3. Implement Azure security

└ **3.2 Implement secure Azure solutions**

└ **3.2.3 Implement Managed Identities for Azure resources**

1. What are managed identities and what problem do they solve?
2. What is the difference between system-assigned and user-assigned managed identities?
3. How do you enable a managed identity on an Azure resource?
4. How do you assign RBAC roles to a managed identity?
5. How do you authenticate to Azure services using managed identities in code?
6. How do you use managed identity with Azure Key Vault?
7. How do you troubleshoot managed identity access issues?
8. How do managed identities behave during resource deletion or scaling?
9. What services support managed identities?
10. What are best practices when using managed identities in cloud apps?

---

1. What are managed identities and what problem do they solve?

Managed identities provide Azure-hosted identities for applications to access Azure resources securely without storing credentials in code or config.

---

2. What is the difference between system-assigned and user-assigned managed identities?

- *System-assigned*: Tied to the resource lifecycle; deleted with the resource.
- *User-assigned*: Standalone; reusable across multiple resources; managed separately.

---

3. How do you enable a managed identity on an Azure resource?

Via Azure CLI:

```
az webapp identity assign --name <app-name> --resource-group <rg>
```

Or in ARM/Bicep: identity: { type: 'SystemAssigned' }

---

4. How do you assign RBAC roles to a managed identity?

Use Azure CLI:

```
az role assignment create \
  --assignee <clientId-or-objectId> \
  --role <role-name> \
  --scope <resource-scope>
```

---

5. How do you authenticate to Azure services using managed identities in code?

Use DefaultAzureCredential from Azure SDK:

```
var client = new SecretClient(new Uri(kvUrl), new DefaultAzureCredential());
```

Automatically uses the managed identity of the running resource.

---

6. How do you use managed identity with Azure Key Vault?

1. Enable managed identity on the resource
2. Assign Key Vault Secrets User role to the identity at Key Vault scope
3. Use DefaultAzureCredential in code to access secrets

7. How do you troubleshoot managed identity access issues?

- Verify identity is enabled
- Confirm role assignment at correct scope
- Check az role assignment list and Key Vault diagnostics logs
- Ensure DefaultAzureCredential is used correctly in code

---

8. How do managed identities behave during resource deletion or scaling?

- *System-assigned*: Deleted when the resource is deleted
- *User-assigned*: Must be manually managed; survives resource deletion
  Scaling (e.g., in App Service) automatically reuses the same identity

---

9. What services support managed identities?

Supported in:

- App Service, Functions
- VMs, VMSS
- Logic Apps
- Azure Container Apps
- Azure Kubernetes Service (AKS)
- Azure Data Factory, and more

---

10. What are best practices when using managed identities in cloud apps?

- Prefer system-assigned for single-resource use
- Use user-assigned for cross-resource or lifecycle-independent needs
- Always use RBAC for access control
- Avoid storing credentials; rely on identity + DefaultAzureCredential