntryPointJwt · Unauthorized error: Bad  
credentials

**HansKim**

July 16, 2020 at 1:58 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3766>)

check UserDetailsImpl.java Constructor

```
public UserDetailsImpl(Long id, String username,  
                      Collection authorities)  
{  
    this.id = id;  
    this.username = username;  
    this.email = email;  
    this.password = password;      <===== this one  
    this.authorities = authorities;  
}
```



## Request

```
{  
    "username": "mod",  
    "email": "[email protected] (/cdn-cgi/l/email-protection)",  
    "password": "123456",  
    "role": ["mod", "user"]  
}
```



## Response

```
{  
    "timestamp": "2020-06-04T03:31:19.290+0000",  
    "status": 500,  
    "error": "Internal Server Error",  
    "message": "Error: Role is not found.",  
    "path": "/api/auth/signup"  
}
```

PS : I'm running the code you have provided in github.

**bezkoder**

June 4, 2020 at 1:23 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3104>)

Hi, you should use SQL script to insert 3 rows in `roles` table first.

June 1, 2020 at 6:21 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3065>)

Thank you very much for the effort to create such a jewel.

I've been looking for something like this for a long time.

There are many who have dealt with this subject, but most of the time they have left it half, incomplete or not refreshed.

Man, you found a niche that is highly sought after and appreciated by boot-spring developers.

The secret is the richness of details and the fact that it works!

I recommend adding Lombok and keeping it updated.

Success!

**Lakbir**

May 31, 2020 at 2:31 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3054>)

thank you very much for your work! Great Tuto

**sharad**

May 28, 2020 at 10:25 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3012>)

**bezkoder**

May 29, 2020 at 12:32 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3013>)

Hi, it is in the Source Code section.

**Bohdan**

May 26, 2020 at 10:21 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2956>)

Hi! This is an amazing article, thank you for your job.

I've written all the same. But I have one problem. If I want to get properties of the current user in my own method, I write this:

```
CustomerDetailsImpl currentCustomer
Authentication authentication = SecurityContextHolder.getContext().getAuthentication();
if (!(authentication instanceof AnonymousAuthenticationToken))
    currentCustomer = (CustomerDetailsImpl) authentication;
```

In headers from Postman I send my bearer token after login.

But in authentication I get anonymousUser.

Why is this happening?

May 26, 2020 at 11:11 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2958>)

Solved, my mistake 😊



### Enrique Gordon

November 19, 2020 at 8:42 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5473>)

how did you do it?



### David Montellano

February 5, 2022 at 12:34 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-19132>)

Hi @Enrique Gordon, My solution was:

```
Authentication authentication =  
SecurityContextHolder.getContext().getAuthenticati  
on();  
UserDetailsImpl currentUser = (UserDetailsImpl)  
authentication.getPrincipal();
```

This way you can get data current user in any class  
or method.



**Hussain**

May 19, 2020 at 8:40 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2659>)

You're the best teacher that I've ever come across in my coding journey who can explain in detail what spring security and jwt are all about. I don't know how I can describe you but you're genius. This is an inspirational tutorial and I look forward to see more powerful tutorial of this type. Once again, I will say big thank you

**Karthi**

May 19, 2020 at 6:09 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2648>)

Hi, how can I add a cascading dropdown in the signup page , and the data's in the dropdown from the database. How it is possible in the backend and frontend?

**bezkoder**

May 19, 2020 at 10:56 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2663>)

Hi, please take a look at following tutorial (also related fullstack

**Soumendu Das**

May 19, 2020 at 12:36 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2631>)

I've implemented jwt with spring security and work's fine, but when I'm trying to access the API those have @preAuthorize it's giving me unauthorized when hitting from angular(v9) but it's working fine in postman!

I have ensured that the login role and preAuthorize api roles are same.

Why is this happening? I have no idea...

Please help...

**bezkoder**

May 19, 2020 at 2:04 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2638>)

Hi, you can follow this Angular 8 JWT Auth tutorial (<https://bezkoder.com/angular-jwt-authentication/>). 😊

NTD

Hello, Thank you so much for this great tutorial !

I have implemented this tutorial, but I have a problem.

when i call:

POST: http://localhost:8080/api/auth/signin

(http://localhost:8080/api/auth/signin)

I filled out :

```
{  
    "username": "user",  
    "password": "123456"  
}
```

and postman response:

```
{  
    "timestamp": "2020-05-14T13:30:02.827+0000",  
    "status": 401,  
    "error": "Unauthorized",  
    "message": "Error: Unauthorized",  
    "path": "/api/auth/signin"  
}
```

and eclipse response: "Unauthorized error:User account is locked"

Although I have filled in the correct information.

How can I fix it? thanks

bezkoder

Hi, "User account is locked" happens after a number of failed authentication events. So you need to check the issue why you couldn't login with this account.

**Kenyodarz**

May 21, 2020 at 4:36 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2733>)

In UserDetailsImpl.java:

Switch from false to true by the following methods:

```
public boolean isAccountNonExpired()  
public boolean isAccountNonLocked()  
public boolean isCredentialsNonExpired()  
public boolean isEnabled()
```

Because if you automatically generated them by default they are false.

PD: Sorry, I'm really not very good at English

**Qwerty**

May 14, 2020 at 2:03 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2417>)

how can i implement the same in oracle database. i have some issues

**Sascha**

May 12, 2020 at 9:29 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2337>)

Hey. Thank for the tutorial but I think there might be one problem or it's only happening for me. But maybe you can reproduce it. I'm using the full tutorial of Angular and Spring Boot and if I register and directly login with this account I get 2020-05-12 23:20:09.452 ERROR 13476 — [nio-8080-exec-7] d.example.project.security.jwt.JwtUtil : Invalid JWT token: JWT strings must contain exactly 2 period characters. Found: 0 The JWT token is undefined. So the typical Angular behaviour if something is not loaded. Is that a special case and should somehow be validated in Spring Boot? Or is anything wrong in Angular? The method doFilterInternal() in AuthTokenFilter.java is causing this.

**Sascha**

May 12, 2020 at 9:56 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2338>)

Okay I could fix it in a not very acceptable manner. In the authInterceptor in Angular you are using the line const token = this.token.getToken(); and this is giving me undefined and is not null and gives it to the Spring Boot backend. The only way to handle it is if (token != null && token != "undefined") {

**Bogdan**

June 18, 2021 at 2:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9159>)

If I clone your repo, the app is working fine. If I try to build the app after yours, I got Invalid JWT token: JWT strings must contain exactly 2 period characters. Found: 0. I look forward to hear from you with a solution. Thank you

**Shraddha**

January 3, 2021 at 4:47 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6059>)

I could not understand why token value is undefined as everything is running fine in postman

**Bogdan**

June 18, 2021 at 1:54 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-9158>)

Hello! Were you able to resolve the error caused by Invalid JWT token? I have the same problem and I couldn't solve it with (token != null && token != "undefined"). Thank you

**Hq**

May 12, 2020 at 7:48 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2295>)

Hello, Thank you so much for this great tutorial!

