AZ-204 - 3. Implement Azure security -> 3.1 Implement user authentication and authorization -> 2.2.1 Authenticate and authorize users and apps by using Microsoft Entra ID

1. What is Microsoft Entra ID and how is it used in authentication and authorization?
2. What are managed identities and when should you use them?
3. How do you assign roles to users and apps in Entra ID?
4. How do you implement role-based access control (RBAC)?
5. How do you use Microsoft Graph to check user roles or group membership?
6. How do you authenticate using client credentials (app-only access)?
7. How do you configure an app to use a managed identity?
8. How do you restrict access to Azure resources using Entra ID?
9. How do you authorize apps to access APIs on behalf of a user?
10. What are best practices for securing app access via Entra ID?

---

1. What is Microsoft Entra ID and how is it used in authentication and authorization?
Microsoft Entra ID (formerly Azure AD) is Microsoft's cloud-based identity service.

- **Authentication**: Verifies user or app identity.
- **Authorization**: Controls access via roles, groups, or policies to Azure and custom resources.

---

2. What are managed identities and when should you use them?
System- or user-assigned identities created in Entra ID for Azure resources (e.g., App Service, Functions).
Use them to authenticate without secrets when calling Entra-secured resources like Key Vault or Graph.

---

3. How do you assign roles to users and apps in Entra ID?

- Go to the Azure resource → Access control (IAM) → Add role assignment.
- Assign roles (e.g., Reader, Contributor) to users, groups, or service principals.
  Use az role assignment create to script this.

---

4. How do you implement role-based access control (RBAC)?
Use Entra ID roles (built-in or custom) and assign them to identities.
Access is enforced based on assigned role scopes (e.g., resource group, subscription).

---

5. How do you use Microsoft Graph to check user roles or group membership?
Use Graph API endpoint /me/memberOf or /users/{id}/getMemberGroups.
Requires Group.Read.All or similar delegated/app permission.
Example:

```
GET https://graph.microsoft.com/v1.0/me/memberOf
```

6. How do you authenticate using client credentials (app-only access)?
Register the app in Entra ID → Generate a client secret or certificate → Grant API permissions.
Use MSAL or REST to request a token with client_id, client_secret, tenant_id, and scope.
Flow: OAuth 2.0 client credentials grant.

---

7. How do you configure an app to use a managed identity?

- Enable system-assigned identity in the Azure resource (App Service, Function, VM).
- Assign RBAC role to that identity (e.g., Key Vault Reader).
- Access tokens via Azure SDK's DefaultAzureCredential or IMDS endpoint.

---

8. How do you restrict access to Azure resources using Entra ID?
Use RBAC:

- Assign specific roles (e.g., Reader) to Entra identities.
- Scope can be subscription, resource group, or individual resource.
- Enforced via Entra token claims and role assignments.

---

9. How do you authorize apps to access APIs on behalf of a user?
Use **delegated permissions** via OAuth 2.0 authorization code flow.
The app receives an access token with the user's identity.
Ensure scopes like `User.Read` are consented to during sign-in.

---

10. What are best practices for securing app access via Entra ID?

- Use managed identity instead of storing secrets.
- Assign minimum required RBAC roles.
- Use conditional access policies where applicable.
- Validate token issuer, audience, and scopes in APIs.