**Gitleaks** es una herramienta de código abierto utilizada para detectar **secretos y credenciales expuestos en repositorios de Git**. Esta herramienta es extremadamente útil en la **seguridad de la información**, especialmente para desarrolladores, equipos DevOps y expertos en ciberseguridad que quieren asegurarse de que su código y repositorios no contienen información sensible que pueda ser explotada por atacantes. Aquí te explico cómo puede ayudar en diferentes escenarios:

**1. Detección de secretos expuestos**

Gitleaks escanea el historial de commits en un repositorio de Git (local o remoto) buscando patrones de información sensible como:

* Claves API
* Tokens de acceso
* Credenciales de bases de datos
* Contraseñas
* Certificados privados

Esto es crucial porque los desarrolladores a veces accidentalmente incluyen estos secretos en el código y los suben al repositorio, lo que puede exponer a la empresa o proyecto a vulnerabilidades serias.

**2. Automatización en pipelines de CI/CD**

Gitleaks puede integrarse en **pipelines de CI/CD** (Integración Continua y Entrega Continua) para escanear cada commit o push antes de que se suba al repositorio. Esto garantiza que ningún código con secretos comprometidos llegue al repositorio principal.

**3. Auditorías de seguridad**

Si estás realizando una **auditoría de seguridad** o un análisis de vulnerabilidades, Gitleaks te permite verificar si en algún momento de la historia del proyecto se subieron credenciales o información sensible. Incluso si se ha eliminado después, el historial de Git conserva la información, por lo que aún podría estar accesible si alguien lo busca.

**4. Explotación en ataques**

Desde el punto de vista de la explotación, si un atacante tiene acceso a un repositorio, **Gitleaks** puede ser utilizada para escanear rápidamente y encontrar posibles credenciales filtradas. Un secreto expuesto podría darle acceso a servicios críticos como bases de datos, almacenamiento en la nube o infraestructuras.

**5. Cumplimiento normativo**

Para las empresas que deben cumplir con normativas como **GDPR**, **HIPAA** o **PCI-DSS**, es importante asegurarse de que no se exponen datos sensibles en los repositorios. Gitleaks ayuda a garantizar que los repositorios están limpios de información privada o confidencial, apoyando en las auditorías de cumplimiento.

**6. Prevención proactiva**

Aparte de detectar vulnerabilidades existentes, Gitleaks ayuda a los equipos de desarrollo y operaciones a ser proactivos y evitar incidentes futuros, asegurando que no suban accidentalmente información sensible al repositorio.

Comandos disponibles:

complete Genera el script de autocompletado para el shell especificado

detect detecta secretos en el código

help Ayuda sobre cualquier comando

protect protege secretos en el código

version muestra la versión de gitleaks

Banderas:

-b, --baseline-path cadena ruta a la línea base con problemas que se pueden ignorar

-c, --config cadena ruta del archivo de configuración

orden de precedencia:

1. --config/-c

2. env var GITLEAKS\_CONFIG

3. (--source/-s)/.gitleaks.toml

Si no se utiliza ninguna de las tres opciones, gitleaks utilizará la configuración predeterminada

--exit-code int código de salida cuando se han encontrado fugas (predeterminado 1)

-h, --help ayuda para gitleaks

-l, --log-level cadena nivel de registro (trace, debug, info, warn, error, fatal) (predeterminado "info")

--max-target-megabytes int los archivos más grandes que este se omitirá

--no-banner suprimir el banner

--redact eliminar secretos de los registros y la salida estándar

-f, --report-format cadena formato de salida (json, csv, sarif) (predeterminado "json")

-r, --report-path cadena archivo de informe

-s, --source cadena ruta a la fuente (predeterminado: $PWD) (predeterminado ".")

-v, --verbose mostrar la salida detallada del análisis

Use "gitleaks [comando] --help" para obtener más información sobre un comando.