Guía de Seguridad de las TIC
CCN-STIC 1011

Procedimiento de Empleo Seguro
Prowler 5.13.0



Noviembre 2026

Edita:

CCN
c=ES, l=MADRID, o=CCN,
ou=SELLO ELECTRONICO,
2.5.4.97=VATES-S2800155J,
serialNumber=S2800155J,
cn=CCN
2026.11.04 14:07:01 +01:00

Fecha de Edición: Noviembre de 2026

LIMITACIÓN DE RESPONSABILIDAD
El presente documento se proporciona de acuerdo con los términos en él recogidos, rechazando expresamente cualquier tipo de garantía implícita que se pueda encontrar relacionada. En ningún caso, el Centro Criptológico Nacional puede ser considerado responsable del daño directo, indirecto, fortuito o extraordinario derivado de la utilización de la información y software que se indican incluso cuando se advierta de tal posibilidad.

AVISO LEGAL
Quedan rigurosamente prohibidas, sin la autorización escrita del Centro Criptológico Nacional, bajo las sanciones establecidas en las leyes, la reproducción parcial o total de este documento por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, y la distribución de ejemplares del mismo mediante alquiler o préstamo públicos.

Esta publicación se distribuye bajo licencia CC BY-NC-ND 4.0 que permite compartir el material en cualquier medio o formato con la condición de reconocer adecuadamente la procedencia «Edita: Ministerio de Defensa. Secretaría General Técnica.» No se puede modificar. No se puede utilizar con fines comerciales.

ÍNDICE

1. INTRODUCCIÓN 3
2. OBJETO Y ALCANCE 3
3. ORGANIZACIÓN DEL DOCUMENTO 3
4. FASE PREVIA A LA INSTALACIÓN 4
4.1. ENTREGA SEGURA DEL PRODUCTO 4
4.2. ENTORNO DE INSTALACIÓN SEGURO 4
4.3. REGISTRO Y LICENCIAS 5
4.4. CONSIDERACIONES PREVIAS 5
4.5. COMPONENTES DEL ENTORNO DE OPERACIÓN 5
5. FASE DE INSTALACIÓN 7
6. FASE DE CONFIGURACIÓN 8
6.1. MODO DE OPERACIÓN SEGURO 8
6.2. AUTENTICACIÓN 10
6.3. ADMINISTRACIÓN DEL PRODUCTO 11
6.3.1. ADMINISTRACIÓN LOCAL Y REMOTA 11
6.3.2. CONFIGURACIÓN DE ADMINISTRADORES 11
6.4. CONFIGURACIÓN DE INTERFACES, PUERTOS Y SERVICIOS 12
6.5. GESTIÓN DE CERTIFICADOS 12
6.6. SERVIDORES DE AUTENTICACIÓN 13
6.7. SINCRONIZACIÓN 13
6.8. ACTUALIZACIONES 13
6.9. AUTO-CHEQUEOS 13
6.10. ALTA DISPONIBILIDAD 14
6.11. AUDITORÍA 14
6.11.1. REGISTRO DE EVENTOS 14
6.11.2. ALMACENAMIENTO LOCAL 14
6.11.3. ALMACENAMIENTO REMOTO 14
6.12. BACKUP 14
7. REFERENCIAS 15
8. ABREVIATURAS 16

1. INTRODUCCIÓN

Prowler es una herramienta de seguridad en la nube de código abierto que evalúa la postura de seguridad de entornos cloud sobre distintos proveedores (AWS, Azure, Google Cloud, Kubernetes y otros), ejecutando cientos de comprobaciones automatizadas y asociando los resultados a marcos de cumplimiento reconocidos, entre ellos el Esquema Nacional de Seguridad (ENS), CIS, NIST, ISO 27001, PCI-DSS y SOC 2. El producto evaluado es Prowler Local Server en su modalidad autoalojada, esto es, un despliegue local (on-premise) sobre infraestructura propia del organismo: un despliegue en contenedores formado por una Consola web (UI), una API, procesos de trabajo en segundo plano, una base de datos PostgreSQL y una caché Valkey, orquestados con Docker Compose.

