Akamai

WHITE PAPER

Fundamentos de seguridad de API: amplíe sus conocimientos y proteja su empresa

Introducción

Las API han evolucionado rápidamente, pasando de ser un mero detalle de la implementación hasta convertirse en un factor estratégico para la innovación digital. Cada vez que un cliente, un partner o un proveedor interactúan digitalmente con una empresa, existe una API entre bastidores que facilita un intercambio eficiente de información.

A medida que las API proliferan, también lo hacen sus riesgos. En la carrera por crear y lanzar rápidamente nuevas aplicaciones y servicios mejorados con IA, las API subyacentes presentan con demasiada frecuencia errores de configuración, carecen de controles de seguridad y son vulnerables a ataques de ejecución sencilla.

Como resultado, las API han pasado a ser uno de los principales vectores de ataque y, debido a ello, los equipos de seguridad deben modernizar sus estrategias de seguridad al respecto. Por lo tanto, proteger las API se está convirtiendo rápidamente en una de las principales prioridades estratégicas de los responsables de TI y seguridad.

Tanto si desea familiarizarse con los aspectos básicos de seguridad de API como si quiere saber qué preguntas debe hacerse al contratar un servicio, en esta guía encontrará los datos que necesita, entre los que se incluyen:

- **Los diferentes tipos de API.**
- **La importancia de la seguridad de las API para las empresas actuales.**
- **Las prácticas recomendadas para combatir los riesgos de seguridad de las API.**
- **Métodos de ataque y abuso habituales en materia de API.**

Si desea pasar directamente a las prácticas recomendadas de seguridad de API, avance hasta la página 10.

Índice

Conceptos básicos sobre las API

¿Qué es una API web?

Una interfaz de programación de aplicaciones (API) web está formada por uno o más terminales en un sistema de mensajes de solicitud-respuesta definido, normalmente expresado en formato JSON o XML, que se exponen públicamente a través de la web, generalmente mediante un servidor web basado en HTTP.

En otras palabras, una API web es lo que la mayoría de la gente identifica con el término "API": un conjunto de terminales. Los terminales constan de rutas de acceso a recursos, las operaciones que se pueden realizar en esos recursos y la definición de los datos de recursos (en JSON, XML, Protobuf u otro formato).

Las API web son distintas a otras API, como las expuestas por el sistema operativo o las bibliotecas de aplicaciones que se ejecutan en la misma máquina. No obstante, al utilizar el término "API" de forma general, se suele hacer referencia a las API web, basadas en HTTP, especialmente en el contexto de la transformación digital empresarial y la seguridad de API.

¿Cuáles son los tipos de API más comunes?

En la siguiente tabla, se muestran distintos términos que hacen referencia a diferentes modelos de uso y enfoques técnicos de la implementación de API. Las API web se basan en HTTP y suelen dividirse en cuatro tipos principales: RESTful, SOAP, GraphQL y gRPC. En la tabla se definen todos ellos, así como algunos otros.

Modelo de uso de la API	Descripción
API pública	Una API que está disponible y se comparte libremente con todos los desarrolladores a través de Internet.
API externa	Término que a menudo se utiliza indistintamente con el de API pública, es una API expuesta a Internet.
API privada	API que se implementa en un centro de datos protegido o un entorno de nube para su uso por parte de desarrolladores de confianza.
API interna	Término que a menudo se utiliza indistintamente con el de API privada.
API de terceros	Proporciona acceso programático a funciones especializadas o datos de un origen de terceros para su uso en una aplicación.
API de partner	Tipo de API de terceros que se pone a disposición de forma selectiva para los partners comerciales autorizados.
API autenticada	API a la que solo pueden acceder los desarrolladores a los que se han otorgado credenciales (o los atacantes que han obtenido acceso no autorizado a ellas).
API no autenticada	API a la que se puede acceder programáticamente sin necesidad de credenciales específicas.
API HTTP	API que utiliza el protocolo de transferencia de hipertexto como protocolo de comunicación para llamadas API.

API RESTful	La transferencia de estado representacional (RESTful) es el tipo más común de API web que utiliza texto sin formato, HTML, XML, YAML o JSON para transferir datos. Las API RESTful son fáciles de utilizar por los marcos de front-end modernos (por ejemplo, React y React Native) y facilitan el desarrollo de aplicaciones web y móviles. Se han convertido en el estándar para cualquier API web, incluidas las utilizadas para la transferencia empresa a empresa (B2B).
GraphQL	Las API GraphQL son el nuevo estándar desarrollado por Facebook que proporciona acceso a bases de datos mediante un único terminal POST (normalmente /graphql). Resuelve un problema común de las API RESTful, que requieren varias llamadas para rellenar una sola página de interfaz de usuario.
SOAP	SOAP utiliza el lenguaje de marcado extensible (XML) detallado para llamadas a procedimientos remotos (RPC). Aún se puede encontrar entre las API heredadas.
XML-RPC	XML-RPC es un método para realizar llamadas a procedimientos a través de Internet que utiliza una combinación de XML para la codificación y HTTP como protocolo de comunicaciones.
gRPC	Las API gRPC son un protocolo binario de alto rendimiento desarrollado por Google sobre HTTP/2.0 y se utilizan principalmente para la comunicación este-oeste en redes internas.
OpenAPI	OpenAPI es una descripción y especificación de documentación para API. Cabe destacar que Swagger hace referencia a la especificación original, mientras que OpenAPI es el estándar abierto desarrollado por la iniciativa homónima.

¿Cuál es la diferencia entre las API y los terminales?

