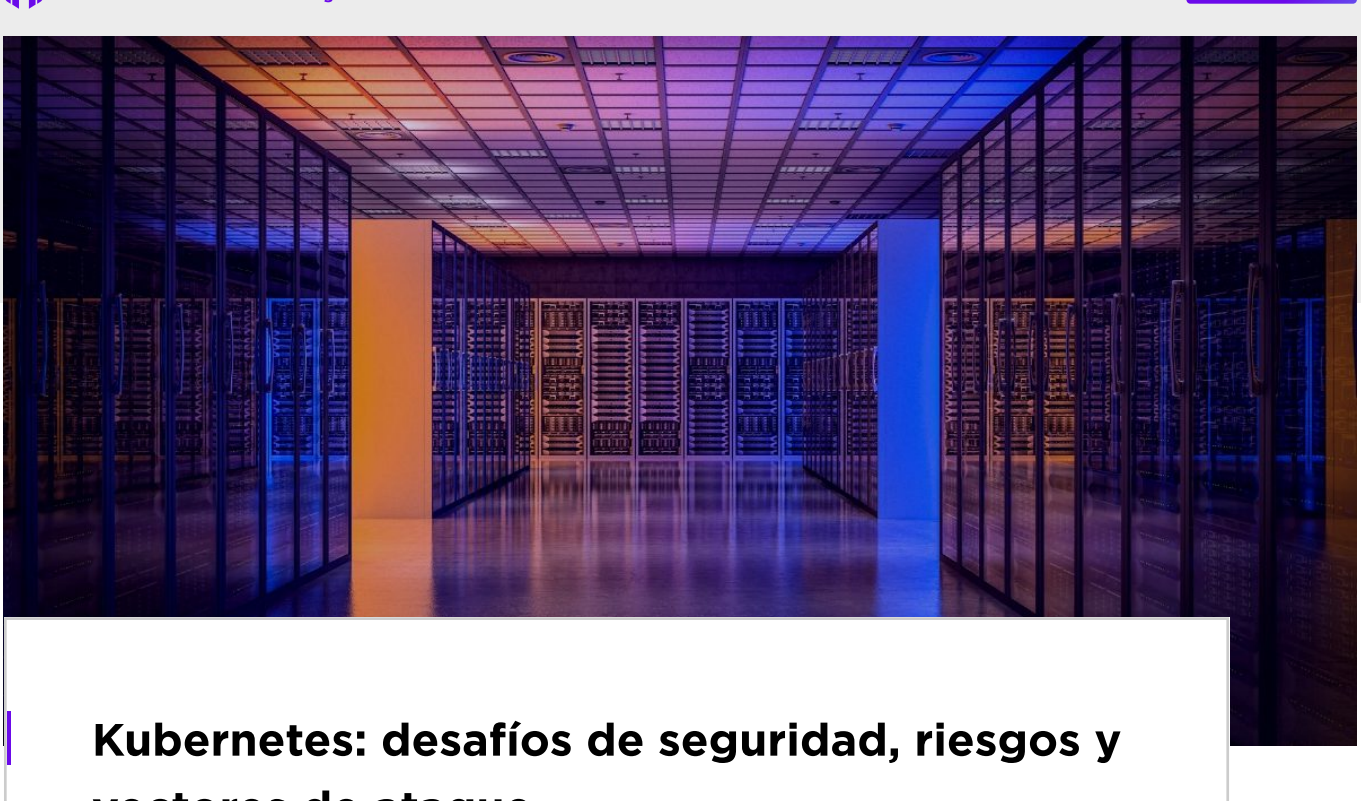


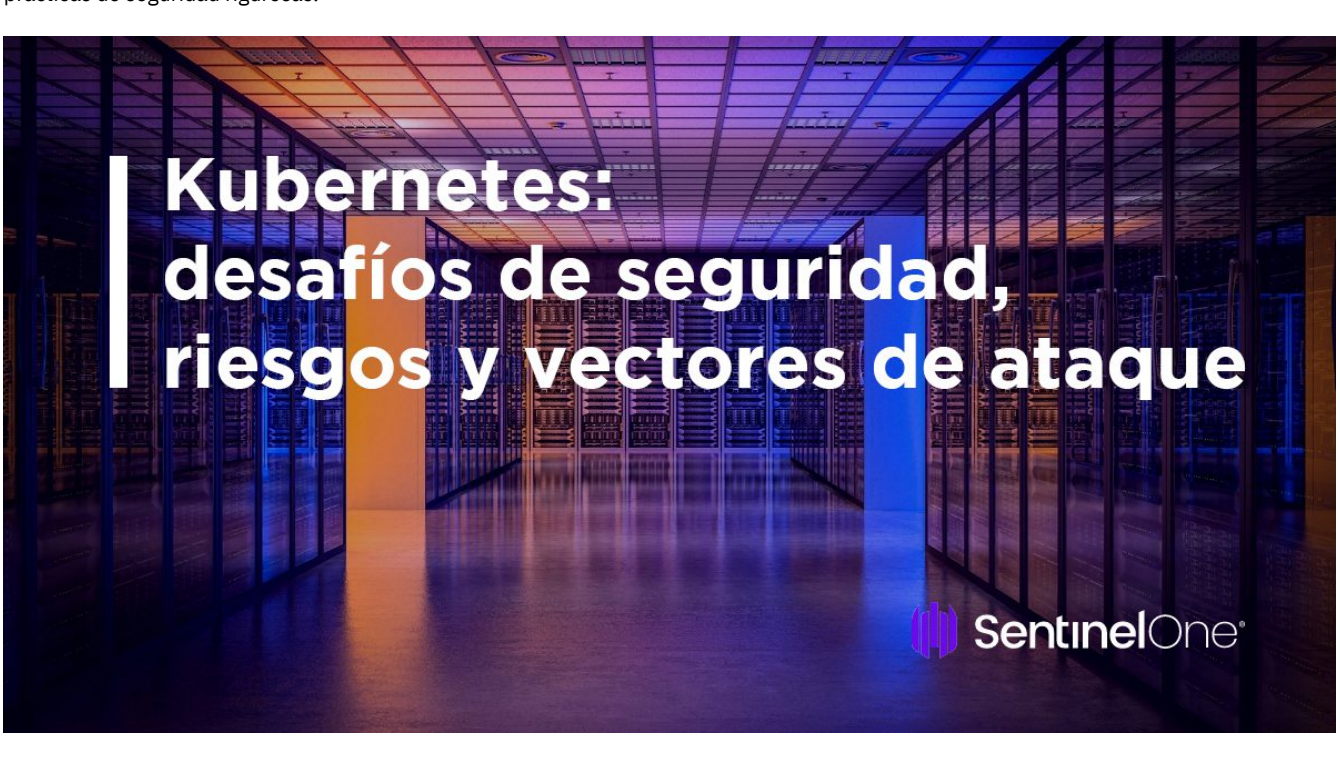
Kubernetes: desafíos de seguridad, riesgos y vectores de ataque

julio 16, 2020
by SentinelOne

La popularidad de los contenedores y de Kubernetes (K8s) ha supuesto un cambio fulgurante en el mundo de las tecnologías de la información. En solo siete años, hemos pasado de una máquina virtual a contenedores y de ahí a una plataforma de orquestación de contenedores (la primera versión de Docker se lanzó en 2013). Mientras que algunas *startups* aún intentan descubrir cómo aprovechar estos nuevos recursos, algunas de las empresas más consolidadas se plantean migrar sus sistemas existentes hacia infraestructuras más eficaces.

Si bien, como demuestra su rápida adopción, los contenedores y Kubernetes han supuesto una revolución, es innegable que estas tecnologías también han generado nuevos problemas de seguridad. Su gran popularidad y la ausencia de medidas de seguridad adecuadas en gran cantidad de empresas han hecho de la contenedorización y de Kubernetes un objetivo perfecto para los ciberdelincuentes.