I have implemented this tutorial, but I have a problem.

When localhost:8080/api/expense/all called, it's successfully as expected. But when localhost:8080/api/expense/76 (76 is ID for an expense), the expenseService always returns null (500 server error).

What could be the cause? and how to fix it?

**LC**

May 10, 2020 at 2:39 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2172>)

Hello, thank you for the tutorial. I keep getting "Error: Unauthorized" when I go to /user , /mod, or /admin and /all. Everything else seems to work okay. Do you have any idea as to why I might be getting this error?

**bezkoder**

**Khatri (<http://bezkoder.com>)**

May 13, 2020 at 6:01 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2369>)

Hey there can u please help me with error ....  
"internal server error"

when i m passing the jason body it is showing error as:  
Internal server Error  
and error message as : "Role not found"

**bezkoder**

May 13, 2020 at 9:36 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2372>)

Hi, maybe you forgot to insert 3 records into roles table 😊

**Adrian**

May 9, 2020 at 4:57 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2154>)

Amazing work, I used this implementation and everything works fine.

Thanks for sharing.

However, I have a question because I want to add the option  
remember me to login and I am not sure how to implement it in this  
case.



I have API in spring boot and frontend in react. I tried to add  
rememberMe() function at configure(HttpSecurity http) function at  
WebSecurityConfig class, but after that i don't see a remember-me  
cookie in the browser.

Do you have any advice?

Greetings

Adrian

**Mpho Vic**

May 8, 2020 at 10:47 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2129>)

main] o.s.b.web.embedded.tomcat.TomcatStarter : Error starting  
Tomcat context. Exception:  
org.springframework.beans.factory.UnsatisfiedDependencyException.

Am I the only one ?

May 8, 2020 at 9:18 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2100>)

Apart from component scanning , if you missed web-starter dependency , it could cause a ripple , that is what i've done and failed me many times.

**Dennis Nyoni**

May 8, 2020 at 6:40 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2120>)

Oh ok let me check. thanks for the response!

**Dennis Nyoni**

May 7, 2020 at 11:08 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2055>)

Hello Sir

Thank you so much for such a detailed tutorial. You made life very easy for more in learning spring authentication.

However, I am facing two challenges; 1. I cant add more than one users because the in the user\_roles table the ids for both user and the role are unique. I get an exception:

`java.sql.SQLIntegrityConstraintViolationException: Duplicate entry '1'`



2. After registering a user I get another exception on trying to login:

2020-05-07 13:05:36.007 WARN 12224 — [nio-8080-exec-7]

o.s.s.c.bcrypt.BCryptPasswordEncoder : Encoded password does not look like BCrypt

2020-05-07 13:05:36.008 ERROR 12224 — [nio-8080-exec-7]

z.o.m.security.jwt.AuthEntryPointJwt : Unauthorized error: Bad credentials

How best can I avoid roles column in the user\_roles table to be non unique or is there a different approach.

**bezkoder**

May 7, 2020 at 11:42 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2073>)

Hi, I don't really understand your question.

"I cant add more than one users because the in the user\_roles table the ids for both user and the role are unique."

You can add user by sending HTTP request to  
`/api/auth/signup` endpoint (with `role` array in the payload).

**Dennis Nyoni**

its about the relationship table. the one that has two columns for either ids (role\_id and user\_id). Since we created a table with 3 roles that have ids 1,2, and 3 in the table the ids are unique say if my first user has id=1 and role is ROLE\_USER which has id=1 then if i wish to add another user with id=2 and role ROLE\_USER again when added to the relationship table where role\_id = 1 already exists. this will create a scenario like:

user_id		role_id
1		1
2		1

which then complains of constraint violation issues since role\_id must be unique. well i had to create new roles with new ids.

my question number 2

After registering a user I get another exception on trying to login:

```
2020-05-07 13:05:36.007 WARN 12224 — [nio-8080-exec-7] o.s.s.c.bcrypt.BCryptPasswordEncoder : Encoded password does not look like BCrypt  
2020-05-07 13:05:36.008 ERROR 12224 — [nio-8080-exec-7] z.o.m.security.jwt.AuthEntryPointJwt : Unauthorized error: Bad credentials
```

What could be the cause?

1. The relationship between User entity and Role entity is Many to Many.

We don't add any record to `user_roles` table directly. Please look at how we signup user in `AuthController`.

I will write a tutorial for Spring JPA Many to Many Association when having time.

2. I don't really know what was wrong. You can debug yourself and notice these lines:

```
User user = new User(signUpRequest.getUser()
                     signUpRequest.getEmail()
                     encoder.encode(signUpRequest.getPassword()))
```



**Dennis Nyoni**

May 8, 2020 at 6:38 pm

(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2119>)

Thanks very much I will do that and check whats causing the exception

June 20, 2020 at 7:34 pm

(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3377>)

Hey Bezkoder,

i am not able to understand what might be  
the root of Empty coded password, i  
searched it everywhere, user has already  
been registered, but i am facing the above  
issue and error is coming as Bad credentials

**Jessy**

May 7, 2020 at 10:54 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2054>)

Hello,

I'm trying to follow your tutorial but I'd like to have a connexion without username (just email/password), what is the workaround for this? I'm in trouble when I need to implement the UserDetailsService because there is no loadByEmail method or something like this.

**bezkoder**

May 7, 2020 at 11:33 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2067>)

**Mayur**

May 7, 2020 at 8:24 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2050>)

Best Tutorial!

I have learned lot of new things.

Can you explain to me the functionality of every role?

**bezkoder**

May 7, 2020 at 11:30 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2066>)

Hi, it depends on your use cases. The roles in this tutorial is used for illustrating authority 😊

**Volker**

May 6, 2020 at 6:04 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2034>)

Really great tutorial, it helped me a lot, thank you very much! 😊

Read more comments

We never used the string ADMIN anywhere, only ROLE\_ADMIN as part of the enum.

**bezkoder**

May 6, 2020 at 11:47 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2040>)

Hi, the 'ROLE\_' prefix gets added automatically for `hasRole` . If you want to work with `ROLE_ADMIN` , just use `hasAuthority('ROLE_ADMIN')` .

**Volker**

May 11, 2020 at 6:11 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2249>)

ah, thank you so much, very simple explanation and it makes sense! I didn't take a course on Spring Security yet, so I didn't know that.

**GM**

Thanks for this amazing course. It was really easy to understand it.