A menudo, se utiliza el término "API" cuando de lo que se habla realmente es de un único terminal de API. Las API, a veces denominadas servicios o productos de API, son grupos de terminales que sirven a una función empresarial. Por otro lado, un terminal es un recurso (o ruta de acceso de recursos, también conocida como URI o identificador de recursos uniforme) y la operación que se realiza en él (crear, leer, actualizar o eliminar). En las API RESTful, estas operaciones se asignan normalmente a los métodos HTTP (POST, GET, PUT y DELETE).

¿Qué son las API norte-sur?

Se trata de API que las organizaciones ponen a disposición de usuarios externos, principalmente para interactuar con sus partners comerciales; es lo que se denomina exposición de API. Por ejemplo:

> **Los bancos que adoptan la banca abierta pueden exponer sus datos a otras organizaciones de servicios financieros o tecnología financiera a través de API.**

> **Las organizaciones del sector sanitario pueden exponer los registros de los pacientes a compañías de seguros y otras organizaciones médicas a través de API.**

> **Las empresas del sector de la hostelería pueden exponer sus sistemas de reservas a agentes de viajes o agregadores a través de API.**

Las API son el tejido conectivo que permite intercambiar datos entre distintas organizaciones. Las API norte-sur suelen considerarse seguras porque el acceso está autorizado y autenticado. Normalmente, son el tipo de API de mayor volumen y crecimiento, por lo que suponen también la mayor superficie de ataque de la mayoría de las organizaciones.

¿Qué son las API este-oeste?

Se trata de API que las organizaciones utilizan internamente y que no deben ser accesibles para nadie fuera de la empresa. Estas API conectan aplicaciones internas o unidades de negocio o departamentos. No obstante, existe la posibilidad de que un desarrollador cometa un error que las haga accesibles por accidente. No están pensadas para que entidades externas puedan acceder a ellas, ni incluso conocerlas, pero se producen vulnerabilidades cuando los atacantes encuentran API este-oeste accesibles a través de Internet.

¿Qué diferencia a las API B2C de las API B2B?

Las API de empresa a cliente (B2C) respaldan las aplicaciones web y móviles. Normalmente, los clientes de front-end modernos las utilizan para permitir a los usuarios finales autenticados acceder a la funcionalidad empresarial.

Las API B2B son aquellas que una empresa ofrece a otras organizaciones para el desarrollo de las actividades comerciales y, en ocasiones, para proporcionar valor a clientes conjuntos.

Estas API ayudan a optimizar la forma en que una empresa trabaja con sus proveedores, distribuidores y otros partners, y a proporcionar mejores experiencias a sus clientes.

Entre los ejemplos de API B2B se incluyen:

API de banca abierta

API de gestión de la cadena de suministros

API de facturación y pagos electrónicos entre partners comerciales

Dado que los usuarios de las API difieren enormemente, los controles de seguridad disponibles para proteger estas API también varían. El sector se ha centrado en casos de uso de B2C hasta hace poco, pero incluso en esos casos, no se ha centrado en proteger las API B2C, sino en proteger las aplicaciones web. Las herramientas y controles de seguridad que se utilizan normalmente para proteger las aplicaciones web B2C ofrecen ciertas ventajas (por ejemplo, firewalls de aplicaciones web [WAF] y protección de API y aplicaciones web [WAAP]), pero no pueden proporcionar el grado de visibilidad, supervisión en tiempo real ni protección necesarios para proteger las API B2C de los ataques.

Proteger las API B2B es cada vez más difícil. Estas API suelen ser objetivos fáciles para los atacantes, ya que a menudo carecen de los mecanismos de protección necesarios. Las herramientas de seguridad de API antiguas tenían una visibilidad limitada de las API B2B y presentaban problemas a la hora de proteger aquellas API que facilitaban el acceso en bloque a los datos en nombre de usuarios compartidos (como sucede con la banca abierta, donde las empresas de tecnología financiera y las instituciones financieras comparten consensualmente datos de los clientes). No obstante, las soluciones más recientes ofrecen análisis de comportamiento y son capaces de reconocer actividades anómalas para superar de forma eficaz estos problemas.

¿Cuál es la diferencia entre las API privadas y las públicas?

Las API privadas, a veces también denominadas API internas, están dirigidas a los desarrolladores y contratistas de la empresa. A menudo forman parte de una iniciativa de arquitectura orientada a servicios (SOA) y están pensadas para agilizar el desarrollo interno al permitir que diferentes departamentos o unidades de negocio accedan a los datos de los demás de forma eficiente y eficaz.

Por el contrario, las API públicas, también conocidas como API externas, están expuestas a usuarios externos a la empresa. En su manifestación más extrema, como API abiertas, cualquiera puede usarlas libremente, pero, en cualquier caso, requieren una gestión estricta y una documentación óptima para que puedan utilizarlas ingenieros externos a la empresa.

Es importante tener en cuenta que las API privadas a las que se puede acceder a través de Internet no son realmente privadas, en el sentido estricto de la palabra. Tomemos, por ejemplo, la API B2C de ACME, utilizada únicamente por las aplicaciones móviles de ACME y desarrollada internamente por sus ingenieros. Podría parecer una API privada, pero como el tráfico llega a esta API desde Internet (fuera de la empresa), no lo es, simplemente no está publicada al público externo. Los hackers atacan estas API con regularidad interceptando el tráfico y aplicando ingeniería inversa a las aplicaciones móviles para encontrar sus API correspondientes.

Explicación de la seguridad de API

¿Qué es la seguridad de API?