Un clúster de Kubernetes es un conjunto de máquinas gestionadas por un nodo maestro (y sus réplicas). Puede cubrir millones de máquinas y servicios, y, por consiguiente, convertirse en un vector de ataque fundamental. Es por eso que resulta indispensable adoptar prácticas de seguridad rigurosas.



Protección del clúster

Un clúster de Kubernetes contiene numerosos elementos que deben protegerse adecuadamente. Sin embargo, es imposible garantizar la seguridad de un clúster en un solo proceso. La seguridad de todo el clúster requiere la implementación de una serie de mejores prácticas y de un equipo de seguridad competente.

En este artículo, abordaremos varios vectores de ataque de Kubernetes diferentes, así como las mejores prácticas para garantizar la seguridad de su clúster de Kubernetes.

Actualización de Kubernetes y sus nodos

Kubernetes es un sistema de código abierto que se actualiza continuamente. Su repositorio GitHub es uno de los más activos de la plataforma y, como tal, es objeto de la constante introducción de nuevas funciones, optimizaciones y actualizaciones de seguridad.

Cada cuatro meses, se publica una nueva versión importante de Kubernetes. Cada nueva versión incluye funciones destinadas a mejorar el servicio, pero también puede introducir nuevos problemas de seguridad o errores, algo a lo que todo programa de software es susceptible, sobre todo si se actualiza con frecuencia.

Sin embargo, se pueden encontrar fallos de seguridad incluso en las versiones antiguas. Resulta, por lo tanto, esencial comprender cómo gestiona el equipo de Kubernetes las actualizaciones en las versiones antiguas. A diferencia de las distribuciones de Linux u otras plataformas, Kubernetes no tiene una versión LTS con soporte a largo plazo. En lugar de eso, el sistema Kubernetes intenta incorporar las correcciones de los problemas de seguridad en las tres últimas versiones principales distribuidas.

Por este motivo es esencial mantener su clúster en una de las tres versiones principales más recientes, aplicar regularmente los parches de seguridad y planificar actualizaciones a la última versión principal, al menos cada doce meses.

Además de sus componentes principales, Kubernetes también gestiona los nodos que ejecutan la carga de trabajo asignada al clúster. Estos nodos pueden ser máquinas físicas o virtuales con un sistema operativo instalado. Cada nodo representa un vector de ataque potencial que debe actualizarse para eliminar posibles problemas de seguridad. Los nodos deben, por tanto, estar lo más limpios posible para reducir la superficie de ataque.

Control de acceso de los usuarios

El control de acceso basado en roles es una de las mejores formas de controlar qué usuarios tienen acceso al clúster y cómo. Permite definir un conjunto de permisos pormenorizados para cada usuario. Como las reglas se agregan unas a otras, es necesario definir explícitamente cualquier permiso. Gracias al control de acceso basado en roles, es posible limitar los permisos (visualización, lectura o escritura) para cada objeto de Kubernetes, desde los pods (la unidad de computación más pequeña de Kubernetes) a los espacios de nombres.

El control de acceso basado en roles también puede adjuntarse a otro servicio de directorios mediante tokens de conexión de OpenID. Esto permite centralizar la administración de usuarios y grupos para un uso más amplio dentro de la empresa.

El permiso de acceso no se limita a Kubernetes. A veces, los usuarios pueden necesitar, por ejemplo, acceso a un nodo del clúster para identificar los problemas. En esos casos, es preferible crear usuarios temporales para resolver esos problemas y después eliminarlos.

Mejores prácticas para contenedores

Docker, la principal tecnología de contenedores, está compuesta de capas: la capa interna es la estructura más primitiva, mientras que la capa externa es la más específica. Así, todas las imágenes de Docker empiezan por algún tipo de soporte de distribución o idioma, y cada nueva capa añade o modifica la función anterior hasta la última capa. El contenedor dispone entonces de todo lo que necesita para gestionar la aplicación.