I have one question, given this example code, how could I implement an oauth2 login as well?

E.g. : when the user visits `/signin` I would like to provide the options of logging in either with jwt or oauth2(github/google). Can you give me a hint?

**bezkoder**

May 4, 2020 at 2:16 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1905>)

Hi, you can read this article  
(<https://spring.io/guides/tutorials/spring-boot-oauth2/>).

**nico**

May 3, 2020 at 6:44 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1890>)

Thx for this tutorial, I looked through a lot of tutorials and couldn't understand any of them until I found yours.

Thank you so much



May 2, 2020 at 9:49 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1879>)

Great tutorial!

I have a question, How can I modify it to authenticate a user using its email and password instead of username and password ?



**bezkoder**

May 2, 2020 at 11:58 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1883>)

Hi, you can see the email as username in the signup process.

**TnCoders**

January 26, 2021 at 7:13 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-6563>)

Thank you for This Awesome Gift Tutorial 😊

I have many other Application that use the same Table and there is no possible way to change table structure as it's in production.

We have email,password Field so can you please show me How to force UsernamePasswordAuthenticationToken to attempt auth using email instead of username please.



**Aashna**

April 28, 2020 at 11:24 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1718>)

Hello,

I'm getting the following error, whenever I try to create a signup POST request:

ERROR: insert or update on table "user\_roles" violates foreign key constraint "fkrhfovciq1l558cw6udg0h0d3"  
Detail: Key (role\_id)=(2) is not present in table "role".

I've already inserted the 3 roles you mentioned.

Do you know why I'm getting this error? Thank you for your help! 😊

**bezkoder**

April 29, 2020 at 8:29 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1758>)

Hi, I think you can drop `user_roles` table, then drop `users` and `roles` table.

Just do everything from the beginning. Maybe you miss or make mistake in a certain step.

**Filipe**

April 25, 2020 at 1:51 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1603>)

Can you post here your headers in the request? I got i problem and i think the header is the cause.

**bezkoder**

April 26, 2020 at 9:23 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1664>)

Hi, you need to set `content-type: application/json` for HTTP request header.

**Akshatha**

April 27, 2020 at 4:43 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1674>)

It has been set. Still facing the issue with same error.

April 27, 2020 at 4:42 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1673>)

```
{  
    "username": "alphabank",  
    "password": "alphabank"  
    "role": ["mod", "user"]  
}
```

this is the request i am sending, and the header is content-type-application/json

**bezkoder**

April 27, 2020 at 6:53 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1675>)

You should have 'email' in the payload.

**sai**

April 24, 2020 at 2:21 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1421>)

For admin while signing up i have given the role as user and for signin it s coming good but for `http://localhost:8080/api/test/admin` (`http://localhost:8080/api/test/admin`) it is coming as 403 Forbidden

**bezkoder**

April 24, 2020 at 8:08 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1527>)

Hi, please make sure that you set the correct Authorization Header in HTTP request.

**Etienne BELEMNEGRE**

April 22, 2020 at 8:32 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1276>)

very good and detailed tutorial; thanks a lot

**Haibara**

April 14, 2020 at 6:10 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-859>)

Hello!

it's giving me this: "message": "Error: Role is not found." to refer to this one: case "admin":

```
Role adminRole = roleRepository.findByName(ERole.ROLE_ADMIN)
.orElseThrow(() -> new RuntimeException("Error: Role is not found."));
roles.add(adminRole);
```



**bezkoder**

April 14, 2020 at 6:49 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-861>)

Hi, maybe you forgot to run ~~SQL~~ insert 3 rows of Role table. 😊

**Larry**

August 15, 2020 at 10:27 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4331>)

I run SQL script but the issue still remains

**bezkoder**

August 16, 2020 at 3:07 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4335>)

Please follow step by step in the video demo:

<https://youtu.be/o8DEk4XGcZw>

(<https://youtu.be/o8DEk4XGcZw>)

I loved this Spring Boot Auth tutorial! I actually read your blog fairly often, and you're constantly coming out with some great tutorials here.

▲  
**Misho**

April 12, 2020 at 11:37 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-844>)

Very helpful post. Could you add how to revoke tokens?

▼  
**bezkoder**

April 14, 2020 at 3:36 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-857>)

Hi, I will write the tutorial for revoke tokens when having time



▼  
**Foliveira**

April 8, 2020 at 6:03 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-795>)

Excellent! It worked perfectly. That's not the first time I found here

good quality and useful tutorials. Thank you for the good work!

**Jorge**

April 8, 2020 at 12:28 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-772>)

Awesome job, thanks for snaring. 

**Ujitha**

April 5, 2020 at 12:07 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-739>)

Hi BezKoder,

Thank you so much for you posting such superb tutorial. And could you explain a little how this JWT token has secured in this api.

**bezkoder**

April 6, 2020 at 2:19 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-747>)

Hi, JWT does not hide, obscure, secure data at all.  
You can see that the process of generating JWT (Header, Payload, Signature) only encode & hash data, not encrypt data.

The purpose of JWT is to prove that the data is generated by an

**Radek**

April 5, 2020 at 9:29 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-736>)

Ok, now I've got it. Testing this endpoint I tried to send token from Headers tab, instead using Authorisation tab. Everything is ok 😊

**Radek**

April 4, 2020 at 11:53 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-730>)

Hi bezkoder, thank you very much, you've done a great job!

I have one problem with Postman.

After signing in as "mod" I try to access resource: GET

<http://localhost:8080/api/test/user>

(<http://localhost:8080/api/test/user>).

I send value: Bearer {my token} with key Authorization (in section Header) and I receive this error:

```
{  
  "timestamp": "2020-04-04T23:51:06.706+0000",  
  "status": 401,  
  "error": "Unauthorized",  
  "path": "/api/test/user"
```

**bezkoder**

April 5, 2020 at 12:29 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-731>)

Hi, I've checked GET `http://localhost:8080/api/test/user` with 'mod' role and it works.  
Please make sure that you use the token correctly.

**Taeco**

June 14, 2020 at 1:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3264>)

Can you post me your PostMan requests? I still get the unauthorized error  
Thank you!

**jiji**

May 31, 2020 at 10:26 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3050>)

Hello, i have the some problem, did you solve it? can u help me

**H.L**

July 6, 2020 at 4:56 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3638>)

Hello, Can anyone solve this problem please !

**bezkoder**

July 6, 2020 at 11:32 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3640>)

Hi, you should check the Authorization header.

**User9888**

March 29, 2020 at 5:51 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-679>)

Hi bezkoder i have a problem when i create a user and give him some roles on angular when i'm back in springBoot the role array is empty can you help please!!!

Best regards !!



Hi, please show me the HTTP request payload.

