1)What is spring security ?

* Spring security is a separate module of the spring framework that focuses on providing authentication and authorization methods in java applications.
* It also takes care of most of the common security vulnerabilities such as CSRF attacks.
* To use spring security in web applications ,we can get started with the simple annotation @EnableWebSecurity.

2)What are some essential features of spring security?

* Supports authentication and authorization in a flexible and comprehensive manner.
* Java Authentication and Authorization service (JAAS) is used for authentication purposes.
* Allows single sign-on so that users can access multiple applications with just one account (username and password)

3)What is spring security authentication and authorization?

* Authentication : Verifying the identity of the user, using the credentials provide when accessing certain restricted resources.
* An example is logging into a website with a username and a password.
* Authorization : This ensures that users can only access the parts of a resource that they are authorized to access.
* An example is Role is assigned to login user like ADMIN, CUSTOMER,GUEST.

4)What do you mean by basic authentication?

* We send a username and password using the HTTP [Authorization ] header to enable us to access the resource.
* Usernames and passwords are encoded using base64 encoding (not encryption) in Basic Authentication.
* The encoding is not secure since it can be easily decoded.

Synatx

Value =username : password

Encoded Value = base64(Value)

Authorization Value = Basic<Encoded Value>

5)Explain securityContext and SecurityContext Holder ?

* SecurityContext : In this , information/data about the currently authenticated user(also known as the principal) is stored.
* SecurityContextHolder : Retrieving the currently authenticated principal is easiest via static call to the SecurityContextHolder.
* As a helper class, it provides access to the security context.

6)Explain spring security OAuth2.

* A simple authorization framework , OAuth 2.0 , permits client applications to access protected resources via an authorization server.
* Using it, a client application (third party) can gain limited access to an HTTP service on behalf of the resources owner or on its own behalf.
* Resource Owner/user , client, Authorizaion and Resource Server are involved in this concept.

7)What is method security and why do we need it ?

* The role of the user is used to determine which user is authorized to access the resource.
* A security measure applied to a method prevents unauthorized users and only allows authentic users.
* To prevent unauthorized users form performing activities beyond their privileges and roles.
* Method level security is implemented using AOP (Aspect-Oriented-Programming).

8)What do you mean by HASHING in spring security?

* Plain text passwords not good to be stored in your database
* Store encrypted passwords in a database. This is called password hashing.
* Encoding a string using the hashing algorithms like : MD4,MD5,SHA(SECURITY HASHING ALGORITHM) like SHA256,SHA128 ,etc..

9)What is PasswordEncoder?

* Password encoding is provided b spring security using the passwordEncoder interface. This interface defines two methods.
* Encode() : It converts a plain password into an encoded form.
* Matches() : It compares an encoded password from the database with a plain password (input by the user) that’s been encoded using the same salting and hashing algorithm as the encoded password.

10)Name security annotations that are allowed to use SpEL(spring expression language)

* @PreAuthorize, @PreFilter,@PostAuthorize and @PostFilter
* These provide expression-based access control.
* In spring security, @PreAuthorize is one of the most powerful annotations that allows you to use SpEL.
* But the old @Secured annotation cannot use it, for example you cannot write @Secured (“hashRole(‘ROLEADMIN’)”), but you can do @PreAuthorize(“hasRole(‘ROLEADMIN’)”).

11) What is AuthenticationManager in spring security ?

* It says “How authentication will happen”.
* AuthenticationManager contains references to all the AuthenticationProviders.
* AuthenticationManagers can perform one of the three actions in their authenticate() method : If valid return authenticated = true, invalid throw AuthenticationException unable to find valid input returns null.

12)What are some predefined filters used by spring scecurity ?

* The spring security filter chain is a very complex and flexible chain of filters.
* securityContextPersistenceFilter : This filter restores Authentication from the JSESSIONID cookie.
* UsernamePasswordAuthenticationFilter : This filter performs authentication.

13)Can you add custom filters in spring security’s filter chain ?

* Yes,you can add or replace individual filters with your own logic in spring’s security filter chain
* You may need to implement new functionality depending upon your project’s requirement and this can be done by creating a new filter to use in the chain.
* We do use Filter type like OncePerRequestFilter and register with security configuration.

14)How to implement a custom filter in spring security?

* We can implement a custom filter in spring security by using the org.springframework.web.filter.GenericFilterBean class.
* The GenericFilterBean is a simple javax.servlet.Filter implementation which is spring aware. You can override doFilter(ServletRequest req,ServletResponse res, FilterChain chain ) to implement your own logic..

15)What does @ and # is used for in spring Expression Language (EL)?

* The “@” symbol in spring EL is used to reference a spring bean
* The “#” symbol in spring EL allows you to reference a parameter on the method you are securing.
* Ex: #{authorization.name} prints username of current user logged-in

16)What do you mean by session management in spring security ?

* Session management relates to securing and managing multiple user’s sessions against their request.
* To control HTTP sessions, spring security uses the following options : SessionManagementFilter and SessionAuthenticationStrategy.
* SessionAuthenticatinStrategy takes care of session timeout ,session ids etc..

17)Explain salting and it’s usage?

* Spring security automatically applies salting since version 3.1.
* Salting is the process of combining random data with a password before password hashing.
* Salt improves hashing by increasing it’s uniqueness and complexity.
* Hashed passwords are then stored in a database , along with salt.

18)Which Servlet Filter , intercepts all the incoming requests sent to an application?

* Filter implementation named DelegatingFilterProxy that allows bridging between the servlet container’s lifecycle and spring’s ApplicationContext.
* The Servlet container allows registering Filters using it’s own standards.
* DelegatingFilterProxy can be registered via standard Servlet container mechanisms.

19)To secure which layer @EnableGlobalMethodSecurity annotation is used?

* We can use EnableGlobalMethodSecurity annotation to secure your service layer.
* It provides support for JSR-250 annotation security as well as the framework’s original @Secured annotation
* From 3.0 you can also make use of new expression-based annotations.

20) How to implement spring Boot + Basic Authentication ?

* Add spring-boot-starter-security maven dependency in pom.xml
* Configure the user and password in application.yml
* Configure spring security by extending WebSecurityConfigureAdapter.