2. OBJETO Y ALCANCE

Esta guía describe el procedimiento de empleo seguro para:

- Producto: Prowler (Prowler Local Server, en despliegue autoalojado local u on-premise).
- Versión: 5.13.0.
- Imágenes de los componentes:
  - API: `prowlercloud/prowler-api`
  - Consola (UI): `prowlercloud/prowler-ui`
- Plataforma: host Linux con Docker Engine y Docker Compose (v2). El producto no exige una distribución de sistema operativo concreta; requiere un host con las herramientas Docker Engine y Docker Compose instaladas y soportadas, según los requisitos descritos en la guía pública de instalación [REF2]. El producto no se ancla a una distribución concreta: se ejecuta en contenedores y es independiente del sistema operativo anfitrión, siempre que este disponga de versiones soportadas de Docker Engine y Docker Compose. Ahora bien, la seguridad del despliegue depende de toda la pila, por lo que el host y el motor de contenedores deben configurarse de forma segura conforme al apartado 4.2.

El producto Prowler ha sido incluido en el catálogo CPSTIC; para conocer el listado, la categoría o nivel y la familia consulte el sitio web del CPSTIC [REF1].

3. ORGANIZACIÓN DEL DOCUMENTO

a) Apartado 4. En este apartado se recogen los aspectos y recomendaciones a considerar antes de proceder a la instalación del producto.

b) Apartado 5. En este apartado se recogen las recomendaciones a tener en cuenta durante la fase de instalación del producto.

c) Apartado 6. En este apartado se recogen las recomendaciones a tener en cuenta durante la fase de configuración del producto, para lograr una configuración segura.

4. FASE PREVIA A LA INSTALACIÓN

4.1. ENTREGA SEGURA DEL PRODUCTO

Prowler se distribuye como imágenes de contenedor publicadas en Docker Hub y como código fuente en el repositorio público de GitHub. La integridad de los artefactos desplegados se verifica mediante el digest de la imagen (huella SHA-256), identificador de confianza de cada imagen. Las imágenes evaluadas y certificadas (arquitectura x86-64) son:

- prowler-api:
  `sha256:0bc85d6a7d453f0915743a1d0d56fc87cca5ba0c3a5bfc9bf9554f8fee3e49c1`
- prowler-ui:
  `sha256:dca3566a311cfa1749ce7a2ffd21a400b1262ea0d82e3b000097277a55383769`

El procedimiento de verificación es el siguiente:

1. Descargar cada imagen por digest, no por etiqueta:

```
docker pull prowlercloud/prowler-api@sha256:0bc85d6a7d453f0915743a1d0d56fc87cca5ba0c3a5bfc9bf9554f8fee3e49c1

docker pull prowlercloud/prowler-ui@sha256:dca3566a311cfa1749ce7a2ffd21a400b1262ea0d82e3b000097277a55383769
```

2. Comprobar que el digest local coincide con el certificado:

```
docker image inspect --format '{{index .RepoDigests 0}}' <imagen>
```

Descargue siempre por digest para garantizar que se despliega exactamente la imagen certificada. La etiqueta de versión `5.13.0` en Docker Hub corresponde a la release certificada; no obstante, el digest es el identificador de confianza y debe verificarse tras la descarga.

Las imágenes se distribuyen actualmente sin firma criptográfica. La descarga por digest garantiza que se despliega exactamente la imagen certificada. La firma de imágenes (por ejemplo, con cosign) queda identificada como mejora futura y no es necesaria para la configuración certificada.

4.2. ENTORNO DE INSTALACIÓN SEGURO

El host que ejecuta Prowler Local Server debe residir en un entorno controlado, con acceso restringido a administradores autorizados. Los componentes de base de datos y caché no deben ser accesibles desde redes no confiables (véase el apartado 6.4). El acceso administrativo al host y a la Consola se limita al personal autorizado.

La seguridad del despliegue no depende únicamente del producto, sino de toda la pila sobre la que se ejecuta. El producto se distribuye en contenedores y no impone una distribución de sistema operativo concreta, por lo que la configuración segura del sistema operativo anfitrión y del motor de contenedores es responsabilidad del organismo y condición necesaria para la operación segura. Con carácter mínimo:

- Sistema operativo anfitrión: aplicar la guía CCN-STIC de seguridad correspondiente a la distribución empleada, manteniendo el sistema actualizado, reduciendo al mínimo los servicios y paquetes instalados y restringiendo el acceso administrativo al personal autorizado.
- Motor de contenedores: aplicar la guía CCN-STIC de seguridad en entornos de contenedores. En particular, no exponer el socket del demonio Docker por red, restringir la pertenencia al grupo docker al personal autorizado, y desplegar únicamente imágenes verificadas por digest conforme al apartado 4.1.
- Actualizaciones: mantener actualizados el sistema operativo anfitrión y el motor de contenedores conforme al apartado 6.8.
- Acceso y aislamiento: el host debe residir en un entorno controlado y los servicios de base de datos y caché no deben ser accesibles desde redes no confiables, conforme al apartado 6.4.

