Spring security - II

Teaching Faculty: Dr. Muhyieddin Al-Tarawneh

Prepared by Muhyieddin **AL-TARAWNEH**, Umur **INAN**

Refresh Tokens

- APIs may return two tokens
  - Access token with an expiration time
  - Refresh token with no expiration time

- Refresh token used to get a new access token
  - No additional authentication required

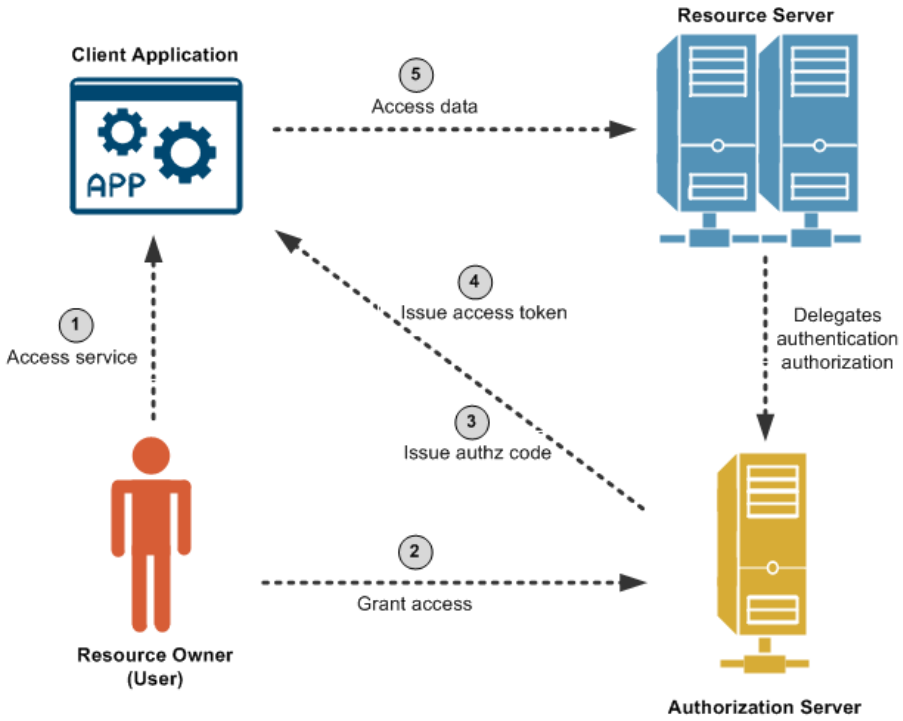
OAuth 2.0

- OAuth 2.0 is the industry-standard protocol for authorization.
- OAuth 2.0 focuses on client developer simplicity while providing specific authorization flows for web applications, desktop applications, mobile phones, and living room devices.
- It works by delegating user authentication to the service that hosts a user account and authorizing third-party applications to access that user account.