**Hugo**

March 27, 2020 at 5:59 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-662>)

Thank you for that great tutorial sooooooooooooo usefull !

Great piece of work!

**Alberto**

March 18, 2020 at 1:25 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-563>)

Hi, im new with java language and this helped me a lot. Are there any way to support and appreciate your work?

Thanks so much !!

**bezkoder**

March 18, 2020 at 2:11 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/>)

I'm so happy to know that this tutorial helps many people like you.

If you want to donate, I have a Paypal account: [email protected]  
(/cdn-cgi/l/email-

protection#c4b0ada1aaeaaa3b0b6b1abaaa384a3a9a5ada8eaa  
7aba9) 😊

Best Regards,  
bezkoder.

**MensiGhaith**

March 17, 2020 at 2:27 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-553>)

Hello Bezkoder thank you , you did a great job , but `hasRole(..)` it should be changed to `hasAuthority` right? because at first `hasRole` doesn't work after that i tried with `hasAuthority` and it works..

**bezkoder**

March 17, 2020 at 2:39 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-554>)

Hi, Roles and Authorities are similar in Spring.

The difference is that the `ROLE` is usually added

**MensiGhaith**

March 17, 2020 at 4:13 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-555>)

thank you bezkoder , i have a question i didn't add the prefix ROLE\_ when i added my roles (because i didn't use enumarted strings ), so i think that's why she doesn't work in the first time with hasRole so did you think if that will causes a problems??

**bezkoder**

March 17, 2020 at 4:32 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-556>)

You must add the prefix ROLE\_ to make it work 😊  
Please follow step by step in the tutorial.

**MensiGhaith**

March 17, 2020 at 4:45 pm  
(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-557>)

```
//tokenStore.removeAccessToken(accessToke  
n);  
//tokenStore.removeRefreshToken(refreshTo  
ken);
```

**bezkoder**

March 18, 2020 at 12:07 am  
(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-561>)

We're gonna implement logout on frontend side (by removing token from localStorage/sessionStorage for example). So we don't need logout endpoint here.

**abdeljabbar**

March 16, 2020 at 1:05 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-545>)

where are payload classes, i didn't find them  
`com.bezkoder.springjwt.payload.*`



i find it, thanks

**julian**

March 9, 2020 at 8:11 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-479>)

Hi, thanks for the tutorial I have this error that the postman shows me what do you know?

```
"error": "Bad Request",
"message": "Invalid JSON input: Cannot deserialize instance of `java.util.HashSet` out of VALUE_STRING token; nested exception is com.fasterxml.jackson.databind.exc.MismatchedInputException: Cannot deserialize instance of `java.util.HashSet` out of VALUE_STRING token\n at [Source: (PushbackInputStream); line: 5, column: 10]
(through reference chain:
com.ramajudicial.Solicitud.de.Informacion.payload.request.SignupRequest[\"role\"])"
```

**bezkoder**

March 9, 2020 at 11:47 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-481>)

Hi please show me your POST request 😊

**julian**

March 10, 2020 at 4:03 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-484>)

Hi, thanks for answering. I had something wrong with the POST, now this error is generating, I am working with a sql server database

Error: Cannot insert the value NULL into column 'userId', table 'lramajudicialpru.dbo.user'; column does not allow nulls. INSERT fails.

Do you know why you are sending me null?

**bezkoder**

March 11, 2020 at 1:45 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-488>)

Hi, this Spring Boot Application is configured to work with MySQL or PostgreSQL database, so you need to modify `application.properties` file for SQL Server Database.

Do you have any idea how to configure it? I have this configuration connected to the database but I don't know why it gives the previous error

```
spring.datasource.url=
spring.datasource.username=
spring.datasource.password=
spring.datasource.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDriver
spring.jpa.show-sql=true
spring.jpa.hibernate.dialect=org.hibernate.dialect.SQLServer2012Dialect
spring.jpa.hibernate.ddl-auto = validate
spring.jpa.hibernate.naming_strategy =
org.hibernate.cfg.EJB3NamingStrategy
```

thanks.

**bezkoder**

March 11, 2020 at 3:19 am

(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-490>)

You should fill url , username and password .

Server now.

I will check it and write another tutorial when  
having time 😊

**Pradeep Balachandran**

March 4, 2020 at 1:01 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-457>)

Amazing tutorial.

**Juan Pablo**

March 3, 2020 at 2:25 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-453>)

Amazing Tutorial,

Thank you!

**Rick Meels**

March 2, 2020 at 7:30 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-448>)



Thanks for the great tutorial, but i have one problem. When sending a request to the API with the roles "mod", "user" it always throws an exception that it could not find a role.

This even happens when using your source code.

```
{  
    "username": "henk",  
    "email": "[email protected] (/cdn-cgi/l/email-protection)",  
    "password": "keukentafel",  
    "role": ["mod", "user"]  
}
```

**Rick Meels**

March 2, 2020 at 7:33 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-449>)

Nevermind, as I was writing this comment I found the problem.  
Forgot to put the roles in the postgres database

**bezkoder**

March 3, 2020 at 2:15 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-452>)

**khouloud**

March 30, 2020 at 11:07 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-685>)

i have the same error.

**bezkoder**

March 30, 2020 at 11:17 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-686>)

Hi, you must insert 3 rows in `roles` table as tutorial instruction first.

**lokesh**

October 23, 2020 at 1:25 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5164>)

sir,can't we do that within the code itself.

**vikram**

August 15, 2020 at 6:26 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5164>)

```
{  
    "username": "henk",  
    "email": "[email protected] (/cdn-cgi/l/email-protection)",  
    "password": "keukentafel",  
    "role": ["henk", "user"]  
}
```

please use above one. (insted of mod —> henk)

**juan**

March 2, 2020 at 7:29 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-447>)

Hello, I am a beginner in the code do you create the fields of the database tables? or do you do manual in mysql?

**bezkoder**

March 3, 2020 at 2:14 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-451>)

Hi, Spring Data will do it automatically 😊

Thank you for your response, and if I have a database already created with the same data, should your code work well? Or do I have to skip somewhere?

**bezkoder**

March 3, 2020 at 11:11 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-456>)

If you have the same database table, you don't need to run SQL queries for inserting records to roles table 😊

**OneCentral**

February 28, 2020 at 10:45 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-430>)

Thanks a lot for this tutorials. Been looking for this for a long time!

**SB**

February 28, 2020 at 4:21 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-428>)

**bezkoder**

February 29, 2020 at 1:15 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-435>)

Hi, I think that you can find the answer here  
(<https://security.stackexchange.com/questions/108662/why-is-bearer-required-before-the-token-in-authorization-header-in-a-http-re>) 😊

**Alex**

February 25, 2020 at 8:50 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-410>)

Hello, many thanks for these great tutorials!

