Title: Tekton Pipelines: A Kubernetes-Native CI/CD Revolution – Replacing Jenkins and Ansible in Modern DevOps Workflows

Abstract: This paper introduces Tekton, a Kubernetes-native CI/CD framework that is rapidly gaining traction as a modern replacement for traditional tools like Jenkins and Ansible. As organizations increasingly adopt Kubernetes, Tekton provides a cloud-native, scalable, and secure way to define and run continuous integration and continuous delivery pipelines. The paper explores the architecture of Tekton, its advantages over Jenkins and Ansible, and how organizations benefit by adopting it.

1. Introduction: Jenkins and Ansible have been industry-standard tools for CI/CD and configuration management. However, with the evolution of cloud-native environments and Kubernetes, these tools often fall short in scalability, security, and Kubernetes-native integration. Tekton was developed to address these challenges by offering a CRD-based CI/CD framework designed to run directly on Kubernetes.

2. What is Tekton? Tekton is an open-source CI/CD system based on Kubernetes custom resources. It allows users to create and run pipelines natively within Kubernetes using standard Kubernetes APIs. Core components include: - **Task**: A series of steps run in a container - **Pipeline**: A sequence of Tasks - **TaskRun/PipelineRun**: Instantiation and execution of Tasks and Pipelines - **Workspaces**: Shareable volumes for Tasks - **Tekton Triggers**: Webhooks to start pipelines - **Tekton Chains**: Supply chain security and provenance

3. Comparison with Jenkins and Ansible:

3.1 Jenkins vs Tekton: | Feature | Jenkins | Tekton | |--------|---------|--------| | Platform | Standalone, plugin-based | Kubernetes-native | | Scalability | Manual agent management | Auto-scaling pods | | Pipeline Definition | Groovy scripts | YAML CRDs | | Security | Requires hardening | Uses Kubernetes RBAC and secrets | | GitOps Support | Limited | Strong GitOps integration |

3.2 Ansible vs Tekton: | Feature | Ansible | Tekton | |--------|---------|--------| | Focus | Configuration management | CI/CD pipeline orchestration | | Execution | SSH-based tasks | Container-based tasks | | Idempotency | Strong | Depends on container design | | K8s Integration | External tool | Native execution in K8s |

4. Why Replace Jenkins and Ansible with Tekton? - **Kubernetes-native**: Operates entirely within the Kubernetes ecosystem - **Simplified CI/CD management**: No need for external agents or servers - **Cloud-native security**: Leverages K8s secrets and RBAC - **Improved scalability and maintainability** - **Declarative pipelines and GitOps alignment**

5. Organizational Benefits: - **Faster time-to-market** with parallel and auto-scaled execution - **Lower operational overhead** with native Kubernetes deployment - **Greater security and compliance** through native audit logging and secret management - **Better integration** with modern tools (ArgoCD, Flux, GitLab, etc.) - **Future-proof architecture** for hybrid and multi-cloud strategies

6. Real-World Use Case: A financial institution replaced Jenkins and Ansible with Tekton across microservices. Tasks such as build, test, and deploy were containerized as Tekton Tasks. The transition resulted in: - 40% reduction in pipeline execution time - Improved deployment reliability - Unified GitOps and CI/CD platform

7. Migration Strategy: - Audit current Jenkins/Ansible usage - Convert Jenkinsfiles and Ansible tasks into Tekton Tasks - Define Pipelines and use Triggers for automation - Gradually shift services to Tekton-based pipelines - Monitor and refine with Tekton Dashboard and Results

8. Conclusion: Tekton is a powerful and extensible Kubernetes-native alternative to Jenkins and Ansible. Its alignment with cloud-native practices, declarative pipelines, and scalable architecture makes it ideal for modern DevOps needs. Organizations seeking to optimize their CI/CD pipelines and move toward GitOps and Kubernetes-native development should strongly consider adopting Tekton.

References: 1. Tekton Documentation - https://tekton.dev 2. CD Foundation Projects 3. CNCF GitOps Working Group 4. Jenkins and Ansible Official Docs 5. Real-world Tekton adoption case studies