4. Security and compliance

└ 4.3 Secret and sensitive data management

└ 4.3.1 Azure Key Vault secrets/keys/certs

---

1. What are the types of objects stored in Azure Key Vault?
2. How do you create and manage secrets in Azure Key Vault?
3. What is the difference between keys and secrets in Key Vault?
4. How do you control access to Key Vault using RBAC vs. access policies?
5. How do you integrate Azure Key Vault with GitHub Actions or Azure Pipelines?
6. How are secrets retrieved securely in CI/CD pipelines?
7. What are best practices for rotating secrets and certificates?
8. How is soft-delete and purge protection used in Key Vault?
9. How do managed identities access Azure Key Vault?
10. How do you audit access and usage of Azure Key Vault?

---

1. What are the types of objects stored in Azure Key Vault?

- Secrets (e.g., passwords, tokens),
- Keys (RSA/EC keys),
- and Certificates (with optional auto-renewal).

---

2. How do you create and manage secrets in Azure Key Vault?

Use Azure CLI:

`az keyvault secret set --vault-name <name> --name <secretName> --value <value>`

You can also manage via Azure Portal, REST API, or ARM/Bicep templates.

---

3. What is the difference between keys and secrets in Key Vault?

- **Secrets:** Store plain strings (e.g., passwords).
- **Keys:** Used for cryptographic operations (encrypt, sign) and can't be exported.

---

4. How do you control access to Key Vault using RBAC vs. access policies?

- **RBAC:** Assign roles like "*Key Vault Secrets User*" at Azure resource level.
- **Access policies:** Grant specific permissions (e.g., get, list) directly in Key Vault settings. RBAC is recommended for new deployments.

---

5. How do you integrate Azure Key Vault with GitHub Actions or Azure Pipelines?

Use AzureLogin action or service connection to authenticate, then access secrets via:
`az keyvault secret show` in a script step or with built-in `AzureKeyVault@2` task.

---

6. How are secrets retrieved securely in CI/CD pipelines?

Use secure service connections and managed identities. Avoid hardcoding secrets; pull them at runtime using secure pipeline tasks (e.g., *AzureKeyVault@2* or azure/login + CLI).

---

7. What are best practices for rotating secrets and certificates?

- Enable expiration dates.
- Use Event Grid with automation (e.g., *Logic Apps*, *Azure Functions*) for rotation.
- For certificates, enable auto-renew with integrated CAs.

8. How is soft-delete and purge protection used in Key Vault?

- **Soft-delete:** Keeps deleted items for a retention period (default 90 days).
- **Purge protection:** Prevents permanent deletion until retention expires. Both are essential for regulatory compliance.

---

9. How do managed identities access Azure Key Vault?

Assign the identity a role (RBAC) or access policy in *Key Vault* with required permissions. No credentials are needed—Azure handles auth transparently.

---

10. How do you audit access and usage of Azure Key Vault?

Enable diagnostic settings to send logs to *Log Analytics, Storage,* or *Event Hubs*. Monitor events like *SecretGet, KeySign* using *Azure Monitor* or *KQL*.