Gobernanza y Seguridad de Contenedores con Harbor y OPA Gatekeeper

Alejandro Liáñez Frutos – 2º ASIR – IES Gonzalo Nazareno.

Objetivos del Proyecto

1 **Entorno Seguro**

Crear un entorno seguro, privado y gobernado para contenedores.

2 **Herramientas Open Source**

Utilizar herramientas open source como Kubernetes (Minikube), Harbor y OPA Gatekeeper.

3 **Buenas Prácticas**

Reforzar las buenas prácticas en despliegue de aplicaciones contenerizadas.

Arquitectura del Entorno

- Kubernetes local con Minikube sobre KVM.
- Registro privado Harbor con autenticación y escaneo (Trivy).
- Aplicación de políticas con OPA Gatekeeper.

¿Qué es Harbor?

Registro Privado y Seguro

Harbor es un registro de contenedores privado, robusto y seguro para empresas.

Características Clave

- Autenticación basada en roles (RBAC).
- Escaneo de vulnerabilidades con Trivy.
- Replicación de imágenes.
- Interfaz web completa y auditable.

Entornos Empresariales

Ideal para entornos empresariales o educativos que requieren control estricto.

Seguridad en Harbor

Escaneo Automático

Trivy escanea las imágenes automáticam ente al subirlas.

Control de Acceso

Acceso a proyectos por roles definidos (administrador, desarrollador).

Registro de Actividad

Auditoría completa de logs y actividades de usuarios.

¿Qué es OPA Gatekeeper?

Extensión de OPA

Es una extensión de Open Policy Agent para Kubernetes.

Webhook de API

Se integra como webhook en el API Server de Kubernetes.

Políticas de Seguridad

Permite definir y aplicar políticas de seguridad y gobernanza.

Lenguaje Rego

Las reglas se escriben en el lenguaje Rego.

Beneficios de Políticas con OPA

Seguridad Reforzada

Políticas que impiden despliegues no conformes.

Gobernanza Centralizada

Control unificado sobre el despliegue de recursos.

Cumplimiento

Asegura las buenas prácticas de seguridad y etiquetado.

Flujo de Trabajo Integrado

Desarrollador Sube Imagen

La imagen es creada y subida a Harbor.

Harbor Escanea

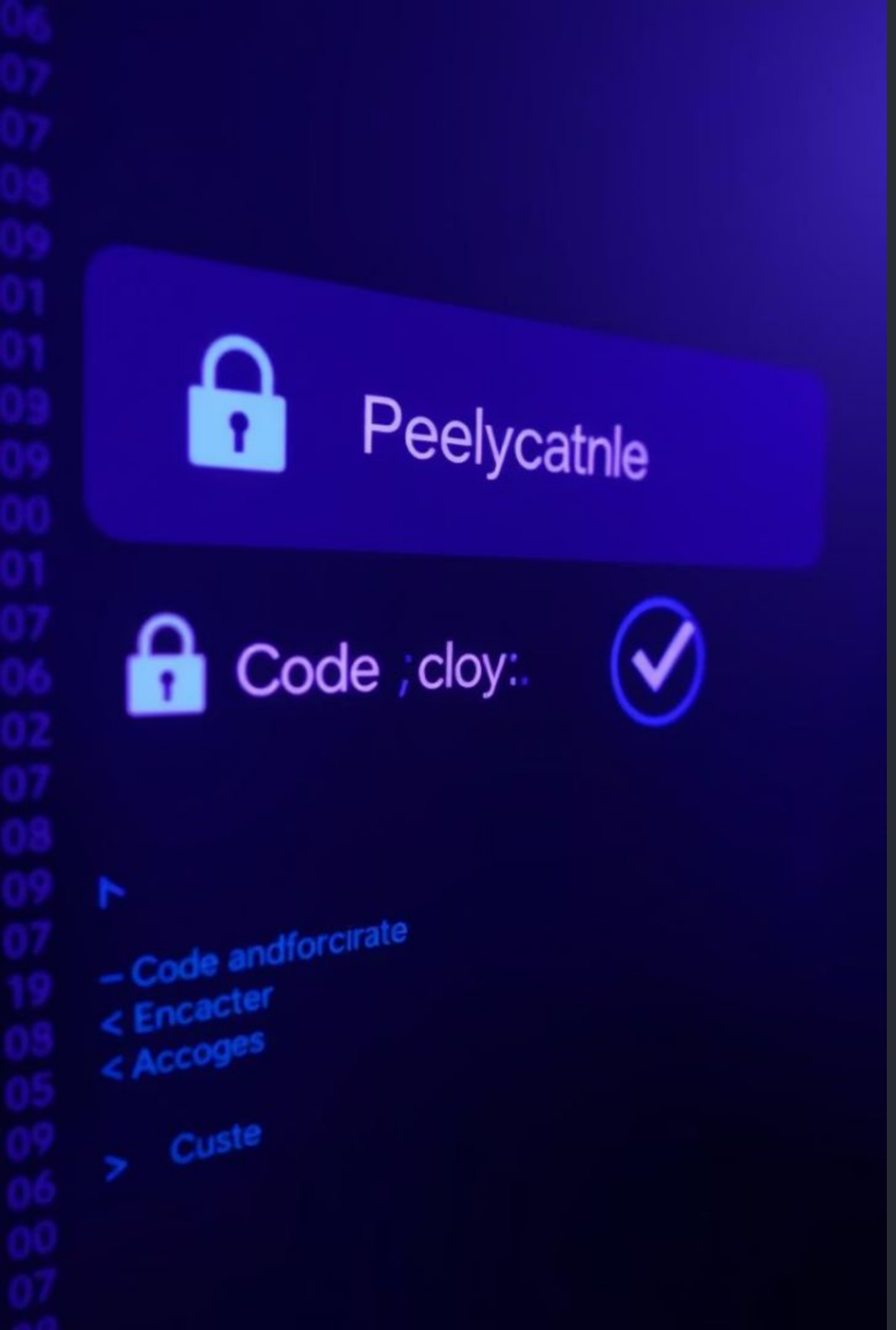
Harbor automáticamente escanea con Trivy.

