Google Cloud

Partner Certification Academy

Professional Cloud Developer

v2309

Quiz questions*

Application Default Credentials (ADC)

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

Question: What is the primary purpose of Application Default Credentials (ADC) in Google Cloud?

A. To provide a static API key for all Google Cloud services.
B. To automatically find credentials based on the application environment.
C. To enable two-factor authentication for Google Cloud services.
D. To provide a unique user ID for each Google Cloud service.

Question: What is the recommended way to store and access application secrets and keys when using ADC?

A. Hardcoding them in the application.
B. Storing them in a public repository.
C. Using Google Cloud Secret Manager.

D. Sharing them via email.

Question: Which of the following is NOT a feature of ADC?

A. Automatically finding credentials based on the application environment.
B. Providing a static API key for all applications.
C. Authenticating applications running on Google Cloud services.
D. Using the credentials of the service account associated with the GCP project.

Question: Which command is used to provide user credentials to ADC in a local development environment?

A. gcloud auth application-default set
B. gcloud auth application-default login
C. gcloud auth application-default init
D. gcloud auth application-default user-login

Question: Which of the following is a security risk if not managed correctly when setting up ADC?

A. User credentials
B. Service account impersonation
C. Service account keys
D. Environment variables

Question: When using ADC, where does the environment variable GOOGLE_APPLICATION_CREDENTIALS point to?

A. The path of the JSON file that contains your user credentials.
B. The path of the JSON file that contains your service account key.
C. The path of the ADC configuration file.

D. The path of the Google Cloud SDK installation.

Question: Which command allows you to use service account impersonation to set up a local Application Default Credentials (ADC) file?

A. gcloud auth application-default impersonate SERVICE_ACCT_EMAIL
B. gcloud auth application-default login --use-service-account SERVICE_ACCT_EMAIL
C. gcloud auth application-default login --impersonate-service-account SERVICE_ACCT_EMAIL
D. gcloud service-account impersonate ADC

Question: In which scenarios is it recommended to use service account impersonation for ADC?

A. When you want to hardcode credentials in the application.
B. When you want to use static API keys for authentication.
C. When you want to set up ADC with credentials from a service account without exposing the private key.
D. When you want to use user credentials for all Google Cloud services.

Question: What is a potential security risk when setting up ADC using service account keys?

A. The keys are automatically rotated by Google Cloud.
B. The keys can be easily revoked using the Google Cloud Console.
C. The keys are stored in a highly encrypted format.
D. If not managed correctly, service account keys can be exposed, leading to unauthorized access.

Question: When running code in Google Cloud, which of the following is the recommended method for authenticating applications to access Google Cloud services?

A. Hardcoding the credentials in the application.
B. Using a static API key.
C. Attaching a service account to the resource.
D. Setting the environment variable GOOGLE_APPLICATION_CREDENTIALS to the path of the service account key.

Correct Answer: C. Attaching a service account to the resource.

Question: When your application is running on Google Cloud services like Compute Engine or Cloud Run and you haven't set the GOOGLE_APPLICATION_CREDENTIALS environment variable, what does ADC use to authenticate your application?

A. The default service account credentials stored in a well-known location on your file system.
B. The credentials provided by the user during runtime.
C. The credentials from the metadata server associated with the attached service account.
D. The credentials obtained by the gcloud auth application-default login command.

Question: Which of the following is a characteristic of the Compute Engine default service account?

A. It is automatically created when the Compute Engine API is enabled.
B. It is attached by default to all VMs created using third-party tools.
C. It is automatically granted the IAM basic Owner role.
D. It is automatically granted the IAM basic Viewer role.

Question: What is the primary purpose of a service account in Google Cloud?

A. To represent human users for browser-based sign-in.
B. To represent non-human users for accessing resources or performing actions without end-user involvement.
C. To manage Google Cloud resources using static API keys.
D. To provide SSL and mTLS authentication for Google Cloud services.

Question: When using service accounts with VMs in Compute Engine, which of the following is true?

A. A single VM can have multiple service accounts attached to it.
B. If you attach the same service account to multiple VMs, changes to the service account do not affect any VMs.
C. Service accounts are associated with a particular employee.
D. You can attach the same service account to multiple VMs, but changes to the service account affect all VMs using it.

Question: Which of the following is the preferred method for finding credentials in a production environment on Google Cloud?

