Google Cloud

Partner Certification Academy

Professional Cloud Developer

v2309

Quiz questions: OAuth2*

** These are for practice only and are not actual exam questions*

In the OAuth 2.0 framework, what is the purpose of the token obtained from the Authorization Server when using Google APIs?

a) It represents the user's login credentials.
b) It allows access to the resource server.
c) It contains the user's personal data.
d) It defines the application's branding information.

Where can you obtain OAuth 2.0 credentials, including a client ID and client secret, for authenticating users with Google's APIs?

a) Google OAuth 2.0 Playground

b) Google API Console
c) Google Identity Services
d) Google Drive

What is the purpose of setting a redirect URI in the API Console when configuring OAuth 2.0 credentials for Google authentication?

a. To display user consent screen information
b. To specify the user's email address
c. To validate the identity of the user
d. To receive the response from Google

Which flow is recommended for authenticating users when a client-side application needs to access Google APIs directly, such as a JavaScript app running in the browser?

a. Server flow
b. Implicit flow
c. Token flow
d. Code flow

What is the recommended approach for implementing an implicit flow for user authentication when using Google's OAuth 2.0 authentication system for a client-side application?

a. Use Google Identity Services
b. Implement your own custom authentication flow\
c. Employ the server flow with a client-side application

d. Use the Google APIs client library for PHP

Answers:

In the OAuth 2.0 framework, what is the purpose of the token obtained from the Authorization Server when using Google APIs?

(b) It allows access to the resource server. Google APIs utilize and expand upon the OAuth 2.0 framework, which defines multiple authentication methods or "flows." Typically, an application provides credentials, representing either a user or a service account, to an Authorization Server. The Authorization Server replies with a token, serving as authentication for accessing a service and its resources. This token specifies one or more scopes, indicating the permissions granted to the application. The application subsequently presents this token to a resource server to gain access to the desired resources. In essence, OAuth 2.0 within Google APIs ensures secure authentication and resource access.
https://cloud.google.com/docs/authentication/

Where can you obtain OAuth 2.0 credentials, including a client ID and client secret, for authenticating users with Google's APIs?

(b) Google API Console. Before your application can use Google's OAuth 2.0 authentication system for user login, you must set up a project in the Google API Console to obtain

OAuth 2.0 credentials, set a redirect URI, and (optionally) customize the branding information that your users see on the user-consent screen. https://developers.google.com/identity/openid-connect/openid-connect

What is the purpose of setting a redirect URI in the API Console when configuring OAuth 2.0 credentials for Google authentication?

d) To receive the response from Google. The redirect URI that you set in the API Console determines where Google sends responses to your authentication requests. https://developers.google.com/identity/openid-connect/openid-connect

Which flow is recommended for authenticating users when a client-side application needs to access Google APIs directly, such as a JavaScript app running in the browser?

(b) Implicit flow. The most commonly used approaches for authenticating a user and obtaining an ID token are called the "server" flow and the "implicit" flow. The server flow allows the back-end server of an application to verify the identity of the person using a browser or mobile device. The implicit flow is used when a client-side application (typically a JavaScript app running in the browser) needs to access APIs directly instead of via its back-end server.

https://developers.google.com/identity/openid-connect/openid-connect
https://cloud.google.com/docs/authentication/

What is the recommended approach for implementing an implicit flow for user authentication when using Google's OAuth 2.0 authentication system for a client-side application?

(e) Use Google Identity Services
https://developers.google.com/identity/openid-connect/openid-connect
https://cloud.google.com/docs/authentication/
https://developers.googleblog.com/2021/07/launching-our-new-google-identity-services-apis.html