Del nodo hacia arriba: guía completa sobre la seguridad en Kubernetes

Con Prisma Cloud

paloalto NETWORKS | PRISMA CLOUD BY PALO ALTO NETWORKS

Contenido

3 **Introducción**

4 **Capítulo 1: Aspectos básicos de la seguridad en Kubernetes**

4 Kubernetes es un gigante multicapa

5 Protecciones nativas de Kubernetes

6 **Capítulo 2: Protección de la infraestructura de Kubernetes**

7 Entornos de desarrollo integrados

7 Integración continua

7 Gestión de la configuración

8 **Capítulo 3: Protección de imágenes de contenedor para su ejecución en Kubernetes**

8 Escritorio de desarrollo

8 Integración continua

9 Registros de contenedores

10 **Capítulo 4: Seguridad del tiempo de ejecución en Kubernetes**

10 Visibilidad

10 Protección en tiempo de ejecución

10 Protección de la red

12 Supervisión del SO de los nodos

12 Cumplimiento normativo y auditorías de seguridad en Kubernetes

13 **Conclusión**

Introducción

La mayor parte del debate sobre la seguridad en Kubernetes® se centra en lo complejo que resulta proteger un clúster. Cuentan que Kubernetes solo dispone de un puñado de funciones de seguridad nativas, así que resulta tremendamente difícil proteger todas las capas de estos entornos.

Es cierto que Kubernetes ofrece pocas herramientas de seguridad integradas y que, para proteger esta plataforma, hay que afrontar numerosos tipos de vulnerabilidades potenciales en diferentes capas de la infraestructura. Sin embargo, eso no significa que haya plantear la seguridad de Kubernetes como un desafío imposible.

De hecho, que Kubernetes sea una plataforma con tanto alcance y tantas integraciones abre una puerta: facilita la creación de un conjunto de procesos automatizados y sistemáticos capaz de integrar la seguridad en el corazón del proceso de creación e implementación de Kubernetes. Como resultado, se obtiene una estrategia de seguridad integrada que mitiga las amenazas en todas la capas y niveles de la infraestructura.

En este libro electrónico se explica cómo diseñar una estrategia de seguridad que refuerce el resto de los procesos basados en Kubernetes, en lugar de entorpecerlos. En él, se identifican los problemas que plantea la seguridad de Kubernetes desde el nodo y se señalan soluciones específicas para cada uno, con especial atención a las estrategias automatizadas y ampliables que mantienen protegidas las cargas de trabajo de Kubernetes independientemente del tamaño del clúster o del tipo de infraestructura utilizada para alojarlo (instalación local, nube pública o servicio gestionado).

Capítulo 1: Aspectos básicos de la seguridad en Kubernetes

Antes de ahondar en los problemas de seguridad específicos que plantea Kubernetes y en las estrategias para afrontarlos, repasemos brevemente las consideraciones generales esenciales sobre la seguridad de Kubernetes.

Kubernetes es un gigante multicapa

En primer lugar, es importante comprender que Kubernetes es una plataforma compleja que consta de más de media docena de componentes distintos. Cuenta con API Server para que las distintas partes del clúster se comuniquen entre sí, un programador que gestiona la distribución de las cargas de trabajo y controladores para gestionar el estado de Kubernetes en sí. También dispone de un agente que se ejecuta en cada nodo (o servidor) dentro del clúster, así como de un almacén clave-valor que aloja la información de configuración del clúster.