A. Using the GOOGLE_APPLICATION_CREDENTIALS environment variable.
B. Hardcoding the credentials in the application.
C. Using user credentials set up by the Google Cloud CLI.
D. Using the credentials from the attached service account via the metadata server.

Answers to Quiz questions

Application Default Credentials (ADC)

Question: What is the primary purpose of Application Default Credentials (ADC) in Google Cloud?

E. To provide a static API key for all Google Cloud services.
F. To automatically find credentials based on the application environment.
G. To enable two-factor authentication for Google Cloud services.
H. To provide a unique user ID for each Google Cloud service.

Correct Answer: B. To automatically find credentials based on the application environment.

Explanation: ADC is a strategy used by Google authentication libraries to automatically find credentials depending on the application's environment.
Resource Link: Set up Application Default Credentials - Google Cloud

Question: What is the recommended way to store and access application secrets and keys when using ADC?

E. Hardcoding them in the application.
F. Storing them in a public repository.
G. Using Google Cloud Secret Manager.
H. Sharing them via email.

Correct Answer: C. Using Google Cloud Secret Manager.

Explanation: Google Cloud Secret Manager provides a secure and recommended way to store and access application secrets and keys.
Resource Link: Authenticate to Secret Manager

Question: Which of the following is NOT a feature of ADC?

E. Automatically finding credentials based on the application environment.
F. Providing a static API key for all applications.
G. Authenticating applications running on Google Cloud services.

H. Using the credentials of the service account associated with the GCP project.

Correct Answer: B. Providing a static API key for all applications.

Explanation: ADC does not provide static API keys. Instead, it finds credentials based on the application's environment.
Resource link: Set up Application Default Credentials

Question: Which command is used to provide user credentials to ADC in a local development environment?

E. gcloud auth application-default set
F. gcloud auth application-default login
G. gcloud auth application-default init
H. gcloud auth application-default user-login

Correct Answer: B. gcloud auth application-default login

Explanation: To provide your user credentials to ADC in a local development environment, you use the command gcloud auth application-default login.
Resource Link: Set up Application Default Credentials - Google Cloud

Question: Which of the following is a security risk if not managed correctly when setting up ADC?

E. User credentials
F. Service account impersonation
G. Service account keys
H. Environment variables

Correct Answer: C. Service account keys

Explanation: Service account keys are a security risk if not managed correctly. It's essential to choose a more secure alternative to service account keys whenever possible. Resource Link: Set up Application Default Credentials - Google Cloud

Question: When using ADC, where does the environment variable GOOGLE_APPLICATION_CREDENTIALS point to?

E. The path of the JSON file that contains your user credentials.
F. The path of the JSON file that contains your service account key.
G. The path of the ADC configuration file.
H. The path of the Google Cloud SDK installation.

Correct Answer: B. The path of the JSON file that contains your service account key.

Explanation: When using a service account key for ADC, you need to set the environment variable GOOGLE_APPLICATION_CREDENTIALS to the path of the JSON file that contains your service account key.
Resource Link: Set up Application Default Credentials - Google Cloud

Question: Which command allows you to use service account impersonation to set up a local Application Default Credentials (ADC) file?

E. gcloud auth application-default impersonate SERVICE_ACCT_EMAIL
F. gcloud auth application-default login --use-service-account SERVICE_ACCT_EMAIL
G. gcloud auth application-default login --impersonate-service-account SERVICE_ACCT_EMAIL
H. gcloud service-account impersonate ADC

Correct Answer: C. gcloud auth application-default login --impersonate-service-account SERVICE_ACCT_EMAIL

Explanation: The command gcloud auth application-default login --impersonate-service-account SERVICE_ACCT_EMAIL allows you to use service account impersonation to set up a local ADC file.
Resource Link: Set up Application Default Credentials - Google Cloud

Question: In which scenarios is it recommended to use service account impersonation for ADC?

E. When you want to hardcode credentials in the application.
F. When you want to use static API keys for authentication.
G. When you want to set up ADC with credentials from a service account without exposing the private key.
H. When you want to use user credentials for all Google Cloud services.

Correct Answer: C. When you want to set up ADC with credentials from a service account without exposing the private key.

Explanation: Service account impersonation allows you to set up ADC without exposing the private key of the service account, providing a more secure authentication mechanism.
Resource Link: Set up Application Default Credentials - Google Cloud

Question: What is a potential security risk when setting up ADC using service account keys?

