Securing the Kubernetes Ecosystem: A Comprehensive Multi-Level Framework

Dec 5, 2024

Agenda

Securing the Kubernetes Ecosystem: key Highlights

1. About me

2. Statistical report – Alarming security Facts

3. Goal

4. A Multi-Level Approach & Architecture

5. Final touch

About me

Thiyagarajan Aramudhan

Cloud Services Manager

- IT leader with 20+ years of experience
- Focus on Architecture, SRE, Cloud Solution and Gen AI.

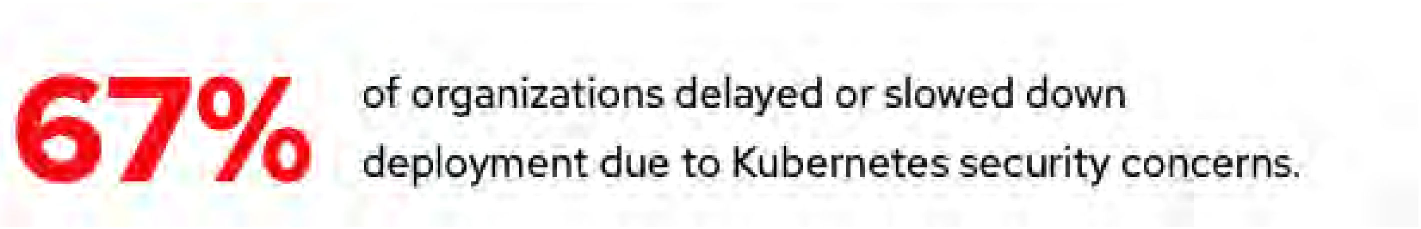
Connect with me via Linked in
www.linkedin.com/in/thiyagarajan-aramudhan-ba8bb9ab/

Statistical Report – Reveals Alarming Trends

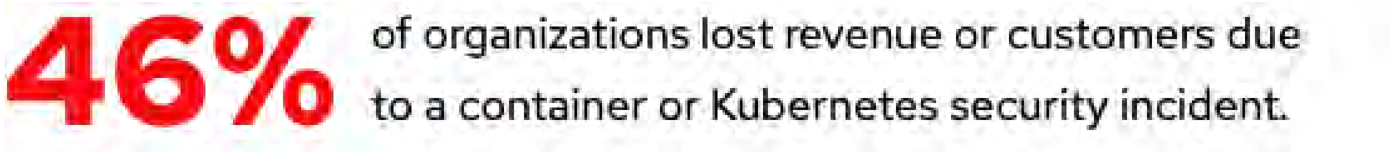
Red Hat

The state of Kubernetes security report

2024 edition



67% of organizations delayed or slowed down deployment due to Kubernetes security concerns.



46% of organizations lost revenue or customers due to a container or Kubernetes security incident.

Goal

A multi-level approach to secure the Kubernetes ecosystem is crucial to achieve robust security.

