OAuth2.0协议定义了用于获得授权的四种主要授权类型。

Authorization code

标准的Server授权模式，非常适合Server端的Web应用。一旦资源的拥有者授权访问他们的数据之后，他们将会被重定向到Web应用并在URL的查询参数中附带一个授权码（code）。在客户端里，该code用于请求访问令牌（access_token）。并且该令牌交换的过程是两个服务端之前完成的，防止其他人甚至是资源拥有者本人得到该令牌。另外，在该授权模式下可以通过refresh_token来刷新令牌以延长访问授权时间。

Implicit Grant

该模式是所有授权模式中最简单的一种，并为运行于浏览器中的脚本应用做了优化。当用户访问该应用时，服务端会立即生成一个新的访问令牌（access_token）并通过URL的#hash段传回客户端。这时，客户端就可以利用JavaScript等将其取出然后请求API接口。该模式不需要授权码（code），当然也不会提供refresh token以获得长期访问的入口。

Resource Owner Password Credentials

这种模式要求用户提供用户名和密码来交换访问令牌（access_token）。该模式仅用于非常值得信任的用户，例如API提供者本人所写的移动应用。虽然用户也要求提供密码，但并不需要存储在设备上。因为初始验证之后，只需将OAuth的令牌记录下来即可。如果用户希望取消授权，因为其真实密码并没有被记录，因此无需修改密码就可以立即取消授权。token本身也只是得到有限的授权，因此相比最传统的username/password授权，该模式依然更为安全。

Client Credentials

一种基于APP的密钥直接进行授权，因此APP的权限非常大。它适合像数据库或存储服务器这种对API的访问需求。