DUMPSBOSS.

Salesforce Certified Identity and Access Management Architect

Salesforce Identity-and-Access-Management-Architect

Version Demo

Total Demo Questions: 47

Total Premium Questions: 476

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, Identity and Access Management (IAM) Concepts	64
Topic 2, Identity and Access Management (IAM) Architecture	70
Topic 3, Identity and Access Management (IAM) Implementation	315
Topic 4, Identity and Access Management (IAM) Operations	27
Total	476

QUESTION NO: 1

Universal Containers (UC) uses Salesforce to allow customers to keep track of the order status. The customers can log in to Salesforce using external authentication providers, such as Facebook and Google. UC is also leveraging the App Launcher to let customers access an of platform application for generating shipping labels. The label generator application uses OAuth to provide users access. What license type should an Architect recommend for the customers?

A. Customer Community license

B. Identity license

C. Customer Community Plus license

D. External Identity license

ANSWER: D

Explanation:

External Identity license is the appropriate choice because UC's users are external customers whose primary requirements are identity-oriented: authentication through social identity providers such as Facebook and Google, access to Salesforce-hosted customer experiences for order-status visibility, and access to an external shipping-label application through the App Launcher and OAuth. External Identity is designed for business-to-consumer and other external-user identity scenarios, allowing organizations to manage customer identities in Salesforce while providing authentication, social sign-on, connected-app access, and single sign-on capabilities.

This licensing approach supports a unified customer identity without assigning the broader Salesforce application access intended for full customer-community use cases. Salesforce can act as the identity provider and authorization layer for the label-generation connected app, while OAuth safely grants the application the authorized access appropriate to each authenticated customer. The App Launcher provides a convenient entry point to connected applications after the customer signs in. Salesforce documents External Identity as an external-user license for identity services and connected-app access; see External Identity Licenses and Salesforce Connected Apps.

QUESTION NO: 2

Universal Containers is budding a web application that will connect with the Salesforce API using JWT OAuth Flow.

Which two settings need to be configured in the connect app to support this requirement?

Choose 2 answers

A. The Use Digital Signature option in the connected app.

B. The "web" OAuth scope in the connected app,

C. The "api" OAuth scope in the connected app.

D. The "edair_api" OAuth scope m the connected app.

ANSWER: A C

Explanation:

The Use Digital Signature option in the connected app is required because the JWT bearer OAuth flow authenticates the client application with an X.509 certificate. Salesforce uses the certificate uploaded to the connected app to validate the signature on the JWT assertion. The application signs the assertion with the corresponding private key, proving that it possesses the registered credential without sending a client secret or requiring an interactive user login. The certificate must therefore be associated with the connected app through its digital-signature configuration.

The "api" OAuth scope in the connected app is also required because it authorizes the access token issued through the JWT flow for Salesforce API access. The application presents the signed JWT assertion to Salesforce's token endpoint, receives an access token for the authorized user, and uses that token to invoke REST, SOAP, Bulk, or other supported Salesforce APIs within the permissions granted to that user. Salesforce documents that JWT bearer flow relies on a connected app

certificate and that the connected app's OAuth scopes determine the token's permitted access. See Salesforce JWT Bearer OAuth Flow documentation and Salesforce connected app OAuth configuration documentation.

QUESTION NO: 3

Universal Container's (UC) is using Salesforce Experience Cloud site for its containerwholesale business. The identity architect wants to an authentication provider for the new site.

Which two options should be utilized in creating an authentication provider?

Choose 2 answers

A. A custom registration handier can be set.

B. A custom error URL can be set.

C. The default login user can be set.

D. The default authentication provider certificate can be set.

ANSWER: A B

Explanation:

A custom registration handier can be set because an Authentication Provider can invoke an Apex registration handler after the external identity provider has authenticated a user. The handler implements Salesforce's `Auth.RegistrationHandler` interface and controls how Salesforce creates a new user or updates an existing user using the identity attributes returned by the provider. This is especially useful for an Experience Cloud site, where the organization may need to map external claims to Contact, Account, profile, or community-user attributes and apply its provisioning rules consistently.

A custom error URL can be set because Authentication Provider configuration supports an error URL for failures that occur in the authentication flow. Supplying a site-appropriate error destination lets UC present a branded, useful page to external users rather than leaving them at an unhelpful authentication failure endpoint. The page can provide next steps such as trying again, using another supported sign-in method, or contacting support. Together, the registration handler and error URL address lifecycle handling and user-facing failure handling for federated authentication. Salesforce documents the registration-handler contract in the Auth.RegistrationHandler interface reference and describes Authentication Provider setup in Salesforce Help.

QUESTION NO: 4

Northern Trail Outfitters (NTO) utilizes a third-party cloud solution for an employee portal. NTO also owns Salesforce Service Cloud and would like employees to be able to login to Salesforce with their third-party portal credentials for a seamless expenence. The third-party employee portal only supports OAuth.

What should an identity architect recommend to enable single sign-on (SSO) between the portal and Salesforce?

A. Configure SSO to use the third party portal as an identity provider.

B. Create a custom external authentication provider.

C. Add the third-party portal as a connected app.

D. Configure Salesforce for Delegated Authentication.

ANSWER: B

Explanation:

Create a custom external authentication provider. Salesforce can use an authentication provider to redirect a user to an external identity system, receive the OAuth authorization response at a callback URL, and establish the corresponding

authenticated Salesforce user session. This is the appropriate integration pattern when the external employee portal is the credential authority and supports OAuth rather than a Salesforce SAML single sign-on configuration.

A custom authentication provider is appropriate when the portal's OAuth implementation requires integration logic that is not covered by Salesforce's predefined provider types. The implementation can perform the OAuth authorization-code exchange, retrieve the identity attributes needed to identify the employee, and map that external identity to a Salesforce user. This lets employees authenticate with their existing portal credentials while Salesforce relies on the result of that OAuth authentication to grant access to Service Cloud.

Salesforce documents custom authentication-provider implementations through the `Auth.AuthProviderPluginClass` Apex interface, including the methods used to initiate authorization, process the callback, and obtain user information. Review the Salesforce Apex reference for Auth.AuthProviderPluginClass and Salesforce's Authentication Providers documentation when designing the provider, callback URL, user mapping, and required OAuth scopes.

QUESTION NO: 5

A financial services company uses Salesforce and has a compliance requirement to track information about devices from whichusers log in. Also, a Salesforce Security Administrator needs to have the ability to revoke the device from which users log in.

What should be used to fulfill this requirement?

A. Use multi-factor authentication (MFA) to meet the compliance requirement totrack device information.

B. Use the Activations feature to meet the compliance requirement to track device information.

C. Use the Login History object to track information about devices from which users log in.

D. Use Login Flows to capture device fromwhich users log in and store device and user information in a custom object.

ANSWER: B

Explanation:

Use the Activations feature to meet the compliance requirement to track device information. Salesforce activations provide an administrative record of devices and browsers that users have activated or verified for access. This gives Security Administrators visibility into user-device associations, including relevant activation details, which supports the organization's compliance need to monitor the endpoints used to access Salesforce.

Activations also provide the required administrative control: an administrator can revoke an activation when a device is lost, compromised, no longer approved, or otherwise should no longer be trusted. Revocation removes that device's activation, so the user must complete the applicable verification or activation process again before using it to access Salesforce. This makes activations appropriate when the requirement combines endpoint tracking with the ability to withdraw a specific device's trusted access, rather than merely reporting historical login events.

Salesforce documents device activation management as part of its identity-verification and security controls. See Salesforce Help: Multi-Factor Authentication and Salesforce Help: Device Activation.

QUESTION NO: 6

A public sector agency is setting up an identity solution for its citizens using a Community built on Experience Cloud and requires the new user registration functionality to capture first name, last name, and phone number. The phone number will be used for passwordless login.

Which feature should an identity architect recommend to meet the requirements?

A. Integrate with social websites (Facebook, LinkedIn, Twitter)

B. Use Login Discovery

C. Create a custom Lightning Web Component

D. Use an external Identity Provider

ANSWER: B

Explanation:

Use Login Discovery is correct because Login Discovery supports an identifier-first authentication experience in which an Experience Cloud user supplies an identifier, such as a mobile phone number, before Salesforce determines the appropriate authentication flow. This is well suited to a citizen-facing portal where the registered phone number is the identity attribute used to initiate passwordless access. The registration process can collect the required first name, last name, and mobile number, while the login experience uses the mobile number to locate the user and continue into the configured passwordless verification process.

Salesforce provides the `Auth.LoginDiscoveryHandler` interface to implement Login Discovery behavior. A handler receives the identifier entered by the user and can return the appropriate login method, including a passwordless flow. This separation of user identification from authentication is valuable for public-sector citizen experiences because it enables a streamlined sign-in journey without requiring citizens to remember a password. The phone number should be stored and managed as a verified, normalized mobile identifier so that it can reliably be used for passwordless verification and account recovery processes. Salesforce's Login Discovery interface is documented at Auth.LoginDiscoveryHandler. Additional passwordless authentication implementation guidance is available in the Headless User Discovery Handler documentation.

