Spring Security

Introduction

Spring Security is a framework that helps:

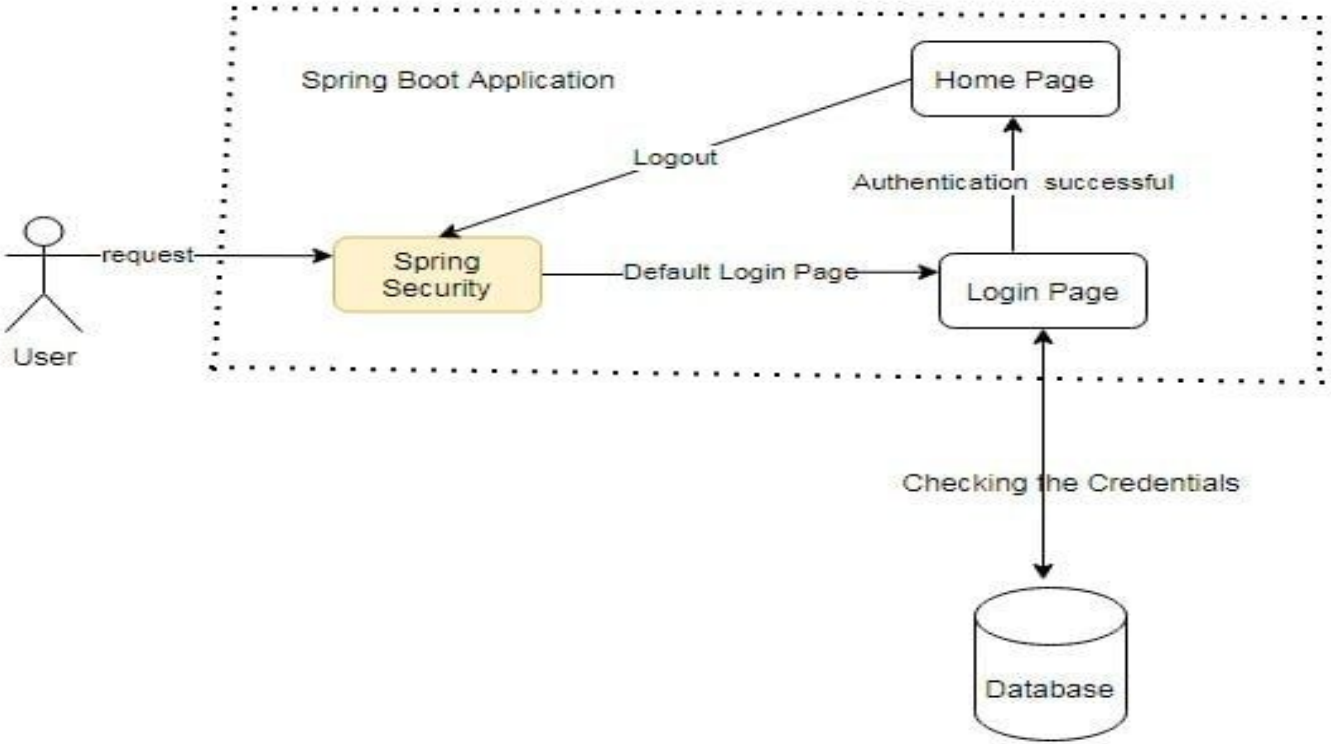
- Secure enterprise applications
- Creating powerful and highly customizable authentication and access-control framework

Web Application Security

There are 3 important concepts in web security.