Securing Your Kubernetes Eco-system: A Multi-Level Approach

Level 1: Infrastructure

Level 2: Cluster

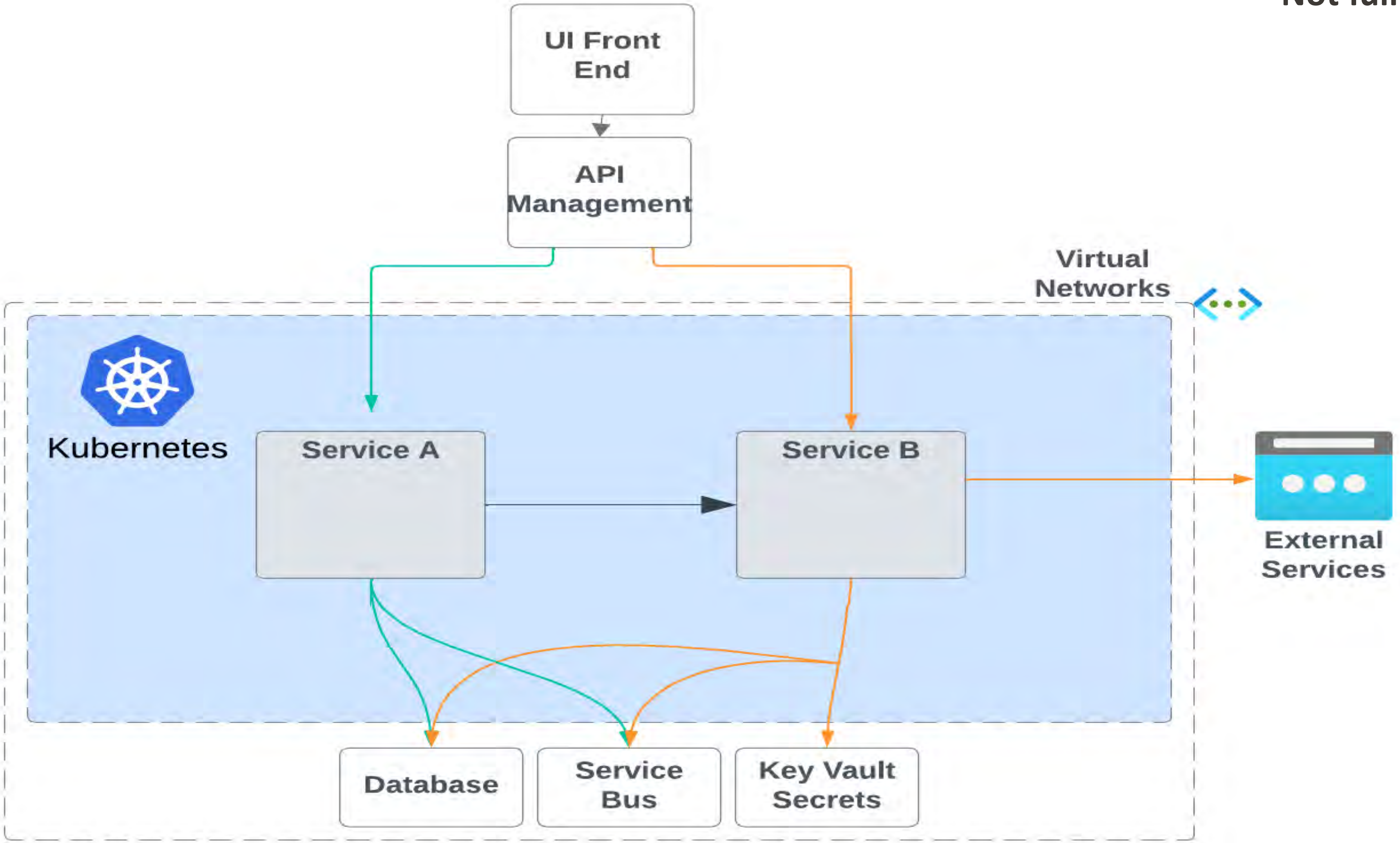
Level 3: Container

Level 4: Application

Level 5: Code

Generic API and Web App – Architecture

Not fully Secured!



