Investigadores en ciberseguridad han revelado una vulnerabilidad que afecta a los Servicios de Kubernetes de Microsoft Azure, la cual, si es explotada con éxito, podría permitir a un atacante aumentar sus privilegios y obtener acceso a las credenciales de los servicios utilizados por el clúster.

> *«Un atacante con la capacidad de ejecutar comandos en un Pod que se esté ejecutando dentro de un clúster de Servicios de Kubernetes de Azure afectado podría descargar la configuración utilizada para aprovisionar el nodo del clúster, extraer los tokens de arranque de seguridad de la capa de transporte (TLS) y llevar a cabo un ataque de arranque TLS para acceder a todos los secretos dentro del clúster»,* afirmó Mandiant, una empresa propiedad de Google.

Se ha identificado que los clústeres que utilizan «Azure CNI» para la «Configuración de red» y «Azure» para la «Política de red» son vulnerables a este problema de escalada de privilegios. Microsoft ha solucionado este problema tras haber sido informado de manera responsable.

La técnica de ataque desarrollada por la firma de inteligencia de amenazas se basa en acceder a un componente poco conocido llamado Azure WireServer para solicitar una clave que se usa para cifrar los valores de configuración protegidos («wireserver.key») y usarla para decodificar un script de aprovisionamiento que contiene varios secretos, como:

- KUBELET_CLIENT_CONTENT (Clave TLS genérica del nodo)
- KUBELET_CLIENT_CERT_CONTENT (Certificado TLS genérico del nodo)
- KUBELET_CA_CRT (Certificado CA de Kubernetes)
- TLS_BOOTSTRAP_TOKEN (Token de autenticación de arranque TLS)

> *«El contenido de KUBELET_CLIENT_CONTENT, KUBELET_CLIENT_CERT_CONTENT y KUBELET_CA_CRT se puede decodificar en Base64 y escribir en el disco para usar con la herramienta de línea de comandos de Kubernetes `kubectl` y autenticar al usuario en el clúster»,* señalaron los investigadores Nick McClendon, Daniel

> McNamara y Jacob Paullus.
>
> *«Esta cuenta tiene permisos mínimos en Kubernetes dentro de los clústeres de Servicios de Kubernetes de Azure (AKS) desplegados recientemente, pero puede listar los nodos en el clúster.»*

Por otro lado, el token TLS_BOOTSTRAP_TOKEN podría utilizarse para llevar a cabo un ataque de arranque TLS y, eventualmente, obtener acceso a todos los secretos utilizados por las cargas de trabajo en ejecución. Este ataque no requiere que el pod se ejecute con permisos de root.

> *«Mandiant sugiere adoptar un proceso para crear NetworkPolicies restrictivas que solo permitan acceso a los servicios necesarios, lo que previene este tipo de ataques en su totalidad. La escalada de privilegios a través de un servicio no documentado se evita cuando no se puede acceder a dicho servicio»*, añadió Mandiant.

Esta revelación llega en un momento en que la plataforma de seguridad Kubernetes, ARMO, ha destacado una nueva vulnerabilidad de alta severidad en Kubernetes (CVE-2024-7646, con una puntuación CVSS de 8.8) que afecta al controlador ingress-nginx y que podría permitir a un atacante obtener acceso no autorizado a recursos sensibles del clúster.

> *«La vulnerabilidad surge debido a un fallo en la forma en que ingress-nginx valida las anotaciones en los objetos Ingress»*, explicó el investigador de seguridad Amit Schendel.
>
> *«Este fallo permite a un atacante inyectar contenido malicioso en ciertas anotaciones, eludiendo las comprobaciones de validación previstas. Esto podría dar*

> *lugar a la inyección arbitraria de comandos y al acceso potencial a las credenciales del controlador ingress-nginx, que, en configuraciones por defecto, tiene acceso a todos los secretos del clúster.»*

Este hallazgo también se suma al descubrimiento de una falla de diseño en el proyecto Kubernetes git-sync, que podría permitir la inyección de comandos en Amazon Elastic Kubernetes Service (EKS), Azure Kubernetes Service (AKS), Google Kubernetes Engine (GKE) y Linode.

> *«Esta falla de diseño puede resultar en la exfiltración de datos de cualquier archivo dentro del pod (incluidos los tokens de cuentas de servicio) o en la ejecución de comandos con los privilegios del usuario git_sync. Para explotar esta vulnerabilidad, un atacante solo necesita aplicar un archivo YAML en el clúster, lo cual es una operación que requiere pocos privilegios»*, explicó Tomer Peled, investigador de Akamai.

No se planean parches para corregir esta vulnerabilidad, lo que hace que sea esencial que las organizaciones auditen sus pods de git-sync para identificar qué comandos se están ejecutando.

> *«Ambos vectores se deben a la falta de saneamiento de entradas, lo que subraya la importancia de una defensa sólida en cuanto al saneamiento de entradas de usuario. Los equipos de seguridad deben estar atentos a cualquier comportamiento inusual proveniente del usuario gitsync en sus organizaciones»,* indicó Peled.