Estas capas (también llamadas imágenes) pueden estar disponibles públicamente en el Docker Hub o a título privado en otro registro de imágenes. La imagen puede expresarse de dos maneras: como un nombre más una etiqueta (por ejemplo, node:latest) o con su identificador SHA inmutable (por ejemplo, sha256:`d64072a554283e64e1bfeb1bb457b7b293b6cd5bb61504afaa3bdd5da2a7bc4b` para la misma imagen en el momento de la redacción de este artículo).

El propietario del repositorio puede modificar en cualquier momento la imagen asociada a la etiqueta; la etiqueta *latest* indica la última versión disponible. También significa que, al crear una nueva imagen o ejecutar una imagen con etiqueta, la capa interior puede cambiarse de repente, sin previo aviso.

Esta estrategia por supuesto plantea algunos problemas: (1) Pierde el control de lo que pasa en su instancia de Kubernetes, ya que puede actualizarse una capa superior y añadir un conflicto, o (2) la imagen puede modificarse de forma intencionada para introducir un fallo de seguridad.

Para evitar el primer problema, evite utilizar la etiqueta *latest*, y opte por una etiqueta más específica de la versión (por ejemplo, node:14.5.0). Para evitar el segundo problema, elija imágenes oficiales, clone la imagen de su repositorio privado o utilice el valor SHA.

Otro enfoque consiste en utilizar una herramienta de detección de vulnerabilidades para analizar continuamente las imágenes utilizadas. Estas herramientas pueden ejecutarse conjuntamente con canalizaciones de integración continua y pueden supervisar el repositorio de imágenes para identificar problemas no detectados previamente.

Al crear una nueva imagen, es importante recordar que cada imagen debe contener un solo servicio. La imagen completa debe crearse de manera que solo tenga las dependencias necesarias para esa aplicación y nada más. De esta forma, se reduce la superficie de ataque a únicamente los componentes esenciales del servicio. El hecho de tener solo una aplicación por imagen facilita igualmente la actualización a una nueva versión y la asignación de recursos en el módulo de orquestación.

Seguridad de red

Las recomendaciones de la sección anterior se centran en la reducción de la superficie de ataque, y el mismo principio se aplica a las redes. Kubernetes contiene redes virtuales dentro del clúster que pueden restringir el acceso entre los pods y controlar el acceso externo de manera que solo sean accesibles los servicios autorizados. Se trata de una solución primitiva que funciona bien en los clústeres pequeños.

Sin embargo, los clústeres de mayor tamaño que contienen varios servicios desarrollados por equipos diferentes son mucho más complejos, y un enfoque centralizado puede ser imposible de administrar. En dicho caso, las mallas de servicios son en la actualidad el mejor método disponible. Las mallas de servicios crean una capa de cifrado de red que permite a los servicios comunicarse entre sí de forma segura. Funcionan generalmente como un agente *sidecar* adosado a cada pod y que garantiza la comunicación entre servicios. El papel de las mallas de servicios no se limita a la seguridad; también permiten el descubrimiento de servicios, la supervisión/rastreo/registro, y la prevención de interrupciones de servicio haciendo el papel de disyuntor, por ejemplo.

Establecimiento de cuotas de recursos

Como las aplicaciones se actualizan constantemente, las medidas para proteger su clúster descritas son por sí solas insuficientes, ya que existe siempre el riesgo de violación de la seguridad.

El uso de cuotas de recursos, mediante las que Kubernetes limita la protección contra interrupción de servicio según las restricciones establecidas, constituye otra medida importante. Si las restricciones están bien diseñadas, impedirán que todos los servicios del clúster dejen de estar disponibles debido a un agotamiento de los recursos.

También pueden evitar tener que pagar una factura de nube astronómica a final de mes.