I think I'll go this way to implement a simple app, but I have some questions.

Is there a way to encrypt password in the POST request? I can see it is encrypted in database but requests can be intercepted and I'm new to angular.

Do you plan to make a tutorial on deployment of a Spring boot app? I heard of solutions like heroku, azure, amazon.. but it's still a bit confusing, I'd really like a guide made by you on this (you make things

**bezkoder**

February 26, 2020 at 1:26 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-414>)

Hi, I will write the tutorials when having time 😊

**OneCentral**

February 28, 2020 at 10:59 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-433>)

Definitely look into Heroku, which is part of Salesforce Inc apparently. They are by far the easiest and most user friendly hosting platform for Spring boot applications that I know of.

**Martin**

April 25, 2020 at 1:13 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1602>)

You encrypt POST requests by using HTTPS with a valid certificate. If you do that, the whole POST request body will be encrypted. In order for your requests to not be intercepted, you should never send them over HTTP. This tutorial is good for setting it up in development, but when you go into live

**proctor07**

May 21, 2020 at 10:44 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2722>)

Hello, how can we do that easily ? Is there some simple tutorials like that ?

**Kam Hieu**

February 22, 2020 at 9:19 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-393>)

Hello. Which version of Java uses this tutorial? Is it completely Java 8?

**bezkoder**

February 22, 2020 at 10:13 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-394>)

Yeah, Java 8 😊

**Anton (<http://antonthomas.be>)**

February 9, 2020 at 1:34 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-395>)

Thank you for the great tutorial. I do have a question, though. The AuthenticationManager in the AuthController is not able to be autowired as there is no bean found for the AuthenticationManager. Do you have any idea as how to solve it?

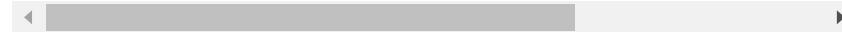
Thanks in advance!

**bezkoder**

February 9, 2020 at 1:48 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-330>)

Hi, we have a bean in WebSecurityConfig :

```
public class WebSecurityConfig extends WebSecurityCon  
    ...  
    @Bean  
    @Override  
    public AuthenticationManager authenticationManagerB  
        return super.authenticationManagerBean();  
    }  
    ...
```



Hi bezkoder, maybe you have any resources, where I can will read about refresh token?

**bezkoder**

August 8, 2021 at 11:20 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10551>)

Hi, here you are: Spring Boot Refresh Token with JWT example  
(<https://www.bezkoder.com/spring-boot-refresh-token-jwt/>)

**Pascal Dantinger**

February 6, 2020 at 3:20 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-312>)

Hi, great Tutorial. But if I want to register a user i get this Error:  
"timestamp": "2020-02-06T15:14:41.823+0000",  
"status": 405,  
"error": "Method Not Allowed",  
"message": "Request method 'GET' not supported",  
"path": "/api/auth/signup"

The same for "path": "/api/auth/signin".  
Could you pls help me. Thank you very much

**bezkoder**

February 7, 2020 at 3:08 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-325>)

Hi, please make sure you make a **POST** HTTP request (with payload) instead of **GET** request to /api/auth/signin endpoint.

**Theodora**

February 6, 2020 at 12:19 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-309>)

Hi bezkoder,

Thank you for the tutorial!

I have followed your instructions. However, for some reason in my WebSecurityConfig.java, I get an error on the following code block:

```
@Override  
public void configure(AuthenticationManagerBuilder auther  
    authenticationManagerBuilder.userDetailsService(userDet  
    }  
  
◀ ▶
```

which underlines the userDetailsService in the brackets and says:

reason: no instance(s) of type variable(s) exist so that  
UserDetailsServiceImpl conforms to UserDetailsService

Do you have an idea why I might be getting this one and how could I fix it?

Thank you in advance!

**Theodora**

February 6, 2020 at 9:16 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-318>)

Nvm, please ignore me. I managed to fix it.

**bezkoder**

February 7, 2020 at 9:23 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-320>)

Great! 😊

**Springy**

February 22, 2020 at 6:10 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/>)

How did you fix it?? Please in future comment the fix as well (for other people who have the same problem, like me)

↑  
**Phil**

March 25, 2020 at 10:10 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-639>)

I have the same issue. Whats this error?

↓  
**bezkoder**

March 26, 2020 at 1:26 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-641>)

Hi, make sure that you've add `@Service` annotation for `UserDetailsService` class.

▼  
**Tendo**

May 6, 2020 at 12:06 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2007>)

**Rajko**

June 15, 2020 at 4:25 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3285>)

I have the same issue. Could you help me to fix it?

**demka**

April 22, 2020 at 11:17 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-1222>)

bonjour comment avez-vous résolu ce problème ??

**Talib**

July 19, 2020 at 8:48 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-3838>)

How did you fix it let me know

**sekenuw**

July 28, 2020 at 1:45 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4030>)

That worked for me.

**vaftre**

December 29, 2020 at 5:42 pm  
(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-5978>)

Thank you!

**sss**

February 5, 2020 at 9:54 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-299>)

here is an overview of its code application.properties:

```
spring.datasource.url= jdbc:mysql://localhost:3306/testdb?  
useSSL=false  
spring.datasource.username= root  
spring.datasource.password=  
  
spring.jpa.hibernate.ddl-auto= update  
spring.jpa.properties.hibernate.dialect =  
org.hibernate.dialect.MySQL5Dialect  
  
# App Properties
```

**bezkoder**

February 5, 2020 at 11:19 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-301>)

Have you created MySQL ~~testdb~~ database manually?

**Karthi**

May 19, 2020 at 6:29 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2650>)

How to create database automatically?

**sss**

February 5, 2020 at 8:12 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-296>)

But it doesn't generate the table.

**bezkoder**

February 5, 2020 at 8:45 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-297>)

**sss**

February 5, 2020 at 7:47 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-294>)

Great tutorial but i did not understand what my database table was not generated at startup

**bezkoder**

February 5, 2020 at 7:52 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-295>)

Hi, you must create `testdb` database with *username* & *password* first.

Our Hibernate configuration only generates tables.

**Mcgill Dias**

February 3, 2020 at 6:53 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-291>)

Simplesmente fantástico, o seu tutorial. Parabéns

January 31, 2020 at 3:10 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-281>)

Hi bezkoder! thank you very much for this awesome tutorial! I have a little question though.. It all worked perfectly, but when i create an user with 2 roles (as input I wrote: "role": ["ROLE\_MODERATOR", "ROLE\_USER"] the app only associates the user with user role, but not with moderator role. I tried with my own code (following ur instructions) but when it didnt work, i cloned your repo, but i still have the same problem.. do u know what it could be?

Thanks again for this amazing tutorial!