Level 1: Infrastructure Level Security - Laying the Foundation

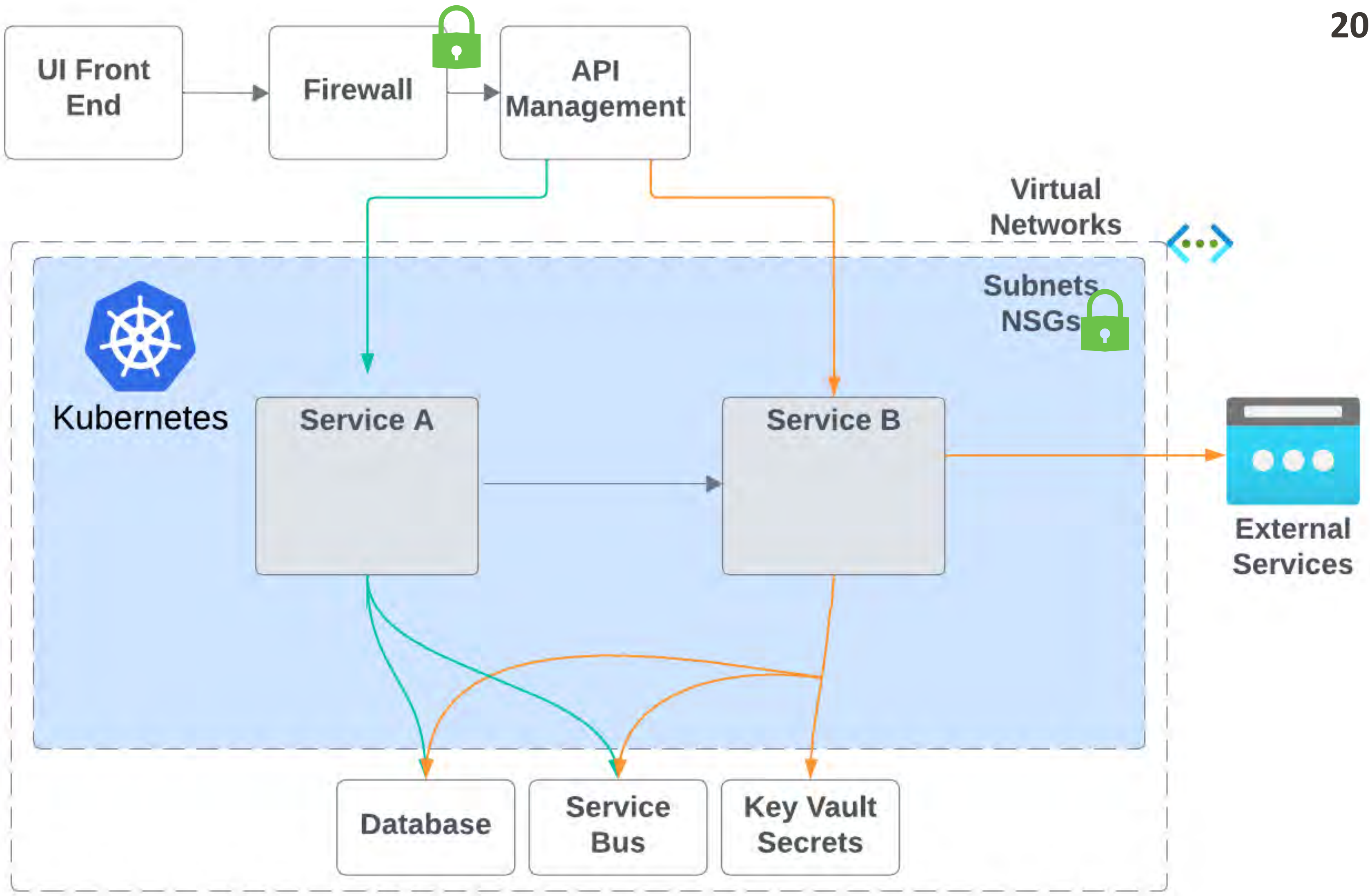
- ❑ Secure your foundation (Azure, AWS, on-premises).
- ❑ Implement firewalls.
- ❑ Regularly apply security patches.
- ❑ Leverage cloud provider security features like Azure Security Groups, AWS Security Groups & Network ACLs .

AKS	AWS	Open Source
Azure Security Center, NSGs, Azure Firewall	AWS Security Hub, Security Groups, AWS WAF	Firewall, OS hardening

Level 1: Infrastructure Level Security

✓

20% Secured



Level 2: Cluster Level Security - Protecting the Kubernetes Core

□ Implement Role-Based Access Control (RBAC).

□ Enforce network policies using Calico.