Las guías CCN-STIC aplicables al sistema operativo anfitrión y a los entornos de contenedores se encuentran publicadas por el Centro Criptológico Nacional [REF9].

4.3. REGISTRO Y LICENCIAS

Prowler es software de código abierto y no requiere registro de licencia para Prowler Local Server autoalojado. La instalación está documentada públicamente [REF2].

4.4. CONSIDERACIONES PREVIAS

Antes de la instalación, el administrador debe:

- Decidir la topología de despliegue y preparar un proxy inverso que termine TLS por delante de la Consola y la API (véanse los apartados 4.5 y 6.1).
- Generar los valores secretos de la configuración segura (véase el apartado 6.1): el material de clave de firma de tokens, los secretos de la aplicación y las credenciales de la base de datos.
- Decidir el modelo de autenticación: credenciales locales con la política de contraseñas exigida, o delegación en un proveedor de identidad (IdP) SAML externo (véanse los apartados 6.2 y 6.6).

4.5. COMPONENTES DEL ENTORNO DE OPERACIÓN

La operación segura del producto se apoya en los siguientes componentes:

- Proxy inverso: termina TLS y expone el extremo HTTPS externo, reenviando el tráfico a la Consola y la API (véanse los apartados 6.1 y 6.5).
- Proveedor de identidad (IdP) SAML externo (opcional): cuando se requiere autenticación y política de sesión centralizadas (véase el apartado 6.6).
- Fuente de tiempo (NTP): proporcionada por el sistema operativo del host (véase el apartado 6.7).
- Colector de registros (opcional): un destino syslog remoto para el almacenamiento centralizado de auditoría (véase el apartado 6.11.3).

La figura siguiente resume los componentes del producto, su ubicación respecto del host y del motor de contenedores, y los elementos del entorno de operación con los que interactúa.

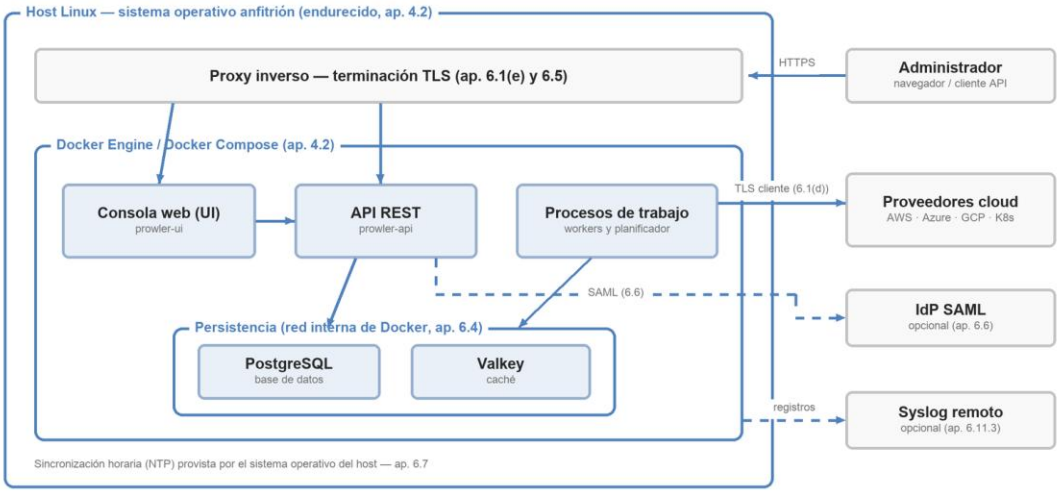


Figura 1: Arquitectura de componentes y entorno de operación de Prowler Local Server.

