

Security

1. Container Security:

- **Image Scanning:**
 - Ensuring container images are free from vulnerabilities before deployment.
 - Tools like **Clair**, **Trivy**, or **Anchore** can scan for known vulnerabilities in container images.
- **Immutable Images:**
 - Using read-only images and containers to prevent modification at runtime.

2. Pod Security:

- **Pod Security Policies (PSP)** (deprecated in Kubernetes 1.21, but still relevant in older versions):
 - Define security controls for Pods (e.g., allow or disallow privileged containers, restrict running as root, enforce read-only file systems).
 - Controls what actions can be performed on Pods, such as running containers as root or accessing sensitive host files.
- **Pod Security Standards (PSS):**
 - New set of security controls for Pods replacing PSP, with three levels: **Privileged**, **Baseline**, and **Restricted**.
- **Admission Controllers:**
 - The **PodSecurityAdmission** (PSA) admission controller enforces the Pod Security Standards (PSS).
 - You can configure it at the namespace level with annotations that specify which PSS level (Privileged, Baseline, Restricted) is required.

3. Role-Based Access Control (RBAC):

- **Roles and RoleBindings:**
 - Define who can access Kubernetes resources and what operations they can perform.
 - Granular permissions on resources like Pods, Services, ConfigMaps, and more.

- **ClusterRoles and ClusterRoleBindings:**
 - Used for setting permissions across the entire Kubernetes cluster.

4. Service Accounts and Identity Management:

- **Service Accounts:**
 - Kubernetes uses service accounts to grant applications running in Pods permissions to access Kubernetes API resources.
- **Identity and Access Management (IAM):**
 - Integration with external identity providers (e.g., AWS IAM, Azure AD) to control access to Kubernetes resources.

5. Network Security:

- **Network Policies:**
 - Define rules for controlling the communication between Pods (e.g., allow traffic only from specific Pods, block access from certain networks).
 - Ensure secure communication within the cluster.
- **TLS Encryption:**
 - Encrypting communication between services and between Pods using **Transport Layer Security (TLS)**.
- **Service Mesh (e.g., Istio):**
 - Provides secure communication between microservices by enforcing mTLS, along with fine-grained access control.

6. Secrets Management:

- **Kubernetes Secrets:**
 - Store sensitive information like API keys, tokens, passwords in Kubernetes resources.
 - Secrets can be mounted as volumes or injected as environment variables into containers.
- **Encryption at Rest:**
 - Ensuring that Secrets and sensitive data are encrypted when stored in etcd.
- **Third-Party Tools for Secret Management:**
 - Integration with tools like **HashiCorp Vault**, **SealedSecrets**, or **AWS Secrets Manager** for enhanced security.

7. Container Runtime Security:

- **Security Contexts:**
 - Define security settings for Pods and containers (e.g., setting user IDs, restricting privileges, setting capabilities).
- **RunAsUser & RunAsGroup:**
 - Define the user and group ID under which a container should run, limiting privileges.
- **Privilege Escalation:**
 - Prevent containerized applications from gaining escalated privileges (e.g., running as root).

8. Audit Logging:

- **Audit Logs:**
 - Kubernetes provides audit logs to record access and actions performed on the API server.
 - Helps with monitoring and detecting suspicious or unauthorized access.

9. Supply Chain Security:

- **Secure Supply Chain:**
 - Ensuring security in every step of the container supply chain, from image creation to deployment.
 - Using signed images (e.g., via **Notary** or **Cosign**) to ensure image integrity.

10. Vulnerability Management:

- **Kubernetes Security Contexts:**
 - Controls around running containers with restricted privileges (e.g., no access to the host network or storage).
- **Runtime Security (e.g., Falco):**
 - Tools like **Falco** that monitor container behavior during runtime to detect security incidents like unauthorized network access or privilege escalation.