La seguridad de API es una estrategia para obtener visibilidad, hacer pruebas rigurosas y proteger todas las API de una empresa, algunas de las cuales son partes integrales de las aplicaciones, los procesos empresariales y las cargas de trabajo en la nube. Sin embargo, dado que tanto las API internas como las externas se producen a gran velocidad y a gran escala, puede resultar difícil conocer de forma integral el panorama de API de su organización. Muchas empresas carecen de visibilidad para saber cuántas API tienen realmente y cuáles devuelven datos confidenciales cuando reciben una llamada. La identificación y la mitigación de los riesgos de seguridad de las API requieren controles lo suficientemente sofisticados como para obtener el nivel de visibilidad y análisis de datos necesarios. Las API que necesitan protección pueden incluir:

- API que facilitan el acceso a los datos por parte de clientes o partners comerciales.
- API utilizadas por los partners comerciales.
- API que se implementan y utilizan internamente para que la funcionalidad y los datos de las aplicaciones estén disponibles para diversos sistemas e interfaces de usuario de una manera estandarizada y escalable.

Una estrategia de seguridad de API eficaz debe incluir técnicas sistemáticas para evaluar el riesgo y las posibles consecuencias, así como para aplicar las medidas de mitigación adecuadas. El primer paso para evaluar el riesgo es crear un inventario de todas las API autorizadas y no autorizadas publicadas y utilizadas por la organización. Este inventario debe incluir atributos como los siguientes:

- Clasificaciones de datos que, como mínimo, distingan entre datos "no confidenciales", "confidenciales" y "muy confidenciales".
- Indicadores de riesgo, como vulnerabilidades de API y errores de configuración.

Además, las medidas de visibilidad y mitigación de riesgos de las API deben tener en cuenta un conjunto diverso de posibles amenazas, entre las que se incluyen las siguientes:

- Detectar e impedir el uso de API en la sombra no autorizadas (consulte el recuadro lateral).
- Identificar y corregir las vulnerabilidades y los errores de configuración de las API que los atacantes podrían aprovechar.
- Prevenir casos de uso indebido de las API, como el abuso de la lógica empresarial y el scraping de datos.

¿En qué se diferencia la seguridad de las API de la seguridad de las aplicaciones?

Si bien la seguridad de las API y la seguridad de las aplicaciones tradicional son disciplinas relacionadas, la primera plantea un reto distinto por dos razones clave: la escala y la complejidad del problema.

Mayor escala

Hay tres factores que contribuyen al rápido crecimiento del uso de las API:

1. Está creciendo el uso de microservicios, una arquitectura que exige el uso de API para la comunicación servicio a servicio.
2. En el canal de usuario directo, los marcos de aplicaciones front-end modernos, como React, Angular y Vue, utilizan API y están sustituyendo a las aplicaciones web heredadas.
3. Se añaden API para abordar canales completamente nuevos (por ejemplo, partners, Internet de las cosas [IoT] y automatización empresarial).

Flexibilidad que genera complejidad

A diferencia de las aplicaciones web, las API están diseñadas para utilizarse mediante programación de muchas formas diferentes, lo que hace que diferenciar el uso legítimo de los ataques y el abuso sea extremadamente difícil.

¿Hay alguna taxonomía de API que los equipos de seguridad deberían comprender?

A continuación se muestran las categorizaciones y descripciones comunes de las API que pueden aparecer en un contexto de seguridad.

API autorizadas

API publicada (con documentación de Swagger o similar)

API no autorizadas

- API en la sombra
- API no aprobada
- API zombi
- API oculta

API desfasadas

- API obsoleta
- API heredada
- API zombi
- API huérfana

¿Cuáles son las prácticas recomendadas para proteger las API?

Mejorar la seguridad de API comienza con las siguientes prácticas recomendadas:

- Integrar los estándares y las prácticas de seguridad de API en el ciclo de vida de desarrollo de software (SDLC) de la organización.
- Incorporar documentación de las API y pruebas de seguridad automatizadas en los procesos de integración y entrega continuas (CI/CD).
- Asegurarse de que las API disponen de controles de autenticación y autorización adecuados y eficaces.
- Implementar medidas de limitación de velocidad para evitar el abuso o la sobrecarga de las API.
- Intensificar la limitación de velocidad y el uso de otras medidas a nivel de aplicación con puertas de enlace especializadas o redes de distribución de contenido (CDN) para mitigar el riesgo de ataques distribuidos de denegación de servicio (DDoS).
- Convertir las pruebas de seguridad de API en una parte integral de los procesos de pruebas de aplicaciones más amplios.
- Poner en práctica un proceso de detección continua de API.
- Incorporar un enfoque sistemático para identificar y corregir vulnerabilidades comunes de las API, incluidos los 10 principales riesgos de seguridad de API según OWASP.
- Utilizar la detección y prevención de amenazas basadas en firmas como nivel de referencia de protección contra ataques conocidos de API.
- Ampliar la detección basada en firmas con IA y análisis de comportamiento para que la detección de amenazas de API sea más escalable, precisa, relevante para el negocio y resistente frente a nuevas amenazas.
- Asegurarse de que el proceso de supervisión y análisis de la seguridad de API se prolongue durante varias semanas y sesiones de API.
- Complementar la supervisión y las alertas de seguridad de API con acceso bajo demanda a los datos de actividad e inventario de API para que los utilicen los investigadores de amenazas, los desarrolladores, los equipos de DevOps y el personal de soporte.

Su capacidad para implementar estas prácticas recomendadas depende de cuánto haya conseguido consolidar su estrategia de seguridad de API (consulte el recuadro lateral).

