3. Build and release pipelines

└ 3.5 Implement deployment solutions

└ 3.5.4 Configuration Management Tools & Strategies

---

1. What are configuration management tools, and why are they important in DevOps pipelines?
2. Which configuration management tools are commonly used with Azure DevOps?
3. How do you integrate tools like Ansible, Chef, or Puppet into Azure Pipelines?
4. What are best practices for managing application configuration in CI/CD?
5. How do you secure sensitive configuration data in deployment pipelines?
6. What is Infrastructure as Code (IaC), and how does it relate to configuration management?
7. How can you automate configuration drift detection and remediation?
8. How do you handle configuration for different environments (dev, test, prod)?
9. What Azure-native configuration management options exist?
10. How do you monitor and audit configuration changes across environments?

---

1. **What are configuration management tools, and why are they important in DevOps pipelines?**

Tools that automate

- the deployment,
- configuration,
- and maintenance

of infrastructure and applications. They ensure

- consistency,
- repeatability,
- and compliance

across environments.

---

2. **Which configuration management tools are commonly used with Azure DevOps?**

Ansible, *Chef*, *Puppet*, *PowerShell DSC*, and *SaltStack*. Azure-native tools include *Azure Automation State Configuration* and *Azure Automanage*.

---

3. **How do you integrate tools like Ansible, Chef, or Puppet into Azure Pipelines?**

Use marketplace tasks or custom scripts within pipeline steps. Install tool dependencies, authenticate to targets, and invoke playbooks/manifests/recipes as pipeline tasks.

---

4. **What are best practices for managing application configuration in CI/CD?**

- Store configuration as code,
- use source control,
- parameterize settings,
- avoid hardcoding secrets,
- validate changes in lower environments,
- and document configuration decisions.

---

5. **How do you secure sensitive configuration data in deployment pipelines?**

- Use *Azure Key Vault* or pipeline secret variables,
- avoid storing secrets in code or repositories,
- restrict access to secrets,
- and audit secret usage in pipeline logs.

6. **What is Infrastructure as Code (IaC), and how does it relate to configuration management?**
   - IaC is managing infrastructure (servers, networks, etc.) using code and automation.
   - Configuration management ensures resources are provisioned & configured to desired state.

---

7. **How can you automate configuration drift detection and remediation?**
Use tools like *Azure Automation State Configuration*, *PowerShell DSC*, or monitoring solutions to regularly check for drift and automatically reapply the desired state if differences are detected.

---

8. **How do you handle configuration for different environments (dev, test, prod)?**
Use environment-specific parameter files, variable groups, or templates. Separate configuration data from code, and apply correct settings via pipeline variables at deployment time.

---

9. **What Azure-native configuration management options exist?**
Azure Automation State Configuration (DSC), *Azure Automanage*, *Azure App Configuration*, and *Azure Policy* for enforcing compliance.

---

10. **How do you monitor and audit configuration changes across environments?**
Enable activity logs and change tracking (*Azure Monitor*, *Log Analytics*), use version control for configuration code, and review audit trails for pipeline and resource changes.