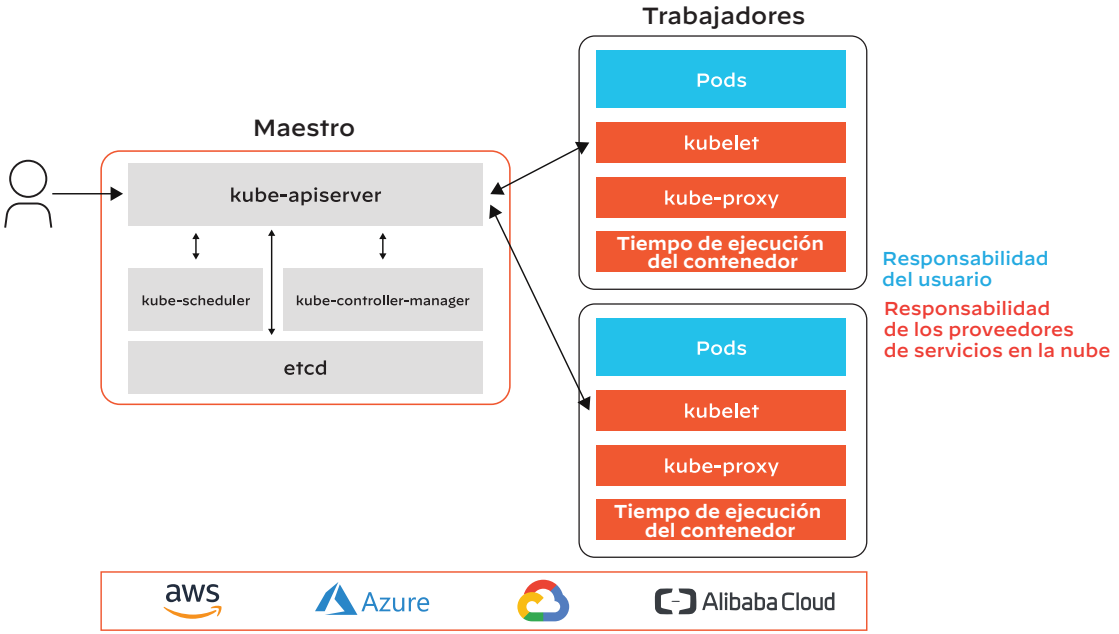

Figura 1: Arquitectura de servicios gestionados de Kubernetes

Eso no son más que los componentes principales de Kubernetes en sí. Un clúster funcional también depende de muchas partes móviles, como el tiempo de ejecución de los contenedores, algún tipo de solución de almacenamiento persistente, una herramienta de creación de logs o sistemas operativos para cada nodo, entre otras.

Cada una de estas partes del clúster de Kubernetes suma sus propias posibles vulnerabilidades. Los tiempos de ejecución de los contenedores pueden tener defectos de codificación que permitan la escalada de privilegios dentro del contenedor. El API Server de Kubernetes podría estar mal configurado, de forma que los atacantes obtengan acceso a recursos que deberían estar bloqueados. Podría haber vulnerabilidades dentro de una aplicación basada en contenedores o en los sistemas operativos que se ejecutan en los nodos de Kubernetes, lo que posibilita los ataques de escalada de privilegios o el acceso a datos confidenciales. Estos son solo algunos ejemplos.

En resumen, proteger Kubernetes pasa por proteger un amplio abanico de componentes distintos, cada uno con sus propias necesidades de seguridad. No existe un solo conjunto de herramientas o procesos que sirva para proteger fácilmente todos los puntos de un clúster de Kubernetes frente a todos los tipos de vulnerabilidad. Se requiere un sistema de defensa de múltiples frentes.

Mecanismos de protección nativos de Kubernetes

Para complicar aún más la seguridad de Kubernetes, esta plafaforma es prácticamente incapaz de protegerse a sí misma sin la ayuda de herramientas externas, a pesar de contar con ciertas funciones de seguridad integradas.

Kubernetes permite a los administradores definir políticas de control de acceso basado en funciones (RBAC, por sus siglas en inglés) para ayudar a evitar el acceso no autorizado a los recursos del clúster. También pueden configurar políticas de seguridad del pod y políticas de red para prevenir determinados tipos de ataque a los pods y a la red que los conecta. Pueden imponer cuotas de recursos para reducir el impacto que puedan provocar los atacantes que entren en una parte del clúster. Gracias a esas cuotas de recursos, el atacante no podrá perpetrar ataques por denegación de servicio (lo que privaría al resto del clúster de suficientes recursos para funcionar), siempre que el ataque no se propague más allá de la parte del clúster donde se origina.

Kubernetes permite a los administradores definir políticas de control de acceso basado en funciones (RBAC, por sus siglas en inglés) para ayudar a evitar el acceso no autorizado a los recursos del clúster.

