**OAuth:**

It stands for Open Authorization. OAuth is an authorization framework that enables applications to obtain limited access to user accounts on an HTTP service, such as Facebook, GitHub. It works user authentication to the service that hosts the user account, and authorizing third-party applications to access the user account. OAuth provides authorization flows for web and desktop applications, and mobile devices.

**OAuth Roles:**

OAuth defines four roles:

* Resource Owner
* Client
* Resource Server (API)
* Authorization Server

### Resource Owner (User)

The user who authorizes an application to access their account. The application’s access to the user’s account is limited to the “scope” of the authorization granted.

### Resource Server (API)

The resource server hosts the protected user accounts, and the authorization server verifies the identity of the user then issues access tokens to the application.

### Client:

The client is application that wants to access the user’s account. Before it may do so, it must be authorized by the user, and the authorization must be validated by the API.

**Application Registration:**

Before using OAuth with your application, you must register your application with the service. This is done through a registration form in the “developer” or “API” portion of the services.

### **Client ID and Client Secret:**

Once our application is registered, the service will issue “client credentials” and a client secret.

Client id is a is used by the service API to identify the application, and is also used to build authorization URLs that are presented to users.

Client Secret is used to authenticate the identity of the application to the service API when the application requests to access a user’s account, and must be kept private between the application and the API.

**Authorization Grant :**

The Abstract Protocol Flow above, the first four steps cover obtaining an authorization grant and access token.

The authorization grant type depends on the method used by the application to request authorization, and the grant types supported by the API.

It defines four grant types,

* **Authorization Code**:

It used with server-side Applications

* **Implicit**:

It used with Mobile Apps or Web Applications

* **Resource Owner Password Credentials**:

It used with trusted Applications, such as those owned by the service itself.

* **Client Credentials**:

It used with Applications API access.