E. The keys are automatically rotated by Google Cloud.
F. The keys can be easily revoked using the Google Cloud Console.
G. The keys are stored in a highly encrypted format.
H. If not managed correctly, service account keys can be exposed, leading to unauthorized access.

Correct Answer: D. If not managed correctly, service account keys can be exposed, leading to unauthorized access.

Explanation: Service account keys are a security risk if not managed correctly. It's essential to choose a more secure alternative to service account keys whenever possible.
Resource Link: Set up Application Default Credentials - Google Cloud

Question: When running code in Google Cloud, which of the following is the recommended method for authenticating applications to access Google Cloud services?

E. Hardcoding the credentials in the application.
F. Using a static API key.
G. Attaching a service account to the resource.
H. Setting the environment variable GOOGLE_APPLICATION_CREDENTIALS to the path of the service account key.

Correct Answer: C. Attaching a service account to the resource.

Explanation: In Google Cloud, when running code, it's recommended to attach a service account to the resource (e.g., VM instance) to provide credentials to applications running on it.
Resource Link: Best practices for using service accounts - Google Cloud

Question: When your application is running on Google Cloud services like Compute Engine or Cloud Run and you haven't set the GOOGLE_APPLICATION_CREDENTIALS environment variable, what does ADC use to authenticate your application?

E. The default service account credentials stored in a well-known location on your file system.
F. The credentials provided by the user during runtime.
G. The credentials from the metadata server associated with the attached service account.
H. The credentials obtained by the gcloud auth application-default login command.

Correct Answer: C. The credentials from the metadata server associated with the attached service account.

Explanation: If the GOOGLE_APPLICATION_CREDENTIALS environment variable is not set and the application is running on Google Cloud services, ADC uses the

metadata server to get credentials associated with the attached service account of the resource.
Resource Link: How Application Default Credentials works - Google Cloud

Question: Which of the following is a characteristic of the Compute Engine default service account?

E. It is automatically created when the Compute Engine API is enabled.
F. It is attached by default to all VMs created using third-party tools.
G. It is automatically granted the IAM basic Owner role.
H. It is automatically granted the IAM basic Viewer role.

Correct Answer: A. It is automatically created when the Compute Engine API is enabled.

Explanation: The Compute Engine default service account has an autogenerated name and email address and is added to your project when you enable the Compute Engine API. Resource Link: Service accounts | Compute Engine Documentation - Google Cloud

Question: What is the primary purpose of a service account in Google Cloud?

E. To represent human users for browser-based sign-in.
F. To represent non-human users for accessing resources or performing actions without end-user involvement.
G. To manage Google Cloud resources using static API keys.
H. To provide SSL and mTLS authentication for Google Cloud services.

Correct Answer: B. To represent non-human users for accessing resources or performing actions without end-user involvement.

Explanation: Service accounts represent non-human users and are intended for scenarios where a workload, such as a custom application, needs to access resources or perform actions without end-user involvement.
Resource Link: Best practices for using service accounts - Google Cloud

Question: When using service accounts with VMs in Compute Engine, which of the following is true?

E. A single VM can have multiple service accounts attached to it.
F. If you attach the same service account to multiple VMs, changes to the service account do not affect any VMs.
G. Service accounts are associated with a particular employee.
H. You can attach the same service account to multiple VMs, but changes to the service account affect all VMs using it.

Correct Answer: D. You can attach the same service account to multiple VMs, but changes to the service account affect all VMs using it.

Explanation: If you attach the same service account to multiple VMs, any subsequent changes you make to the service account, including IAM roles, affect all VMs that use the service account.
Resource Link: [Service accounts | Compute Engine Documentation - Google Cloud]()

Question: Which of the following is the preferred method for finding credentials in a production environment on Google Cloud?

E. Using the GOOGLE_APPLICATION_CREDENTIALS environment variable.
F. Hardcoding the credentials in the application.
G. Using user credentials set up by the Google Cloud CLI.
H. Using the credentials from the attached service account via the metadata server.

Correct Answer: D. Using the credentials from the attached service account via the metadata server.

Explanation: In a production environment on Google Cloud, the preferred method for finding credentials is to use the credentials from the attached service account, which can be obtained via the metadata server.
Resource Link: [How Application Default Credentials works - Google Cloud]()