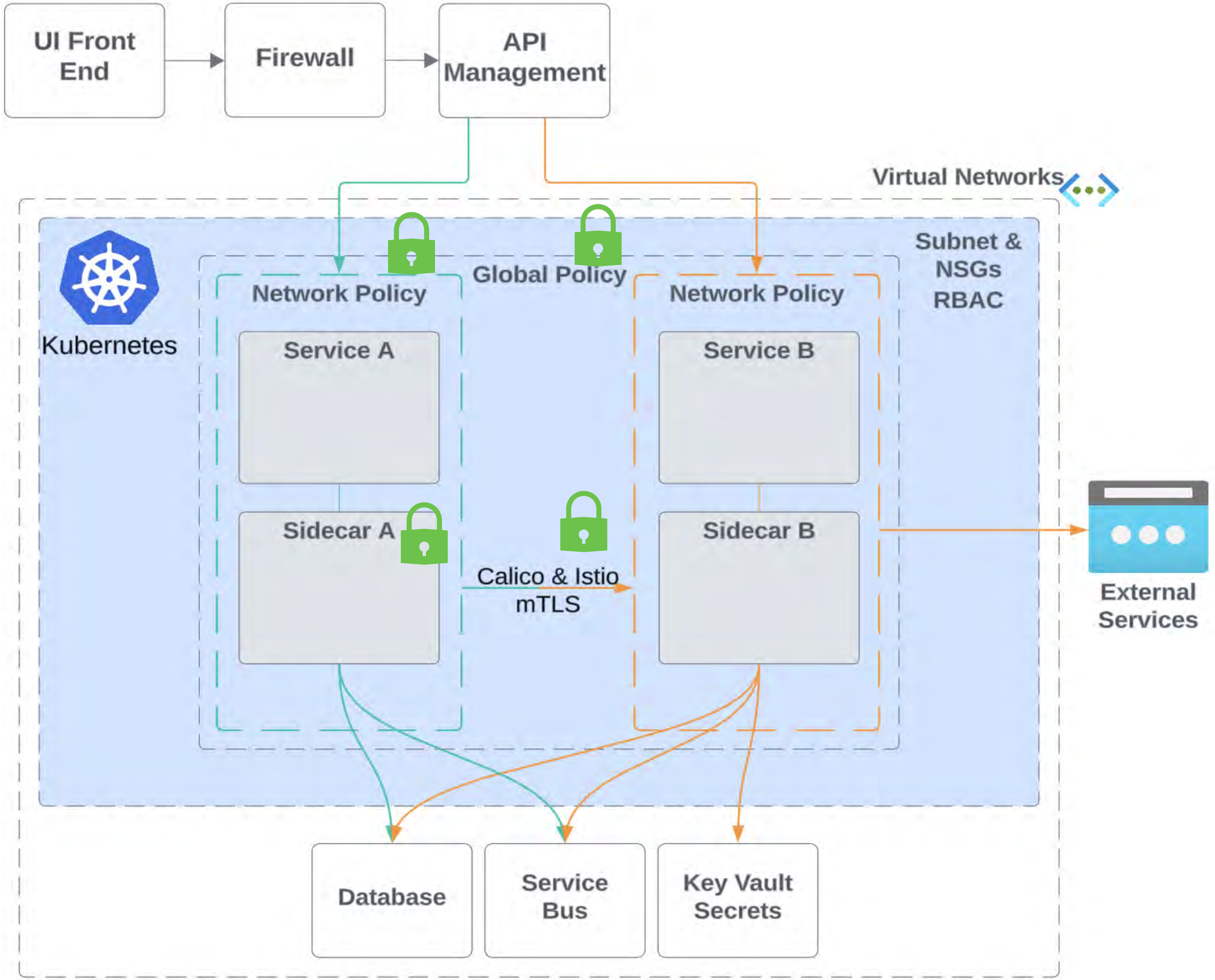
□ Encrypt the communication between the control plane and data plane components using Istio

Azure	AWS	Open Source
Azure RBAC, Azure Policy	IAM Roles, AWS Security Hub	RBAC, Network Policies (Calico)

