GammaCSOC-CERT
By Gamma Ingenieros

GAMMA INGENIEROS

Boletín de Ciberseguridad Semanal

Edición °4226

TLP:GREEN
TRAFFIC LIGHT PROTOCOL
https://www.first.org/tlp/

BOLETÍN CIBERSEGURIDAD
SEMANAL EDICIÓN °4226

BOLETÍN DE CIBERSEGURIDAD SEMANAL

Bienvenido a su boletín semanal de ciberseguridad generado por Gamma Ingenieros. A través de esta publicación, tenemos el propósito de compartir contenidos relevantes para nuestras operaciones y para nuestros aliados de negocio.

Aquí encontrará información sobre vulnerabilidades, alertas tempranas, noticias del sector, incidentes de ciberseguridad, tecnologías y servicios relacionados, entre otros.

VISTA RÁPIDA

	CRÍTICO	URGENTE	IMPORTANTE
VULNERABILIDADES	3	0	0
MALWARE	0	1	0
NOTICIAS DE CIBERSEGURIDAD	0	1	0

VULNERABILIDADES

GITLAB CE/EE – VULNERABILIDAD CRÍTICA DE DESBORDAMIENTO DE ENTERO EN COMPILADOR DE EXPRESIONES REGULARES PERMITE EJECUCIÓN REMOTA DE CÓDIGO (CVE-2026-110264)

GitLab ha publicado una actualización de seguridad crítica de emergencia para corregir una vulnerabilidad de máxima gravedad (CVSS 10.0) que afecta a sus versiones autoalojadas Community Edition (CE) y Enterprise Edition (EE). La falla se debe a un desbordamiento de entero en el motor de expresiones regulares Onigmo (integrado dentro del intérprete de Ruby empaquetado por GitLab). Bajo ciertas condiciones, un atacante remoto no autenticado puede lograr la ejecución remota de código (RCE) arbitrario en el servidor mediante el envío o procesamiento de expresiones regulares especialmente manipuladas, afectando a todas las versiones de la plataforma lanzadas desde 2014 (desde la versión 7.0).

Resumen técnico:

- Identificador principal: CVE-2026-110264 (Boletín de Parche Crítico del 21 de octubre de 2026 / INCIBE-2026-766).
- Severidad: Crítica (CVSS v3.1: 10.0 Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H).
- Causa raíz: Desbordamiento de entero (CWE-190), escritura fuera de límites (CWE-787) y control indebido de generación de código (CWE-94) en el compilador de expresiones regulares de la librería Onigmo.
- Mecanismo de falla:
- La librería Onigmo utiliza un operando de 16 bits con signo para almacenar los identificadores de repetición de un patrón, contándolos como un entero.
- Al procesar un patrón de expresión regular manipulado con más de 32.767 repeticiones, el identificador sufre un desbordamiento convirtiéndose en un valor negativo, lo que provoca una escritura fuera de los límites del búfer en tiempo de coincidencia (match time).
- Un vector principal de entrada sin autenticación se produce a través del renderizador de archivos con extensión .org (org-ruby), los cuales construyen expresiones regulares a partir de líneas de encabezado no escapadas; al visualizar un archivo .org en un repositorio público o forzar el renderizado vía vista previa del editor, un visitante no autenticado desencadena la falla y la ejecución de código en el servidor.
- Estado de explotación: Divulgada oficialmente el 21 de octubre de 2026. No se registra evidencia de explotación activa masiva en el entorno real ni figura en el catálogo KEV de la CISA al momento de la divulgación, aunque existen reglas de detección publicadas por GitLab (framework TLDR en formato Sigma) para identificar intentos de explotación.
- Versiones afectadas: Instalaciones autoalojadas de GitLab CE/EE en todas las versiones desde la 7.0 hasta anteriores a la 18.11.13, 19.0 anteriores a 19.0.10, 19.1 anteriores a 19.1.9, 19.2 anteriores a 19.2.9, 19.3 anteriores a 19.3.6, 19.4 anteriores a 19.4.3 y 19.5 anteriores a 19.5.1. (Los entornos GitLab.com y GitLab Dedicated cuentan con la corrección aplicada por el fabricante).