QUESTION NO: 7

A large consumer company is planning to create a community and will requ.re login through the customers social identity. The following requirements must be met:

1. The customer should be able to login with any of their social identities, however salesforce should only have one user per customer.

2. Once the customer has been identified with a social identity, they should not be required to authonze Salesforce.

3. The customers personal details from the social sign on need to be captured when the customer logs into Salesforce using their social Identity.

3. If the customer modifies their personal details in the social site, the changes should be updated in Salesforce .

Which two options allow the Identity Architect to fulfill the requirements?

Choose 2 answers

A. Use Login Flows to call an authentication registration handler to provision the user before logging the user into the community.

B. Use authentication providers for social sign-on and use the custom registration handler to insert or update personal details.

C. Redirect the user to a custom page that allows the user to select an existing social identity for login.

D. Use the custom registration handler to link social identities to Salesforce identities.

ANSWER: B D

Explanation:

Using authentication providers for social sign-on and a custom registration handler enables Salesforce Experience Cloud users to authenticate through supported social identity providers without maintaining a separate Salesforce username-and-password sign-in process. The authentication provider obtains the identity data supplied by the social provider, and the registration handler receives that data during sign-in. The handler can locate the existing Salesforce user for the customer and either create a user when no match exists or update the existing user's mapped personal attributes. Consequently, profile details such as name or email can be refreshed whenever the customer signs in after changing them at the social provider.

The custom registration handler is also the appropriate extensibility point for linking social identities to Salesforce identities. Its logic can use stable external identity information and a customer-matching strategy to associate additional social-provider identities with the same Salesforce user, rather than creating a separate community user for each identity. This provides a single Salesforce user per customer while preserving social sign-on. Salesforce documents the registration handler interface and its create-user and update-user lifecycle methods in the Auth.RegistrationHandler Apex reference. Authentication-provider configuration and social sign-on behavior are described in Salesforce Authentication Providers documentation.

QUESTION NO: 8

Universal Containers (UC) is building an integration between Salesforce and a legacy web application using the canvas framework. The security for UC has determined that asigned request from Salesforce is not an adequate authentication solution for the Third-Party app. Which two options should the Architect consider for authenticating the third-party app using the canvas framework? Choose 2 Answers

A. Utilize the SAML Single Sign-on flow to allow the third-party to authenticate itself against UC's IdP.

B. Utilize Authorization Providers to allow the third-party application to authenticate itself against Salesforce as the Idp.

C. Utilize Canvas OAuth flow to allow the third-party application to authenticate itself against Salesforce as the Idp.

D. Create a registration handler Apex class to allow the third-party application to authenticate itself against Salesforce as the Idp.

ANSWER: A C

Explanation:

"Utilize the SAML Single Sign-on flow to allow the third-party to authenticate itself against UC's IdP." and "Utilize Canvas OAuth flow to allow the third-party application to authenticate itself against Salesforce as the Idp." are appropriate alternatives when a Canvas signed request alone does not meet the required authentication assurance. Salesforce Canvas supports multiple authentication approaches, including OAuth and SAML. A Canvas OAuth flow allows the canvas application to obtain an access token through Salesforce's OAuth authorization process. Depending on the application architecture, this can use an interactive authorization flow and lets Salesforce authenticate the user and authorize the application before an access token is issued. The token can then be used by the third-party application to make authenticated calls within its approved scope.

SAML single sign-on is also a suitable approach where UC's identity provider is the trusted authority for user authentication. In that model, the third-party application can rely on a SAML assertion issued by the identity provider, providing federated authentication rather than relying solely on the Canvas signed request payload. This is especially useful when UC needs centralized enterprise authentication policies, such as multifactor authentication, conditional access, or a common session across applications. Salesforce documents Canvas authentication methods and its support for OAuth-based authorization at Salesforce Canvas App Authentication. Salesforce's SAML federation overview is available at Single Sign-On Overview.

QUESTION NO: 9

Universal Containers (UC) has an e-commerce website where customers can buy products, make payments, and manage their accounts. UC decides tobuild a Customer Community on Salesforce and wants to allow the customers to access the community from their accounts without logging in again. UC decides to implement an SP-initiated SSO using a SAML-compliant Idp. In this scenario where Salesforce is theService Provider, which two activities must be performed in Salesforce to make SP-initiated SSO work? Choose 2 answers

A. Configure SAML SSO settings.

B. Create a Connected App.

C. Configure Delegated Authentication.

D. Set up My Domain.

ANSWER: A D

Explanation:

For SP-initiated SAML single sign-on, Salesforce must be configured as the service provider and must know how to communicate with the external identity provider. **Configure SAML SSO settings.** is required because the SAML Single Sign-On Settings record establishes the trust relationship with the identity provider. It includes the identity provider's issuer/entity ID, SSO login endpoint, certificate used to validate SAML assertions, and the user-identification mapping Salesforce uses to locate the customer's user record after a successful assertion.

Set up My Domain. is also required because My Domain provides the branded, org-specific Salesforce URL used for authentication and Experience Cloud access. It enables the organization to use SAML authentication and gives the customer community a stable domain and login-routing context. When a customer initiates access from the community or an associated Salesforce URL, Salesforce can generate the SAML authentication request, redirect the customer to the configured identity provider, and process the returned signed assertion before granting access to the community.

Salesforce documentation on SAML single sign-on and My Domain describes these foundational configuration requirements.

QUESTION NO: 10

Universal Containers (UC) needs to call an external service that requires mutual TLS authentication. Which two actions should an Architect recommend to configure Salesforce as the TLS client? Choose 2 answers

A. Create or import a client certificate that includes its private key.

B. Configure the outbound callout to use the client certificate and configure the external service to trust its public certificate.

C. Upload the external service private key to Salesforce.

D. Disable TLS server certificate validation for the external endpoint.

ANSWER: A B

Explanation:

UC must create or import a **certificate that includes a private key** into Salesforce Certificate and Key Management. Salesforce uses the private key to prove possession of the client certificate during the TLS handshake. A certificate without an associated private key cannot be used by Salesforce as a client-authentication certificate.

UC must configure the outbound callout authentication to **use the client certificate**, and the external service must be configured to trust the public certificate presented by Salesforce. A named credential can centralize the endpoint and authentication configuration for the callout. Mutual TLS supplements normal HTTPS server validation by allowing the external service to authenticate Salesforce as the calling client. See Certificate and Key Management and Named Credentials.

QUESTION NO: 11

Northern Trail Outfitters would like to use a portal built on Salesforce Experience Cloud for customer self-service. Guests of the portal be able toself-register, but be unable to automatically be assigned to a contact record until verified. External Identity licenses have been purchased for the project.

After registered guests complete an onboarding process, a flow will create the appropriate accountand contact records for the user.

Which three steps should an identity architect follow to implement the outlined requirements?

Choose 3 answers

A. Enable "Allow customers and partners to self-register".

B. Select the "Configurable Self-Reg Page" optionunder Login & Registration.

C. Set jp an external login page and call Salesforce APIs for user creation.

D. Customize the self-registration Apex handler to temporarily associate the user to a shared single contact record.

E. Customize me self-registrationApex handler to create only the user record.

ANSWER: A B E

Explanation:

Enable "Allow customers and partners to self-register" to make self-registration available from the Experience Cloud site's login and registration experience. This is the foundational site-level setting that permits unauthenticated visitors to begin the registration process and have Salesforce create an external user identity.

Select the "Configurable Self-Reg Page" optionunder Login & Registration to use Salesforce's configurable registration-page framework. This allows the implementation to collect the registration and onboarding information required for the organization's process without requiring an externally hosted login or registration implementation.

Customize me self-registrationApex handler to create only the user record so that registration creates an External Identity user without automatically establishing the eventual business Account and Contact relationship. The handler is the appropriate extension point for controlling self-registration provisioning behavior. After the guest completes verification and onboarding, the planned Flow can create the Account and Contact records and associate them according to the organization's approved process. This design cleanly separates initial identity creation from later customer-record provisioning, which is important when a registrant must not be treated as a verified contact at sign-up. Salesforce documents Experience Cloud self-registration configuration at Enable Self-Registration and the Apex registration extension interface at Site RegistrationHandler Interface.

QUESTION NO: 12

Universal Containers is creating a web application that will be secured by Salesforce Identity using the OAuth 2.0 Web Server Flow uses the OAuth 2.0 authorization code grant type).

Which three OAuth concepts apply to this flow?