Las funciones de protección nativas de Kubernetes cubren ciertas carencias de seguridad de los clústeres. No obstante, resultan poco o nada eficaces frente a muchos otros tipos de riesgos de seguridad, como los exploits que afectan a los sistemas operativos de los nodos o al tiempo de ejecución de los contenedores.

Para crear una estrategia de seguridad holística en Kubernetes, es preciso mirar más allá de sus escasas funciones de seguridad integradas. Esas funciones se pueden y deben usar donde corresponda para mitigar los riesgos para la seguridad, pero por sí solas no tienen, ni de lejos, la funcionalidad que se requiere para proteger un clúster.

En los siguientes capítulos, exploraremos las condiciones necesarias para elaborar una estrategia de seguridad completa para Kubernetes que vaya mucho más allá de las limitadas funciones integradas en la plataforma.

Capítulo 2: Protección de la infraestructura de Kubernetes

La primera capa general que hay que proteger en los entornos basados en Kubernetes es la capa de desarrollo: un conjunto de herramientas que usan los desarrolladores para crear el código que se ejecutará en el entorno de Kubernetes.

Estas herramientas no son parte de Kubernetes en sí. Sin embargo, dado que la seguridad del clúster depende de lo seguro que sea el código que se ejecuta en él, es preciso dar los pasos necesarios para proteger el código antes incluso de implementarlo.

En este capítulo se explica cómo proteger las compilaciones de Kubernetes, con hincapié en las tres uniones principales en las que se incorpora la seguridad dentro de la canalización de compilación automatizada: entornos de desarrollo integrados, gestión de la configuración e integración continua.

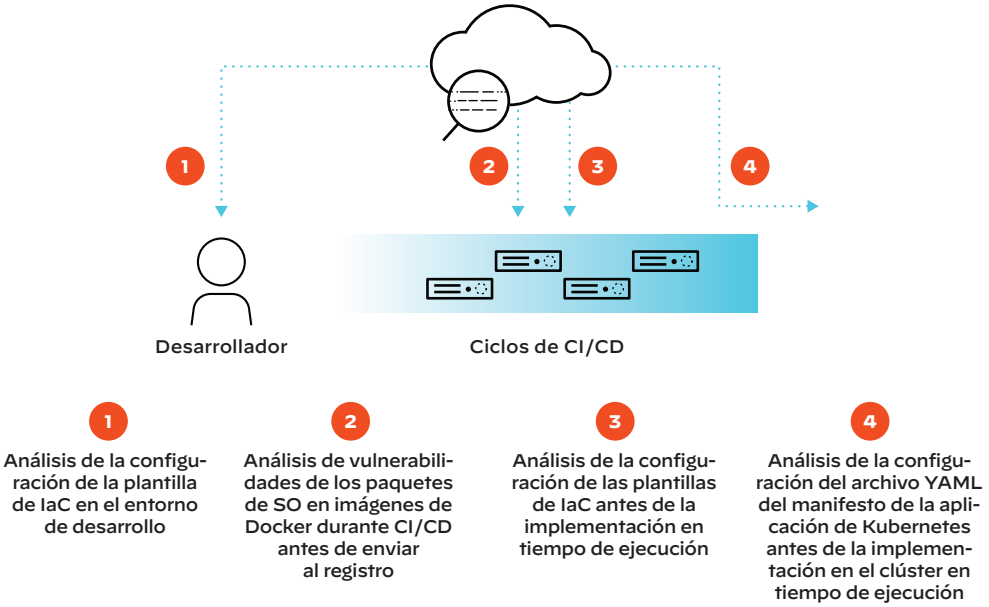

Figura 2: Seguridad en la canalización de compilación

Entornos de desarrollo integrados

Los entornos de desarrollo integrados (IDE, por sus siglas en inglés) son herramientas que usan los desarrolladores para escribir el código fuente de las aplicaciones. Como se trata de la herramienta que inicia la canalización de implementación de aplicaciones, el IDE es el punto de partida en la búsqueda de vulnerabilidades. La mayoría de los IDE se integran con ciertos escáneres de vulnerabilidades de código fuente de terceros que sirven para detectar posibles fallos de seguridad en el código de las aplicaciones.