Impacto potencial:

- Ejecución remota de código (RCE) no autenticada: Otorga a un atacante no autenticado la capacidad de ejecutar comandos arbitrarios en el sistema operativo del servidor GitLab con los privilegios de la cuenta del servicio (git).
- Exfiltración masiva de código fuente y secretos corporativos: Facilita el acceso no autorizado a repositorios de código privado, variables de entorno de CI/CD, tokens de Runner, claves API y secretos de despliegue almacenados en la plataforma.
- Compromiso total de la cadena de suministro de software (Supply Chain): Permite alterar código fuente, inyectar código malicioso en tuberías de integración continua (pipelines) y manipular artefactos distribuidos hacia entornos de producción.
- Pivoteo y movimiento lateral hacia infraestructura interna: Habilita el uso del servidor de GitLab comprometido como punto de entrada para pivoteo hacia la red interna corporativa, clústeres de despliegue y servidores de bases de datos adyacentes.

Recomendaciones de mitigación:

1. Actualización inmediata de la instancia de GitLab: Desplegar con carácter de máxima prioridad el parche correspondiente a la rama en uso: 19.5.1, 19.4.3, 19.3.6, 19.2.9, 19.1.9, 19.0.10 o 18.11.13. Instancias en versiones 18.10 o inferiores deben planificar una migración por etapas hacia una rama soportada.
2. Planificación de ventana de mantenimiento para migraciones de BD: En instalaciones de un solo nodo (Single-Node), considerar que la aplicación de los parches 19.4.3 y 19.3.6 requiere la ejecución de migraciones de base de datos que implicarán tiempo de inactividad (downtime); en entornos multinodo, aplicar procedimientos de actualización sin tiempo de parada (Zero-Downtime).
3. Restricción de exposición de red y visibilidad pública: Como medida de contención temporal antes de aplicar el parche, restringir el acceso público a la instancia a través de firewall/VPN y reducir la visibilidad pública de proyectos, ya que la vía no autenticada requiere el renderizado de archivos .org en proyectos públicos.
4. Caza de amenazas e implementación de reglas de detección: Configurar e inspeccionar los registros de la aplicación (exceptions_json.log y puma_stderr.log) aplicando las reglas Sigma de GitLab para detectar ráfagas de errores al renderizar archivos .org (Org mode render error burst) o caídas del proceso trabajador en Onigmo (Onigmo native worker crash).

Prioridad: Crítica.

Ampliar información:

- https://docs.gitlab.com/releases/patches/patch-release-gitlab-19-5-1-released/
- https://thehackernews.com/2026/10/critical-gitlab-flaw-cvss-100-could.html
- https://ciberseguridad.blog/gitlab-rce-org-mode-onigmo/
- https://www.incibe.es/incibe-cert/alerta-temprana/avisos/multiples-vulnerabilidades-en-gitlab-ceee-de-gitlab
- https://devopsity.com/runbooks/gitlab-patch-cve-2026-110264/

PHP2SHELL – VULNERABILIDADES CRÍTICAS DE EJECUCIÓN REMOTA DE CÓDIGO (RCE) Y DENEGACIÓN DE SERVICIO EN EL MOTOR CORE DE PHP (CVE-2026-109618 Y CVE-2026-109619)

El equipo de seguridad de PHP y diversos grupos de investigación (entre ellos Calif, Wiz Research y OpenAI) han divulgado dos vulnerabilidades críticas de severidad máxima (CVSS 9.2) que afectan al núcleo del lenguaje de programación PHP. El conjunto de fallas, denominado colectivamente PHP2Shell, proviene de problemas de liberación de memoria no válida (Use-After-Free) desencadenados por errores fatales o de agotamiento de memoria (Out-of-Memory / OOM) mediante funciones como longjmp(). La explotación exitosa permite a un atacante remoto no autenticado lograr la ejecución remota de código (RCE) y la divulgación no autorizada de memoria del servidor (pudiendo filtrar cookies de sesión PHPSESSID, tokens API, claves de cifrado TLS o contraseñas) en aplicaciones web populares como Magento, WordPress, Drupal, Laravel, Symfony y phpMyAdmin.