Etapas de la consolidación de la seguridad de API

Fase 1: Visibilidad y detección

Está en proceso de detectar todas sus API y los microservicios que admiten siguiendo un enfoque automatizado. Es fundamental contar con una cobertura amplia, ya que las API que se pasan por alto (por ejemplo, las que ya no se utilizan) son uno de los principales objetivos de los atacantes.

Fase 2: Pruebas

Prueba todas las API para asegurarse de que están codificadas correctamente y de que realizan su función. Las pruebas que se realizan antes de implementar una API son el último escalón de esta etapa de consolidación; se elimina el riesgo antes de que la API entre en producción y cualquier corrección necesaria es exponencialmente menos costosa.

Fase 3: Auditoría de riesgos

Audita constantemente todo su entorno de API para identificar problemas de configuración u otros errores. En esta fase también se asegura de que todas las API cuentan con la documentación adecuada y determina si contienen datos confidenciales o si carecen de los controles de seguridad adecuados.

Fase 4: Protección del tiempo de ejecución

Utiliza una solución con protección automatizada del tiempo de ejecución, que detecta cualquier actividad anómala de las API. Gracias a esta forma de supervisar las interacciones de las API, puede detectar aquellos comportamientos que supongan una amenaza en tiempo real.

Fase 5: Respuesta

Dispone de soluciones para responder a comportamientos sospechosos de las API, como un WAF o una puerta de enlace de API que bloquea el tráfico sospechoso antes de que pueda acceder a recursos críticos. Sus soluciones incorporan reglas personalizadas y automatizadas.

Fase 6: Búsqueda de amenazas

Con regularidad, realiza análisis forenses a partir de datos de amenazas anteriores para saber si se identificaron correctamente los riesgos y si han surgido patrones que permiten la búsqueda proactiva de amenazas mediante una combinación de herramientas sofisticadas e inteligencia humana.

Riesgos de seguridad y abuso de las API

¿Qué es una vulnerabilidad de API?

Una vulnerabilidad de API es un error de software o de configuración del sistema que puede explotar un atacante para acceder a funciones o datos confidenciales de una aplicación o para hacer algún otro uso indebido de una API. Los 10 principales riesgos de seguridad de API según OWASP ofrecen una descripción útil de algunas de las vulnerabilidades de API que más se explotan y que las organizaciones deben intentar identificar y corregir.

¿Se incluyen todas las vulnerabilidades de API en los 10 principales riesgos de seguridad de API según OWASP?

Los 10 principales riesgos de seguridad de API según OWASP son un excelente punto de partida para las organizaciones que quieren mejorar su estrategia de seguridad de API. Sus categorías cubren una amplia gama de posibles riesgos de API, pero también son bastante generales, por lo que es importante profundizar en las subáreas de cada una. Los atacantes que apuntan a las API a menudo intentan aprovechar los problemas de autorización (cubiertos ampliamente por OWASP), pero también existen riesgos de API que quedan completamente fuera de los 10 principales riesgos de seguridad de API según OWASP, como el abuso de errores lógicos.

¿Cómo se pueden vulnerar las API?

Las API se pueden atacar y vulnerar de distintas formas, pero algunas de las más comunes incluyen:

- **Explotación de vulnerabilidades:** las vulnerabilidades técnicas de la infraestructura subyacente pueden comprometer el servidor. Aquí entrarían desde las vulnerabilidades de Apache Struts (CVE-2017-9791 y CVE-2018-11776) hasta las de Log4j (CVE-2021-44228).
- **Abuso de la lógica empresarial:** el abuso de la lógica se produce cuando un atacante aprovecha los defectos de diseño o implementación de una aplicación para generar un comportamiento inesperado y no autorizado. Estos escenarios resultan problemáticos para los directores de seguridad de la información y sus equipos porque los controles de seguridad tradicionales son ineficaces.
- **Acceso no autorizado a los datos:** otra forma común de abuso de API es aprovechar mecanismos de autorización comprometidos para acceder a datos que no deberían ser accesibles. Estas vulnerabilidades tienen muchos nombres, como autorización a nivel de objeto comprometida (BOLA), referencia de objeto directo no segura (IDOR) y autorización a nivel de función comprometida (BFLA).

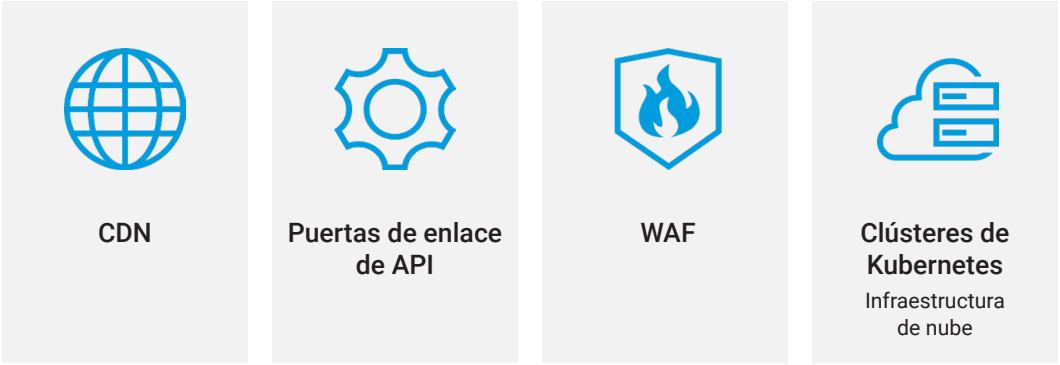
- **Robo de cuentas:** después de un robo de credenciales o incluso de un ataque de scripts entre sitios (XSS), se puede tomar el control de una cuenta. Cuando esto sucede, el abuso incluso de la API mejor redactada y mejor protegida es posible. El uso de una solución de seguridad de API que ofrezca análisis de comportamiento le permite diferenciar la actividad autenticada del uso ilegítimo.
- **Scraping de datos:** cuando las organizaciones ponen a disposición conjuntos de datos a través de API públicas, los atacantes pueden consultar de forma agresiva estos recursos para realizar una captura indiscriminada de grandes volúmenes de información valiosa.
- **Ataque de denegación de servicio (DoS) empresarial:** al pedir al back-end que realice tareas pesadas, los atacantes de API pueden provocar una erosión del servicio o un DoS completo en la capa de aplicación (una vulnerabilidad muy común en GraphQL, pero que puede suceder con cualquier implementación de terminal de API que utilice muchos recursos).