Choose 3 answers

A. Verification URL

B. Client Secret

C. Access Token

D. Scopes

ANSWER: B C D

Explanation:

The OAuth 2.0 Web Server Flow is the authorization-code grant intended for a confidential web application that can securely retain its credentials on the server. The application is represented by a connected app and uses its client credentials, including the **Client Secret**, when it exchanges the authorization code at Salesforce's token endpoint. This server-to-server exchange proves the identity of the client and returns an **Access Token**. The application then presents that access token when calling Salesforce APIs or other protected Salesforce resources on behalf of the authorized user. **Scopes** are also integral to this flow: the connected app requests scopes that define the permissions and types of resources the issued token can access, while the authorization process applies the user's approval and the organization's connected-app policies. Together, the client secret, access token, and scopes provide client authentication, delegated authorization, and authenticated API access for a server-based application. Salesforce documents the authorization-code sequence, including the code-to-token exchange, in its OAuth 2.0 Web Server Flow documentation. The underlying authorization-code grant and token behavior are defined in RFC 6749, section 4.1.

QUESTION NO: 13

ON NO: 12

A group of users try to access one of Universal Containers' Connected Apps and receive the following error message: " Failed: Not approved for access." What is the most likely cause of this issue?

A. The Connected App settings "All users mayself-authorize" is enabled.

B. The Salesforce Administrators have revoked the OAuth authorization.

C. The Users do not have the correct permission set assigned to them.

D. The User of High Assurance sessions are required for the Connected App.

ANSWER: C

Explanation:

The correct answer is **"The Users do not have the correct permission set assigned to them."** A Connected App can be configured so that only administrator-approved users may access it. In that model, Salesforce uses the Connected App's permitted-users policy together with profile and permission-set assignments to determine which users are pre-authorized. A user who has not been included through an assigned profile or permission set is blocked during authorization and can receive the "Failed: Not approved for access" message.

For a group of affected users, the architect or administrator should review the Connected App's OAuth policy and confirm that the appropriate permission set is assigned to every intended user. The permission set must grant access to that specific Connected App, and any user-level prerequisites, such as an active user record and applicable login restrictions, must also be met. Using permission sets for Connected App access supports least privilege and makes access easier to administer consistently as users join, change roles, or leave the organization.

Salesforce documents Connected App access policies and assignment through profiles or permission sets in Manage Access to a Connected App and describes OAuth Connected App policies in the Salesforce Developer documentation.

QUESTION NO: 14

Northern Trail Outfitters wants to implement a partner community. Active community users will need to review and accept the community rules, and update key contact information for each community member before their annual partner event.

Which approach will meet this requirement?

A. Create tasks for users who need to update their data or accept the new community rules.

B. Create a custom landing page and email campaign asking all community members to login and verify their data.

C. Create a login flow that conditionally prompts users who have not accepted the new community rules and who have missing or outdated information.

D. Add a banner to the community Home page asking users to update their profile and accept the new community rules.

ANSWER: C

Explanation:

Create a login flow that conditionally prompts users who have not accepted the new community rules and who have missing or outdated information. A login flow runs after a user authenticates and before they access Salesforce or the Experience Cloud site, making it an effective enforcement point rather than a passive reminder. Northern Trail Outfitters can associate the flow with the profile used by its partner community users and use flow decisions to evaluate fields that record acceptance of the community rules and the completeness or currency of required contact data. When action is needed, the flow can display screens requiring the user to acknowledge the rules and provide updated information, then save those values before allowing the user to continue to the community. Users whose acceptance and contact information are already current can bypass the update screens through the conditional path. This meets the requirement for active users to complete these steps before participating in the annual event while avoiding unnecessary prompts for users with valid information. Salesforce documents that login flows can direct users through a flow immediately after login, and screen flows can collect input and update records. See Salesforce Help: Login Flows and Trailhead: Build Screen Flows.

QUESTION NO: 15

universal container plans to develop a custom mobile app for the sales team that will use salesforce for authentication and access management. The mobile app access needs to be restricted to only the sales team. What would be the recommended solution to grant mobile app access to sales users?

A. Use a custom attribute on the user object to control access to the mobile app

B. Use connected apps Oauth policies to restrict mobile app access to authorized users.

C. Use the permission set license to assign the mobile app permission to sales users

D. Add a new identity provider to authenticate and authorize mobile users.

ANSWER: B

Explanation:

Use connected apps Oauth policies to restrict mobile app access to authorized users. A custom mobile application that authenticates through Salesforce should be registered as a connected app, which defines the OAuth integration and the access scopes the application can request. To limit use of that app to the sales team, an administrator can configure the connected app's permitted-user policy as "Admin approved users are pre-authorized" and then grant access through a profile or permission set assigned only to sales users. This creates a centrally managed authorization boundary: users must both authenticate successfully and be explicitly pre-authorized to use the connected app. The administrator can also apply appropriate OAuth scopes, session policies, IP restrictions, and token controls to meet the organization's mobile-access security requirements. This approach uses Salesforce's intended mechanism for governing which internal users can authorize and access a specific OAuth client, while retaining normal Salesforce identity controls such as login policies and multifactor authentication. Salesforce documents connected-app OAuth configuration and permitted-user policies at Manage OAuth Access for Connected Apps and provides guidance for assigning connected-app access through profiles and permission sets at Configure Profiles and Permission Sets for Connected Apps.

QUESTION NO: 16

Northern Trail Outfitters manages application functional permissions centrally as ActiveDirectory groups. The CRM_Superllser and CRM_Reportmg_SuperUser groups should respectively give the user the SuperUser and Reportmg_SuperUser permission set in Salesforce. Salesforce is the service provider to a Security Assertion Markup Language (SAML) identity provider.

Mow should an identity architect ensure the Active Directory groups are reflected correctly when a user accesses Salesforce?

A. Use the Apex Just-in-Time handler to query standard SAML attributes and set permission sets.

B. Use the Apex Just-in-Time handler to query custom SAML attributes and set permission sets.

C. Use a login flow to query custom SAML attributes and set permission sets.

D. Use a login flow to query standard SAML attributes and set permission sets.

ANSWER: B

Explanation:

Use the Apex Just-in-Time handler to query custom SAML attributes and set permission sets. The identity provider can include a custom attribute in its SAML assertion that represents the user's Active Directory group memberships. During SAML single sign-on, Salesforce passes SAML attributes to an Apex class that implements the `Auth.SamlJitHandler` interface. The handler can inspect the group attribute, determine whether it contains values such as `CRM_SuperUser` or `CRM_Reporting_SuperUser`, and create or remove the appropriate `PermissionSetAssignment` records for the Salesforce user.

This approach keeps Active Directory as the central source for functional authorization while applying Salesforce permission sets at sign-in. It also supports automatic updates when group membership changes: the next successful SAML login

evaluates the current assertion and synchronizes the user's assigned permissions. The identity provider must be configured to send the needed group information as a custom SAML attribute, and the Apex handler should be implemented to safely handle group values, user creation or updates, and permission-set assignment changes. Salesforce documents the handler contract and the SAML attributes available to it in the Auth.SamlJitHandler Apex reference and its SAML Just-in-Time provisioning documentation.

QUESTION NO: 17

Universal containers (UC) is building a mobile application that will make calls to the salesforce REST API. Additionally UC would like to provide the optimal experience for its mobile users. Which two OAuth scopes should UC configure in the connected App? Choose 2 answers

A. Refresh token

B. API

C. full

D. Web

ANSWER: A B

Explanation:

API is required because it grants the connected app permission to access Salesforce APIs, including the REST API used by the mobile application. In Salesforce connected apps, the API scope is specifically intended for clients that need to make authenticated programmatic calls to Salesforce resources. The user who authorizes the application must still have the relevant object, field, and record permissions, so OAuth scope enables API access without bypassing Salesforce's normal authorization controls.

Refresh token supports a better mobile experience by allowing the app to obtain new access tokens after an access token expires, without requiring the user to authenticate interactively every time. Salesforce presents this scope in connected-app configuration as "Perform requests on your behalf at any time (refresh_token, offline_access)." A mobile app can securely retain the refresh token and use it to maintain a session across launches or periods of inactivity, subject to the organization's connected-app and session policies. Together, these scopes provide the necessary REST API authorization and the ability to maintain a low-friction, durable sign-in experience for users.

See Salesforce's OAuth connected app documentation and the REST API authentication guide.

QUESTION NO: 18

A farming enterprise offers smart farming technology to its farmer customers, which includes a variety of sensors for livestock tracking, pest monitoring, climate monitoring etc. They plan to store all thedata in Salesforce. They would also like to ensure timely maintenance of the Installed sensors. They have engaged a salesforce Architect to propose an appropriate way to generate sensor Information In Salesforce.

Which OAuth flow should the architect recommend?

A. OAuth 2.0 Asset Token Flow

B. OAuth 2.0 Device Authentication Row

C. OAuth 2.0 JWT Bearer Token Flow

D. OAuth 2.0 SAML Bearer Assertion Flow

ANSWER: A

Explanation:

OAuth 2.0 Asset Token Flow is the appropriate choice for connected smart-farming sensors that need to submit telemetry or sensor information to Salesforce. This flow is designed for Internet of Things scenarios, where an individual physical asset, such as a livestock tracker, climate monitor, or pest sensor, needs its own identity and authorized API access. Salesforce can associate the device with an asset record and issue an asset token that the device uses when calling Salesforce APIs. This provides a practical model for ingesting sensor data while allowing access to be managed at the device level rather than requiring an interactive user to sign in on each sensor.