Resumen técnico:

- Identificador principal: CVE-2026-109618 y CVE-2026-109619 (PHP2Shell).
- Severidad: Crítica (CVSS v4.0: 9.2).
- Causa raíz: Uso de memoria previamente liberada (Use-After-Free - CWE-416) y gestión inadecuada de condiciones excepcionales/OOM (Out-Of-Memory) mediante la interrupción abrupta de flujos por longjmp().
- Mecanismo de falla:
- CVE-2026-109618 (UAF en des-serializador de progreso de carga / OOM): Ocurre cuando una solicitud supera el límite de memoria del servidor (memory_limit) durante el procesamiento de datos multipart/form-data. La salida abrupta vía longjmp() deja las estructuras internas parcialmente inicializadas; la limpieza posterior procesa esta memoria como datos válidos, filtrando fragmentos del heap del servidor o permitiendo RCE.
- CVE-2026-109619 (UAF en punteros de rastreo y estado global / OPcache): Un error fatal interrumpe la ejecución pero deja referencias a regiones de memoria ya liberadas dentro del estado global de los procesos trabajadores (workers) de PHP-FPM o mod_php. Una petición posterior manejada por el mismo trabajador reutiliza ese estado no válido antes de su reinicio, derivando en confusión de tipos y RCE.
- Vectores de activación: Peticiones HTTP POST con cuerpo multipart/form-data manipulado (p. ej., gran volumen de campos vacíos o encabezados anidados) dirigidas a cualquier script que procese solicitudes web.
- Estado de explotación: Divulgadas oficialmente el 21 de octubre de 2026. Confirmada la explotación activa en el entorno real en el catálogo KEV de la CISA, así como en redes de señuelos (honeypots) y registros de seguridad perimetral (Cloudflare y Amazon).
- Versiones afectadas: Todas las versiones de PHP desde la 5.4.0 hasta anteriores a la 8.2.35, 8.3.36, 8.4.27 y 8.5.12 (afecta a arquitecturas PHP-FPM, Apache mod_php, php-cgi, FrankenPHP y LiteSpeed).

Impacto potencial:

- Ejecución remota de código (RCE) no autenticada: Concede a un atacante la capacidad de ejecutar comandos arbitrarios en el servidor web subyacente con los privilegios del proceso de PHP (por ejemplo, www-data o nobody).
- Fuga y divulgación masiva de memoria (Memory Disclosure): Permite extraer fragmentos de la memoria del heap del servidor a través de cookies PHPSESSID manipuladas o respuestas HTTP, expuesto datos sensibles como credenciales, claves privadas TLS, tokens de sesión y claves API de otras solicitudes.
- Compromiso total de plataformas CMS y marcos de desarrollo: Permite tomar el control de aplicaciones web legítimas construidas en PHP (tales como Magento, WordPress, Drupal, Joomla, Laravel, Symfony y Nextcloud) independientemente de que el código propio de la aplicación esté libre de fallas.
- Denegación de servicio (DoS) y agotamiento de recursos: Facilita el colapso masivo de trabajadores (worker processes) de PHP-FPM o Apache debido a errores de segmentación (SIGSEGV) provocados por la corrupción recurrente de memoria.

Recomendaciones de mitigación:

1. Actualización prioritaria del entorno de ejecución PHP: Actualizar inmediatamente el binario o contenedor de PHP a las versiones corregidas oficiales: 8.2.35, 8.3.36, 8.4.27 o 8.5.12. En versiones sin soporte oficial (EOL como 7.4, 8.0 u 8.1), aplicar los paquetes corregidos distribuidos por los mantenedores del sistema operativo (p. ej. Ondřej Surý, Remi, Ubuntu Pro, CloudLinux) y realizar un reinicio explícito del servicio web (p. ej., php-fpm, Apache2 o Nginx).
2. Bloqueo temporal de solicitudes multipart/form-data en WAF o Proxy: Si no es posible aplicar el parche de inmediato, configurar reglas en el servidor web (Nginx / Apache) o WAF para bloquear temporalmente peticiones POST con encabezado Content-Type: multipart/form-data en rutas que no requieran la carga de archivos.
3. Ajustes de mitigación temporal en php.ini o PHP-FPM: Deshabilitar el seguimiento de progreso de carga mediante session.upload_progress.enabled = 0, ajustar los límites de entrada (max_input_vars, max_file_uploads) y forzar el reinicio de procesos trabajadores configurando pm.max_requests = 1 en el archivo de contexto del pool de PHP-FPM.
4. Auditoría de memoria y caza de amenazas (Threat Hunting): Inspeccionar los logs del servidor web en busca de cookies PHPSESSID con longitudes anómalas (superiores a 256 bytes), picos de errores HTTP 500/404 intermitentes o creación impredecible de procesos secundarios (sh, bash, curl, wget) iniciados desde los servicios del servidor web.

