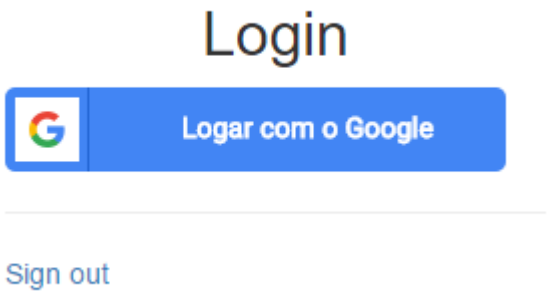
OAuth

Diferente do modelo de autenticação cliente-servidor, o OAuth é uma API de autorização que possibilita que uma aplicação faça autenticação em outra aplicação. Isso ocorre quando uma aplicação solicita acesso para o usuário sem que a mesma tenha acesso às senhas do usuário. Contudo, fica a cargo do usuário conceder esse acesso à aplicação ou não, mas caso ele conceda, o mesmo ainda pode anular a permissão da aplicação quando bem entender. No caso de o usuário desejar alterar a senha, não há necessidade de reconfirmar a permissão à aplicação uma vez que a mesma continuará válida. O OAuth é empregado em diferentes formatos de autenticação. Um bom exemplo de uso prático do protocolo são as solicitações de autenticação pela conta do Google ou do Facebook, que, para evitar realizar um cadastro completo quando estamos em uma página ou aplicativo de terceiros, sugere logar pela conta de uma dessas aplicações. Quando o usuário opta por uma dessas opções de autenticação, ele está permitindo que a página ou aplicativo em questão utilize ferramentas da própria aplicação e dá acesso aos seus dados de forma limitada por meio do HyperText Transfer Protocol, ou HTTP.



1. Para que serve e para o que foi criado o protocolo OAuth?

Serve para autorizar que uma aplicação faça autenticação em terceiras. Foi criado na intenção de adicionar um nível a mais de autorização com o intuito de distanciar o papel do usuário do recurso proprietário, de modo que o usuário corra menos riscos ao realizar a autenticação em páginas de terceiros.

2. Descreva o fluxo do protocolo OAuth na versão 2.0"

Quais os agentes envolvidos?

Resource owner (Proprietário do recurso): Usuário que dá a autorização de acesso aos dados.

Client (Aplicação): Responsável por interagir com o proprietário do recurso.

Authorization server (Servidor de autorização): Realiza a autenticação do proprietário do recurso e concede os tokens de acesso.

Resource server (Servidor de recurso): Detentor dos recursos protegidos. Concede acesso a esses recursos por meio do token de acesso concedido pelo servidor de autorização.

Qual o fluxo de informação entre estes agentes?

1º a aplicação solicita autorização ao proprietário do recurso para acessar os seus recursos.

2º caso o proprietário do recurso autorize a solicitação, a aplicação recebe a Concessão de Autorização do proprietário do recurso.

3º a aplicação utiliza a própria identidade e a Concessão de Autorização para conseguir um token de acesso do servidor de autorização.

4º o Servidor de Autorização realiza a autenticação da identidade da aplicação e da Concessão de Autorização e em seguida devolve um token de acesso à aplicação.

5º a aplicação apresenta o token de acesso ao servidor de recurso e solicita um recurso.

6º o servidor de recurso autentica o token de acesso e na sequência providencia o recurso solicitado pela aplicação.

3. Descreva como este serviço, se mal utilizado, pode trazer problemas de segurança para uma empresa.

Em certos casos a URL de retorno de uma página redireciona o usuário para outra que não possui o domínio autorizado configurado, enganando então a verificação. Isso acontece porque em certas formas de uso o protocolo concede o token de acesso às páginas desautorizadas sem realizar uma verificação. Essa “brecha” pode colocar os usuários do sistema à mercê de qualquer um que queira acessar as contas dos usuários no servidor da empresa no caso.

4. Cite pelo menos 10 serviços, de grandes empresas provedoras de autorização que utilizam, este protocolo.

Amazon Cognito – Amazon

Apple Developer – Apple

Discord Developer – Discord

Dropbox – Dropbox

Facebook Developers – Facebook

Github Developer – GitHub

Gmail – Google

Instagram Platform – Facebook

Hotmail – Microsoft

Xbox - Microsoft