Container Security

Docker Container Security

Buenas prácticas:

1. No exponer puertos innecesarios
2. No iniciar contenedores en modo privilegiado
3. Eliminar capacidades al iniciar contenedores
4. Establecer cuotas de recursos
5. Ejecutar procesos como un usuario no root
6. Evitar que los contenedores escalen privilegios
7. Utilizar RO del sistema de archivos
8. Utilizar un gestor de secretos dedicado

3. Eliminar capacidades al iniciar contenedores

- El conjunto predeterminado de capacidades Linux es demasiado permisivo para producción: cambiar UIDs y GIDs de archivos, matar procesos y eludir comprobaciones de permisos de lectura, escritura y ejecución de archivos.

- Las opciones --cap-drop y --cap-add del comando docker run te permiten eliminarlas y otorgarlas.

Ejemplo, elimina todas las capacidades y añade CHOWN:

$ docker run --cap-drop=ALL --cap-add=CHOWN image:tag

4. Establecer cuotas de recursos

- Docker no aplica automáticamente ninguna restricción de recursos a tus contenedores. Los procesos en contenedores pueden usar CPU y memoria ilimitadas, lo que podría afectar a otras aplicaciones en su host.

- Establecer límites para estos recursos ayuda a defenderse contra ataques de denegación de servicio (DoS).

Ejemplo, límite de memoria y CPU

$ docker run -m=128m --cpus=2 imagen:tag

5. Ejecutar procesos como un usuario no root

- Los contenedores se ejecutan por defecto como root. Usa otro usuario para lanzar los procesos y minimizar el riesgo.

Ejemplo,

$ docker run --user=1000 image:tag

6. Evitar que los contenedores escalen privilegios

- Los contenedores normalmente pueden escalar sus privilegios llamando a los binarios setuid y setgid. Esto es un riesgo de seguridad porque el proceso en contenedor puede usar setuid para convertirse en root.

- Para prevenir esto, debes establecer la opción de seguridad no-new-privileges cuando inicies tus contenedores:

Ejemplo,

$ docker run -security-opt=no-new-privileges:true imagen:tag

7. Utilizar RO del sistema de archivos

- Si el contenedor no necesita escribir en el sistema de ficheros, activa el modo de sólo lectura para evitar modificaciones en el sistema de archivos, excepto en las ubicaciones de montaje de volumen designadas.
- Esto bloqueará a un intruso de hacer cambios maliciosos al contenido dentro del contenedor, como por ejemplo reemplazando binarios o archivos de configuración.

Ejemplo,
$ docker run --read-only imagen:tag

8. Utilizar un gestor de secretos dedicado

- Los datos confidenciales requeridos por sus contenedores, como claves API, tokens y certificados, deben almacenarse en una solución de gestión de secretos dedicada.
- Esto reduce el riesgo de exposición accidental que surge cuando se utilizan variables de entorno o archivos de configuración normales.