Prioridad: Crítica.

Ampliar información:

- https://www.zebra.com/es/es/support-downloads/lifeguard-security/php2shell-vulnerability-cve-2026-109618-cve-2026-109619.html
- https://www.tenable.com/blog/php2shell-cve-2026-109618-cve-2026-109619-frequently-asked-questions-about-critical-php-remote
- https://security.paloaltonetworks.com/CVE-2026-109618
- https://cloud.google.com/blog/products/identity-security/responding-to-core-php-vulnerabilities-cve-2026-109618-cve-2026-109619
- https://sansec.io/research/php2shell

NODE2SHELL / HTTP2SHELL – VULNERABILIDAD CRÍTICA EN DESCOMPRESOR BROTLI PERMITE EJECUCIÓN REMOTA DE CÓDIGO (RCE) Y DENEGACIÓN DE SERVICIO EN NODE.JS Y OTRAS PLATAFORMAS (CVE-2026-111497)

El equipo de seguridad de Node.js y diversos grupos de investigación (incluidos Imperva, Aikido y HeroDevs) han divulgado una vulnerabilidad de severidad alta/crítica (CVSS v4.0: 9.2) atribuida a la librería de descompresión Brotli empaquetada en el núcleo de Node.js y otros entornos de ejecución. El fallo, denominado Node2Shell (y posteriormente HTTP2Shell debido a su alcance generalizado en protocolos HTTP), se origina por una escritura fuera de límites (Out-of-Bounds Write) al procesar secuencias de datos comprimidos manipulados. La explotación exitosa permite a un atacante remoto no autenticado provocar la caída del servicio (Denegación de Servicio - DoS), filtrar fragmentos de memoria del proceso o lograr la ejecución remota de código (RCE) arbitrario en servidores que descompriman peticiones o respuestas manipuladas con codificación Brotli.

Resumen técnico:

- Identificador principal: CVE-2026-111497 (Node2Shell / HTTP2Shell).
- Severidad: Crítica / Alta (CVSS v4.0: 9.2 Vector: CVSS:4.0/AV:N/AC:H/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N).
- Causa raíz: Escritura fuera de límites (Out-of-Bounds Write - CWE-787) y valor de retorno no verificado (CWE-252) en el decodificador nativo de Brotli (decode.c).
- Mecanismo de falla:
- El descompresor nativo de Brotli no maneja correctamente una transición de cambio de bloque inválida en el flujo de datos comprimidos.
- Al alcanzar un estado imposible en la transmisión, el decodificador interpreta erróneamente la condición como una solicitud de más datos de entrada y continúa la ejecución en un estado interno no válido.
- Esto provoca que el motor escriba más allá del límite del búfer reservado en memoria (heap), permitiendo la corrupción de memoria, caídas del proceso (SIGSEGV) o la ejecución de código.

- Vectores de entrada: Solicitudes HTTP entrantes con encabezado Content-Type: br o Content-Encoding: br hacia aplicaciones que utilicen middleware de análisis de cuerpo (como Express 5 con body-parser 2.x), o respuestas HTTP salientes procesadas automáticamente mediante la función nativa fetch() o librerías de cliente en Python (urllib3, httpx).
- Estado de explotación: Divulgada oficialmente el 21 de octubre de 2026. Equipos de investigación de Imperva demostraron la ejecución de código (PoC funcional) en imágenes oficiales de Node.js 24.22.0 (Linux arm64). No se reporta explotación activa masiva en el entorno real al momento de la publicación, pero existen múltiples pruebas de concepto públicas.
- Versiones afectadas:
- Node.js: Ramas 26.x (anteriores a 26.12.1), 24.x (anteriores a 24.22.1) y 22.x (anteriores a 22.23.4). Todas las ramas fuera de soporte (EOL como 20.x, 18.x, 16.x) contienen el componente vulnerable y no recibirán parches oficiales.
- Bun: Versiones anteriores a 1.4.4.
- Ecosistema Python: Paquetes PyPI brotli (versiones 1.0.3 a 1.2.0) y brotlicffi (versiones anteriores a 1.2.1.0).
- Infraestructura adicional: Envoy, Istio y contenedores que compilen estáticamente la librería Brotli.