**Vicente Ortiz**

January 31, 2020 at 3:25 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-282>)

Nevermind my question, I saw what my mistake was.. thank you again!

**bezkoder**

January 31, 2020 at 3:54 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-283>)

Yeah, you should use: ["mod", "user"].

**Jiji**

May 19, 2020 at 9:03 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2623>)

Hello, can you help me to resolve this problem, i can't assign ROLE\_ADMIN or ROLE\_MODERATOR to user.when i add a user even i specify admin or role in DataBase it's user role !!

**bezkoder**

May 19, 2020 at 9:49 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2626>)

Hi, you need to add `role:["admin", "mod"]` in the payload 😊

**Jiji**

May 19, 2020 at 6:57 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2652>)

i did but i have the some problem :( .in user\_roles table i have:  
user\_id role\_id  
1 1



4 1

5 1

THX U for your answer (f)

**bezkoder**

May 19, 2020 at 10:52 pm

(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2662>)

We have 3 tables, and each row in  
`user_roles` was generated automatically 😊

So you should create user by sending HTTP  
request with payload containing `role` array.

**jiji**

May 21, 2020 at 11:47 am

(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-2726>)

big thx ,it's works (y)



Pls can you tell me where to add role : [“admin”, “mod”] in payload . because even I am facing the same problem only one role is getting assign to user.



**Fastspeed**

July 31, 2020 at 3:33 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-4109>)

Can you tell me how did you resolved the mistake

**adebowale**

January 21, 2020 at 2:55 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-239>)

Great tutorial but my database table was not generated at startup



```
spring.datasource.url= jdbc:mysql://localhost:3306/testdb?
spring.datasource.username= root
spring.datasource.password= 123456

spring.jpa.properties.hibernate.dialect= org.hibernate.dialect.
spring.jpa.hibernate.ddl-auto= update

# App Properties
bezkoder.app.jwtSecret= bezKoderSecretKey
bezkoder.app.jwtExpirationMs= 86400000
```

and any pointer on how to generate refresh token. thanks

**bezkoder**

January 21, 2020 at 3:10 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-241>)

Hi, you must create **testdb** database with username & password first.

Our Hibernate configuration only generates tables.

For refresh token, I will write a tutorial when having time. 😊

Insane tuto, exactly what i needed. thx.

**Michal**

January 10, 2020 at 9:37 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-172>)

Thank you for this tutorial. It really helped a lot! Great job.

**Shimul**

January 8, 2020 at 10:55 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-158>)

Hello,

Thank you very much for your step by step tutorial.

I got the following error: "cannot resolve method getName()" in the following method:

```
public static UserDetailsImpl build(User user) {  
    List authorities = user.getRoles().stream()  
        .map(role -> new SimpleGrantedAuthority(ro:  
        .collect(Collectors.toList());  
  
    return new UserDetailsImpl(  
        user.getId(),  
        user.getUsername(),  
        user.getEmail(),  
        user.getPassword(),  
        authorities);  
}
```

Initially, I thought it happened maybe I did some mistake during my typing. But I copy-paste your code for ERole, User, Role Model and also the UserDetailsImpl.java but it still shows that error.

How can I fix that?

**bezkoder**

January 8, 2020 at 3:25 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-159>)

Hi, I'm sorry, syntax highlight plugin causes the Typo.

You can check `Role` and `User` class again to make sure that all the fields are correct.  
Another way is to download the source code.

**Donnie**

December 23, 2019 at 6:59 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-111>)

WOW, easy to understand your Tutorial. You make complicated things like Spring Security simple.

**justice**

December 22, 2019 at 11:58 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-107>)

Hello, I would like to ask you something.

The spring security oauth 2.0 resource serve has been deprecated.

Is this tutorial the way that it should be done now that the `EnableResourceServer` is deprecated?

I would like to migrate my project that is using it, but in their migration guide there are not explanation about the resource server since they say that is out of the scope in the migration guide



thanks

**bezkoder**

December 23, 2019 at 1:35 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-108>)

Hi, I think that `EnableResourceServer` will not affect the code in this tutorial.

For Spring Security OAuth 2.0, I will write the tutorial in near future.

**Yaman M Nur (<http://yamanmnur.github.io>)**

December 11, 2019 at 6:49 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-91>)

Excellent, I have tried this tutorial and it worked!, I'm really thankful for you, I hope you are going to be successful because sharing useful knowledge for many people.

**bezkoder**

**Yaman M Nur (<http://vamanmnur.github.io>)**

December 12, 2019 at 1:32 am  
(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-93>)

I have a question, how to keep the tokens from expiring  
(token does not expire)?

**KaitoKid**

December 12, 2019 at 9:08 am  
(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-94>)

Thanks for your post. I'm also waiting for your  
Spring Boot JWT with Refresh Token tutorial.

**bezkoder**

August 8, 2021 at 11:21 am  
(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-10552>)

Hi, here you are: [Spring Boot Refresh Token with JWT example](#)

**bezkoder**

December 17, 2019 at 3:19 pm

(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-100>)

Hi, you can implement Refresh Token.

**umsu**

December 4, 2019 at 9:45 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-86>)

Hello, I'm a beginner, so I have a lot of questions. I hope you will answer this question. I'm so frustrated because I don't understand. In the repository, we divided the user and the role. What function does the user and the role ? For example, if you type id/pass and then go to the repository, I don't understand why you use id/email in the user. I know the reason why I want to use the role in the repository for automation, but I don't know why I use the repository user.

**bezkoder**

December 6, 2019 at 3:19 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-88>)

- `userRepository.existsByEmail`
- `userRepository.save()`
- `userRepository.findByUsername()`

**umsuhyun(엄수현)**

December 4, 2019 at 1:54 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-84>)

Thank you for good writing. But I don't understand what's the difference between database and repository. I don't know exactly the difference between the two, so I'm asking you a question.

**bezkoder**

December 4, 2019 at 3:21 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-85>)

Hi,

A **Repository** is a database that stores the metadata for Designer objects. In other words, the data associated with objects like Entities, Modules, Table Definitions, etc. is stored as rows in tables that are part of the Repository.

The Repository also includes a PL/SQL Application Programming Interface (API) for maintaining this data.

A **Database** is just a place to store data, or an application database is a place to store the data for a particular computer application.

The Repository is itself an application database, but it also stores data about (ie. metadata – <sup>data about data</sup>) other application databases. These other databases might be co-located with the Repository, but normally are stored somewhere else.

A Repository is a central common point of file storage of a related nature, a Database is a structured file or set of files that store and organize data.

---

I found this great answer here: <https://qr.ae/TeQcol>  
(<https://qr.ae/TeQcol>).