¿Qué es una API zombi?

Debido a los cambios en los requisitos empresariales y en el mercado, las API están en constante evolución. A medida que se lanzan nuevas implementaciones de terminales para satisfacer las nuevas necesidades empresariales, corregir errores e introducir mejoras técnicas, las versiones más antiguas de estos terminales van quedando obsoletas. La gestión del proceso de retirada de terminales antiguos no es una tarea sencilla. A menudo, las implementaciones de terminales que deberían haberse retirado permanecen activas y son accesibles; es lo que se denomina terminales zombis.

¿Cómo se pueden identificar los distintos tipos de API en la sombra?

Una de las maneras de descubrir API en la sombra a nivel empresarial es analizar e incorporar el tráfico de API en su red. Entre los ejemplos de fuentes de tráfico de API se incluyen:



Una vez que se hayan recopilado los datos sin procesar de todas las fuentes disponibles, se pueden utilizar técnicas de IA para transformarlos en un inventario integral de todas las API, los terminales y los parámetros. A partir de ahí, se pueden realizar análisis adicionales para clasificar estos elementos e identificar las API en la sombra que deben eliminarse o incorporarse a los procesos de control formales.

¿Cómo se deben proteger las API internas y las B2B?

Realmente depende de la definición de "internas". Algunos equipos se refieren a las API de las aplicaciones web y móviles de su propia organización que están expuestas a Internet como "API internas". Aunque la documentación de estas API solo puede ser accesible para empleados y contratistas de la empresa, los hackers son expertos en el análisis de aplicaciones y la ingeniería inversa de las API mediante kits de herramientas de desmontaje de aplicaciones y proxies como Burp Suite.

Sin embargo, si las "API internas" se definen como API este-oeste, a las que no se puede acceder desde fuera de la organización, el riesgo principal se reduce a las amenazas internas. La seguridad de sus API este-oeste y B2B es tan importante como la del resto: comience por proteger el SDLC y, a continuación, asegúrese de que el acceso está autenticado y autorizado. También puede implementar la gestión de cuotas, límites de velocidad y detenciones de picos, así como proteger sus API frente a amenazas conocidas mediante WAF/WAAP. Para las API B2B, considere la posibilidad de utilizar mecanismos de autenticación estrictos, como mTLS, ya que las transacciones suelen incorporar datos confidenciales y masivos.

Le recomendamos que utilice análisis de comportamiento tanto para las B2B como para las este-oeste, especialmente si tiene muchas entidades implicadas, lo que puede dificultar el proceso de distinguir entre comportamientos legítimos e ilegítimos. Por ejemplo:

> **¿Cómo saber si las credenciales de API de un usuario concreto han sido vulneradas?**

> **¿Cómo podría saber si su API de facturación está sufriendo un ataque por parte de un partner que itera números de factura para robar datos de cuentas?**

La protección de API B2B y este-oeste requiere un contexto empresarial que no se puede obtener analizando elementos técnicos, como direcciones IP y tokens de API, por sí solos. El uso de aprendizaje automático (ML) y análisis de comportamiento para obtener visibilidad de las entidades relevantes para el negocio es la única forma de comprender y gestionar eficazmente los riesgos. El contexto empresarial y los puntos de referencia históricos para el uso normal de las API por parte de entidades específicas, como sus usuarios o partners, o incluso entidades de procesos empresariales (facturas, pagos, pedidos, etc.), permiten detectar anomalías que, de otro modo, pasarían inadvertidas.

¿Las puertas de enlace de API ofrecen suficiente protección frente a los riesgos?

Muchas organizaciones que siguen un enfoque estratégico para las API utilizan puertas de enlace de API. La mayoría de puertas integran funciones de seguridad completas que las organizaciones deberían aprovechar, como, por ejemplo, la autenticación (y también la autorización, si pueden utilizar OpenID Connect). Sin embargo, no basta con realizar la autenticación, la autorización y la gestión de cuotas en la puerta de enlace de API, por varios motivos:

Las deficiencias en la detección de las puertas de enlace de API: las puertas de enlace de API solo tienen visibilidad y control sobre las API que están configuradas para gestionar, lo que las hace ineficaces a la hora de detectar los terminales y las API en la sombra.

Las brechas de seguridad de las puertas de enlace de API: las puertas de enlace de API pueden aplicar la autenticación y, en cierta medida, los esquemas de autorización, pero no inspeccionan las cargas (como hacen los WAF y los WAAP) ni perfilan el comportamiento para detectar abusos.

¿Cuáles son los fallos de configuración de API más comunes?