Level 2: Cluster Level Security

✓

40% Secured



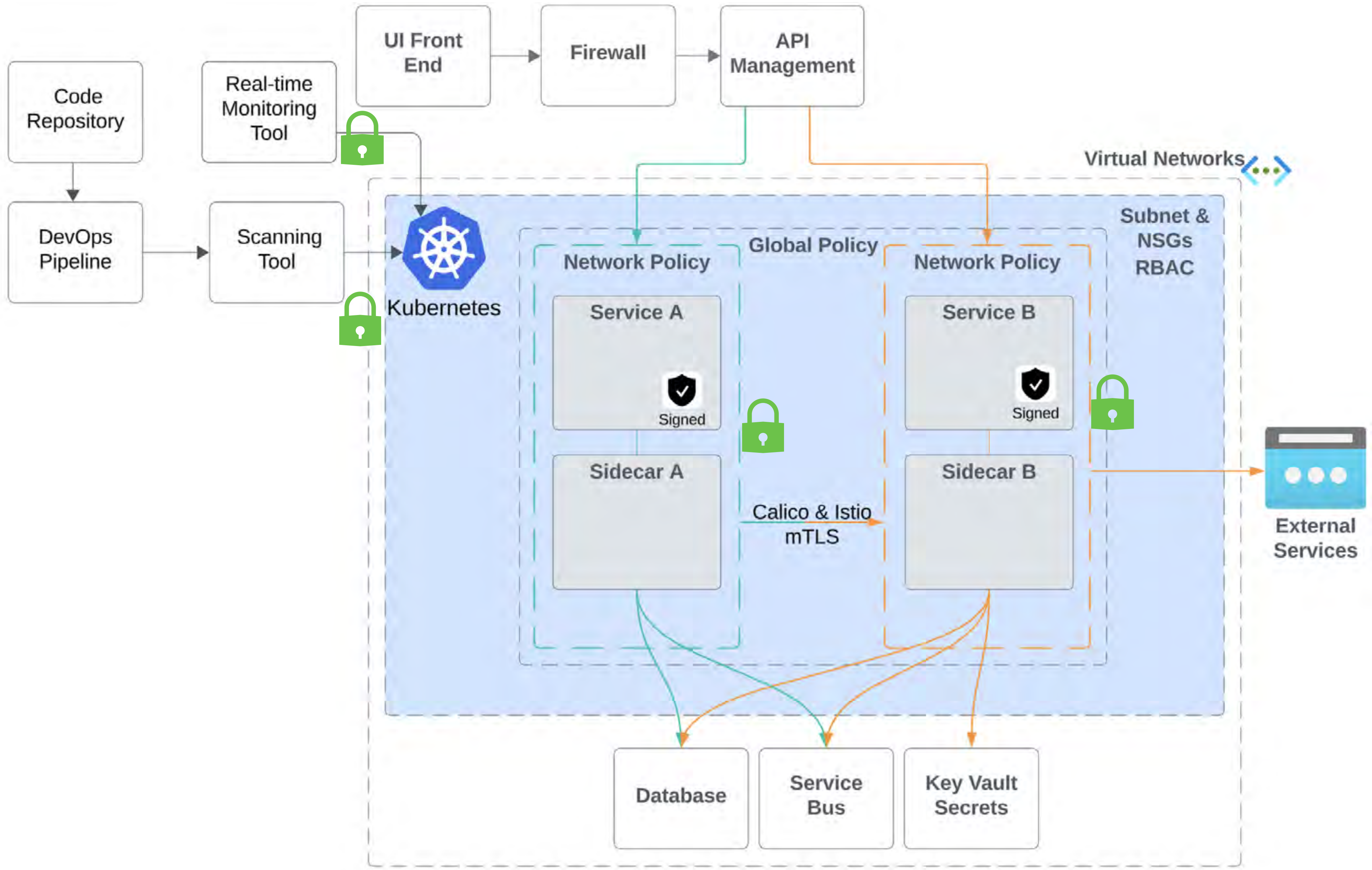
Level 3: Container Level Security - Protecting Your Images and Runtime

- ❑ Use minimal base images.
- ❑ Scan container images for vulnerabilities
- ❑ Utilize real-time monitoring and protection tools
- ❑ Prioritize security in development (DevSecOps).

Azure	AWS	Open Source
Azure Container Registry, AKS Vulnerability scanning	Amazon ECR, AWS Inspector	Container image scanning tools (Ex. Trivy, Clair), Security context

Level 3: Container Level Security

60% Secured



Level 4: Application-Level Security - Securing Your Applications within Kubernetes

❑ Implementing secure coding practices.