**argeta**

November 30, 2019 at 4:26 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-74>)

I am regular reader of your site. All of Spring tutorials are really good.  
This Spring Boot Authentication with JWT tutorial is the best.

Amazing article! It really helped me with a school project.  
Only one question, how can I add a server side logout feature? So the token is immediately invalid

**bezkoder**

November 23, 2019 at 1:06 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-58>)

Hi, you can implement a `/logout` endpoint (or logout functionality) where you only have to get the token and remove it from `TokenStore`.

```
tokenStore.removeAccessToken(accessToken);  
tokenStore.removeRefreshToken(refreshToken);
```

The detail is described at: <https://stackoverflow.com/a/42907381> (<https://stackoverflow.com/a/42907381>).

**Sandy**

November 10, 2019 at 9:41 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-39>)

# App Properties

**bezkoder**

November 11, 2019 at 2:07 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-40>)

It's custom properties.

**Sandy**

November 12, 2019 at 1:51 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-41>)

where can i get that? like custom properties according to your design?

**bezkoder**

November 12, 2019 at 3:42 am  
(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-42>)

These custom properties are what you set, personally, and should not be the same as other places.

**Peter Kronenberg**

October 22, 2021 at 12:23 pm

(<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-14258>)

This is a great example. Thanks so much for this code and explanation. You have your timeout set to 10 days (if I did the math right). I assume that normally, you would want the timeout to be some much lower value, such as 10-15 minutes. How do you handle the timeout? You only want the user's login to timeout if he has been inactive. So do you get a new token with every transaction? Or if it has been 10 minutes since the last token?

**Mathieu LC**

October 24, 2019 at 12:00 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-27>)

Hello, im a french dev student, thank you very much for your work! I have a probleme with this method in UserDetailsImpl.

```
=> public static UserDetailsImpl build(User user) {  
    List authorities = user.getRoles().stream()  
        .map(role -> new SimpleGrantedAuthority(ro:  
        .collect(Collectors.toList());  
    }  
    ^  
    return new UserDetailsImpl(  
        user.getId(),  
        user.getUsername(),  
        user.getEmail(),  
        user.getPassword(),  
        authorities);  
}  
=> getName() "cannot resolve method"
```

I'm not very strong with stream map etc, can you explain me this block a bit more?

Thank you for your attention and continue to display this wonderful content.

Mathieu

**bezkoder**

October 24, 2019 at 12:59 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-28>)

- `user.getRoles()` returns a `Set<Role>`, we convert this Set to a `Stream` of `Role`. Imagine that the stream is a collection of `Role` that releases items in turn.

- `map()` changes ~~every Role item above~~ to a <sup>SimpleGrantedAuthority</sup> object, the parameter for the constructor could be one of these Strings: 'ROLE\_USER', 'ROLE\_MODERATOR', 'ROLE\_ADMIN'.

So, what does `role.getName().name()` do?

- `role.getName()` returns `ERole` enum object, let's call it `erole_object`. Look back to this:

```
public enum ERole {  
    ROLE_USER,  
    ROLE_MODERATOR,  
    ROLE_ADMIN  
}
```

- `erole_object.name()` returns the String corresponding to the enum ('ROLE\_MODERATOR' for example).

Please make sure that you define `Role` model in `Role.java` correctly.

Regards,  
bezkoder.

October 24, 2019 at 1:23 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-29>)

Thank you a lot for your explanation! I will follow your site and youtube channel studiously and recommend it to others students!



**bezkoder**

October 24, 2019 at 1:25 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-30>)

Thank you so much! Wish you all the best 😊

**Phuoc**

March 12, 2020 at 3:17 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-499>)

Hi @Mathieu LC,  
I think you miss getter/setter for Name field on Role.java class



**Shivam**

December 31, 2020 at 6:27 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/>)

**Haylton**

October 22, 2019 at 12:42 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-23>)

Great tutorial series. Congratulations for the detail and the good explanation.

**bezkoder**

October 22, 2019 at 1:13 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-24>)

Hi, thank you!

**Mykilchenko**

March 18, 2021 at 9:04 pm (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-7663>)

Thank you, this is the best article on Spring Security & JWT I have ever seen!

September 19, 2021 at 7:18 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-12418>)

Can you please tell me how to delete a user from postman directly?



**Siddhant**

April 3, 2022 at 4:56 am (<https://www.bezkoder.com/spring-boot-jwt-authentication/#comment-20761>)

What if we wanted to update the role for a user on the website ?

Like remove Role\_Moderator and upgrade him to Role\_Admin ?

How to do that ?

Comments are closed to reduce spam. If you have any question, please send me an email.

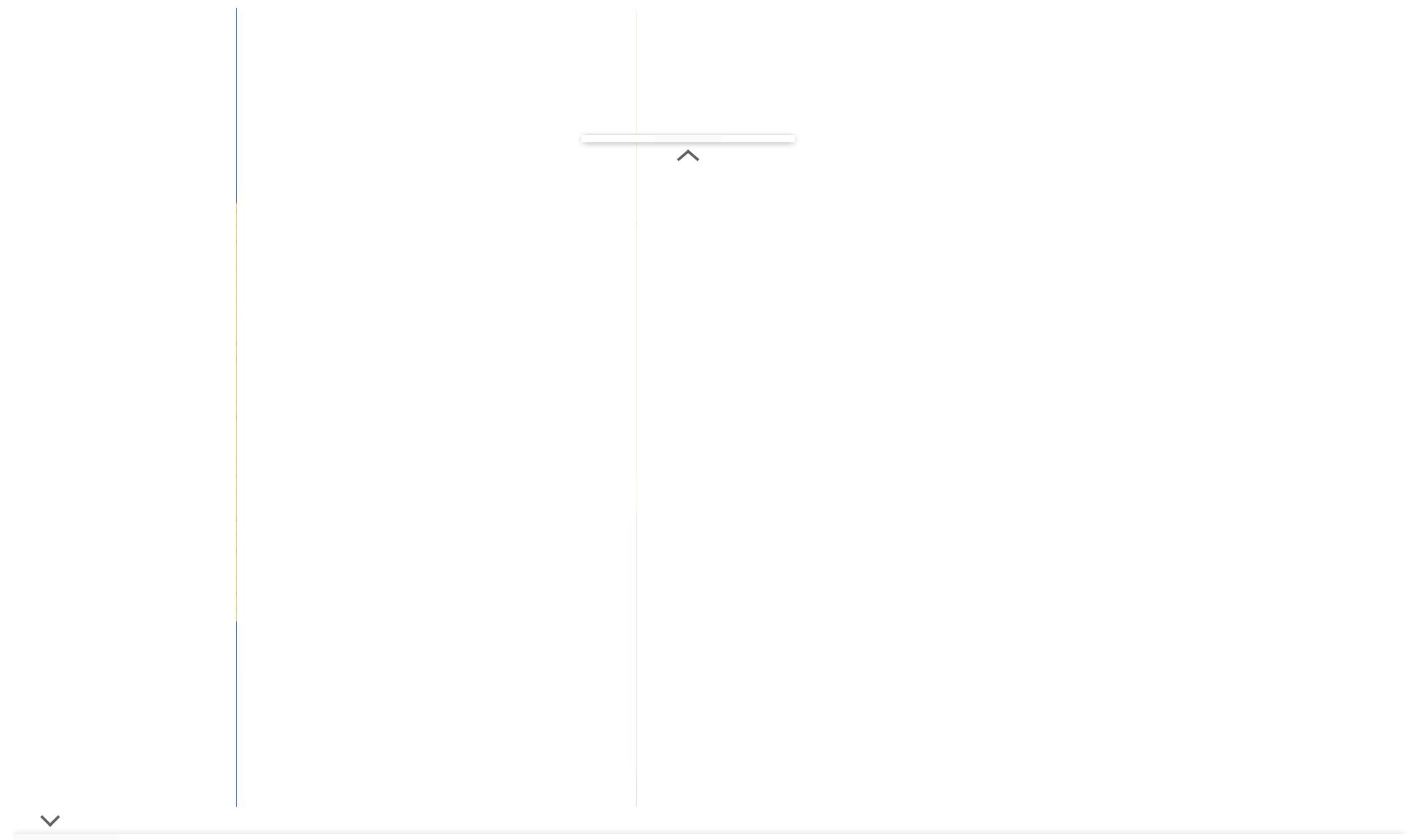
◀ React Firebase CRUD with Realtime Database (<https://www.bezkoder.com/react-firebase-crud/>)