5. FASE DE INSTALACIÓN

Prowler Local Server se instala siguiendo la guía pública de instalación [REF2]. La instalación usa Docker Compose para levantar la Consola, la API, los procesos de trabajo, PostgreSQL y Valkey. Los parámetros de seguridad relevantes se describen en el apartado 6. El host no requiere software base adicional más allá de Docker Engine y Docker Compose.

Como prerrequisito, el host debe disponer de una versión soportada de Docker Engine y de Docker Compose (v2), y encontrarse configurado de forma segura conforme al apartado 4.2 antes de desplegar el producto. La instalación no requiere una distribución de sistema operativo concreta; los requisitos concretos de plataforma y las versiones soportadas se detallan en la guía pública de instalación [REF2].

6. FASE DE CONFIGURACIÓN

6.1. MODO DE OPERACIÓN SEGURO

La configuración segura certificada se alcanza aplicando los siguientes pasos a una instalación por defecto. Estos pasos resuelven los hallazgos identificados durante la evaluación y alinean el despliegue con la guía CCN-STIC 807 [REF6].

a) Clave dinámica de firma de tokens. El producto firma y verifica los tokens de sesión con un par de claves RSA. En la versión certificada, la instalación elimina el par de claves por defecto y genera un par único por instalación, de modo que la clave de firma nunca se comparte entre despliegues [REF2]. Para rotar la clave, elimine los valores `DJANGO_TOKEN_SIGNING_KEY` y `DJANGO_TOKEN_VERIFYING_KEY` y reinicie la API, que generará un nuevo par.

b) Modificar los valores de autenticación por defecto. El fichero `.env` se distribuye con valores por defecto que son públicos en la distribución de código abierto y deben cambiarse antes del primer arranque:

- Generar dos valores aleatorios independientes y asignarlos a `AUTH_SECRET` y `DJANGO_SECRETS_ENCRYPTION_KEY`:

```
openssl rand -base64 32
```

- Cambiar `POSTGRES_ADMIN_USER`, `POSTGRES_ADMIN_PASSWORD`, `POSTGRES_USER` y `POSTGRES_PASSWORD`. Las credenciales deben tener al menos 12 caracteres y combinar mayúsculas, minúsculas, dígitos y símbolos.

c) Limitar los puertos expuestos. La base de datos y la caché solo se comunican internamente por la red de Docker. Comente las declaraciones `ports` de los servicios `postgres` y `valkey` en el fichero `docker-compose.yml` (véase el apartado 6.4).

d) Endurecimiento TLS de la pila cliente. Es la indicación exigida explícitamente para el PES en el anexo de cualificación del expediente PEN-2025-071: la correcta configuración de las ciphersuites descritas en el documento Prowler TLS Hardening [REF7]. Prowler actúa como cliente TLS hacia los extremos de la nube a través de la pila estándar de Python (OpenSSL 3.0). Su selección de mecanismos por defecto se restringe en tiempo de ejecución, sin modificar la imagen evaluada, creando un directorio local `hardening/` con los dos ficheros siguientes, que se montan en el contenedor.

Fichero `openssl.cnf` (elimina el grupo ffdhe2048 de la extensión supported_groups y fija las suites de TLS 1.3):

```
openssl_conf = openssl_init


[openssl_init]
ssl_conf = ssl_configuration


[ssl_configuration]
system_default = tls_system_default


[tls_system_default]
```

```
MinProtocol = TLSv1.2
Groups = x25519:prime256v1:secp384r1
Ciphersuites =
TLS_AES_128_GCM_SHA256:TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256
```

Fichero `sitecustomize.py` (restringe las suites de TLS 1.2 a ECDHE con AES-GCM y ChaCha20-Poly1305):

```
import ssl

_ALLOWED_TLS12_CIPHERS = (
    "ECDHE-ECDSA-AES128-GCM-SHA256:"
    "ECDHE-RSA-AES128-GCM-SHA256:"
    "ECDHE-ECDSA-AES256-GCM-SHA384:"
    "ECDHE-RSA-AES256-GCM-SHA384:"
    "ECDHE-ECDSA-CHACHA20-POLY1305:"
    "ECDHE-RSA-CHACHA20-POLY1305"
)

_orig_new = ssl.SSLContext.__new__


def _hardened_new(cls, *args, **kwargs):
    ctx = _orig_new(cls, *args, **kwargs)
    try:
        ctx.set_ciphers(_ALLOWED_TLS12_CIPHERS)
    except (ssl.SSLError, ValueError):
        pass
    return ctx


ssl.SSLContext.__new__ = staticmethod(_hardened_new)
```