- Authentication
- Authorization
- Servlet filters

Authentication

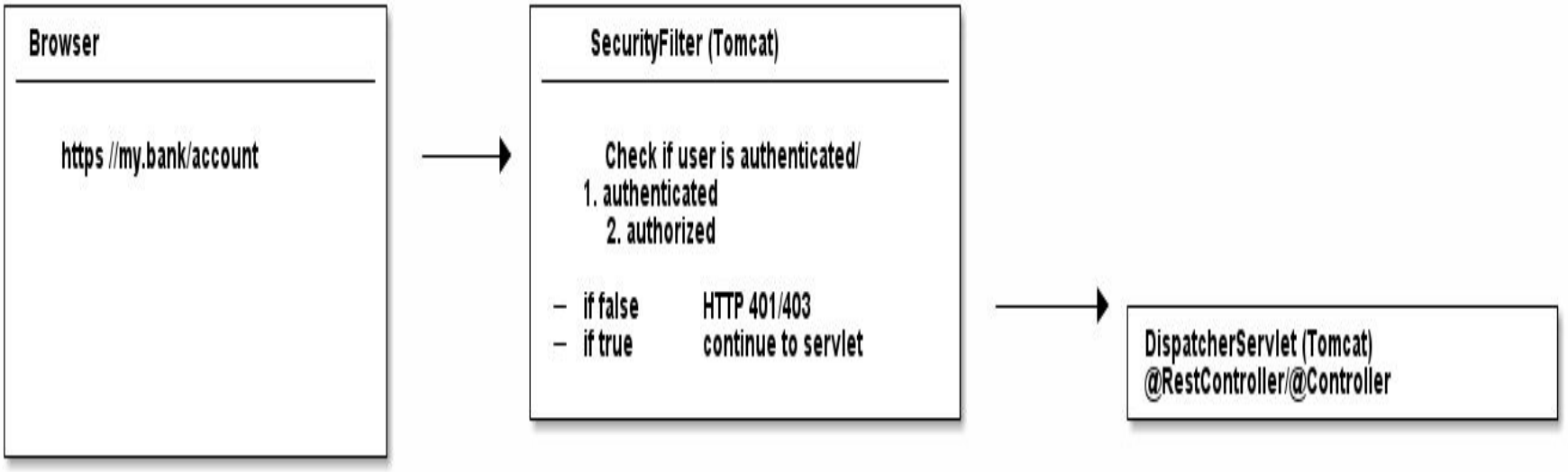


Authorization

Servlet Filters

A servlet filter is an object that is used to intercept HTTP Requests. In Spring Boot, servlet filters can execute in three situations:

- Before sending a request to the controller
- Before sending response to the client.


Browser
https //my.bank/account
SecurityFilter (Tomcat)
Check if user is authenticated/
1. authenticated
2. authorized
– if false HTTP 401/403
– if true continue to servlet
DispatcherServlet (Tomcat)
@RestController/@Controller

Principal

Principal is the person you have identified through the process of authentication. In other words, a principal is the currently logged in user.

It is the unique information or account in the system that you tie to a specific person in the context of an application.

