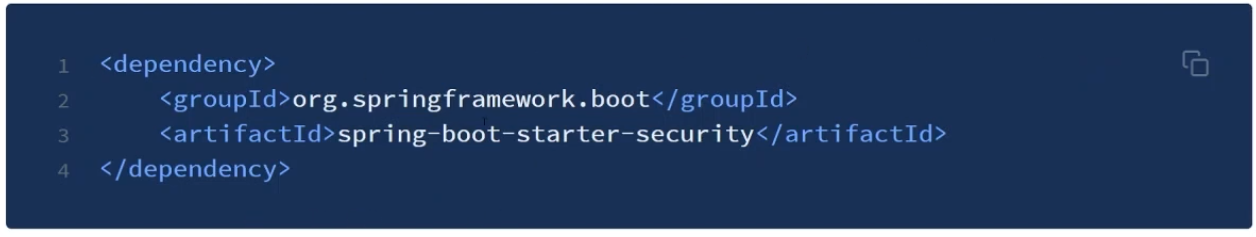
Spring Security

# Terminologia

* **A autenticação** refere ao processo de verificação da identidade de um usuário, com base nas credencias fornecidas. Um exemplo comum é inserir um nome de usuário e uma senha ao fazer login em um site. Você pode pensar nisso como uma resposta à pergunta *quem é você*?
* **Autorização** se refere ao processo de determinar se um usuário tem permissão adequada para executar uma ação específica ou ler dados específicos, supondo que o usuário seja autenticado com êxito. Você pode pensar nisso como uma resposta à pergunta *Um usuário pode fazer/ler isso*?
* **Princípio** refere-se ao usuário autenticado no momento.
* **Autoridade** concedida refere-se à permissão do usuário autenticado
* **Função** refere-se a um grupo de permissões do usuário autenticado.

# Habilitando Segurança

Em um projeto Spring Boot Web você precisa somente incluir a dependência no pom.xml.



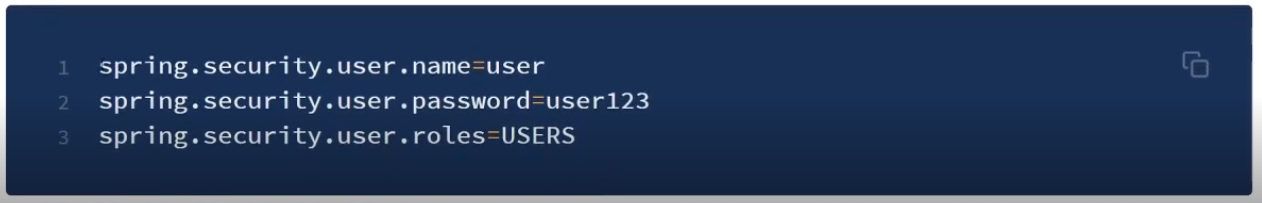
# Autenticação Simples

Spring possuir algumas configurações para definir os usuários na sua camada de segurança.

Como sabemos por padrão o Spring Security habilita um usuário de nome user e gera uma senha aleatoriamente a cada inicialização. Para aplicações em produção esta não é uma abordagem tanto aconselhável, e é por isso que vamos conhecer algumas outras configurações de segurança.

## No application.properties

O Spring Secutiry verifica se existe alguma configuração de segurança no arquivo application.properties.



## Em memória

Está configuração permite criar mais de um usuário e perfis de acesso.

## Configura Adapter

Nos exemplos anteriores já podemos considerar um nível de segurança em nossa aplicação, mas se percebermos o esforço de configuração para as novas funcionalidades poderá não ser uma abordagem tão satisfatória. Para isso vamos tentar deixar parte da configuração centralizada na nossa WebSecurityConfig, removendo configurações adicionais em nossos controllers.

# Auth Database

Depois de contextualizada simples sobre segurança, hora de explorar novos recursos como interação com bando de dados.

Como sabemos o Spring Boot dispõe de uma facilidade para integrar nossa aplicação com bando de dados com o Spring Data JPA.

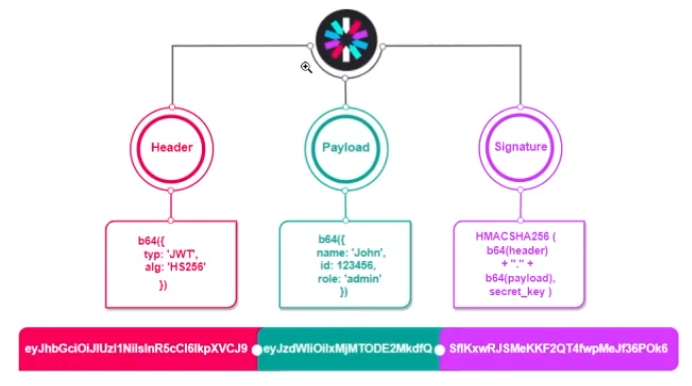
## UserDetailService

A interface UserDetailsService é usada para recuperar dados relacionados ao usuário. Ele possui um método denominado loadUserByUsername() que pode ser substituído para personalizar o processo de localização do usuário.