Impacto potencial:

- Ejecución remota de código (RCE) no autenticada: Concede a un atacante la capacidad de ejecutar comandos arbitrarios en el sistema operativo con los privilegios del proceso que ejecuta Node.js, Bun o la aplicación afectada.
- Denegación de servicio (DoS) por colapso repetido de procesos: Permite provocar fallos de segmentación (SIGSEGV) recurrentes que interrumpen la disponibilidad de la aplicación web ante peticiones entrantes o respuestas salientes manipuladas.
- Fuga de memoria del proceso (Memory Disclosure): Permite la lectura parcial de la memoria del heap, exponiendo variables de entorno, claves API, tokens de sesión y credenciales de bases de datos residentes en la memoria del trabajador.
- Compromiso extendido de la cadena de suministro y clientes HTTP: Afecta no solo a servidores web que reciben tráfico, sino a agentes de integración, raspadores (scrapers), servicios de webhooks o herramientas de desarrollo que consuman peticiones desde URLs de terceros mediante fetch() o clientes HTTP habilitados con Brotli.

Recomendaciones de mitigación:

1. Actualización prioritaria del entorno de ejecución: Actualizar de inmediato los runtimes de Node.js a las versiones corregidas oficiales (26.12.1, 24.22.1 o 22.23.4), Bun a la versión 1.4.4 o superior, y los paquetes de Python brotli a v1.2.1 y brotlicffi a v1.2.1.0. Reconstruir e implementar inmediatamente todas las imágenes de contenedor afectadas.
2. Bloqueo y filtrado perimetral en WAF / Proxy: Si la actualización no puede aplicarse inmediatamente, configurar el WAF o proxy inverso para rechazar o descartar solicitudes entrantes que contengan el encabezado Content-Encoding: br o cuerpos comprimidos en Brotli en rutas no indispensables.
3. Desactivación de descompresión automática en middleware: Configurar las opciones de descompresión en los analizadores de cuerpo de la aplicación (por ejemplo, deshabilitar la opción de inflado en Express/body-parser mediante express.json({ inflate: false })) para evitar invocar el motor de Brotli.
4. Auditoría de contenedores, artefactos y monitoreo de procesos: Realizar inventarios detallados para identificar ejecutables integrados en imágenes Docker o entornos virtuales de Python que no reporten la vulnerabilidad en gestores de paquetes tradicionales, y monitorear reinicios o caídas anómalas en los procesos de Node.js/Bun.

Prioridad: Crítica.

Ampliar información:

- https://thehackernews.com/2026/10/researchers-demonstrate-node2shell-rce.html
- https://www.imperva.com/blog/node2shell-from-brotli-data-to-remote-code-execution/
- https://www.aikido.dev/blog/node2shell-nodejs-brotli-rce-cve-2026-111497
- https://www.herodevs.com/blog-posts/cve-2026-111497-node2shell-node-js-brotli-rce
- https://www.tanium.com/blog/http2shell-find-and-fix-vulnerable-brotli-copies-with-tanium

MALWARE

CAMPAÑA DE ATAQUE A LA CADENA DE SUMINISTRO EN LITELLM 1.107.0 – PAQUETE MALICIOSO EN PYPI Y DOCKER DISTRIBUYE INFOSTEALER Y PUERTA TRASERA RAT EN ENTORNO DE EJECUCIÓN PYTHON

Se ha identificado un compromiso grave de la cadena de suministro de software en la herramienta de código abierto LiteLLM, utilizada ampliamente como pasarela y gateway unificado para modelos de Inteligencia Artificial. El 21 de octubre de 2026, atacantes lograron vulnerar el proceso de publicación del proyecto mediante el uso de tokens robados de GitHub, distribuyendo artefactos comprometidos en la versión 1.107.0 a través de PyPI y los registros de imágenes Docker (v1.107.0, latest, main-latest y main-stable). El paquete malicioso incorpora mecanismos de ejecución automática durante el inicio del intérprete de Python, desplegando un ladrón de información (Infostealer) orientado a la captura de credenciales críticas en la nube, claves API de IA y secretos del sistema, además de una puerta trasera (RAT) para la ejecución remota de código.

