

JWT Interview Revision Sheet

1. JWT (JSON Web Token) is a secure token format used for authentication and authorization. It allows servers to verify user identity without storing session data.
2. A JWT has three parts: Header, Payload, and Signature.
3. Payload contains user data (e.g., userId, role). Signature ensures token integrity and verifies it was signed with the correct secret key.
4. `jwt.sign()` creates and signs a JWT token using a secret key.
5. `jwt.verify()` validates the token, checks the secret key, and ensures it is not expired.
6. JWT secret keys should be stored in a `.env` file to protect sensitive information and prevent exposure in source code.
7. If the verification secret differs from the signing secret, token verification fails with an invalid signature error.
8. The `expiresIn` option sets the token expiration time for security purposes.
9. After successful login, the backend sends the token to the client in the response (JSON or HTTP-only cookie).
10. The client should send the token in the Authorization header as: `Bearer .`
11. If the token is missing, return 401 Unauthorized.
12. If the token is invalid or expired, return 403 Forbidden (or 401 depending on implementation).
13. Authentication verifies who the user is. Authorization determines what the user can access.
14. JWT is stateless because the server does not store session data; all necessary information is inside the token.
15. To protect routes in Express, create middleware that extracts the token, verifies it using `jwt.verify()`, and allows access if valid.