Son necesarios ambos ficheros. El intérprete de Python aplica su propia lista de suites de TLS 1.2 después de cargar la configuración de OpenSSL, por lo que las directivas Groups, Ciphersuites y MinProtocol de `openssl.cnf` son efectivas, pero la restricción de las suites de TLS 1.2 requiere `sitecustomize.py`. Este fichero utiliza el mecanismo estándar del intérprete para aplicar configuración en el arranque (lo carga automáticamente el módulo site), por lo que no supone modificación del producto ni de la imagen evaluada.

Aplicación: monte el directorio `hardening/` en modo solo lectura en los contenedores que ejecutan el motor de Prowler (los basados en la imagen prowler-api) y establezca en ellos las variables de entorno `OPENSSL_CONF=/hardening/openssl.cnf` y `PYTHONPATH=/hardening`. La configuración es reversible retirando ambas variables.

Con esta configuración, la oferta del cliente en la negociación TLS queda restringida a los siguientes mecanismos:

- TLS 1.3: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256.
- TLS 1.2: `TLS_ECDHE_{ECDSA,RSA}_WITH_AES_{128,256}_GCM_SHA{256,384}` y `TLS_ECDHE_{ECDSA,RSA}_WITH_CHACHA20_POLY1305_SHA256`.
- Grupos de intercambio de claves: x25519, secp256r1 (prime256v1, P-256) y secp384r1 (P-384).

Verificación: las suites que ofrece el cliente pueden comprobarse ejecutando, sobre la imagen certificada (véase el apartado 4.1) con la configuración aplicada:

```
docker run --rm \
  -v ./hardening:/hardening:ro \
  -e OPENSSL_CONF=/hardening/openssl.cnf \
  -e PYTHONPATH=/hardening \
  --entrypoint python3 prowlercloud/prowler-api:5.13.0 \
  -c "import ssl; [print(c['name']) for c in
ssl.create_default_context().get_ciphers()]"
```

e) Proxy inverso para la exposición externa. En la configuración por defecto, la Consola y la API se publican en extremos HTTP locales (http://localhost:3000 y http://localhost:8080) y el producto no realiza terminación TLS nativa sobre ellos [REF3]. Mientras el acceso se limite al propio host, este modelo es aceptable. Cuando la Consola o la API deban ser accesibles desde otros equipos, el acceso debe realizarse por HTTPS; para ello se despliega un proxy inverso que termina TLS y reenvía el tráfico internamente a cada servicio. Configure el proxy inverso con mecanismos criptográficos conformes a la guía CCN-STIC 807 (véanse los apartados 6.3.1 y 6.5).

f) Configuración segura del entorno anfitrión. La configuración segura del producto presupone un sistema operativo anfitrión y un motor de contenedores igualmente endurecidos. Antes de dar por alcanzado el modo de operación seguro, verifique que se han aplicado las medidas del apartado 4.2 sobre el host y sobre Docker, y que ambos se mantienen actualizados conforme al apartado 6.8.

6.2. AUTENTICACIÓN

- Autenticación de usuarios: credenciales locales sujetas a la política de contraseñas exigida (véase el apartado 6.3.2), o delegación en un proveedor de identidad (IdP) SAML externo (véase el apartado 6.6). También se admite el inicio de sesión social (Google, GitHub).
- Tokens de sesión: el producto emite JSON Web Tokens firmados con RSA (RS256), usando el par de claves por instalación descrito en el apartado 6.1(a). El token de acceso caduca a los 30 minutos por defecto (`DJANGO_ACCESS_TOKEN_LIFETIME`) y el de refresco a los 1440 minutos, 24 horas, por defecto (`DJANGO_REFRESH_TOKEN_LIFETIME`), ambos configurables por variable de entorno. Los tokens de refresco se rotan y pueden revocarse.