Using asset-specific authorization also supports the operational requirement to maintain installed sensors. The organization can track devices as Salesforce assets, govern their access independently, and manage or revoke access when a sensor is replaced, retired, or compromised. The flow is particularly suitable for unattended devices that cannot present an interactive login experience. Salesforce documents the asset token flow as an OAuth mechanism for connected devices and explains the related authorization architecture in its OAuth implementation guidance. See OAuth 2.0 Asset Token Flow for Connected Devices and Authorize Apps with OAuth.

QUESTION NO: 19

A security architect is rolling out a new multi-factor authentication (MFA) mandate, where all employees must go through a secure authentication process before accessing Salesforce. There are multiple Identity Providers (IdP) in place and the architect is considering how the "Authentication Method Reference" field (AMR) in the Login History can help.

Which two considerations should the architect keep in mind?

Choose 2 answers

A. AMR field shows the authentication methods used at IdP.

B. Both OIDC and Security Assertion Markup Language (SAML) are supported but AMR must be implemented at IdP.

C. High-assurance sessions must be configured under Session Security Level Policies.

D. Dependency on what is supported by OpenID Connect (OIDC) implementation at IdP.

ANSWER: A B

Explanation:

The AMR field is useful because it records authentication-method information that Salesforce receives from the external identity provider during a federated login. This gives the architect visibility in Login History into the authentication context used before the user reached Salesforce, such as whether the IdP performed an MFA-capable authentication flow. Reviewing these values across IdPs helps the organization validate and monitor adoption of its MFA mandate rather than relying solely on the fact that the user authenticated through single sign-on.

AMR information can be supplied for both OpenID Connect and SAML-based SSO integrations. However, Salesforce does not create this upstream authentication evidence on its own: each IdP must be configured to emit the applicable authentication-method or authentication-context information in its protocol response. The architect should therefore confirm consistent AMR support, values, and mapping behavior with every IdP used by employees. This is particularly important where different IdPs or policies apply different MFA methods. Salesforce's identity documentation describes its SSO support and the role of an external IdP in authenticating users; see Salesforce Security Implementation Guide and Salesforce Trailhead: Single Sign-On.

QUESTION NO: 20

Universal containers (UC) has a mobile application that it wants to deploy toall of its salesforce users, including customer Community users. UC would like to minimize the administration overhead, which two items should an architect recommend? Choose 2 answers

A. Enable the "Refresh Tokens is valid until revoked " setting in the Connected App.

B. Enable the "Enforce Ip restrictions" settings in the connected App.

C. Enable the "All users may self-authorize" setting in the Connected App.

D. Enable the "High Assurance session required" setting in the Connected App.

ANSWER: A C

Explanation:

Enabling the "Refresh Tokens is valid until revoked " setting in the Connected App supports a low-administration mobile experience because users can use an OAuth refresh token to obtain new access tokens after short-lived access tokens expire. The user does not need to repeatedly authenticate merely because an access token has expired. Keeping the refresh token valid until it is explicitly revoked is especially useful for a broadly deployed mobile application, provided UC has appropriate revocation procedures for lost devices, terminated users, and suspected compromise. Salesforce documents refresh-token lifetime as an OAuth policy configured for a connected app.

Enabling the "All users may self-authorize" setting in the Connected App also reduces operational work at deployment. It permits eligible users, including appropriately licensed and permitted external community users, to approve the application themselves during the OAuth authorization process instead of requiring an administrator to preauthorize individual users, profiles, or permission sets. UC should still ensure that the connected app's OAuth scopes are limited to what the mobile application actually requires and that its external-user access model is properly configured. Together, self-authorization and renewable access tokens minimize manual provisioning and recurring login support while retaining Salesforce's OAuth consent and token controls. See Salesforce's Connected App OAuth policies documentation and Mobile SDK OAuth overview.

QUESTION NO: 21

Universal containers (UC) has implemented a multi-org strategy and would like to centralize the management of their salesforce user profiles. What should the architect recommend to allow salesforce profiles to be managed from a central system of record?

A. Implement jit provisioning on the SAML IDP that will pass the profile id in each assertion.

B. Create an apex scheduled job in one org that will synchronize the other orgs profile.

C. Implement Delegated Authentication that will update the user profiles as necessary.

D. Implement an Oauthjwt flow to pass the profile credentials between systems.

ANSWER: A

Explanation:

Implement jit provisioning on the SAML IDP that will pass the profile id in each assertion. Salesforce Just-in-Time (JIT) provisioning enables an identity provider, acting as the central system of record, to create and update Salesforce user records when users authenticate through SAML single sign-on. The SAML assertion can include the `ProfileId` attribute, allowing the identity provider to centrally control the Salesforce profile assigned to each user. This supports a multi-org design because each Salesforce organization can trust the same identity provider while the identity provider determines the appropriate user attributes and profile assignment for that organization.

For existing users, JIT provisioning can update supported user fields at login, so profile assignment can be maintained through the centrally managed assertion rather than by manually administering profiles for individual users in every org. The profile IDs must be the appropriate IDs for the target Salesforce org, and the integration must be configured to use SAML JIT provisioning. Salesforce documents the SAML JIT workflow and the assertion attributes used for user creation and updates, including profile assignment, in its Just-in-Time Provisioning documentation and SSO JIT provisioning overview.

QUESTION NO: 22

Universal Containers (UC) has a Customer Community that uses Facebook for of authentication. UC would like to ensure that changes in the Facebook profile are 65. reflected on the appropriate Customer Community user. How can this requirement be met?

A. Use SAML Just-In-Time Provisioning between Facebook and Salesforce.

B. Use information in the Signed Request that is received from Facebook.

C. Develop a scheduled job that calls out to Facebook on a nightly basis.

D. Use the updateUser() method on the Registration Handler class.

ANSWER: D

Explanation:

Use the updateUser() method on the Registration Handler class. When Salesforce uses Facebook as an authentication provider, a registration handler can control how the external identity information is applied to the associated Salesforce user. The `updateUser()` method is invoked for an existing user during authentication, allowing the handler to retrieve the identity data supplied by Facebook and update appropriate User fields, such as name, email address, locale, or other mapped profile attributes. This provides a controlled, supported synchronization point each time the community user signs in, ensuring that profile changes received from the identity provider can be reflected in Salesforce without creating a duplicate user account. The handler can also contain the organization's field-mapping and data-validation logic, so UC can determine exactly which Facebook-sourced attributes are permitted to overwrite Salesforce values. Salesforce documents registration handlers as the mechanism for creating and updating users authenticated through an OAuth-based authentication provider, including social providers such as Facebook. See RegistrationHandler Interface and Authentication Providers.

QUESTION NO: 23

In an SP-Initiated SAML SSO setup where the user tries to access a resource on the Service Provider, What HTTP param should be used when submitting a SAML Request to the Idp to ensure the user is returned to the intended resourse after authentication?

A. RedirectURL

B. RelayState

C. DisplayState

D. StartURL

ANSWER: B

Explanation:

RelayState is the SAML parameter used to preserve application state across an SP-initiated single sign-on flow. When a user requests a protected resource at the service provider, the service provider creates a SAML authentication request and redirects or posts it to the identity provider. It can include a RelayState value that identifies the original target URL or an opaque value mapped by the service provider to that target. After the identity provider successfully authenticates the user, it returns the SAML response together with the same RelayState value to the service provider. The service provider then uses that value to return the authenticated user to the intended resource rather than merely sending them to a generic landing page. The SAML specification defines RelayState as state information maintained between the request and response; implementations should protect it appropriately, such as by validating it against an allowlist or using a signed, server-side mapped value, to prevent unsafe redirects. Salesforce's SAML single sign-on documentation also describes RelayState as the parameter used to direct users to a specific destination after login. See the OASIS SAML Bindings specification and Salesforce SAML Single Sign-On documentation.

QUESTION NO: 24

Northern Trail Outfitters (NTO) is planning to roll out a partner portal for its distributors using Experience Cloud. NTO would like to use an external identity provider (idP) and for partners to register for access to the portal. Each partner should be allowed to register only once to avoid duplicate accounts with Salesforce.

What should a identity architect recommend to create partners?

A. On successful creation of Partners using Self Registration page in Experience Cloud, create identity in Ping.

B. Create a custom page m Experience Cloud to self register partner with Experience Cloud and Ping identity store.

C. Create a custom web page in the Portal and create users in the IdP and Experience Cloud using published APIs.

D. Allow partners to register through the IdP and create partner users in Salesforce through an API.

ANSWER: D

Explanation:

Allow partners to register through the IdP and create partner users in Salesforce through an API is the appropriate approach because the external identity provider becomes the authoritative system for partner identity creation and uniqueness checks. The IdP can enforce a single registration by using a stable unique identifier, such as an immutable subject identifier or an approved email-address policy, before any Salesforce user is provisioned. After registration succeeds, an integration can use Salesforce APIs to locate or create the appropriate Account and Contact, then create the associated Experience Cloud partner User. The integration should store and match the IdP's immutable identifier in Salesforce so that subsequent sign-ins and provisioning events resolve to the existing user rather than creating another account. This pattern also supports lifecycle management: the IdP can control authentication and initiate downstream updates, deactivation, or access changes through the integration. Salesforce supports programmatic User creation through its APIs, subject to the required account, contact, license, profile, and partner-user relationships. Review Salesforce's User object API reference and its partner user documentation when designing the provisioning process.