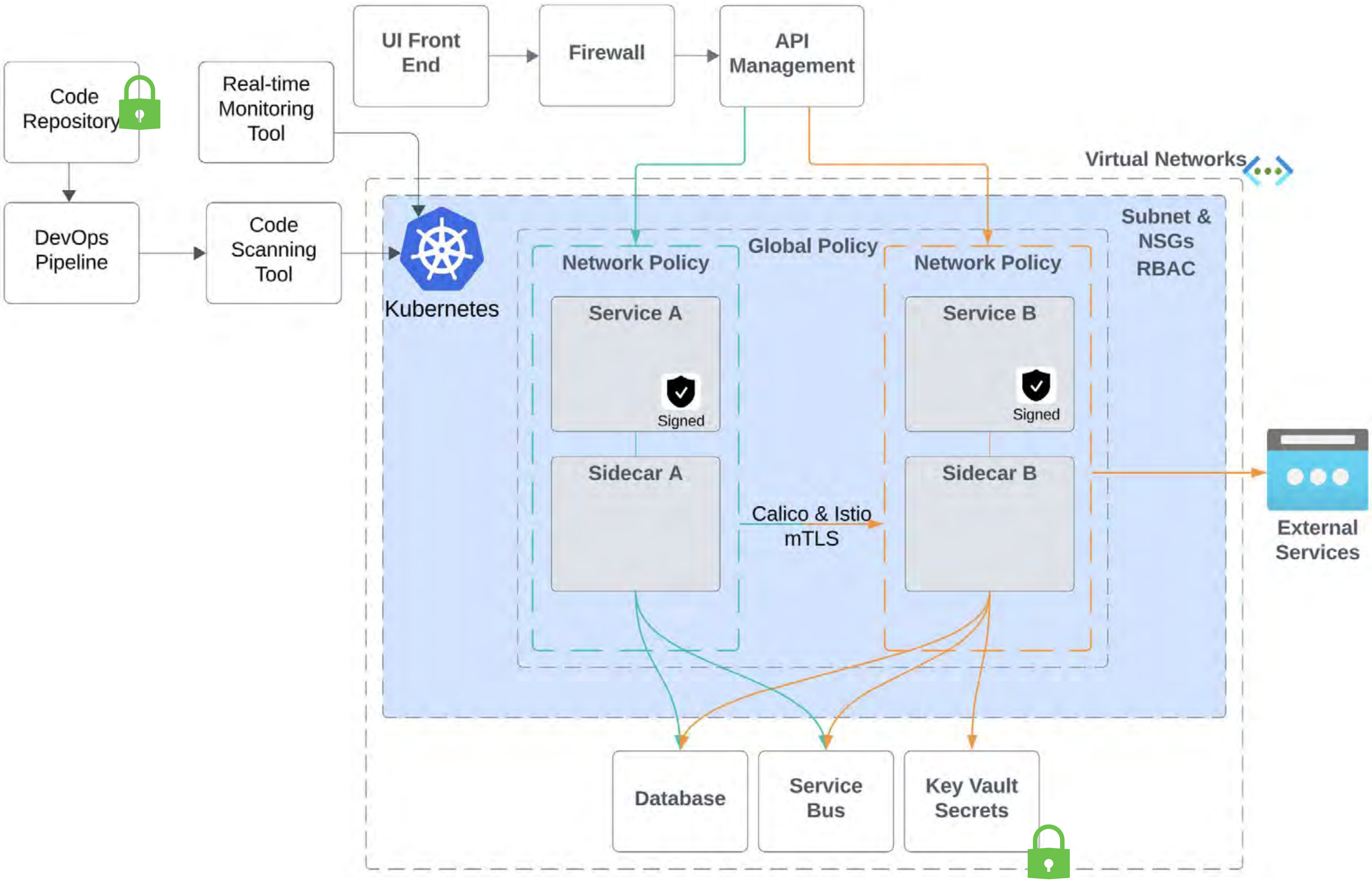
❑ Manage secrets properly

Azure	AWS	Open Source
Azure Key Vault	AWS Secrets Manager, AWS IAM Roles	Secret management tools (Vault, Sealed Secrets), Secure coding practices

