|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1) | |  | | --- | | **What is the primary content of the Header part of JWT?** | | |  |  |  | | --- | --- | --- | |  | 1) | Signature with header, payload and secret key | |  | **2)** | **Encryption Algorithm** | |  | 3) | Secret Key | |  | 4) | User Data | | |
| 2) | |  | | --- | | **What is the syntax for defining Basic HTTP Authentication in HTTP Header?** | | |  |  |  | | --- | --- | --- | |  | **1)** | **Should start with Basic, then a space followed by user:password encoded in Base64** | |  | 2) | Should start with Bearer, then a space followed by user:password encoded in Base64 | |  | 3) | Should start with Basic, then a space followed by user:password encoded in Base68 | |  | 4) | Should start with Bearer, then a space followed by user:password encoded in Base68 | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 3) | |  | | --- | | **Which annotation enables spring security for web application?** | | |  |  |  | | --- | --- | --- | |  | 1) | @EnableMvcSecurity | |  | **2)** | **@EnableWebSecurity** | |  | 3) | @ConfigureWebSecurity | |  | 4) | @SecurityConfig | | |
| 4) | |  | | --- | | **Which method in Jwts checks if a JWT is valid or not?** | | |  |  |  | | --- | --- | --- | |  | 1) | isValid() | |  | 2) | valid() | |  | **3)** | **parseClaimsJws()** | |  | 4) | parseSigningKey() | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5) | |  | | --- | | **What does Spring Boot do when there is no security configuration defined?** | | |  |  |  | | --- | --- | --- | |  | 1) | Creates default user, password and role | |  | **2)** | **Generates a default password and displays it in the log** | |  | 3) | No default configuration. All resources will be restricted for access. | | |
| 6) | |  | | --- | | **Which option in curl command helps defining the HTTP Basic Authentication**  **credentials?** | | |  |  |  | | --- | --- | --- | |  | 1) | -s | |  | **2)** | **-u** | |  | 3) | -a | |  | 4) | -H | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7) | |  | | --- | | **Spring Security is a framework that focuses on providing both authentication and**  **authorization to Java applications** | | |  |  |  | | --- | --- | --- | |  | 1) | FALSE | |  | **2)** | **TRUE** | | |
| 8) | |  | | --- | | **Which encoding is used to encode the three parts of JWT and concatenated using**  **periods to produce the JWT?** | | |  |  |  | | --- | --- | --- | |  | 1) | Bcrypt Encoding | |  | 2) | Binary Encoding | |  | **3)** | **Base64Url Encoding** | |  | 4) | Base32Url Encoding | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 9) | |  | | --- | | **Which method in JwtsBuilder can be used to setting the expiry of a token?** | | |  |  |  | | --- | --- | --- | |  | **1)** | **setExpiration()** | |  | 2) | expire() | |  | 3) | setExpiryTime() | |  | 4) | setExpiry() | | |
| 10) | |  | | --- | | **JWT stands for \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | | |  |  |  | | --- | --- | --- | |  | 1) | Java Web Token | |  | 2) | JSP Web Token | |  | **3)** | **JSON Web Token** | |  | 4) | JAR Web Token | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 11) | |  | | --- | | **Select all options that forms part of the flow of JWT process** | | |  |  |  | | --- | --- | --- | |  | **1)** | **Server validates credentials and creates token** | |  | **2)** | **Server looks for valid credentials in each request** | |  | **3)** | **Client stores the token received from Server and sends it in subsequent requests** | |  | 4) | Client sends the session id received from server in subsequent request | | |
| 12) | |  | | --- | | **Select all options that forms part of JWT** | | |  |  |  | | --- | --- | --- | |  | **1)** | **Signature** | |  | **2)** | **Payload** | |  | **3)** | **Header** | |  | 4) | Footer | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 13) | |  | | --- | | **Which concept helps in validating JWT of a REST API Web Application?** | | |  |  |  | | --- | --- | --- | |  | 1) | Logging | |  | **2)** | **Filters** | |  | 3) | Threading | |  | **4)** | **Include JWT Token validation in each request** | | |
| 14) | |  | | --- | | **Which method of Spring Security Framework can be used to define the REST API**  **URL?** | | |  |  |  | | --- | --- | --- | |  | 1) | hasRole() | |  | 2) | and() | |  | 3) | authorizeRequests() | |  | **4)** | **antMatchers()** | | |