Granted Authority

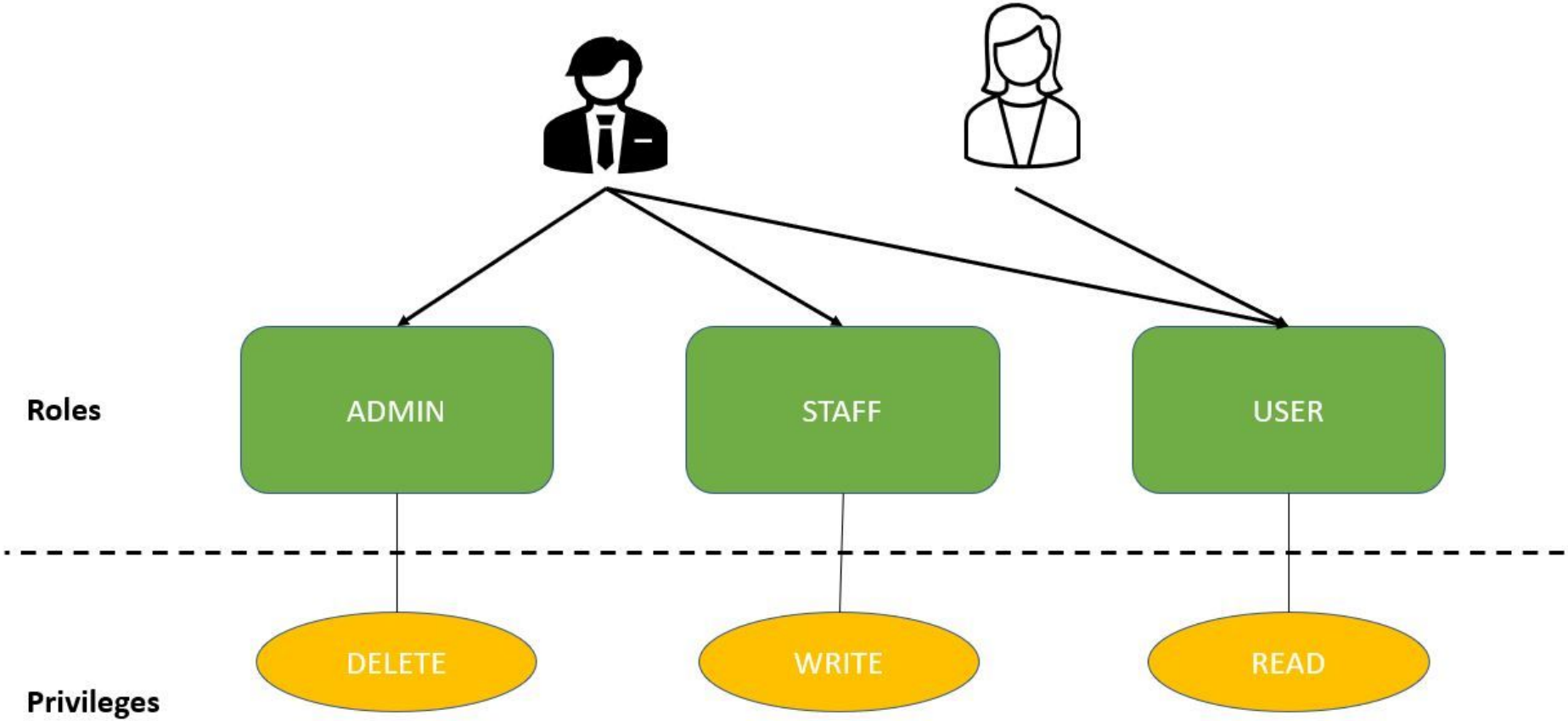
A user trying to do something, so to allow or authorize them to do it only if the user has been granted authority to do so.

We can configure these things in spring security and define what permissions are granted to whom.

Role

Role is pretty much like a group of authorities that are usually assigned together.