- Autenticación multifactor (MFA): el producto no proporciona MFA de forma nativa. Al delegar la autenticación en un IdP SAML externo, el MFA se exige en el IdP. Se recomienda exigir MFA a todos los usuarios, en especial a administradores y cuentas privilegiadas.
- Autenticación entre componentes y con sistemas externos: los componentes se comunican por la red interna de Docker; el acceso a la base de datos y a la caché usa las credenciales establecidas en el apartado 6.1(b). La comunicación con los proveedores de nube usa las credenciales configuradas por el operador.

No se emplean claves precompartidas para la administración interactiva. El acceso externo se protege con TLS según los apartados 6.1(e) y 6.5.

6.3. ADMINISTRACIÓN DEL PRODUCTO

6.3.1. ADMINISTRACIÓN LOCAL Y REMOTA

El producto se administra de forma remota mediante la Consola web (GUI) y la API REST. No existe una consola de administración local independiente más allá de la gestión de contenedores a nivel de host.

El producto no termina TLS de forma nativa en los puertos de la Consola y la API; la terminación TLS se realiza en el proxy inverso (véase el apartado 6.1(e)). El acceso administrativo debe realizarse únicamente por HTTPS, con TLS 1.2 o superior. No deben exponerse HTTP en claro, Telnet, FTP ni versiones heredadas de TLS. La documentación pública describe cómo configurar el acceso seguro [REF2] [REF3].

Los algoritmos criptográficos del acceso administrativo deben cumplir la guía CCN-STIC 807. La configuración certificada emplea:

- TLS 1.2 como mínimo, TLS 1.3 preferente.
- Claves de firma y de certificado: RSA de al menos 3072 bits, o ECDSA con curva P-256 o superior.
- Funciones hash: SHA-256 o superior.
- Cifrado simétrico: AES-128 o superior (AES-GCM) y ChaCha20-Poly1305.
- Intercambio de claves: ECDHE sobre P-256, P-384 o x25519.

6.3.2. CONFIGURACIÓN DE ADMINISTRADORES

En este apartado se recogen los aspectos relativos a la configuración de administradores: roles y privilegios, política de contraseñas, protección frente a ataques de fuerza bruta, parámetros de sesión y controles que deben delegarse en el proveedor de identidad externo.

Roles

El producto implementa control de acceso basado en roles. Cada rol porta permisos explícitos (gestionar usuarios, cuenta, proveedores, integraciones, escaneos y facturación), con un alcance ilimitado, limitado o nulo. Asigne a cada administrador el rol mínimo necesario. La gestión de roles está documentada públicamente [REF5].

Política de contraseñas

La versión certificada exige, para las credenciales locales, una longitud mínima de 12 caracteres y la presencia de al menos una mayúscula, una minúscula, un dígito y un carácter especial.

Protección frente a fuerza bruta

La versión certificada aplica limitación de tasa en el extremo de autenticación, configurable por variable de entorno, con un valor por defecto de 50 peticiones por minuto, lo que hace inviable la adivinación de contraseñas en línea.

Parámetros de sesión

Los tokens de sesión tienen vigencia acotada (30 minutos el de acceso y 24 horas el de refresco por defecto, véase el apartado 6.2), y los de refresco se rotan y pueden revocarse.

El producto no implementa de forma nativa el historial de contraseñas, la caducidad, la edad mínima, el bloqueo de cuenta (distinto de la limitación de tasa), el cierre de sesión por inactividad ni el aviso legal de acceso. Cuando la autenticación se delega en un proveedor de identidad externo mediante SAML, estos controles deben configurarse en dicho proveedor, o mediante procedimiento organizativo si este no los soporta, con los siguientes valores mínimos:

- Historial de contraseñas: no reutilización de las últimas cinco (5) contraseñas.
- Caducidad de contraseñas: sesenta (60) días.
- Edad mínima de la contraseña: diez (10) días.
- Cierre de sesión por inactividad: cinco (5) minutos.
- Bloqueo de cuenta: tres (3) intentos fallidos en un periodo de cinco (5) minutos.
- Aviso legal de acceso: mostrado antes de la autenticación, con contenido aprobado por la organización.

6.4. CONFIGURACIÓN DE INTERFACES, PUERTOS Y SERVICIOS

Restrinja los puertos expuestos al mínimo. Los servicios PostgreSQL y Valkey solo necesitan comunicarse internamente por la red de Docker. En el fichero `docker-compose.yml`, comente las declaraciones `ports` de los servicios `postgres` y `valkey` y guarde el fichero. Solo el proxy inverso debe exponer una interfaz externa. El fichero docker-compose.yml de referencia, con los servicios que levanta y los puertos que publica cada uno, se describe en la guía pública de instalación [REF2]; la sintaxis de la declaración ports está documentada en la referencia oficial del formato Compose [REF8].