QUESTION NO: 25

Northern Trail Outfitters (NTO) uses Salesforce for Sales Opportunity Management. Okta was recently brought in to Just-in-Time (JIT) provision and authenticate NTO users to applications. Salesforce users also use Okta to authorize a Forecasting web application to access Salesforce records on their behalf.

Which two roles are being performed by Salesforce?

Choose 2 answers

A. SAML Identity Provider

B. OAuth Client

C. OAuth Resource Server

D. SAML Service Provider

ANSWER: C D

Explanation:

SAML Service Provider is correct because Salesforce is the application that users are signing in to after Okta authenticates them. In this arrangement, Okta acts as the SAML identity provider: it authenticates the user and sends a signed SAML assertion to Salesforce. Salesforce consumes and validates that assertion, can use its attributes for just-in-time user provisioning, and establishes the user's Salesforce session. Salesforce therefore performs the service-provider role for this federation flow.

OAuth Resource Server is also correct because Salesforce owns and protects the Opportunity and other Salesforce records that the Forecasting web application needs to access. After a user authorizes access through Okta, the application presents an access token when calling Salesforce APIs. Salesforce validates the applicable authorization and token context and then serves the protected resources according to the user's permissions and the granted scopes. That protected-API role is the OAuth resource server role; the Forecasting web application is the OAuth client requesting access on the user's behalf.

Salesforce supports SAML single sign-on and just-in-time provisioning as a service provider, as documented in Salesforce SAML Single Sign-On documentation. Salesforce's role in exposing protected data through OAuth-connected applications is described in Salesforce OAuth flow documentation.

QUESTION NO: 26

Universal Containers (UC) currently uses Salesforce Sales Cloud and an external billing application. Both Salesforce andthe billing application are accessed several times a day to manage customers. UC would like to configure single sign-on and leverageSalesforce as the identity provider. Additionally, UC would like the billing application to be accessible from Salesforce.A redirect is acceptable.

Which two Salesforce tools should an identity architect recommend to satisfy the requirements?

Choose 2 answers

A. salesforce Canvas

B. Identity Connect

C. Connected Apps

D. App Launcher

ANSWER: C D

Explanation:

Connected Apps and **App Launcher** satisfy the stated requirements. A connected app is Salesforce's integration definition for an external application and can support federated access patterns, including SAML-based single sign-on. UC can configure the billing application as a SAML service provider, configure Salesforce as the identity provider, and establish the application's SAML settings, such as its ACS URL and entity ID. After users authenticate to Salesforce, Salesforce can issue a SAML assertion to the billing application, allowing users to reach it without separately authenticating.

The App Launcher provides the Salesforce-side launch experience for users. Administrators can make the connected application available to the appropriate users so it appears as an app tile in the launcher. Selecting the tile starts the configured application launch and federation flow. Because the requirement explicitly permits a redirect, this approach provides an appropriate user experience without requiring the billing user interface to be embedded within Salesforce. Access can also be governed through connected-app policies and user assignments, ensuring that only authorized Salesforce users can launch the billing application. See Salesforce documentation on connected apps and the App Launcher.

QUESTION NO: 27

Universal containers (UC) have a custom, internal-only, mobile billing application for users who are commonly out of the office. The app is configured as a connected App in salesforce. Due to the nature of this app, UC would like to take the appropriate measures to properlysecure access to the app. Which two are recommendations to make the UC? Choose 2 answers

A. Disallow the use of single Sign-on for any users of the mobile app.

B. Require high assurance sessions in order to use the connected App

C. Use Google Authenticator as an additional authentication factor.

D. Set login IP ranges to the internal network for all of the app users profiles.

ANSWER: B C

Explanation:

Requiring high-assurance sessions for the connected app is appropriate because it ensures users meet a stronger authentication requirement before accessing a sensitive billing application. In Salesforce, a high-assurance session is established through stronger authentication, such as multi-factor authentication or an acceptable federated authentication flow that meets the organization's configured assurance requirements. The connected app can enforce this requirement through its session policy, providing protection even when users access the app remotely from unmanaged or changing networks.

Using Google Authenticator as an additional authentication factor is also appropriate. Salesforce supports time-based one-time password authenticator applications as a verification method for multi-factor authentication. A user must provide both their normal login credential and a time-sensitive verification code generated on their registered device. This materially reduces the risk that a compromised password alone could be used to access the connected mobile application. These controls provide strong, user-centered protection while still supporting employees who routinely work outside the corporate office. See Salesforce's guidance on session security levels and getting started with multi-factor authentication.

QUESTION NO: 28

A client is planning to rollout multi-factor authentication (MFA) to its internal employees and wants to understand which authentication and verification methods meet the Salesforce criteria for secure authentication.

Which three functions meet the Salesforce criteria for secure MFA?

Choose 3 answers

A. Username and password = security key

B. Lightning Login

C. Username and password = SMS passwords

D. Third-party single sign-on with Mobile Authenticator app

E. Username & password = Email Verification Code

ANSWER: A B D

Explanation:

Salesforce MFA requires users to verify their identity with more than one factor. "Username and password = security key" meets this standard because a registered security key provides a possession factor and uses phishing-resistant authentication based on the FIDO/WebAuthn standard. The key must be physically available to the user, making it a strong method for protecting Salesforce access.

"Lightning Login" is a Salesforce passwordless login experience that verifies a login request through the Salesforce Authenticator app on a registered mobile device. The user approves the request on a trusted device, providing secure device-based verification for the login flow.

"Third-party single sign-on with Mobile Authenticator app" can meet the requirement when the organization's identity provider enforces MFA before issuing the SSO assertion to Salesforce. In this architecture, Salesforce delegates authentication to the identity provider, while the mobile authenticator app supplies the additional verification factor. Architects should ensure that the identity provider's MFA policy is consistently applied for every relevant Salesforce login path and that the SSO integration is properly configured. Salesforce documents supported MFA methods and SSO considerations in its Multi-Factor Authentication overview and Single Sign-On Trailhead module.

QUESTION NO: 29

Universal Containers allows employees to use a mobile device to access Salesforce for daily operations using a hybrid mobile app. This app uses Mobile software development kits (SDK), leverages refresh token to regenerate access token when required and is distributed as a private app.

The chief security officer is rolling out an org wide compliance policy to enforcere-verification of devices if an employee has not logged in from that device in the last week.

Which connected app setting should be leveraged to comply with this policy change?

A. Scope - Deny refresh_token scope for this connected app.

B. Refresh Token Policy - Expire the refresh token if it has not been used for 7 days.

C. Session Policy - Set timeout value of the connected app to 7 days.

D. Permitted User - Ask admins to maintain a list of users who are permitted based on last login date.

ANSWER: B

Explanation:

Refresh Token Policy - Expire the refresh token if it has not been used for 7 days. directly implements the stated device re-verification requirement. In the OAuth flow used by a Mobile SDK hybrid app, the refresh token lets the application obtain new short-lived access tokens without requiring the employee to authenticate again. Configuring the connected app's refresh-token policy to expire an unused token after seven days means that a device which has not used its refresh token during that period can no longer silently renew access. The next attempted access requires the employee to complete authentication again, which provides the required re-verification of that device.

This is a device-appropriate control because refresh tokens are issued and stored in the app's authorization context on the individual device. Active users who continue to access Salesforce from their device within the seven-day window retain the expected mobile experience, while inactive devices must re-establish authorization before they can resume access. Salesforce provides connected-app refresh-token policies specifically to control refresh-token lifetime and inactivity behavior, enabling an organization to balance persistent mobile access with compliance requirements. See Salesforce's Manage OAuth Access for Connected Apps documentation and the Mobile SDK OAuth Refresh Token Flow guide.

QUESTION NO: 30

Universal Containers (UC) wants to build a few applications that leverage the Salesforce REST API. UC has asked its Architect to describe how the API calls will be authenticated to a specific user. Which two mechanisms can the Architect provide? Choose 2 Answers

A. Authentication Token

B. Session ID

C. Refresh Token

D. Access Token

ANSWER: B D

Explanation:

Session ID and **Access Token** can authenticate Salesforce REST API requests in the context of a particular Salesforce user. A REST client supplies the credential as a Bearer token in the HTTP `Authorization` header, for example `Authorization: Bearer <token>`. Salesforce identifies the user associated with that credential and evaluates the request using that user's object permissions, field-level security, sharing access, assigned permission sets, session policies, and other applicable access controls. An OAuth access token is the standard credential returned after a successful OAuth authorization flow and is designed for use in API requests. A valid Salesforce session ID can likewise be used as the session credential for API access, including REST API calls, subject to the organization's session-security configuration and the user's permissions. This allows integrations to act on behalf of a specific authenticated user rather than as an anonymous caller. Salesforce documents the REST API authorization-header pattern and recognizes session IDs as access tokens for API authentication. See the Salesforce REST API authentication documentation and the Salesforce REST API developer guide.