Resumen técnico:

- Identificador principal: Compromiso de la Cadena de Suministro en LiteLLM 1.107.0 (PyPI / Docker) / Alerta de Seguridad del 21 de octubre de 2026.
- Tipo de amenaza: Ataque a la Cadena de Suministro (Software Supply Chain Attack) / Ladrón de credenciales (Infostealer - PRIVCOLLECT) / Troyano de Acceso Remoto (RAT).
- Mecanismo de acceso e infección:
- Los atacantes manipularon la configuración de compilación de Maturin/Rust (pyproject.toml) para empaquetar de forma silenciosa el directorio cook/ dentro de los archivos wheel subidos a PyPI.
- La instalación deposita los archivos cook.pth, cook.py y sitecustomize.py en la carpeta site-packages del entorno Python.

- Al iniciar cualquier proceso de Python (incluso ejecutando comandos administrativos como pip), la directiva cook.pth fuerza la ejecución inmediata de cook.py sin requerir que la aplicación importe explícitamente el paquete litellm.
- En paralelo, sitecustomize.py se carga automáticamente al arrancar el intérprete, levantando un proceso secundario en segundo plano que consulta periódicamente infraestructura C2 para descargar y ejecutar código Python arbitrario.
- Capacidades operativas y exfiltración:
- Ladrón de credenciales (cook.py): Recopila credenciales de proveedores cloud (AWS, GCP, Azure, endpoints IMDS de metadatos), claves API de IA (OpenAI, Anthropic, Claude Code, Copilot, Cursor, Gemini CLI), claves SSH privadas, tokens de Kubernetes, secretos de CI/CD, bases de datos de navegadores y registros de la base de datos interna de LiteLLM Proxy (LiteLLM_CredentialsTable y LiteLLM_ProxyModelTable).
- Cifrado y canal C2: Los datos robados son cifrados con AES-256-CBC y RSA-4096 y exfiltrados mediante solicitudes HTTP POST hacia el dominio controlado por el atacante hxxp://www[.]litellm[.]dev/healthcheck.
- Estado de actividad y exposición: Divulgado el 21 de octubre de 2026. El paquete malicioso estuvo disponible en PyPI aproximadamente durante dos horas (11:39 a 13:23 UTC) y en registros Docker (11:46 a 14:20 UTC), acumulando decenas de miles de descargas. Se ha confirmado el uso posterior de credenciales robadas en ataques de LLMjacking y reconocimiento en entornos cloud.
- Entornos y versiones afectadas: Entornos de ejecución de Python con la versión litellm==1.107.0 instalada vía PyPI, e imágenes Docker que hayan descargado los digests maliciosos en los tags v1.107.0, latest, main-latest o main-stable. (La distribución privada LiteLLM Enterprise no se vio afectada).

Impacto potencial:

- Robo masivo y exfiltración de secretos corporativos y de IA: Permite la sustracción inmediata de claves API de modelos de IA, credenciales maestras de infraestructuras cloud, llaves SSH privadas, tokens de integración de CI/CD y variables de entorno del sistema.

- Ejecución remota de código (RCE) y persistencia persistente: Facilita el control remoto del equipo afectado mediante la puerta trasera sitecustomize.py, la cual consulta dominios externos de forma horaria para descargar e interpretar nuevas instrucciones maliciosas en memoria.
- Consumo no autorizado de recursos cloud y fraude (LLMjacking): Permite a los atacantes reutilizar las credenciales de AWS, GCP o proveedores de IA exfiltradas para realizar llamadas masivas a modelos de lenguaje y desplegar infraestructura no autorizada a expensas de la organización afectada.
- Reactivación maliciosa en tareas de mantenimiento: Al apoyarse en archivos .pth del intérprete, cualquier intento de diagnosticar o manipular el entorno mediante comandos de Python o pip install vuelve a desencadenar la carga del payload.