6.5. GESTIÓN DE CERTIFICADOS

La terminación TLS y la gestión de certificados se realizan en el proxy inverso situado por delante del producto. Utilice certificados y claves de servidor conformes a la guía CCN-STIC 807:

- RSA con claves de, al menos, 3072 bits de longitud.
- ECDSA con curvas P-256 o superior.
- Funciones hash SHA-256 o superior para las firmas.

No deben usarse certificados autofirmados; utilice la PKI de la organización. Siga la documentación del proxy inverso para generar la solicitud de certificado (CSR), importar el certificado de servidor e importar los certificados de las CA.

6.6. SERVIDORES DE AUTENTICACIÓN

El producto admite autenticación externa mediante SAML, además de proveedores de inicio de sesión social (Google, GitHub). Al usar un IdP SAML externo, siga la guía pública de configuración [REF4] y exija en el IdP la política de contraseñas, sesión y bloqueo de la organización. Use TLS 1.2 o superior para la comunicación con el IdP.

6.7. SINCRONIZACIÓN

La sincronización horaria la proporciona el sistema operativo del host. Configure un servicio NTP de confianza en el host para que todos los contenedores hereden un reloj correcto y sincronizado, siguiendo la documentación de NTP del sistema operativo del host. Active la autenticación NTP donde esté disponible.

6.8. ACTUALIZACIONES

Una actualización debe aplicar de forma conjunta las imágenes de contenedor, el fichero `docker-compose.yml` y el fichero `.env`. Las versiones nuevas pueden incorporar servicios y variables de configuración adicionales, por lo que las imágenes de una versión no son compatibles con el `docker-compose.yml` ni el `.env` de una versión anterior.

Procedimiento: obtenga la nueva versión completa (imágenes por digest según el apartado 4.1, junto con su `docker-compose.yml` y su `.env` de referencia), traslade al nuevo `.env` los valores de seguridad propios del despliegue (véase el apartado 6.1(b): claves, secretos y credenciales) y reinicie el despliegue. La API aplica las migraciones de esquema de base de datos en el arranque. Siga la documentación pública de actualización [REF2] y verifique la integridad de las imágenes según el apartado 4.1.

El fabricante publica las nuevas versiones de Prowler a través de sus canales oficiales (repositorio público y notas de versión). La versión evaluada y objeto de esta guía es la 5.13.0, conforme a lo indicado en el apartado 2.

Además de los componentes del producto, deben mantenerse actualizados los componentes subyacentes del entorno de operación: el sistema operativo del host y el motor de contenedores (Docker Engine y Docker Compose). Las actualizaciones de seguridad publicadas por sus respectivos proveedores deben aplicarse conforme a la política de mantenimiento de la organización, verificando tras cada actualización que el despliegue conserva la configuración segura descrita en el apartado 6.

6.9. AUTO-CHEQUEOS

El producto es una aplicación software y no implementa auto-chequeos criptográficos en tiempo de ejecución, al no ser un módulo criptográfico de tipo FIPS con pruebas de arranque. La integridad del software se garantiza en la entrega mediante la verificación del digest de imagen (véase el apartado 4.1). Las operaciones criptográficas las proporcionan bibliotecas de plataforma contrastadas (OpenSSL 3.0 a través de la biblioteca estándar de Python y el paquete cryptography). En tiempo de ejecución no se realiza ninguna modificación de la imagen evaluada.

6.10. ALTA DISPONIBILIDAD

La alta disponibilidad no es una característica intrínseca del producto, sino una decisión de topología de despliegue (por ejemplo, varios contenedores de Consola y API tras un balanceador, con replicación de base de datos). Para el despliegue certificado de host único no aplica. Cuando una organización la requiera, se implementa en la capa de infraestructura sin alterar la configuración segura del producto.