Integración continua

Las herramientas de integración continua (CI) albergan código fuente y lo convierten en archivos binarios que se pueden implementar en Kubernetes. En este momento también se debe escanear el código en busca de vulnerabilidades. Al igual que los IDE, los servidores de CI son compatibles con distintos escáneres de vulnerabilidades.

Gestión de la configuración

En la actualidad, la mayoría de las canalizaciones de compilación e implementación de aplicaciones para Kubernetes dependen de una gestión de la configuración automatizada y basada en políticas con infraestructura como código (IaC, por sus siglas en inglés) y archivos YAML. Este método permite que los administradores de Kubernetes escriban el código que define la configuración del clúster y la infraestructura que lo alberga y luego aplica ese código de forma automática.

Además de simplificar el proceso de aprovisionamiento de un entorno de Kubernetes, las herramientas de gestión de la configuración permiten escanear los archivos de configuración en busca de problemas de seguridad antes de que se usen. Gracias a herramientas como Prisma™ Cloud, esto se puede hacer de forma automática comparando la IaC y los archivos YAML con aquellos que se sabe que son seguros. Prisma Cloud se integra directamente con el sistema de gestión del código fuente, como GitHub® o GitLab®, de forma que resulta sencillo compilar un proceso plenamente automatizado que proteja los archivos de configuración de Kubernetes dentro de las canalizaciones de compilación existentes.

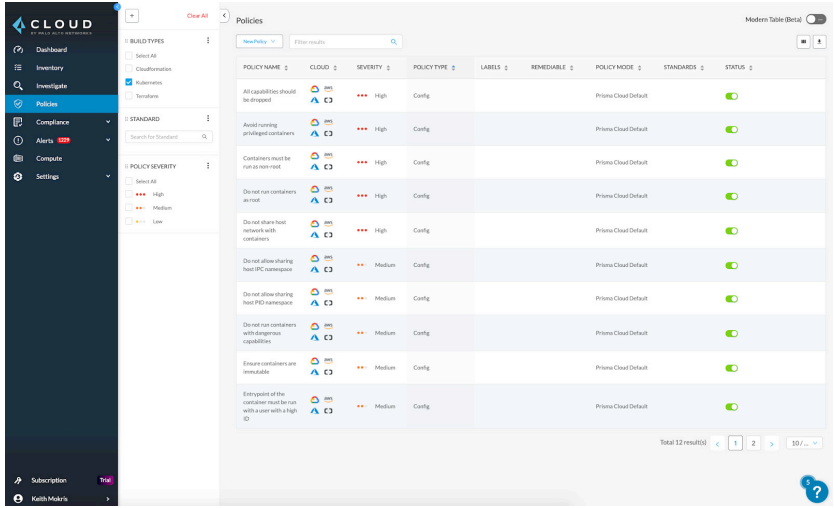

Figura 3: Políticas de Kubernetes en Prisma Cloud

Capítulo 3: Protección de imágenes de contenedor para su ejecución en Kubernetes

En la mayoría de los casos, las aplicaciones se implementan en Kubernetes como imágenes de contenedor. (Es posible gestionar otros tipos de objetos de implementación en Kubernetes, como máquinas virtuales, pero esto es menos frecuente). Las imágenes de contenedor se analizan en busca de vulnerabilidades presentes en el propio código del contenedor, así como en cualquier dependencia ascendente en la que se base la imagen.

Escritorio de desarrollo

Hay dos métodos para realizar el análisis de imágenes de contenedor en busca de problemas de seguridad. En primer lugar, se puede analizar cada imagen de manera manual con herramientas como twistcli. Esto resulta útil cuando queremos realizar una comprobación de seguridad puntual de una imagen.

Cuando se compilan imágenes de contenedor con flujos de trabajo automatizados, es probable que los equipos tengan que integrar análisis de vulnerabilidades y cumplimento normativo.