QUESTION NO: 31

Northern Trail Outfitters (NTO) believes a specific user account may have been compromised. NTO inactivated the user account and needs to perform a forensic analysis and identify signals that could indicate a breach has occurred.

What should NTO's first step be in gathering signals that could indicate account compromise?

A. download the identity provider Event log and contact the details of activities performed by the user.

B. download the Login history and evaluate the details of topics performed by the user.

C. download the Setup Audit Trail and review all recent activities performed by the user.

D. Review the User record and evaluate the login and transaction history.

ANSWER: B

Explanation:

“download the Login history and evaluate the details of topics performed by the user.” is the appropriate first step because Login History is Salesforce’s built-in record of authentication activity. It gives investigators an immediate view of successful and failed login attempts associated with the account, including timestamps, login status, source IP address, browser or client information, platform, and login type. These details provide the primary signals needed to establish whether access originated from an unfamiliar network, geography, device, client, or authentication flow, and to identify a suspicious time window for deeper investigation.

For a potentially compromised identity, establishing the facts around authentication is the essential initial forensic task: determine when the account was accessed, whether attempts failed before succeeding, and what connection and client characteristics were involved. Salesforce makes Login History available in Setup and through the `LoginHistory` object, allowing the information to be reviewed or exported for correlation with network, identity-provider, and security-monitoring records. Salesforce documents the fields captured by Login History in its API reference and explains how administrators can monitor login activity in Setup. See Salesforce LoginHistory object reference and Salesforce Help: Monitor Login Activity.

QUESTION NO: 32

An identity architect has been asked to recommend a solution that allows administrators to configure personalized alert messages to users before they land on the Experience Cloud site (formerly known as Community) homepage.

What is recommended to fulfill this requirement with the least amount of customization?

A. Customize the registration handler Apex class to create a routing logic navigating to different home pages based on the user profile.

B. Use Login Flows to add a screen that shows personalized alerts.

C. Build a Lightning web component (LWC) for a homepage that shows custom alerts.

D. Create custom metadata that stores user alerts and use a LWC to display alerts.

ANSWER: B

Explanation:

Use Login Flows to add a screen that shows personalized alerts. A login flow runs after a user has successfully authenticated and before the user is directed to the intended Salesforce or Experience Cloud destination. This timing directly supports presenting an alert before the Experience Cloud site homepage loads. Administrators can build the flow declaratively with Flow Builder, use screen elements to display a message, and apply flow logic based on attributes such as the user’s profile, user record fields, or other available criteria. This provides a maintainable configuration that avoids the development and ongoing deployment effort associated with building and maintaining a custom Lightning web component or custom Apex-based routing behavior. Login flows can be assigned to profiles, which also enables targeted delivery to the relevant user populations. For personalized content, the flow can retrieve data or evaluate conditions and then show the appropriate screen before allowing the user to continue. Salesforce documents login flows as a way to customize the post-login experience, including presenting information and collecting or acting on user input before users proceed into the application. See Salesforce Help: Login Flows and Salesforce Help: Flows.

QUESTION NO: 33

Universal Containers (UC) wants to implement SAML SSO for their internal of Salesforce users using a third-party IdP. After some evaluation, UC decides NOT to 65« set up My Domain for their Salesforce org. How does that decision impact their SSO implementation?

A. IdP-initiated SSO will NOT work.

B. Neither SP- nor IdP-initiated SSO will work.

C. Either SP- or IdP-initiated SSO will work.

D. SP-initiated SSO will NOT work

ANSWER: B

Explanation:

Neither SP- nor IdP-initiated SSO will work. My Domain is a prerequisite for configuring and using SAML single sign-on in Salesforce. It provides the organization-specific domain infrastructure Salesforce uses to support the SAML authentication flow and to direct users to the appropriate authentication configuration. Therefore, before Universal Containers can enable a third-party identity provider for SAML SSO, it must create and deploy My Domain in the Salesforce org.

This requirement applies regardless of where the authentication flow begins. In an SP-initiated flow, Salesforce must be able to initiate the SAML request using the org's configured domain and SSO settings. In an IdP-initiated flow, Salesforce must likewise receive and process the SAML assertion in the context of the organization's configured SAML and domain setup. Choosing not to set up My Domain prevents the organization from implementing either SAML entry pattern for its internal users.

Salesforce identifies My Domain as required for SAML-based SSO and documents My Domain deployment as part of establishing identity capabilities. See Salesforce Help: My Domain Overview and Salesforce Developer Guide: SAML Single Sign-On.

QUESTION NO: 34

In a typical SSL setup involving a trusted party and trusting party, what consideration should an Architect take into account when using digital certificates?

A. Use of self-signed certificate leads to lower maintenance for trusted party because multiple self-signed certs need to be maintained.

B. Use of self-signed certificate leads to higher maintenance for trusted party because they have to act as the trusted CA

C. Use of self-signed certificate leads to lower maintenance for trusting party because there is no trusted CA cert to maintain.

D. Use of self-signed certificate leads to higher maintenance for trusting party because the cert needs to be added to their truststore.

ANSWER: D

Explanation:

Use of self-signed certificate leads to higher maintenance for trusting party because the cert needs to be added to their truststore. In a certificate-based SSL/TLS relationship, the trusting party must have a trust anchor that enables it to validate the certificate presented by the other party. A certificate issued by a publicly trusted or enterprise certificate authority can generally be validated through an existing trusted root and its certificate chain. By contrast, a self-signed certificate is its own trust anchor and is not inherently trusted by other systems. Each trusting application, server, integration endpoint, or device must explicitly import and trust that certificate (or an equivalent private root certificate). The trusting party must also manage certificate replacement before expiration, distribute the replacement to every relevant truststore, and remove or revoke trust in the prior certificate when appropriate. This operational distribution and lifecycle work is the key maintenance consideration for an Architect designing integrations that use self-signed certificates. Salesforce's certificate guidance describes using certificates to establish trust for integrations, while standard TLS guidance explains the role of trust anchors and certificate validation. See Salesforce Certificates and Key Management and RFC 5280: Internet X.509 Public Key Infrastructure.

QUESTION NO: 35

Universal containers (UC) has a mobile application that it wants to deploy to all of its salesforce users, including customer Community users. UC would like to minimize the administration overhead, which two items should an architect recommend? Choose 2 answers

A. Enable the "Refresh Tokens is valid until revoked " setting in the Connected App.

B. Enable the "Enforce Ip restrictions" settings in the connected App.

C. Enable the "All users may self-authorize" setting in the Connected App.

D. Enable the "High Assurance session required" setting in the Connected App.

ANSWER: A C

Explanation:

Enabling the "Refresh Tokens is valid until revoked " setting in the Connected App minimizes recurring user and support effort for a mobile application. After a user completes the initial OAuth authorization, the application can use its refresh token to obtain new access tokens without requiring the user to sign in again each time an access token expires. Keeping the refresh token valid until it is explicitly revoked supports a low-friction, persistent mobile experience for both internal Salesforce users and customer Community users, while still allowing administrators or users to revoke access when needed.

Enabling the "All users may self-authorize" setting in the Connected App also reduces administrative overhead at broad deployment scale. It allows eligible users to authorize the application themselves rather than requiring an administrator to pre-authorize users through profiles, permission sets, or assigned policies. This is particularly valuable when the intended audience includes a large and changing population of external community users. Salesforce documents refresh-token behavior and connected-app OAuth configuration in its Connected App OAuth documentation and describes authorization choices in the Connected App permitted users documentation.

QUESTION NO: 36

Universal Containers (UC) uses Global Shipping (GS) as one of their shipping vendors. Regional leads of GS need access to UC's Salesforce instance for reporting damage of goods using Cases. The regional leads also need access to dashboards to keep track of regional shipping KPIs. UC internally uses a third-party cloud analytics tool for capacity planning and UC decided to provide access to this tool to a subset of GS employees. In addition to regional leads, the GS capacity planning team would benefit from access to this tool. To access the analytics tool, UC IT has set up Salesforce as the Identity provider for Internal users and would like to follow the same approach for the GS users as well. What are the most appropriate license types for GS Tregional Leads and the GS Capacity Planners? Choose 2 Answers

A. Customer Community Plus license for GS Regional Leads and External Identity for GS Capacity Planners.

B. Customer Community Plus license for GS Regional Leads and Customer Community license for GS Capacity Planners.

C. Identity Licence for GS Regional Leads and External Identity license for GS capacity Planners.

D. Customer Community license for GS Regional Leads and Identity license for GS Capacity Planners.

ANSWER: A B

Explanation:

Customer Community Plus license for GS Regional Leads and External Identity for GS Capacity Planners is appropriate because the regional leads require external access to Cases as well as reporting and dashboard capabilities. Customer Community Plus is an Experience Cloud external-user license designed for more collaborative customer scenarios and includes the sharing and reporting capabilities needed to expose operational case information and KPI dashboards securely. External Identity is purpose-built for external people who primarily need identity services, such as authentication and single sign-on to an external application. It therefore fits the capacity planners' stated need to use Salesforce as the identity provider for the third-party analytics tool without granting unnecessary CRM access.