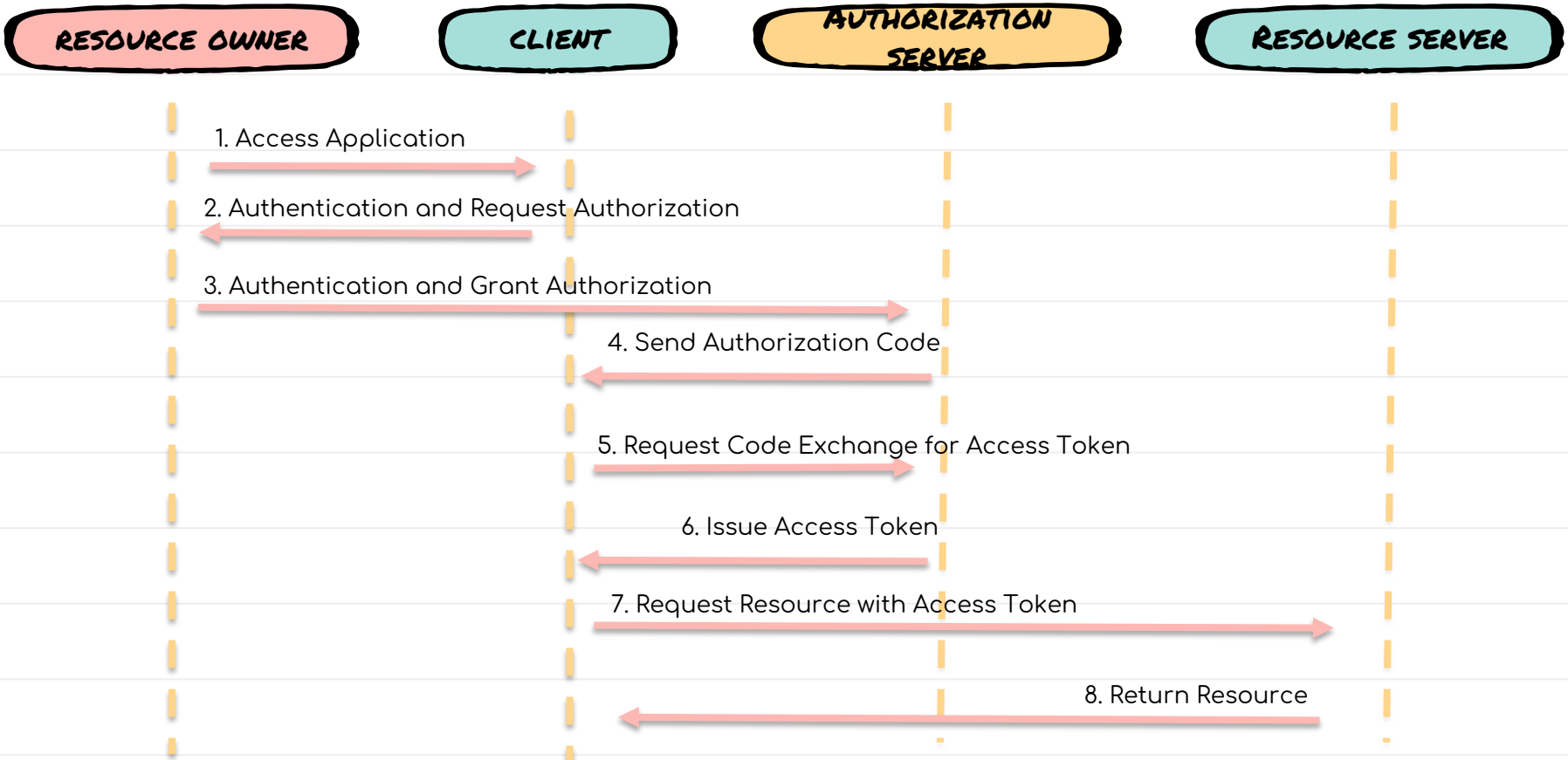
Delegated authorization

- OAuth2 is a standard protocol that solves the delegated authorization problem.
- Users may give permission to "Some App" to access resources on "Another App" so the app can access some resources.
- This is a better way than giving the App our username/password to access resources on our behalf.
- Unfortunately, some apps (Banks, Mint: financial dashboard) are still collecting username/password to access users' data.
- I trust Google but I kind of trust this new App. I want the App to have access to my contacts only.

OAuth 2.0 flow



Authorization code flow



Prepared by Muhyieddin **AL-TARAWNEH**, Umur **INAN**

Resource Owner

- Normally your application's end user that grants permission to access the resource server with an access token.

- A person or system capable of granting access to a protected resource.

Client

- The application that requests the access token

Resource Server

- Accepts the access token and must verify that it's valid. In this case this is your application.
- The resource server is the OAuth 2.0 term for your API server.
- The resource server handles authenticated requests after the application has obtained an access token.
- Large scale deployments may have more than one resource server.

Authorization Server

- The server that issues the access token.

Authorization Code

- The authorization code is obtained by using an authorization server as an intermediary between the client and resource owner.

- The code itself is obtained from the authorization server where the user gets a chance to see what the information the client is requesting and approve or deny the request.

https://www.oauth.com/oauth2-servers/server-side-apps/authorization-code/

Access Token

- An OAuth Access Token is a string that the OAuth client uses to make requests to the resource server.
- Access tokens do not have to be in any particular format, and in practice, various OAuth servers have chosen many different formats for their access tokens.
- Access tokens may be either "bearer tokens" or "sender-constrained" tokens.

Access Token

- Access tokens must not be read or interpreted by the OAuth client. The OAuth client is not the intended audience of the token.
- Access tokens do not convey user identity or any other information about the user to the OAuth client.
- Access tokens should only be used to make requests to the resource server. Additionally, ID tokens must not be used to make requests to the resource server.

Refresh Token

- An OAuth Refresh Token is a string that the OAuth client can use to get a new access token without the user's interaction.
- A refresh token must not allow the client to gain any access beyond the scope of the original grant.
- The refresh token exists to enable authorization servers to use short lifetimes for access tokens without needing to involve the user when the token expires.

Authorization Code Grant

- The Authorization Code grant type is used by confidential and public clients to exchange an authorization code for an access token.

- It is recommended that all clients use the PKCE extension with this flow as well to provide better security.

OpenID Connect

- The OAuth 2.0 framework explicitly does not provide any information about the user that has authorized an application.

- OAuth 2.0 is a delegation framework, allowing third-party applications to act on behalf of a user, without the application needing to know the identity of the user.

OpenID Connect

- OpenID Connect takes the OAuth 2.0 framework and adds an identity layer on top.

- It provides information about the user, as well as enables clients to establish login sessions.

OAuth ++ OpenID Connect

- OAUTH
  - Granting access to the API
  - Getting access to user data in other systems.
  - Authorization
- OpenID Connect
  - Logging the user in
  - Authentication

KEYCLOAK

- Keycloak is an open-source Identity and Access Management solution targeted towards modern applications and services.
    - Single-Sign-On (SSO)
    - Identity Brokering and Social Login
    - User Federation

KEYCLOAK

- Client Adapters
- Admin Console
- Account Management Console

REALMS

- A realm is a space where you manage objects, including users, applications, roles, and groups.
    - A user belongs to and logs into a realm.
- One Keycloak deployment can define, store, and manage as many realms as there is space for in the database.

Main points

- Security underlies an entire enterprise. It provides a shield that makes the application invulnerable.

- Transcendental Consciousness is characterized by the quality of invincibility, which means one cannot be overcome or overpowered.