Integración continua

Cuando las imágenes de contenedor se compilan con flujos de trabajo automatizados en plataformas como Jenkins®, CircleCI® o Azure® DevOps, es probable que los desarrolladores y los equipos de DevOps tengan que integrar análisis de vulnerabilidades y cumplimiento normativo en esos flujos. Las plataformas de seguridad como Prisma Cloud pueden analizar esas imágenes de contenedor e identificar problemas gracias a marcos de trabajo como las directrices del CIS para Docker, además de aplicar normas basadas en los requisitos de la organización o de la aplicación.

Registros de contenedores

Para revisar las imágenes de contenedor de forma automatizada y escalable, no obstante, hay que analizar todas las imágenes presentes en un registro de contenedor de manera periódica. Los registros son repositorios en los que se almacenan imágenes de contenedor. Al incorporar un escáner de vulnerabilidades en el registro, se obtiene total visibilidad de cualquier amenaza que pueda existir en las imágenes de contenedor almacenadas en ese registro.

Una de las dificultades a la hora de analizar los registros de contenedores es que existen distintos tipos. Algunas distribuciones de Kubernetes, como OpenShift® de Red Hat® y los servicios gestionados de Kubernetes que se alojan en nubes públicas, disponen de sus propios registros integrados. Otras permiten que los administradores elijan entre una variedad de registros de terceros.

Esta diversidad de registros y configuraciones hace que sea importante elegir una herramienta de análisis de imágenes de contenedor que se pueda integrar en cualquier tipo de registro. Prisma Cloud brinda esa flexibilidad, ya que ofrece a los administradores una solución integral de análisis de imágenes independientemente de la configuración del clúster de Kubernetes.

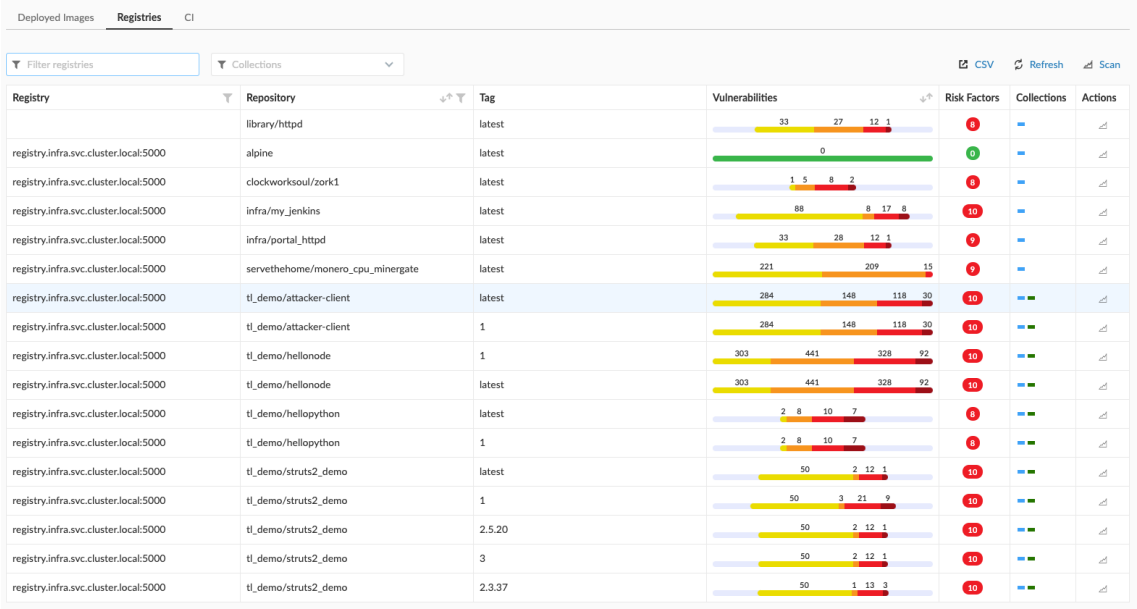

Figura 4: Resultados del análisis del registro de imágenes de contenedor en Prisma Cloud

Capítulo 4: Seguridad del tiempo de ejecución en Kubernetes

El aspecto más complejo de la seguridad de Kubernetes es la protección de las aplicaciones una vez que se han implementado en el clúster. Esto se debe a que hay muchísimos tipos de vulnerabilidades que pueden afectar a la aplicación cuando se está ejecutando, y a que esas vulnerabilidades se pueden explotar desde la propia aplicación y desde Kubernetes.