Recomendaciones de mitigación:

1. Aislamiento y reconstrucción completa del entorno comprometido: Si se confirma la presencia de litellm==1.107.0, aislar inmediatamente el contenedor, máquina virtual o estación de trabajo afectada; evitar ejecutar comandos como pip install en el entorno expuesto y proceder a destruir y reconstruir el entorno desde una imagen base limpia.
2. Rotación obligatoria e inmediata de todas las credenciales accesibles: Revocar y rotar de forma urgente todas las claves API de proveedores de IA, credenciales y roles de AWS/GCP/Azure, claves de bases de datos, tokens de CI/CD y claves SSH que hayan estado presentes o accesibles desde el sistema comprometido.
3. Búsqueda de artefactos e Indicadores de Compromiso (IoCs): Inspeccionar los sistemas en busca de archivos como cook.pth, cook.py, sitecustomize.py, /tmp/.pc_once_72129118c59f5636c0bb o /tmp/remote_code_loader.lock, además de bloquear tráfico saliente hacia el dominio www[.]litellm[.]dev.
4. Reversión a versión segura y fijación de dependencias (Pinning): Revertir la instalación a una versión anterior conocida como segura (por ejemplo, litellm==1.106.2 o posterior corregida), forzando el uso de archivos de bloqueo (lockfiles) con comprobación de hashes en lugar de etiquetas móviles (latest).

Prioridad: Urgente.

Ampliar información:

- https://www.redeszone.net/noticias/seguridad/version-litellm-comprometida-malware/
- https://fixear.net/noticias/litellm-investiga-una-posible-filtracion-de-tokens-de-github-tras-distribuir-una-version-maliciosa-con-robo-de-credenciales
- https://es.aikido.dev/blog/litellm-compromised-infostealer-rat
- https://origami.sa/es/blog/litellm-supply-chain-attack-ai-keys/
- https://www.microsoft.com/en-us/security/blog/2026/10/23/inside-the-litellm-1-107-0-supply-chain-compromise-credential-theft-through-python-startup-hooks/

Recomendaciones generales sobre malware:

1. Actualizar el software y sistemas operativos regularmente, ya que los atacantes suelen explotar vulnerabilidades en software desactualizado.
2. Usar antivirus y herramientas antimalware confiables, manteniéndolos siempre al día.
3. Habilitar la autenticación multifactor (MFA) para proteger cuentas sensibles.
4. Evitar correos electrónicos y enlaces sospechosos, ya que el phishing es una de las principales vías de propagación de malware.
5. Realizar copias de seguridad regulares de archivos importantes para protegerse contra pérdidas en caso de ataques como el ransomware.
6. Limitar los privilegios de usuario, evitando el uso innecesario de cuentas con permisos de administrador.

NOTICIAS DE CIBERSEGURIDAD

DESMANTELAMIENTO DE RED MSP Y DESPLIEGUE DEL MALWARE EN MEMORIA PEERSTITCH A TRAVÉS DE FORTIMANAGER Y FORTIOS (CVE-2024-47575 Y CVE-2025-61624)

Investigadores de Palo Alto Networks Unit 42 y diversas firmas de ciberseguridad han documentado una intrusión avanzada atribuida al clúster de amenaza CL-STA-0954 (también rastreado como UNC5820). La operación comprometió un dispositivo central de gestión FortiManager perteneciente a un proveedor de servicios gestionados (MSP) de telecomunicaciones, permitiendo la exfiltración de configuraciones, credenciales y material criptográfico de más de 500 firewalls FortiGate subordinados. En fases posteriores, los atacantes desplegaron PEERSTITCH, un implante fileless altamente evasivo para arquitectura Linux ARM64 que reside únicamente en la memoria de los procesos nativos del sistema operativo FortiOS.

Resumen técnico:

- Identificador principal: Operación del clúster CL-STA-0954 / UNC5820 (Implante PEERSTITCH y vulnerabilidades CVE-2024-47575 y CVE-2025-61624).
- Tipo de amenaza: Malware en memoria (Fileless / In-Memory Implant) / Ataque a la cadena de suministro en infraestructura de gestión (MSP) / Escalada de privilegios.
- Mecanismo de falla y modo de operación:
- Explotación inicial de FortiManager (CVE-2024-47575 / FortiJump): Registro no autorizado del dispositivo virtual localhost (FMG-VMTM23017412) sobre el puerto TCP 541 (FGFM), logrando la exfiltración del archivo .tm que contenía bases de datos, credenciales y material criptográfico de más de 500 firewalls FortiGate.
- Reentrada y escalada a root (CVE-2025-61624): Acceso a la red interna mediante sesiones VPN SSL no autorizadas utilizando contraseñas descifradas y posterior escalada a privilegios de root mediante un path traversal en el motor de auto-scripts del CLI (../../../../setup).
- Inyección de PEERSTITCH en memoria: Inyección del implante vía ptrace en la memoria de los demonios /bin/miglogd y /bin/iked (reservando búferes mmap con permisos rwxp en /proc/<pid>/mem) y modificación de instrucciones NOP (0xAA0003E0) para desviar la ejecución y omitir validaciones en sesiones VPN etiquetadas.