Level 4: Application-Level Security

✓

80% Secured



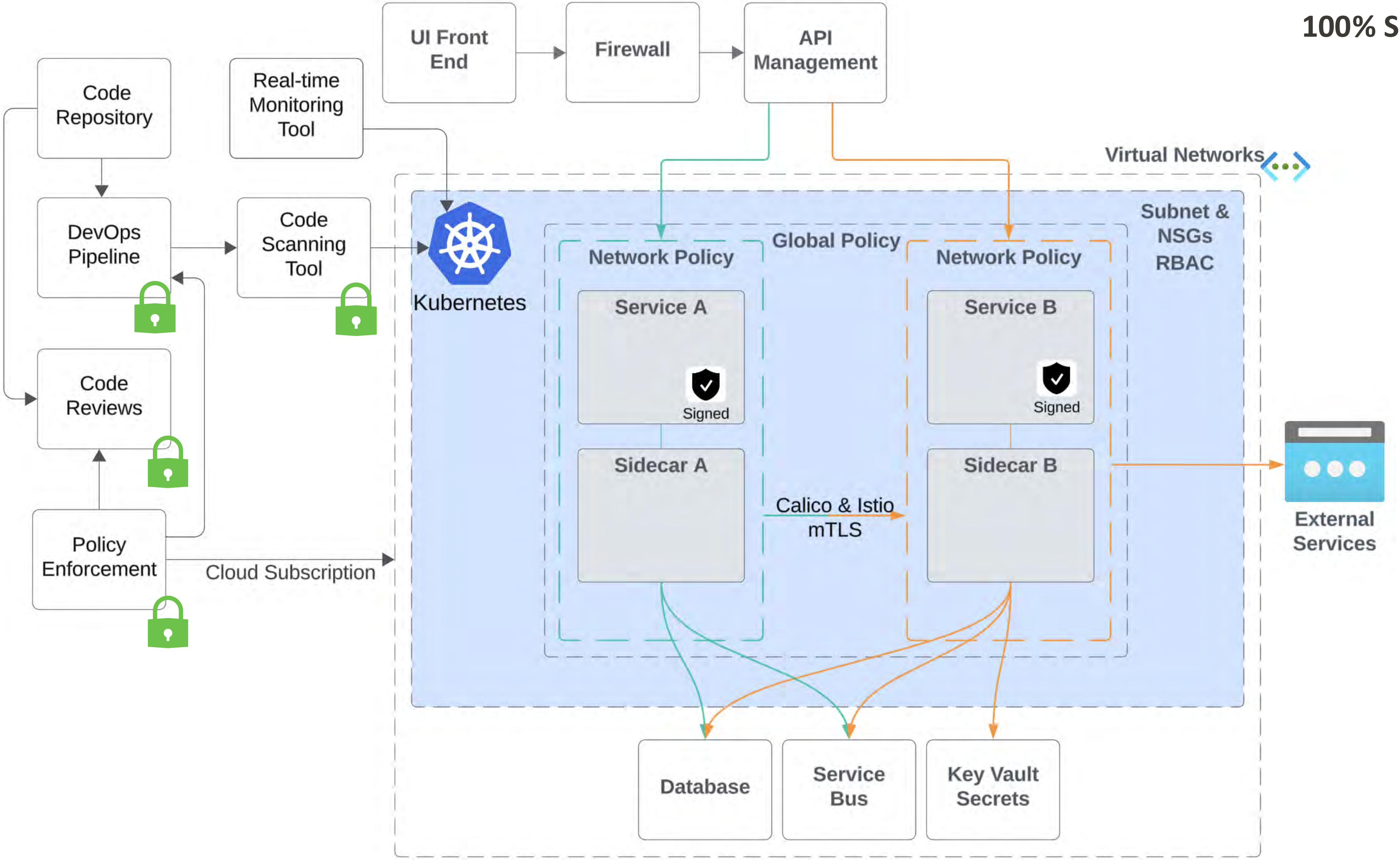
Level 5: Code Level Security- Building Secure Applications from the Ground Up

- ❑ Conduct thorough code reviews.
- ❑ Utilize static and dynamic analysis tools
- ❑ Implement policies for automated compliance checks.

Azure	AWS	Open Source
Azure DevOps security features, Static code analysis tools	AWS Code Pipeline security features, Static code analysis tools	Static code analysis tool (SonarQube), Open Policy Agent (OPA)

Level 5: Code Level Security

100% Secured



Advanced levels….

AI Ops (Artificial Intelligence for IT Operations):

Proactive threat detection

Automated remediation

Zero Trust Security:

- “Never trust, always verify”.

Conclusion

.

By adopting multi-level approach with leveraging the right tools, we can significantly enhance the security of our Kubernetes environments.

Thank you.