Supervisión y registro

La supervisión del clúster, desde el clúster a los pods, es fundamental para descubrir las interrupciones y determinar las causas. Se trata de detectar los comportamientos anómalos. Si el tráfico de red aumenta o si el procesador de los nodos se comporta de forma diferente, conviene investigar las causas para descartar cualquier problema. Si bien la supervisión tiene más que ver con métricas como el procesador, la memoria o el ancho de banda, el registro puede proporcionar información adicional (históricos) que puede ayudar a detectar patrones inusuales o a identificar rápidamente el origen del problema.

El uso conjunto de Prometheus y Graphana permite una supervisión eficaz de Kubernetes. Prometheus es una potente base de datos de series temporales, mientras que Graphana es un panel gráfico capaz de leerlos y sintetizarlos en paneles fáciles de ver.

ElasticSearch es otra herramienta muy útil y una de las más populares para el registro centralizado en tiempo casi real de la aplicación, los nodos y la propia plataforma Kubernetes.

Instalación local o instalación en la nube, desde el punto de vista de la seguridad

Kubernetes puede instalarse in situ o utilizar un servicio gestionado en la nube. En el primer caso, cada etapa de configuración (instalación de nuevas máquinas, configuración de red y protección de la aplicación) deben hacerse *manualmente*. Los servicios gestionados, como Google GKE, AWS EKS o Azure AKS, permiten instalar Kubernetes con una configuración mínima y son compatibles con otros servicios de proveedores de nube.

Desde el punto de vista de la seguridad, las soluciones locales requieren mucha más atención. Como se ha dicho anteriormente, el sistema debe descargar y configurar cada nueva actualización, y también deben actualizarse los nodos. Por lo tanto, se recomienda que solo los equipos con experiencia desplieguen Kubernetes in situ.

Con los servicios gestionados en la nube, por otro lado, el proceso es mucho más simple, ya que Kubernetes ya está instalado y el proveedor de la nube mantiene los nodos actualizados con las últimas funciones de seguridad. En lo que respecta al clúster, la mayoría de los proveedores de nube permiten a los usuarios elegir la versión de Kubernetes entre distintas opciones, y también les ofrecen formas de actualizarla con la nueva versión. Así, aunque este método es más simple, también es menos flexible.

Notas finales

A pesar de las actualizaciones constantes y la gran cantidad de nuevas herramientas en el mercado, resulta difícil mantenerse al día y al corriente de las vulnerabilidades. Las violaciones de la seguridad son inevitables. Con Kubernetes, el desafío es todavía mayor, ya que no se trata de una simple herramienta. Kubernetes es, en realidad, un conjunto de herramientas que administran otras herramientas, máquinas y redes, por lo que su seguridad es fundamental.

Pero a la vista de los numerosos elementos en juego, garantizar la seguridad de Kubernetes no es una tarea trivial, así es que asegúrese de seguir las siguientes recomendaciones:

- Realice análisis periódicos sobre las aplicaciones que se ejecutan en Kubernetes para detectar problemas de seguridad.
- Limite el control de acceso.
- Asegúrese de aplicar los últimos parches de seguridad y supervisar continuamente el clúster para corregir inmediatamente las interrupciones limitando así los daños.

El desafío es incluso mayor en el caso de los despliegues locales, en los que hay que gestionar hardware, crear automatismos y mantener actualizadas varias aplicaciones de software. Sin embargo, si sigue las mejores prácticas descritas en este artículo, puede adelantarse a los ciberdelincuentes y garantizar la seguridad y el buen funcionamiento de su entorno de Kubernetes.

La plataforma SentinelOne admite máquinas físicas y virtuales, Docker, los despliegues de Kubernetes autogestionados y los despliegues de Kubernetes gestionados por un proveedor de servicios en la nube, como AWS EKS. Para obtener más información, solicite una demostración gratuita hoy mismo.

¿Te gusta este artículo? Síguenos en LinkedIn, Twitter, YouTube or Facebook para ver el contenido que publicamos.

Lee acerca de Ciberseguridad

- Ciberdelincuencia y ciberseguridad en un mundo pos-COVID-19
- 10 maneras de proteger su Active Directory