- Persistencia C2 y evasión forense: Creación de un canal C2 persistente mediante el módulo Apache mod_md (/tmp/apache2/conf/md-acme.conf) para ejecutar reverse shells SSH con /bin/node, además del borrado activo de logs y la desvinculación automática del binario cargador en disco.
- Estado de actividad: Campaña activa detectada e investigada en octubre de 2026. La vulnerabilidad CVE-2024-47575 forma parte del catálogo KEV de CISA y CVE-2025-61624 se encuentra confirmada en explotación activa masiva en el entorno real.
- Entidades y sectores afectados: Proveedores de servicios gestionados (MSP) de telecomunicaciones y más de 500 firewalls FortiGate gestionados (afectando tanto a la infraestructura del MSP como a las redes perimetrales de sus clientes corporativos).

Impacto potencial:

- Compromiso masivo de la cadena de suministro de seguridad: Permite acceder a las configuraciones, claves privadas, certificados y credenciales de administración de más de 500 firewalls FortiGate a partir de la vulneración de un único sistema gestor FortiManager.
- Alteración y bypass silencioso de controles en memoria: Otorga la capacidad de interceptar y modificar el comportamiento de los demonios miglogd e iked en FortiOS, permitiendo a los atacantes establecer sesiones VPN SSL no autorizadas sin dejar archivos en disco ni activar alertas en herramientas EDR tradicionales.
- Persistencia criptográfica prolongada post-parcheo: Exfiltra el material criptográfico y las claves de confianza entre la plataforma de gestión y los dispositivos administrados, manteniendo el acceso legítimo de los atacantes hasta que cada clave y certificado sea rotado individualmente en cada firewall.
- Cegado de la supervisión perimetral y evasión forense: Interrumpe deliberadamente los demonios de registro y telemetría, borra registros de caídas (crash logs) y elimina cuentas y objetos de script maliciosos inmediatamente después de su ejecución para obstaculizar el análisis forense.

Recomendaciones para mitigar el riesgo:

1. Actualización urgente de firmware en FortiManager y FortiOS: Aplicar inmediatamente las versiones corregidas de firmware en FortiManager (7.6.1, 7.4.5, 7.2.8, 7.0.13, 6.4.15 o 6.2.13) para corregir CVE-2024-47575, y en FortiOS (7.6.5 o 7.4.10) para corregir CVE-2025-61624.
2. Rotación obligatoria de credenciales y material criptográfico: Realizar la rotación manual e individualizada en cada FortiGate gestionado de todas las contraseñas administrativas, claves precompartidas (PSK) de VPN, certificados SSL, llaves privadas y secretos de integración TACACS, RADIUS y LDAP.
3. Aislamiento de interfaces de gestión (Zero Trust): Restringir la exposición pública de la interfaz FGFM (puerto TCP 541) de FortiManager mediante políticas de acceso locales (local-in policies) y habilitar la opción set fgfm-deny-unknown enable para denegar conexiones de dispositivos no autorizados.
4. Auditoría forense en memoria y cacería de amenazas (Threat Hunting): Inspeccionar la memoria de los procesos miglogd e iked (/proc/<pid>/maps) en busca de regiones ejecutables anónimas rwxp de 16 KB, auditar la presencia del archivo /tmp/apache2/conf/md-acme.conf, identificar dispositivos anómalos registrados como localhost (FMG-VMTM23017412) y aplicar las reglas YARA e IoCs reportados (23.240.77.233, 47.18.167.159).

Prioridad: Urgente.

Ampliar Información:

- https://ciberseguridad.blog/peerstitch-implante-fortigate-fortimanager/
- https://securityaffairs.com/201077/malware/peerstitch-malware-how-attackers-compromised-500-firewalls-through-fortimanager.html
- https://secarma.com/16-10-2026-peerstitch-malware-fortios-cve-2025-61624
- https://gestion.8layer-latam.com/blog/view/280
- https://ciberconcienciadigital.com/noticia.php?id=1216