Gatekeeper Valida

Gatekeeper valida el despliegue en Kubernetes.

Rechazo

Si no cumple, el despliegue es rechazado.



Demos

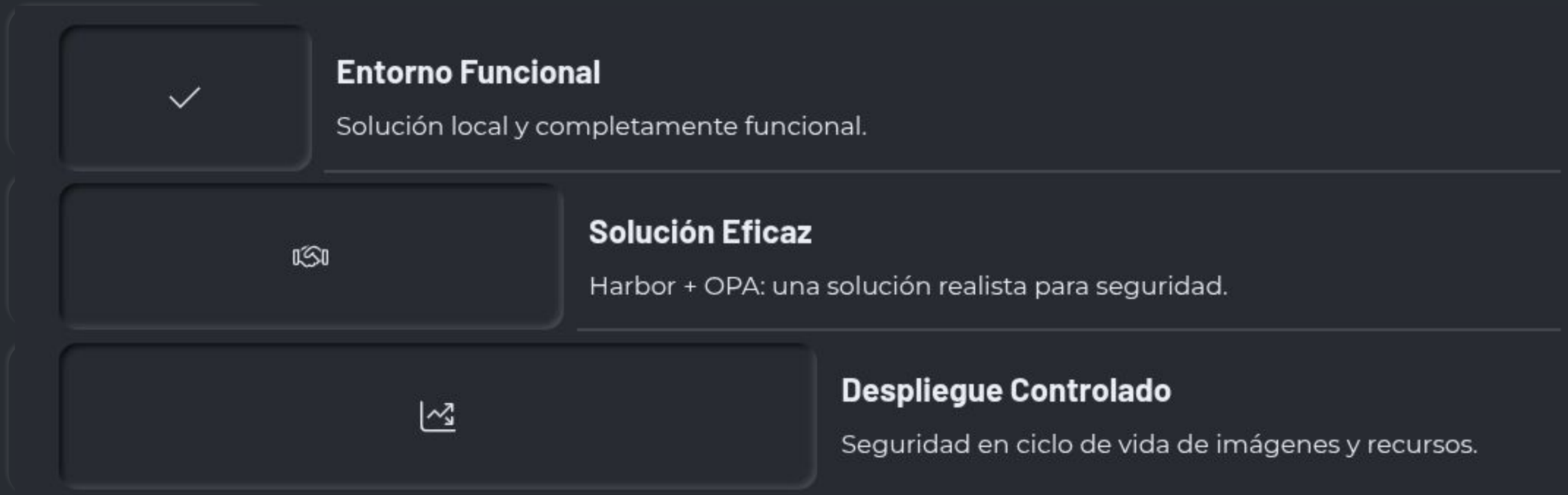
Despliegue de imágenes y escaneo automático; visualización de vulnerabilidades y control de acceso por roles.

Demo 1: Bloqueo de imágenes externas no autorizadas.

Demo 2: Bloqueo de volúmenes hostPath por seguridad.

Demo 3: Bloqueo de pods privilegiados.

Conclusiones



Posibles mejoras futuras incluyen la integración CI/CD, políticas Rego más avanzadas y autenticación externa en Harbor.