El número de posibles errores de configuración de API es casi infinito, dada la multitud de formas en que se utilizan. No obstante, existen algunos aspectos comunes en los errores de configuración:

Autenticación ineficaz o inexistente

La autenticación es fundamental para proteger los datos confidenciales que están disponibles a través de las API. El primer paso consiste en asegurarse de que todas las API que contienen datos confidenciales disponen en primer lugar de autenticación, pero también es importante proteger los mecanismos de autenticación contra ataques de fuerza bruta, Credential Stuffing y el uso de tokens de autenticación robados mediante la limitación de velocidad. A veces pueden producirse errores de configuración que permiten a los usuarios de API omitir los mecanismos de autenticación, a menudo en torno a la gestión de tokens (por ejemplo, algunos problemas notorios de validación de JWT o no comprobar el ámbito de los tokens).

Autorización comprometida

Uno de los usos más comunes de las API es proporcionar acceso a datos o contenido, incluida información confidencial. La autorización es el proceso de verificación de que un usuario de API es apto para acceder a los datos a los que intenta acceder, antes de ponerlos a su disposición. Esto se puede hacer en el nivel de objeto o recurso (por ejemplo, puedo acceder a mis pedidos, pero no a los de otra persona), o en el nivel de función (como suele ocurrir con las capacidades administrativas). La autorización es difícil de implementar correctamente debido al elevado número de casos y condiciones, y debido a los diversos flujos que las llamadas de API pueden tomar entre microservicios. Si no tiene un motor de autorización centralizado, es probable que la implementación de API incluya algunas de estas vulnerabilidades, como BOLA y BFLA.

Errores de configuración de seguridad

Además de los problemas de autenticación y autorización mencionados anteriormente, existen muchos otros errores de configuración de seguridad, como la comunicación no segura (por ejemplo, no usar la capa de sockets seguros [SSL] ni la seguridad de la capa de transporte [TLS], o usar conjuntos de cifrado vulnerables), el almacenamiento en la nube sin protección y las políticas de uso compartido de recursos entre orígenes demasiado permisivas.

Falta de recursos y limitación de velocidad

Cuando las API se implementan sin ningún límite en el número de llamadas que sus usuarios pueden realizar, los ciberdelincuentes pueden saturar los recursos del sistema, lo que provoca la degradación del servicio o ataques de denegación de servicio (DoS) a gran escala. Como mínimo, se deben aplicar límites de velocidad al acceso a cualquier terminal no autenticado, siendo los terminales de autenticación de vital importancia; de lo contrario, los ataques de fuerza bruta, Credential Stuffing y validación de credenciales ocurrirán sin remedio.

¿Qué son los ataques a las API?

Los ataques a las API son intentos de utilizar API con fines maliciosos o no autorizados. Los ataques a las API adoptan muchas formas, entre las que se incluyen:

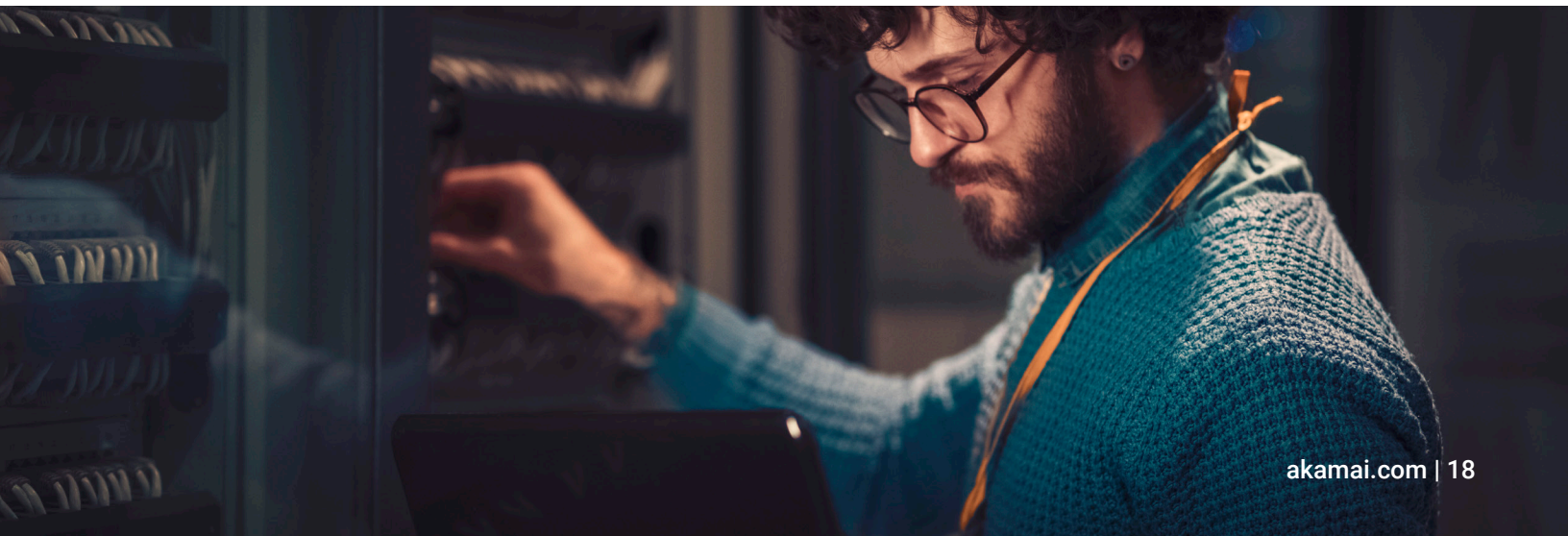
- Explotación de vulnerabilidades técnicas en implementaciones de API.
- Uso de credenciales robadas, así como de otras técnicas de robo de cuentas para hacerse pasar por un usuario legítimo.
- Abuso de la lógica empresarial que permite el uso de API de formas inesperadas.