Para mitigar los riesgos de seguridad una vez implementada la aplicación, se pueden tomar varias medidas (más allá de proteger las aplicaciones antes de su implementación, según los consejos de los capítulos anteriores).

Visibilidad

En primer lugar, resulta fundamental garantizar la visibilidad de los servicios y recursos de Kubernetes en todo momento. Las brechas pueden producirse de diferentes maneras. Cuantos más datos se recopilen sobre el entorno de la aplicación, más probabilidades hay de detectar anomalías que hagan sospechar que se está produciendo una brecha.

Los equipos de seguridad podrían no saber dónde se ejecutan todos sus clústeres, por lo que carecerían de información unificada sobre el estado general de la protección. Con Prisma Cloud, los equipos de seguridad disfrutan de una visibilidad continua de las ubicaciones de los clústeres gracias a los datos de las API de los proveedores de servicios en la nube pública (CPS, por sus siglas en inglés), así como de su estado de cumplimiento normativo y configuración.

Protección del tiempo de ejecución

La recopilación de datos del entorno en Kubernetes es relativamente sencilla. Sin embargo, un desafío importante a la hora de usar esos datos para supervisar el estado de la seguridad es que los clústeres de Kubernetes tienden a cambiar constantemente conforme los nodos se desconectan o apagan, o las aplicaciones se amplían o reducen en función de la demanda, por ejemplo. Por lo tanto, resulta imposible establecer un nivel de actividad «normal» como referencia y usarlo para detectar anomalías.

La alternativa es recurrir a una solución de protección del tiempo de ejecución en Kubernetes como Prisma Cloud, que aprende automáticamente cómo se comportan las aplicaciones implementadas en distintas condiciones. Gracias a esta información, los usuarios pueden distinguir correctamente los cambios normales del comportamiento de las aplicaciones de aquellos que evidencian un problema de seguridad.

Protección de la red

Las amenazas a la seguridad basadas en la red pueden afectar a Kubernetes de dos formas: en redes de cara al público que conectan las aplicaciones con Internet y en redes internas que los contenedores de Kubernetes utilizan para intercambiar datos entre sí.

Por consiguiente, la detección de signos de actividad maliciosa en los dos tipos de red resulta crucial para proteger los recursos de red de Kubernetes. Dado que la actividad de red, como el resto del entorno de Kubernetes, fluctúa constantemente (al igual que las direcciones IP de los contenedores), se requiere un escáner de red que entienda de contenedores y que comprenda las peculiaridades del tráfico de red existente en los clústeres de Kubernetes. Asimismo, se precisa una herramienta cortafuegos que permita definir reglas de protección frente a las amenazas basadas en la red y que emita alertas o bloquee las amenazas automáticamente cuando esas reglas se incumplen. Con el análisis de redes orientado a los contenedores y la función de cortafuegos de capa 4, Prisma Cloud cumple todos estos requisitos.