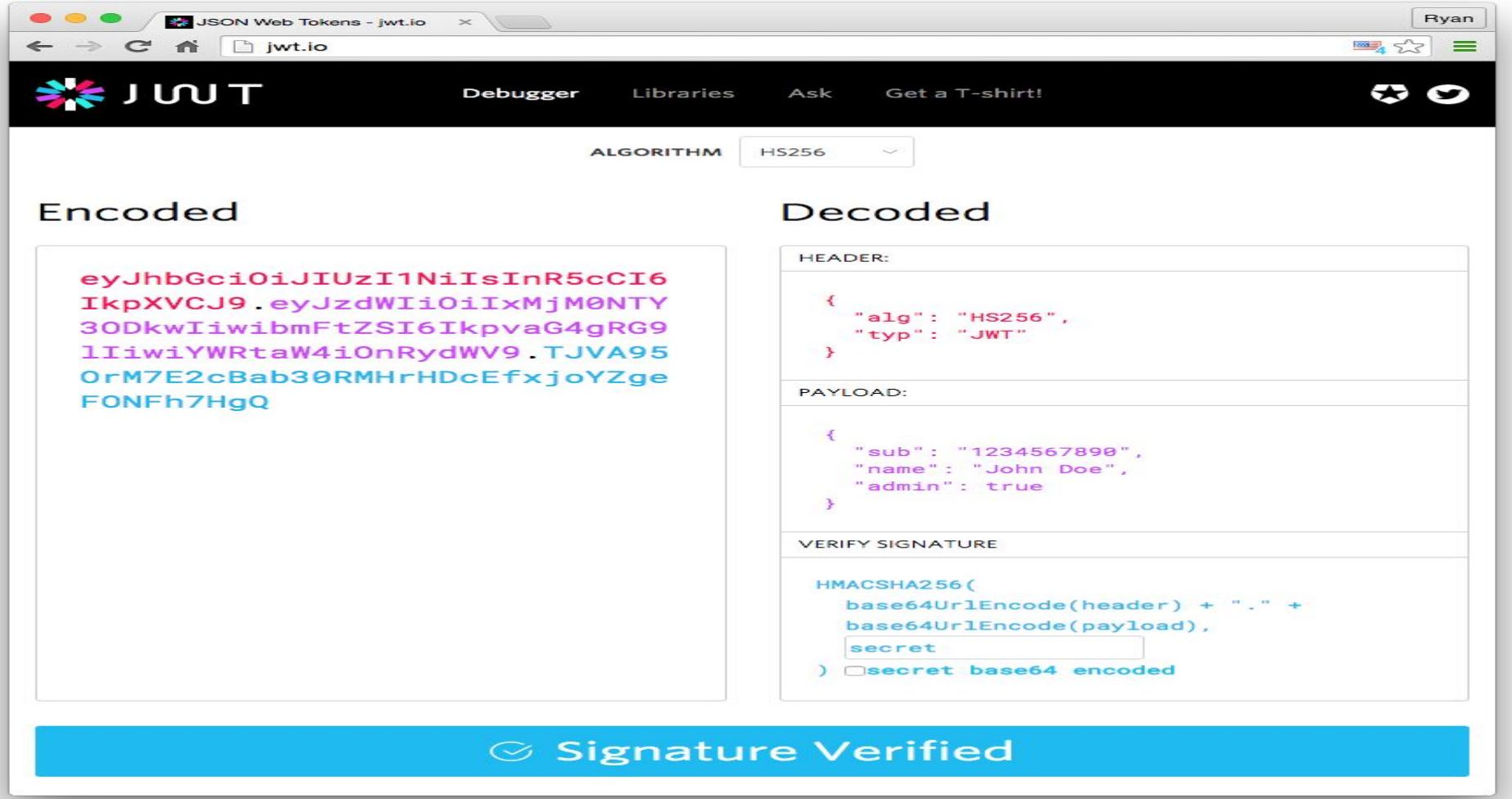
Assigning role to an user, automatically user gets all the authorities for that role.


Roles
ADMIN
STAFF
USER
DELETE
WRITE
READ
Privileges

What is JWT

JWT (JSON Web Token) is a standard that allows the transmission of information between parts in a base64URL encoded JSON. The token encodes three basic (JSON) parts.

1. HEADER
2. PAYLOAD
3. SIGNATURE


JSON Web Tokens - jwt.io
Ryan
jwt.io
JWT
Debugger
Libraries
Ask
Get a T-shirt!
ALGORITHM
HS256
Encoded
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiYWRtaW4iOnRydWV9.TJVA95OrM7E2cBab30RMHrHDcEfxjoYZgeFONFh7HgQ
Decoded
HEADER:
{
"alg": "HS256",
"typ": "JWT"
}
PAYLOAD:
{
"sub": "1234567890",
"name": "John Doe",
"admin": true
}
VERIFY SIGNATURE
HMACSHA256(
base64UrlEncode(header) + "." +
base64UrlEncode(payload),
secret
) secret base64 encoded
Signature Verified

Dependencies

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-security</artifactId>
  </dependency>
<dependency>
  <groupId>io.jsonwebtoken</groupId>
  <artifactId>jjwt-api</artifactId>
  <version>0.11.5</version>
</dependency>
<dependency>
  <groupId>io.jsonwebtoken</groupId>
  <artifactId>jjwt-impl</artifactId>
  <version>0.11.5</version>
  <scope>runtime</scope>
</dependency>
```

References

- https://spring.io/guides/gs/securing-web/
- https://jwt.io/introduction


Thank
You