¿Qué es el Credential Stuffing para las API?

La filtración de información de ID de usuario y contraseña de sitios web y plataformas de software como servicio (SaaS) se ha convertido en algo habitual. A menudo, estos incidentes provocan que se compartan ampliamente en línea grandes conjuntos de credenciales. El Credential Stuffing es la práctica de utilizar credenciales de autenticación filtradas de sitios web que se han vulnerado anteriormente para realizar intentos de inicio de sesión automatizados en otros sitios web. Esta técnica se basa en la premisa de que algún porcentaje de usuarios utiliza las mismas credenciales para varios sitios. Cada vez es más común que los autores de ataques se dirijan directamente a las API y actúen sobre los mecanismos de autenticación. Esto les permite automatizar el ataque más fácilmente, ya que las API se crean para facilitar el uso de programas.

¿Qué es la exfiltración de datos a través de API?

La exfiltración de datos se suele producir cuando los ataques a las API y los abusos a estas consiguen su objetivo. En algunos casos, al hablar de exfiltración de datos se hace referencia a información con un alto nivel de confidencialidad y de carácter privado, robada a través de un ataque a las API. Sin embargo, también se puede aplicar a tipos menos graves de abuso de API, incluido el scraping agresivo de datos disponibles públicamente para recopilar grandes volúmenes de datos valiosos en conjunto.



Soluciones y tendencias de seguridad de API

¿Cuáles son las últimas tendencias en seguridad de API?

A continuación se muestran las tendencias clave que los responsables de seguridad deben tener en cuenta al desarrollar una estrategia de seguridad de API:

Análisis de comportamiento y detección de anomalías: en lugar de intentar predecir posibles ataques y depender únicamente de la detección basada en firmas y de políticas predefinidas (por ejemplo, WAF) para mitigar el riesgo, las organizaciones están añadiendo cada vez más ML y análisis de comportamiento para ver la actividad de API en el contexto empresarial y detectar anomalías.

Transición del entorno local al SaaS: aunque muchos productos de seguridad de API de primera generación se implementaron en el entorno local, los enfoques basados en SaaS son cada vez más populares debido a su velocidad, facilidad de implementación y capacidad para aprovechar el potencial del ML a gran escala.

Análisis de periodos de tiempo más amplios: los enfoques de seguridad de API que solo analizan llamadas de API individuales o actividad de sesión a corto plazo se están sustituyendo por plataformas que analizan la actividad de API a lo largo de días y, a veces, semanas, desde completar la optimización automatizada de políticas WAF básicas hasta realizar análisis de comportamiento y detectar anomalías.

DevSecOps: incorporación de partes interesadas no relacionadas con la seguridad: una de las mejores formas de reducir los riesgos de las API es establecer conexiones más estrechas entre las estrategias y herramientas de seguridad de API, y los desarrolladores y sistemas implicados en crearlas, implementarlas y configurarlas.

Seguridad de API habilitada para API: aunque es fundamental detectar y mitigar los ataques activos a las API y los casos de abuso, las organizaciones con visión de futuro están buscando formas de utilizar el acceso bajo demanda a los datos y la información de seguridad de las API para mejorar la búsqueda de amenazas, la respuesta a incidentes y las prácticas de desarrollo de API.

¿Qué se entiende por seguridad de API basada en firmas?

Las técnicas de seguridad de API basadas en firmas supervisan las características y patrones de ataque conocidos, y, tras ello, generan alertas de seguridad y otras respuestas automatizadas si se observan coincidencias. Es el caso típico de un WAF. El valor: si a una organización se le notifica sobre tráfico de API entrante comprometido o que se comporta de forma anómala, puede utilizar la seguridad de API basada en firmas para bloquearlo inmediatamente.

La mejor opción es un WAF que forme parte de una solución WAAP más grande, que pueda a su vez realizar detecciones avanzadas con ML. De esta forma, aprenderá de los patrones de firma de ataque y seguirá siendo ágil a escala. Una solución WAAP integrada con una solución de seguridad de API que ofrezca análisis de comportamiento y respuestas personalizadas le proporcionará lo mejor de ambos mundos. Juntas, estas soluciones ofrecen visibilidad, detección y respuesta de API completas tanto interna como externamente.

¿Qué es la detección y respuesta de API?

La detección y respuesta de API es una categoría emergente de seguridad de API centrada en un análisis profundo de los datos históricos con los siguientes fines:

- Establecer un estándar de comportamiento para todos los usuarios de API.
- Detectar ataques y anomalías que indiquen un posible abuso y uso incorrecto de las API.

Para ser eficaces a gran escala, las soluciones de detección y respuesta de API deben ofrecerse bajo un modelo SaaS debido al enorme tamaño de los conjuntos de datos que requieren las técnicas de ML, que consumen muchos recursos.

¿Qué es la protección avanzada contra amenazas de API?

La protección avanzada contra amenazas de API es un enfoque basado en SaaS para la seguridad de API que combina los análisis de comportamiento con la búsqueda de amenazas con los siguientes fines:

- Detectar todas las API que utiliza una organización, incluidas las API en la sombra o zombis.
- Aplicar el aprendizaje automático para superponer el contexto empresarial sobre cómo se utilizan las API y se abusa de ellas.
- Realizar análisis de comportamiento y búsquedas de amenazas en las API y sus datos de actividad.

¿Qué es una plataforma de seguridad de API?