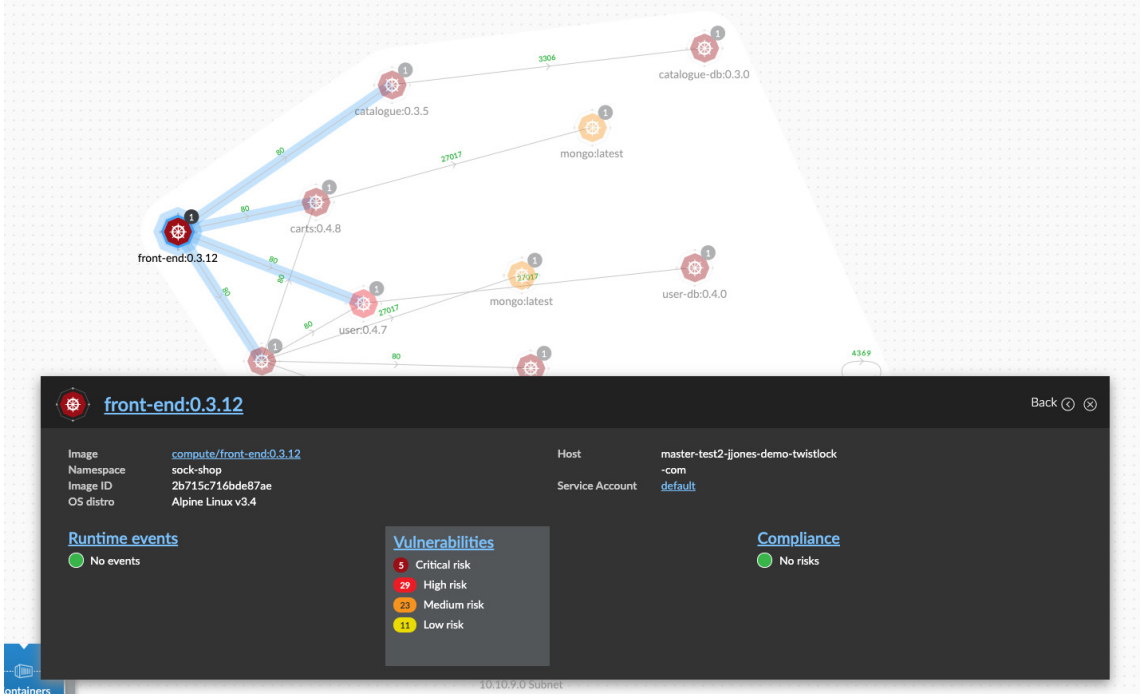

Figura 5: Topología de red y visualización de la seguridad de los contenedores en Prisma Cloud

Supervisión del SO de los nodos

Cualquier atacante que se haga con el control del sistema operativo que se ejecuta en un nodo del clúster de Kubernetes podría causar estragos de todo tipo. Por eso resulta fundamental ampliar la supervisión más allá de los componentes internos de Kubernetes y prestar atención a los sistemas operativos de cada uno de los nodos.

Lo ideal es que esta tarea se pueda realizar con la misma solución que supervisa las aplicaciones de Kubernetes, para que no haya que disponer de distintas herramientas ni supervisar varios paneles de control para detectar las amenazas que se originen en fuentes internas y externas. Prisma Cloud, que puede supervisar cualquier sistema operativo o infraestructura de nube además de Kubernetes, brinda una funcionalidad de supervisión integral.

Cumplimiento normativo y auditorías de seguridad en Kubernetes

Por último, se implementa un proceso de auditoría normal para analizar todas las capas del clúster de Kubernetes y las configuraciones, de modo que se garantice el cumplimiento de los estándares del sector y las prácticas recomendadas. Las auditorías no tienen por qué detectar las amenazas en tiempo real, pero

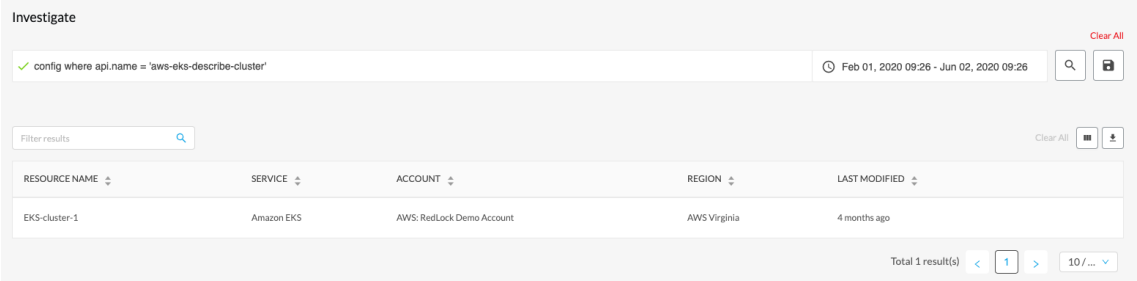

Figura 6: Detalles de la investigación del clúster de Kubernetes en Prisma Cloud

servirán para anticipar problemas de seguridad o errores de configuración que hayan pasado desapercibidos y pudieran abrir la puerta a ataques en el clúster o las aplicaciones.