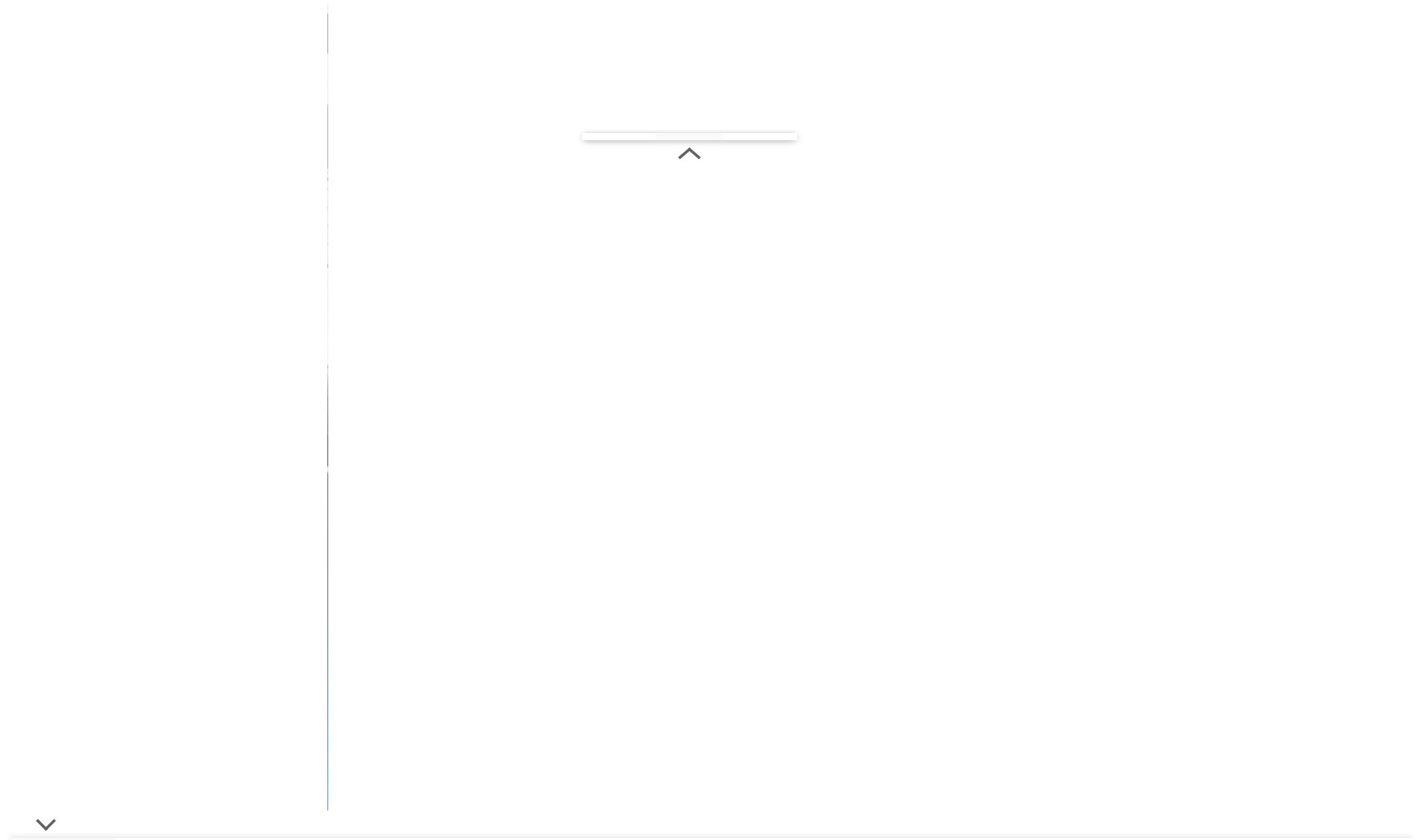
Node.js Express File Upload Rest API example using Multer ➤ (<https://www.bezkoder.com/node-js-express-file-upload/>)



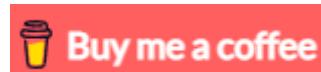
Search ..











([https://www.buymeacoffee.com/  
bezkoder](https://www.buymeacoffee.com/bezkoder))



**FOLLOW US**

**TOOLS**

Json Formatter  
(<https://www.bezkoder.com/json-formatter/>)





(htt

ps://



ww

w.yo

utub

e.co

m/c

han

**f** nel/ 

(htt UCp (htt

ps:// 0mx ps://

face 9RH gith

boo 0Jxa ub.c

k.co Fsm om/



m/b MvK hozk

. Privacy Policy (<https://www.bezkoder.com/privacy-policy/>)

Contact (<https://www.bezkoder.com/contact-us/>) About (<https://www.bezkoder.com/about/>)



(<https://www.dmca.com/Protection/Status.aspx?ID=3f543dd5-c6d8-4208-9a6b-0e92057fd597&refurl=https://www.bezkoder.com/spring-boot-jwt-authentication/>) © 2019-2022 bezkoder.com

