Why we need spring security?

Nowadays almost every client demands for implementation of powerful security. They want their application to be secure, there is no chance of data leak, and they want to maintain them

confidentiality, integrity, and availability of the data.

We have many types of authentication schemes available.

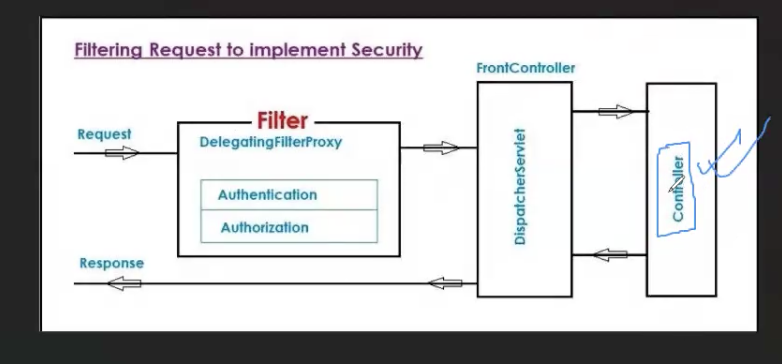
If you have any spring application, and you want to secure that application, you can use spring security, because spring security will provide you all the different type of security mechanism that you can apply on your application based on your requirement.

we have to secure our application. Whether it is a MVC. Application, whether it is res, Api, Application Micro service, or any other project

we have to secure the application.

And for that we need things security.

How does spring works?



So we know that in our application there are many controllers available.

These controllers are responsible for

accepting the client request and returning the response back to the client.

**But before your request reaches your controller, there is another controller in the application. This controller is basically a Dispatcher servlet.** This servlet is responsible for accepting any incoming request and

sending back any response to the client. **It is called Front Controller.**

So when we implement any security on the

application.

It basically

adds a filter, it basically adds a filter. a**nd that this filter will be responsible for securing your application.**

**now the incoming request. It will hit the filter first.st**

It will go through the filter and then it will reach the Controller

after the front controller, the request will reach the Controller.

**You can have many filters. You can have many filters depending on your application configuration. You can have many type of filters.** forbidden.

What does it mean?

**If you get the response**

in 4 0 1 or in 4 0 3. It simply means that

you are not authenticated and authorized to access the resource.

And your request is denied.

what is spring security? Spring security is basically

a chain of filte.

**These filters are responsible for security check in the application.**

If this

filters are passed. If the filters are clear.

then only your request will reach the front end controller from the front end controller. It will reach the Controller.

If your request

doesn't meet the

**authentication and authorization that are responsible**

to check the application security. If your request doesn't pass the

Whenever you add spring security by default, this filter will be applied.

and you can apply both filters. Also.

**let's say you want role based authorization. You want Jwt authentication, authorization, or any other type of filter.**

You can add those filters over here.

Both the terms are related to security.

What is authentication?

**Authentication is the process to identify a user is valid or not.**

It is a process of Validating that user. User is valid or invalid.

Authorization – Authorization is the process of authorizing resources. Permission.

**how do we implement the authorization?**

Mainly, there are 4 authorization, implementation available

i)permiAll 2) authenticated 3)has authority 4)hasAnyAuthority

**Permit all means authentication not required. Authorization not required. Anybody can access it**

just like any website. Homepage. Maybe any website websites you know, public pages

**what type of security implementation spring provides us?**

**Spring provides many types security depends yr need**

**1)in-Memeory security – store username and password in ram. Simplest way**

**2) using database(JDBC) – if u store username and password in db then we validate**

**3) userdetailsservice – username and password from data store**

**4)jwt token based – token based authentication and autorization**

**5)LDAP**

**6)oauth2**

**Spring security:**

**When u create yr project u want to use spring security u need to add spring security dependency**

**The moment u add this, default mechanism is app will ask username and password**