Prisma Cloud permite realizar auditorías de seguridad de Kubernetes exhaustivas en las que se comprueba si alguno de los componentes del clúster se aleja de los valores de referencia y las prácticas recomendadas establecidos, como las directrices del CIS para Kubernetes. Contiene más de 100 comprobaciones integradas personalizables para las configuraciones, las comunicaciones y otros aspectos, definidas para cada aplicación o entorno.

Después, se pueden comparar esas comprobaciones con las plantillas de cumplimiento prediseñadas para marcos frecuentes como la norma de seguridad de datos para el sector de las tarjetas de pago (PCI DSS, por sus siglas en inglés), la Ley de Transferibilidad y Responsabilidad del Seguro Sanitario (HIPAA, por sus siglas en inglés), el RGPD y el SP 800-190 del Instituto Nacional de Estándares y Tecnología estadounidense (NIST, por sus siglas en inglés). Además, se pueden crear comprobaciones propias, compatibles con tecnologías nativas en la nube como Open Policy Agent y Kubernetes AuditSink. Esto puede resultar útil cuando hay que cumplir normativas específicas de un sector o si se implementa una aplicación de línea de negocio personalizada cuyas necesidades de seguridad no quedan cubiertas por las políticas de auditoría generales.

Conclusión

La seguridad en Kubernetes puede resultar intimidante, sobre todo porque hay muchísimas piezas distintas que proteger. Sin embargo, es posible utilizar una herramienta central diseñada para ocuparse de los distintos aspectos de la seguridad en estos entornos. Las funciones de seguridad nativas de Kubernetes hacen parte del trabajo, pero también resulta esencial contar con herramientas externas que cubran ciertas vulnerabilidades (como el código malicioso en las imágenes de contenedor), auditen las configuraciones de Kubernetes en busca de riesgos para la seguridad y ofrezcan una supervisión constante para detectar las amenazas en tiempo real.

Prisma Cloud ofrece el exhaustivo conjunto de funciones de seguridad que los equipos necesitan proteger todos los aspectos de Kubernetes. Además, se integra de forma nativa con Kubernetes y distintas herramientas asociadas, por lo que resulta muy sencillo incorporar seguridad a los procesos de compilación, implementación y gestión de Kubernetes que ya se utilizan.

Si desea obtener más información sobre cómo usar Prisma Cloud para proteger Kubernetes, visite nuestro sitio web.

Prisma Cloud de Palo Alto Networks

Prisma™ Cloud es la plataforma de seguridad nativa en la nube (CNSP, por sus siglas en inglés) más completa del sector por la cobertura que ofrece. Con ella, conseguirá que las aplicaciones, los datos y todas las tecnologías nativas en la nube cuenten con la debida protección y cumplan la normativa, a lo largo de todo el ciclo de vida de desarrollo y en entornos de nube híbrida o de varias nubes.

La plataforma ofrece un enfoque integrado que ayuda a los equipos de DevOps y de operaciones de seguridad a colaborar con eficacia y a desarrollar aplicaciones nativas en la nube seguras en menos tiempo.

Prisma Cloud se integra con los kits de herramientas y las arquitecturas nativas en la nube que protege, para garantizar una cobertura de seguridad completa al tiempo que acaba con la separación de las operaciones de seguridad en todo el ciclo de vida de las aplicaciones, al tiempo que permite la adopción de las operaciones de desarrollo y seguridad y mejora la capacidad de reacción cuando cambian las necesidades de seguridad de las arquitecturas nativas en la nube.

Para obtener más información, visite paloaltonetworks.es/prisma/cloud.

Oval Tower, De Entrée 99 - 197
1101HE Ámsterdam
Países Bajos

Tel.: +31 20 888 1883

www.paloaltonetworks.es

prisma-cloud-complete-guide-kubernetes-ebook-093020-es