Una plataforma de seguridad de API es una oferta basada en SaaS especialmente diseñada con los siguientes fines:

- Crear un inventario continuamente actualizado de todas las API en uso en toda la empresa (tanto si están autorizadas como si no).
- Analizar las API y su uso para descubrir el contexto empresarial y determinar un estándar de comportamiento esperado.
- Detectar anomalías en el uso de las API y, cuando sea necesario, proporcionar alertas y datos de apoyo a los flujos de trabajo de gestión de eventos e información de seguridad (SIEM) y de orquestación, automatización y respuesta de seguridad (SOAR).
- Proporcionar acceso bajo demanda a la información de inventario, actividad y amenazas de API tanto a las partes interesadas del área de seguridad como a las de otros ámbitos.

¿Qué es una empresa de seguridad de API?

Ahora que los responsables de TI y seguridad utilizan las API de forma más estratégica, es posible que tengan que recurrir a partners especializados en API. Los tres tipos más comunes de empresas de API son:

- Empresas que ofrecen puertas de enlace de API y que proporcionan la tecnología necesaria para aceptar las llamadas de API de forma centralizada y dirigirlas a los recursos de back-end y microservicios adecuados.
- Empresas que ofrecen plataformas de seguridad de API y que ayudan a las organizaciones a identificar todas las API activas y sus riesgos, detectar casos de ataques y abusos, realizar pruebas de seguridad completas, y que les proporcionan datos enriquecidos sobre cómo se utilizan esas API.
- Empresas que ofrecen plataformas de seguridad de API y WAAP y que pueden ayudar a transferir datos de tráfico de API sin problemas y, al mismo tiempo, ofrecer la capacidad de identificar las API dentro y fuera de la plataforma, lo que resulta ideal para la consolidación de proveedores y la reducción de las brechas digitales.

¿Qué es la búsqueda de amenazas en las API?

La búsqueda de amenazas implica la búsqueda activa de amenazas desconocidas o no detectadas anteriormente. Este enfoque proactivo es fundamental para identificar amenazas nuevas y emergentes que pueden no haberse visto antes y mitigarlas antes de que puedan causar daños significativos. Una de las técnicas clave utilizadas en la búsqueda de amenazas es el análisis de comportamiento. Esto implica analizar el comportamiento de las API para identificar cualquier actividad sospechosa o anómala. Por ejemplo, que una API solicite de repente miles de registros en un breve espacio de tiempo puede ser señal de que su lógica empresarial se ha vulnerado. Para que los equipos de seguridad identifiquen las posibles amenazas con antelación y pongan en marcha medidas de mitigación, las soluciones de seguridad de API modernas proporcionan funciones específicas para detectarlas.

¿Qué es WAAP?

La protección de API y aplicaciones web (WAAP) es una categorización que la empresa de investigación Gartner utiliza para su cobertura en el sector de soluciones de protección web y API emergentes. Se trata de una evolución de la cobertura anterior del sector en el mercado de los WAF en respuesta a la creciente importancia estratégica de la seguridad de API y al paso de las plataformas WAF a la nube como servicio SaaS gestionado.

¿Qué es un ejemplo de documentación de API?

La forma más común de documentación de API para API RESTful (que son el tipo más común de API web) es una colección de archivos Swagger basados en la especificación OpenAPI. Lo ideal es que los desarrolladores creen la documentación de API cuando se diseña o implementa una API. Sin embargo, la documentación de API suele estar obsoleta, lo que provoca una discrepancia entre esa documentación y el uso de la API en el mundo real. Para solucionar este problema, algunas plataformas de seguridad de API pueden generar archivos Swagger a partir de la actividad real de la API, lo que pone de relieve las brechas entre lo que se documenta y lo que se implementa realmente, un componente integral de cualquier evaluación de riesgos de API.

¿Hay una lista de comprobación de seguridad de API que las empresas deberían seguir?

Una seguridad de API eficaz requiere seguir detenidamente toda una serie de pasos e implementar prácticas continuas de forma específica para cada organización. A continuación se muestra una lista de comprobación que los equipos de seguridad pueden utilizar como punto de partida a medida que avanzan en la seguridad de API:

- ¿Su enfoque de seguridad de API incluye un mecanismo para la detección continua de API en toda la empresa?
- ¿Está integrada la gestión de la situación de las API en las prácticas generales de gestión de riesgos y seguridad de la organización?
- ¿Está implementando un enfoque de seguridad de API de uso general que no le limite a modelos específicos de centro de datos o infraestructura de nube?
- ¿Proporcionará su enfoque a sus equipos el contexto empresarial que necesitan para comprender realmente la actividad de las API y los posibles riesgos que se están observando?
- ¿Cuenta con una estrategia de automatización bidireccional entre su plataforma de seguridad de API y otros procesos empresariales relacionados, como SIEM/SOAR, búsqueda de amenazas, documentación, herramientas de DevOps, etc.?
- ¿Está tomando medidas para hacer partícipes a las partes interesadas que no pertenecen al ámbito de la seguridad, como los desarrolladores, de sus herramientas y procesos de seguridad de API?

La seguridad de Akamai protege las aplicaciones que impulsan su negocio en cada punto de interacción sin comprometer el rendimiento ni la experiencia del cliente. Gracias a la escala de nuestra plataforma global y su visibilidad de las amenazas, colaboramos con usted para prevenirlas, detectarlas y mitigarlas, de forma que pueda generar confianza en la marca y cumplir su visión. Para obtener más información acerca de las soluciones de cloud computing, seguridad y distribución de contenido de Akamai, visite akamai.com y akamai.com/blog, o siga a Akamai Technologies en X, antes conocido como Twitter, y LinkedIn. Publicado en septiembre de 2024.