Desenvolvimento de Componentes Distribuídos - API

Karen Christina Diz – 16/03/21

Descrição da Tarefa:

Pesquise e escreva um texto sobre a API de autorização chamada OAuth.

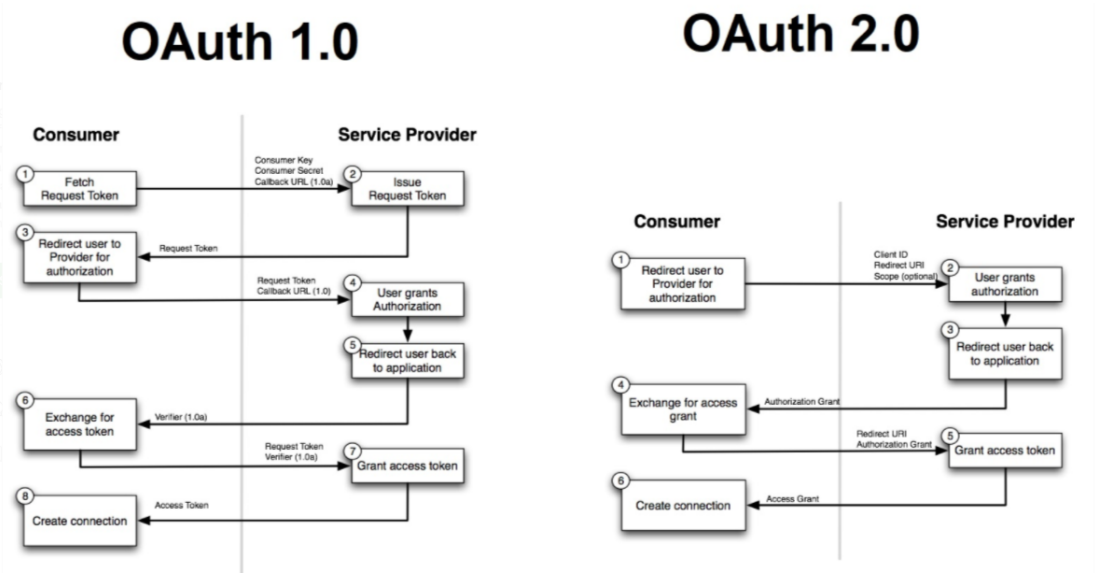
1) Para que serve e para o que foi criado o protocolo OAuth? Utilize de imagens e exemplos.

O OAuth é um protocolo padrão de autorização *Web*, que é utilizado em vários tipos de autenticação como nas telas de login e também na autenticação de APIs. Foi criado para que sites ou aplicativos tenham acesso a recursos protegidos sem que os usuários divulguem suas credencias.

A primeira versão do OAuth foi a 1.0, que foi publicada em 2007, e em seguida já começou a ser amplamente utilizada, mas somente em 2010 que foi publicado como RFC 5849 pelo IEFT. As RFCs são documentos técnicos desenvolvidos e mantidos pelo IETF (Internet Enginnering Task Force), que é uma instituição que especifica os padrões que serão implementados e utilizados em toda a internet.

Mas no mesmo ano a IEFT publicou uma nova versão do OAuth, a 2.0 que foi criada com o intuito de resolver alguns problemas de complexidade e escalabilidade que a versão anterior possuía, como por exemplo a instalação e configuração de bibliotecas para solicitações a API, utilização de usuário e senha, tokens de longa duração.

O OAuth 2.0 é uma versão bastante flexível, extensível, e que possui muitas funcionalidades de segurança para que os usuários permitam acesso aos seus recursos protegidos, permitindo que os desenvolvedores utilizem o protocolo de diversos modos para atender aos requisitos de suas aplicações.



Fonte: https://medium.com/@greekykhs/whats-the-difference-oauth-1-0-and-oauth-2-0-9f1d22e06963

2. Descreva o fluxo do protocolo OAuth na versão 2.0. Quais os agentes envolvidos? Qual o fluxo de informação entre estes agentes?

Para entender como funciona o fluxo do protocolo OAuth na versão 2.0, é necessário primeiramente entender que esse fluxo foi construído em cima de 4 papéis(*roles*) principais que são:

- Proprietário do Recurso (*Resource Owner*): Entidade de acesso aos dados, dono do recurso
- Cliente (*Client*): Aplicação que interage com a entidade de acesso aos dados
- Servidor de Recurso (*Resource Server*): A API que está exposta precisa de proteção dos dados, para conseguir acesso ao seu conteúdo é necessário um token emitido pelo servidor de autorização
- Servidor de Autorização (*Authorization Server*): Responsável por autenticar o usuário e emitir os tokens de acesso, é ele que possui as informações do proprietário do recurso, autentica e interage com o usuário após a identificação do cliente.

Assim o fluxo do OAuth 2.0 funciona da seguinte forma:

1) A aplicação solicita a autorização para acessar os recursos do servidor do usuário
2) Se o usuário permitir a autorização a aplicação recebe uma concessão de autorização
3) A aplicação solicita um token de acesso ao servidor de autorização API através da autenticação da própria identidade e concessão de autorização
4) Se a identidade da aplicação estiver autenticada e a concessão de autorização for valida o servidor de autorização API fornece um token de acesso para a aplicação. Nessa etapa a autorização já está completa e o cliente já possui um token de acesso para gerenciar
5) A aplicação quando precisar solicitar um recurso ao servidor de recursos pode utilizar o token de acesso de autenticação
6) Se o token de acesso for válido o servidor de recurso fornece o recurso para a aplicação.

Fonte: https://www.digitalocean.com/community/tutorials/uma-introducao-ao-oauth-2-pt

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

O protocolo OAuth se mal utilizado pode trazer vários problemas de segurança para uma empresa como:

- A obtenção de recursos confidenciais

- Fazer com que terceiros possam interceptar as solicitações do consumidor e retornar respostas enganosas ou incorretas

- Divulgar os segredos dos tokens

- Ataque de *pishing* para capturar as credenciais do usuário usando um navegador comprometido.

- *Covert Redirect*, ataque de redirecionamento aberto com a intenção de fazer com que o servidor de autorização redirecione a resposta do OAuth para outro local malicioso.

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

Google - autenticação em smartphones

Amazon - integração com outras lojas

GitHub - integração com várias IDEs, ambiente de desenvolvimento integrado

LinkedIn - link com grandes empresas de contratação de emprego

PayPal - autorização para fazer pagamento em diversos sites

Facebook - permite várias aplicações terceiras utilizem o perfil e os dados do usuário

Spotify - integração com vários dispositivos e aplicações, como por exemplo o waze

Discord - integração com SoundCloud, Spotify e Deezer

Dropbox - integração com o WhatsApp para compartilhar arquivos

Instagram - integração aplicações de criadores de conteúdo para gerenciar suas contas