6.11. AUDITORÍA

6.11.1. REGISTRO DE EVENTOS

La Consola, la API y los procesos de trabajo (workers y planificador) emiten registros de operación por la salida estándar, que captura el motor de contenedores. Estos registros comprenden la actividad de la API (peticiones y errores), la ejecución de las tareas de escaneo en los workers (inicio, finalización y errores) y los eventos del marco de aplicación. Cada registro incluye marca de tiempo, módulo o logger de origen, nivel de severidad y contexto de la petición o de la tarea cuando procede, y pueden consultarse mediante el siguiente comando:

```
docker compose logs
```

El producto no dispone de un subsistema de auditoría de acciones de usuario independiente de estos registros; para un producto de la familia CG.1 este nivel de registro es suficiente.

6.11.2. ALMACENAMIENTO LOCAL

Los registros de contenedor se almacenan localmente mediante el driver de logging de Docker en el host. El producto no fija un valor por defecto de rotación ni de retención; ambos deben configurarse en el driver de logging del host conforme a la política de la organización.

6.11.3. ALMACENAMIENTO REMOTO

Para el almacenamiento centralizado de auditoría, configure el driver de logging de Docker de forma que reenvíe los registros de contenedor a un destino syslog remoto sobre un transporte seguro (TLS), siguiendo la documentación de logging del motor de contenedores del host.

6.12. BACKUP

El producto persiste estado en la base de datos PostgreSQL (y en el almacén de grafo, cuando se usa) y en la configuración `.env`. Realice copia de seguridad de la base de datos PostgreSQL con herramientas estándar (por ejemplo, un volcado lógico programado), proteja las copias con cifrado en reposo y almacene de forma segura el fichero `.env` y el material de claves, siguiendo la documentación de copias de seguridad de PostgreSQL. En los despliegues autoalojados las copias de seguridad son responsabilidad del operador; realícelas periódicamente.

7. REFERENCIAS

A lo largo de esta guía se ha hecho referencia a la documentación siguiente:

REF1 Catálogo de Productos y Servicios STIC (CPSTIC). Centro Criptológico Nacional.
https://cpstic.ccn.cni.es/es/catalogo-productos-servicios-stic

REF2 Guía de instalación de Prowler Local Server.
https://docs.prowler.com/getting-started/installation/prowler-app

REF3 Uso básico de Prowler Local Server.
https://docs.prowler.com/getting-started/basic-usage/prowler-app

REF4 Configuración SSO / SAML de Prowler Local Server.
https://docs.prowler.com/user-guide/tutorials/prowler-app-sso
Ejemplo con Google Workspace:
https://docs.prowler.com/user-guide/tutorials/prowler-app-sso-google-workspace

REF5 Control de acceso basado en roles (RBAC) de Prowler Local Server.
https://docs.prowler.com/user-guide/tutorials/prowler-app-rbac

REF6 CCN-STIC 807, Criptología de empleo en el Esquema Nacional de Seguridad.

REF7 Prowler TLS Hardening (restricción de ciphersuites y grupos de intercambio de claves a nivel de contenedor).

REF8 Referencia del formato Compose de Docker (declaración de puertos en servicios).
https://docs.docker.com/reference/compose-file/services/#ports

REF9 Guías CCN-STIC de seguridad para sistemas operativos y para entornos de contenedores. Centro Criptológico Nacional.

8. ABREVIATURAS

API	Interfaz de Programación de Aplicaciones (Application Programming Interface).
CPSTIC	Catálogo de Productos y Servicios de Seguridad de las Tecnologías de la Información y la Comunicación.
CSR	Solicitud de Firma de Certificado (Certificate Signing Request).
ENS	Esquema Nacional de Seguridad.
GUI	Interfaz Gráfica de Usuario (Graphical User Interface).
IdP	Proveedor de Identidad (Identity Provider).
JWT	JSON Web Token.
MFA	Autenticación Multifactor (Multi-Factor Authentication).
NTP	Protocolo de Tiempo de Red (Network Time Protocol).
PES	Procedimiento de Empleo Seguro.
PKI	Infraestructura de Clave Pública (Public Key Infrastructure).
RBAC	Control de Acceso Basado en Roles (Role-Based Access Control).
SAML	Lenguaje de Marcado para Confirmaciones de Seguridad (Security Assertion Markup Language).
TIC	Tecnologías de la Información y la Comunicación.
TLS	Seguridad de la Capa de Transporte (Transport Layer Security).
TOE	Objetivo de Evaluación (Target of Evaluation).