Customer Community Plus license for GS Regional Leads and Customer Community license for GS Capacity Planners also provides a functional external-user licensing model. The Customer Community license can provision GS capacity planners as authenticated external users, allowing them to sign in through Salesforce and participate in the configured identity-

provider flow for the analytics application. Meanwhile, Customer Community Plus continues to supply the regional leads with the Case, reporting, and dashboard access their role requires. UC can further restrict each user population through profiles, permission sets, sharing, and connected-app policies. See Salesforce's Experience Cloud license guidance and External Identity documentation.

QUESTION NO: 37

Universal containers (UC) has decided to use identity connect as it's identity provider. UC uses active directory(AD) and has a team that is very familiar and comfortable with managing ad groups. UC would like to use AD groups to help configure salesforce users. Which three actions can AD groups control through identity connect? Choose 3 answers

A. Public Group Assignment

B. Granting report folder access

C. Role Assignment

D. Custom permission assignment

E. Permission sets assignment

ANSWER: A C E

Explanation:

Identity Connect can use Active Directory group membership as a source for assigning Salesforce access and collaboration structures during user provisioning and synchronization. **Role Assignment** is supported because Identity Connect can map an Active Directory group to a Salesforce role, allowing administrators to maintain role membership through familiar directory-group administration. **Permission sets assignment** is also supported: a mapped group can assign a permission set to users who are members of that directory group, helping centralize entitlement management outside Salesforce. Finally, **Public Group Assignment** is supported through group mapping, so directory-group membership can place users into corresponding Salesforce public groups. This is useful for maintaining group-based sharing and membership consistently as users join or leave Active Directory groups. These mappings are configured in Identity Connect's user provisioning and group-mapping configuration, where administrators associate directory groups with Salesforce roles, permission sets, and public groups. Salesforce documents Identity Connect as a directory-integration solution that provisions and updates Salesforce users and can manage related assignments based on directory data. See Salesforce's Identity Connect documentation and the Identity Connect Developer Guide.

QUESTION NO: 38

After a recent audit, universal containers was advised to implement Two-factor Authentication for all of their critical systems, including salesforce. Which two actions should UC consider to meet this requirement? Choose 2 answers

A. Require users to provide their RSA token along with their credentials.

B. Require users to supply their email and phone number, which gets validated.

C. Require users to enter a second password after the first Authentication

D. Require users to use a biometric reader as well as their password

ANSWER: A D

Explanation:

Two-factor authentication requires evidence from two distinct factor categories, rather than two credentials of the same kind. Requiring users to provide their RSA token along with their credentials combines something the user has—the physical or generated one-time-code token—with something the user knows—their username/password credentials. This creates the required independent factors and can be implemented for Salesforce through an appropriate identity-provider or authentication integration where the organization uses RSA tokens.

Requiring users to use a biometric reader as well as their password likewise combines something the user is, such as a fingerprint or facial biometric verified by a device authenticator, with something the user knows. Salesforce MFA supports authentication methods that can use a device's built-in authenticator, which may rely on biometric verification, in addition to the user's primary login credentials. The organization should deploy the selected method consistently for the critical applications in scope and retain appropriate recovery procedures for users who lose access to their authenticator. Salesforce describes MFA and supported verification methods in its Multi-Factor Authentication overview and its Salesforce MFA documentation.

QUESTION NO: 39

Containers (UC) has an existing Customer Community. UC wants to expand the self-registration capabilities such that customers receive a different community experience based on the data they provide during the registration process. What is the recommended approach an Architect Should recommend to UC?

A. Create an After Insert Apex trigger on the user object to assign specific custom permissions.

B. Create separate login flows corresponding to the different community user personas.

C. Modify the Community pages to utilize specific fields on the User and Contact records.

D. Modify the existing Communities registration controller to assign different profiles.

ANSWER: D

Explanation:

Modify the existing Communities registration controller to assign different profiles. A custom self-registration controller can evaluate the information submitted by a prospective customer, apply the organization's persona rules, and create the external user with the appropriate profile. Profiles are a foundational way to tailor an Experience Cloud user's access because they define the user license and baseline object, field, tab, and application permissions. The selected profile can then be combined with permission sets, sharing configuration, audience targeting, and page variations to deliver an appropriate community experience for each registered persona.

This approach places the decision at the point where the user is provisioned, ensuring that the registration data directly determines the user's access model before the person begins using the community. Salesforce supports customizing Experience Cloud self-registration with Apex when the standard registration behavior does not meet the required business logic. The implementation should validate the submitted data, use only compatible external-user profiles and licenses, and follow secure provisioning practices so users receive only the intended access. Salesforce's guidance on self-registration and Experience Cloud user access is available in Customize Self-Registration for Experience Cloud and Experience Cloud User Profiles.

QUESTION NO: 40

A consumer products company uses Salesforce to maintain consumer information, including orders. The company implemented a portal solution using SalesforceExperience Cloud for its consumers where the consumers can log in using their credentials. The company is considering allowing users to login with their Facebook or LinkedIn credentials.

Once enabled, what role will Salesforce play?

A. Facebook and LinkedIn will be the SPs.

B. Salesforce will be the service provider (SP).

C. Salesforce will be the identity provider (IdP).

D. Facebook and LinkedIn will act as the IdPs and SPs.

ANSWER: B

Explanation:

Salesforce will be the service provider (SP). When an Experience Cloud site is configured to let consumers sign in using Facebook or LinkedIn, Salesforce delegates authentication to the selected external authentication provider. Facebook or LinkedIn validates the consumer's credentials and returns identity information to Salesforce through the configured authentication flow. Salesforce then uses that authenticated identity to establish the user's session and grant access to the Experience Cloud site and the consumer's permitted Salesforce data, such as order information.

In this arrangement, Salesforce is the application and resource environment that relies on an external identity system for authentication, which is the essential role of a service provider. Salesforce supports social sign-on by configuring authentication providers, including providers based on OAuth and OpenID Connect. The authentication provider configuration defines the external provider's endpoints, client credentials, and how Salesforce maps the returned identity to a user record.

For implementation details, see Salesforce's Social Sign-On with Authentication Providers documentation and the Single Sign-On Overview.

QUESTION NO: 41

An Architect needs to advise the team that manages the Identity Provider how to differentiate Salesforce from other Service Providers. What SAML SSO setting in Salesforce provides this capability?

A. IdentityProvider Login URL.

B. Issuer.

C. Entity Id

D. SAML Identity Location.

ANSWER: C

Explanation:

Entity Id is correct because it is Salesforce's service provider identifier in a SAML single sign-on configuration. An identity provider can support multiple service providers, each with different assertion consumer endpoints, certificates, NameID requirements, and attribute mappings. The Entity ID gives the identity provider a stable, unique value with which to recognize that an incoming SAML authentication request is for the Salesforce service-provider configuration rather than for another application.

When Salesforce initiates SAML SSO, it includes its service-provider identity in the SAML request. The identity provider uses that identifier to select the appropriate relying-party or application configuration and produce an assertion intended for Salesforce. The Entity ID configured in Salesforce must therefore match the service-provider identifier registered at the identity provider. This alignment is particularly important where one identity provider serves several Salesforce orgs, Salesforce domains, or non-Salesforce applications.

Salesforce documents the Entity ID as part of the information exchanged when configuring SAML SSO, and identifies it as the service provider's unique identifier. See Salesforce SAML Single Sign-On documentation and the Salesforce Metadata API reference for SAML SSO configuration.

QUESTION NO: 42

Universal Containers (UC) uses Salesforce for its customer service agents. UC has a proprietary system for order tracking which supports Security Assertion Markup Language (SAML) based single sign-on. The VP of customer service wants to ensure only active Salesforce users should be able to access the order tracking system which is only visible within Salesforce.

What should be done to fulfill the requirement?

Choose 2 answers

A. Setup Salesforce as an identity provider (IdP) for order Tracking.

B. Set up the Corporate Identity store as an identity provider (IdP) for Order Tracking,

C. Customize Order Tracking to initiate a REST call to validate users in Salesforce after login.

D. Setup Order Tracking as a Canvas app in Salesforce to POST IdP initiated SAML assertion.

ANSWER: A D

Explanation:

Setting up Salesforce as an identity provider (IdP) for Order Tracking lets Salesforce authenticate the user and issue a SAML assertion to the proprietary application. Because Salesforce controls authentication and only active users can successfully authenticate and establish a Salesforce session, the application can rely on the Salesforce-issued assertion for access. Salesforce can also control which users are entitled to launch the SAML-enabled application through assignment and access configuration, providing an appropriate centralized access-control point for service agents.