# JWT – JSON Web Token

Projeto Spring para demonstração do uso do Springboot, API Rest, Spring Security com JWT e Integração com banco de dados com Spring Data Jpa

O JSON Web Token é um padrão da internet para criação de dados com assinatura opcional e/ou criptografia cujo conteúdo contem o JSON que afirma algum numero de declarações. Os tokens são assinados usando um segredo privado ou uma chave pública/privada.



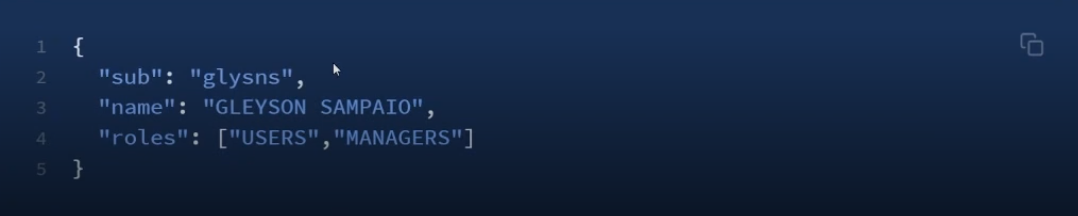
## Header

O header ou cabeçalho normalmente consiste em duas partes: o tipo de token, que é JWT e o algoritmo de assinatura que está sendo utilizado, como HMAC SHA256 ou RSA.



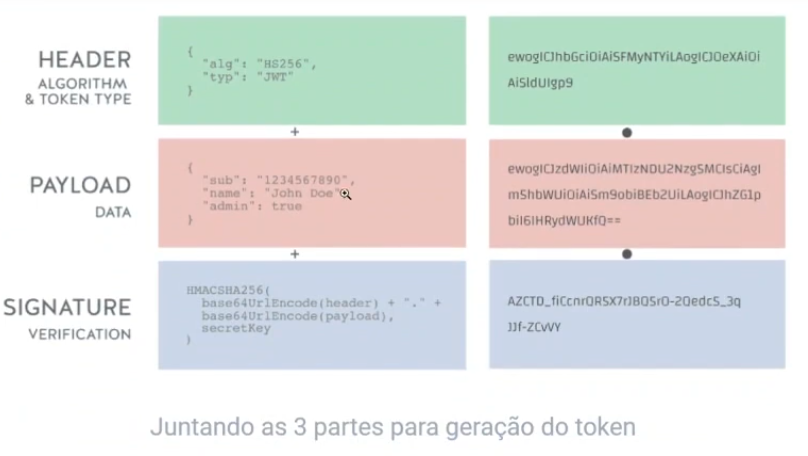
## Payload

De fato, a estrutura do corpo contendo as informações autenticações e autorização de um usuário.



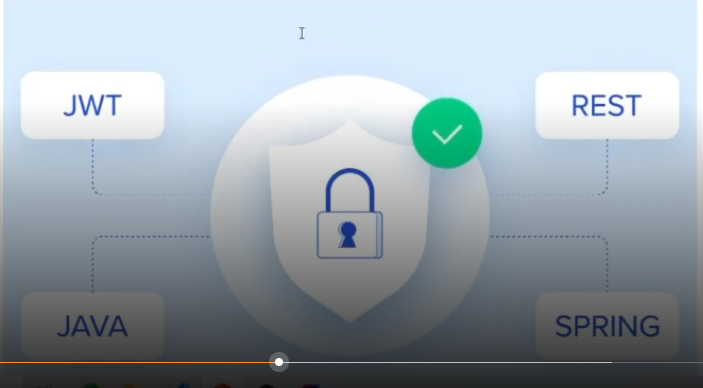
## Signature

Para criar a parte da assinatura, você deve pegar o cabeçalho codificado, o payload codificado, a chave secreta, o algoritmo especificado no cabeçalho e assina-lo.



## Spring Security + JWT

Neste tutorial, desenvolveremos um aplicativo Spring Boot que usa autenticação JWT para proteger uma API REST. Usaremos um usuário com perfis de acesso para geração e validação do token que transferido pelos clientes da nossa API.



# Estrutura do projeto

**Model** - Camada que contém as entidades da aplicação

**Dto** – Cama que contém os dtos da aplicação

**Repository** – Cama que contém os repositórios com base no Spring Data JPA

**Service** – Cama que detém da regra de negocio e comunicação com a base de dados via repositorys

**Controller** – Camada que contem os recursos https expostos na API

**Security** – Camada responsável para toda configuração de segurança