Setting up Order Tracking as a Canvas app in Salesforce provides the required in-Salesforce user experience: agents access the application from within Salesforce rather than navigating directly to a separate application interface. When the application is launched in this context, Salesforce can initiate the federated sign-on flow and POST the SAML assertion to the application. This combines Salesforce-controlled authentication with an embedded application experience, so the order-tracking system is available to authenticated, active Salesforce users in the intended Salesforce workspace.

Salesforce documents its ability to act as a SAML identity provider for external service providers in SAML Single Sign-On. For embedded application integration concepts and Canvas behavior, see the Salesforce Canvas Framework documentation.

QUESTION NO: 43

Universal Containers (UC) is building an authenticated Customer Community for its customers. UC does not want customer credentialsstored in Salesforce and is confident its customers would be willing to use their social media credentials to authenticate to the community. Which two actions should an Architect recommend UC to take?

A. Use Delegated Authentication to call the Twitter login API to authenticate users.

B. Configure an Authentication Provider for LinkedIn Social Media Accounts.

C. Create a Custom Apex Registration Handler to handle new and existing users.

D. Configure SSO Settings For Facebook to serve as a SAML Identity Provider.

E. Configure an Authentication Provider for LinkedIn Social Media Accounts and create a Custom Apex Registration Handler to handle new and existing users.

ANSWER: E

Explanation:

Configure an Authentication Provider for LinkedIn Social Media Accounts and create a Custom Apex Registration Handler to handle new and existing users. This combined recommendation supports social sign-on without requiring Universal Containers to manage customer passwords in Salesforce. An authentication provider establishes the trust and OAuth-based exchange with the external identity provider, allowing a customer to authenticate with their LinkedIn account and return to the Experience Cloud site as an authenticated user. Salesforce receives identity information from the provider rather than validating or retaining the customer's social-media password.

The custom Apex registration handler completes the implementation by defining what Salesforce does after successful external authentication. It can locate an existing user based on information returned by the provider, update that user when appropriate, or create a user and associate the required account/contact records for a first-time community customer. This controlled provisioning logic is especially important for an authenticated customer community, where the external identity must be matched to the correct Salesforce identity and access model. Salesforce documents authentication providers and their use for external authentication in the Auth namespace documentation, and documents the methods used to implement custom post-login user provisioning in the Auth.RegistrationHandler interface reference.

QUESTION NO: 44

Northern Trail Outfitters wants to implement a partner community. Active community users will need to review and accept the community rules, and update key contact information for each community member before performing any further operation on the portal.

Which approach will meet this requirement?

A. Create a custom landing page and email campaign asking all community members to login and verify their data.

B. Add a banner to the community Home page asking users to update their profile and accept the new community rules.

C. Create tasks for users who need to update their data or accept the new community rules.

D. Create a login flow that conditionally prompts users who have not accepted the new community rules and who have missing or outdated information.

ANSWER: D

Explanation:

Create a login flow that conditionally prompts users who have not accepted the new community rules and who have missing or outdated information. Salesforce login flows run immediately after a user authenticates, before the user accesses the Experience Cloud site, so they are suited to gathering required information or presenting required acknowledgments as part of the sign-in experience. The flow can evaluate fields on the User and associated Contact record, such as a rules-acceptance checkbox, acceptance date, and required contact-information fields. When the criteria indicate that action is needed, the flow can display screens requiring the partner user to review the rules, record acceptance, and provide the required updates. The flow then saves those changes before the user proceeds into the portal. Because it is conditional, users whose acceptance and contact data are already current can continue without unnecessary prompts. This makes the process enforceable at the point of access rather than merely notifying users and relying on later voluntary action. Configure the flow in Login & Registration settings for the Experience Cloud site and ensure the flow updates only records and fields the external user is permitted to access. See Salesforce’s Login Flows documentation and Experience Cloud login-flow guidance.

QUESTION NO: 45

Northern Trail Outfitters recently acquired a company. Each company will retain its Identity Provider (IdP). Both companies rely extensively on Salesforce processes that send emails to users to take specific actions in Salesforce.

How should the combined companys' employees collaborate in a single Salesforce org, yet authenticate to the appropriate IdP?

A. Configure unique MyDomains for each company and have generated links use the appropriate MyDomam in the URL.

B. Have generated links append a querystnng parameter indicating the IdP. The login service will redirect to the appropriate IdP.

C. Have generated links be prefixed with the appropriate IdP URL to invoke an IdP-initiated Security Assertion Markup Language flow when clicked.

D. Enable each IdP as a login option in the MyDomain Authentication Service settings. Users will then click on the appropriate IdP button.

ANSWER: D

Explanation:

Enable each IdP as a login option in the MyDomain Authentication Service settings. Users will then click on the appropriate IdP button. A single Salesforce org can support multiple SAML single sign-on configurations, allowing the combined workforce to continue using its respective corporate identity provider while accessing the same Salesforce users, records, collaboration tools, and business processes. In My Domain authentication configuration, the organization can expose the applicable SAML SSO configurations on the Salesforce login page. When a user follows an emailed Salesforce action link

and needs to authenticate, the My Domain login experience presents the available identity-provider choices. The user selects the provider associated with their company, and Salesforce initiates the appropriate service-provider-initiated SAML authentication flow. After successful authentication, Salesforce returns the user to the originally requested action or resource from the email link. This approach keeps Salesforce-generated URLs within the org's single My Domain while providing an explicit, supported authentication path for each population. It also allows each IdP to retain its own authentication policies, such as multifactor authentication and conditional-access requirements, without requiring separate Salesforce orgs or custom URL-routing logic. Salesforce documents SAML SSO configuration at SAML Single Sign-On and My Domain authentication setup at Configure My Domain Authentication.

QUESTION NO: 46

Which two considerations should be made whenimplementing Delegated Authentication?

Choose 2 answers

A. The authentication web service can include custom attributes.

B. It can be used to authenticate API clients and mobile apps.

C. It requires trusted IP ranges at the User Profile level.

D. Salesforce servers receive but do not validate a user's credentials.

E. Just-in-time Provisioning can be configured for new users.

ANSWER: B D E

Explanation:

Delegated Authentication sends the user's Salesforce username and password, along with the source IP address, to a customer-hosted authentication web service. Salesforce does not perform the password validation itself; the external service is responsible for validating the credentials and returning a success or failure result. This makes "Salesforce servers receive but do not validate a user's credentials" a fundamental implementation consideration. The external authentication endpoint must therefore be highly available, secured with a valid TLS certificate, and able to handle authentication traffic reliably.

Delegated Authentication also supports username-and-password login requests made through the SOAP API, allowing applicable API clients and mobile applications that use this login mechanism to authenticate against the external service. Finally, Salesforce supports just-in-time provisioning for Delegated Authentication. An organization can configure its authentication service to provide the information Salesforce needs to create a user record when an eligible user successfully authenticates for the first time. This allows user creation to be driven by the external identity system while retaining Salesforce licensing, profile, and user-management controls. See Salesforce's Delegated Authentication documentation and Just-in-Time Provisioning for Delegated Authentication.

QUESTION NO: 47

A large consumer company is planning to create a community and will requ.re login through the customers social identity. The following requirements must be met:

1. The customer should be able to login with any of their social identities, however salesforce should only haveone user per customer.

2. Once the customer has been identified with a social identity, they should not be required to authonze Salesforce.

3. The customers personal details from the social sign on need to be captured when the customer logs intoSalesforce using their social Identity.

3. If the customer modifies their personal details in the social site, the changes should be updated in Salesforce .

Which two options allow the Identity Architect to fulfill the requirements?

Choose 2 answers

A. UseLogin Flows to call an authentication registration handler to provision the user before logging the user into the community.

B. Use authentication providers for social sign-on and use the custom registration handler to insert or update personal details.

C. C. Redirect the user to a custom page that allows the user to select an existing social identity for login.

D. Use the custom registration handler to link social identities to Salesforce identities.

ANSWER: B D

Explanation:

“Use authentication providers for social sign-on and use the custom registration handler to insert or update personal details” is correct because an authentication provider establishes trust with a supported external social identity provider and lets an Experience Cloud user authenticate through that provider rather than creating a separate Salesforce credential. The authentication response supplies identity and profile attributes, such as name and email, to Salesforce. A custom registration handler can use that data during the sign-in process to create a user when needed or update the customer’s stored details on subsequent sign-ins. Consequently, changed details supplied by the social provider can be synchronized at the customer’s next successful social login.

“Use the custom registration handler to link social identities to Salesforce identities” is also correct. A registration handler implements Salesforce’s `Auth.RegistrationHandler` interface and contains the organization-specific logic for locating an existing Salesforce user and associating the authenticated social identity with that user. For example, it can apply a reliable customer-matching rule before creating a user, preventing separate community users from being provisioned when the same customer signs in through different configured social identities. Once the association exists, the social authentication flow identifies the customer for later access without requiring the customer to separately authorize Salesforce. See Salesforce’s documentation for Authentication Providers and the Auth.RegistrationHandler interface.