INSTEL & SEGURIDAD

Informe consolidado de seguridad

Vulnerabilidades, parches y respuesta operativa
Edición del 24 de octubre de 2026

Elaboración y análisis: Pedro Matías Cacivio
Perito Informático de la Nación

Destinatarios: responsables de sistemas de organismos públicos y empresas.

Resumen ejecutivo

La respuesta requiere atender tanto las vulnerabilidades de ejecución de código en servicios web como la exposición de credenciales en cadenas de suministro. Las prioridades propuestas por Instel son corregir runtimes expuestos, restringir interfaces de administración, revisar la confianza de certificados Java y tratar la ejecución de LiteLLM comprometido como un incidente.

Período y cronología

Ventana de revisión: 23/10/2026 02:17 a 24/10/2026 10:17, America/Buenos_Aires (UTC−03:00), 32 horas. Las actualizaciones con hora comprobada se distinguen de publicaciones fechadas sólo por día y de antecedentes anteriores. Una fecha de consulta no constituye fecha de divulgación.

Tema	Fecha comprobada	Tratamiento
PHP / Google-Mandiant	23/10, 19:20 Argentina	Nueva guía de respuesta dentro de la ventana
LiteLLM	23/10, 18:00 Argentina	Análisis definitivo dentro de la ventana
Palo Alto	Actualización 24/10	Ampliación de firmas; aviso original 21/10
PHP, Java/eMudhra, Ruby y Microsoft	Publicaciones del 23/10	Contexto fechado: las páginas no precisan hora
Node.js y GitLab	21/10	Antecedentes de correcciones disponibles
NGINX	14/10	Antecedente de seguridad; no es parche nuevo

Criterio operativo de Instel

La criticidad técnica debe cruzarse con exposición, función del activo y evidencia de compromiso. Un parche instalado no acredita recuperación de un equipo comprometido. Un WAF habilitado tampoco acredita cobertura si existe acceso directo al origen o tráfico fuera de sus límites de inspección.

Las fuentes primarias se enlazan en cada sección. El descubrimiento y la investigación se atribuyen a sus autores; los procedimientos identificados como recomendaciones de Instel son elaboración propia.

1. PHP: corrección y nueva guía de respuesta

Fallas originales y versiones corregidas — 21/10

PHP publicó 8.2.35, 8.3.36, 8.4.27 y 8.5.12 para CVE-2026-109618 y CVE-2026-109619, que pueden permitir ejecución remota de código sin autenticación. Google/Wiz describen errores de estado de memoria ante falta de memoria y referencias a memoria liberada; el aviso de Google informa severidad crítica, 9,2. La investigación se atribuye a PHP y Wiz Research.

Cambio verificado dentro de las 32 horas

Google actualizó su publicación del 21/10 el 23/10 a las 15:20 PDT, equivalente a las 19:20 de Argentina. Agregó guía de Mandiant sobre monitoreo, parcheo y remediación. Requiere actualizar y redesplegar servicios e imágenes, y trata las reglas WAF como controles compensatorios. La guía refuerza controles de entrada, límites de inspección, contención del proceso y monitoreo de actividad anómala.

Fuente: Google Cloud / Mandiant — publicación y actualización horaria

Aviso PHP del 23/10: CVE-2026-115560

PHP recomienda session.upload_progress.enabled=0 hasta la siguiente corrección ordinaria. El vector requiere que la configuración use php_binary o php_serialize y que la aplicación cambie a php mediante ini_set() antes de session_start(), con el progreso de cargas habilitado. PHP considera poco habitual esa combinación. Mantener la mitigación ya aplicada; no confundirla con una actualización que corrija todas las fallas de PHP2Shell.

Fuente: PHP — avisos oficiales del 21 y 23 de octubre

Procedimiento recomendado por Instel

1. Inventariar el PHP realmente utilizado por cada sitio: binario, servicio, pool, contenedor, imagen base y tareas programadas. La versión del intérprete de consola puede diferir de la del servicio web.

2. Actualizar desde PHP o desde el proveedor de la distribución. Confirmar los backports por el aviso del paquete; no decidir solamente por el número de versión visible. Reiniciar los trabajadores y redesplegar imágenes.

3. Si se demora el cambio, aplicar la configuración temporal en el contexto de arranque apropiado y verificar su valor efectivo. Restringir accesos al origen y funciones de carga que el servicio no necesita; probar que los controles no interrumpan trámites o cargas legítimas.

4. Validar autenticación, sesiones, carga de archivos y tareas críticas. Conservar versión anterior y posterior, hora del reinicio, pruebas funcionales y registros de errores. Si hay indicios de intrusión, preservar evidencia y revisar secretos accesibles al proceso web.

No extrapolar la existencia de una vulnerabilidad a una intrusión concreta. La detección de errores o reinicios de procesos requiere correlación con solicitudes y actividad del sistema.

2. Oracle Java y eMudhra: confianza de certificados

Aviso original — 21/10; guía actualizada — 23/10

CVE-2026-91403 afecta JSSE en Oracle Java SE y GraalVM. Oracle asigna CVSS 3.1 de 9,1, explotable por red sin autenticación, con impacto alto en confidencialidad e integridad. El aviso trata la desconfianza de certificados intermedios de eMudhra; no es una vulnerabilidad de Oracle Database.

Producto	Versiones afectadas enumeradas por Oracle
Oracle Java SE	8u511, 8u511-perf, 11.0.33, 17.0.21, 21.0.13, 25.0.5, 27.0.1
GraalVM for JDK	17.0.21 y 21.0.13
GraalVM Enterprise Edition	21.3.20

Fuente: Oracle — CVE-2026-91403, matriz y ruta de parches

Medidas adicionales de la guía oficial

La guía del 23/10 indica actuar incluso después de actualizar a 27.0.1.0.2, 25.0.5.0.2, 21.0.13.0.2, 17.0.21.0.2, 11.0.33.0.2 u 8u511 b32. Propone retirar las raíces indicadas del almacén de confianza, con respaldo previo, o configurar comprobación de revocación. Deben reiniciarse las instancias y repetirse las acciones para cada runtime.

Para TLS documenta com.sun.net.ssl.checkRevocation=true y com.sun.security.enableCRLDP=true. Para JAR firmados, desde JDK 17, indica verificación con revocación y modo estricto. Proporciona CheckCerts.java para evaluar el runtime. La fecha publicada es 23/10, sin hora: se conserva como contexto técnico fechado, sin asignarle pertenencia horaria exacta.

Fuente: Java Operations — instrucciones eMudhra y comprobador oficial

Aplicación y verificación recomendadas por Instel

1. Localizar todos los JDK/JRE: servidores de aplicaciones, ejecutables empaquetados, servicios Windows, contenedores, agentes y herramientas administrativas. Asociar cada proceso a su runtime y a su almacén real de confianza.

2. Respaldar los almacenes antes de modificarlos y analizar dependencias de certificados. La eliminación de raíces puede interrumpir integraciones legítimas; la comprobación de revocación necesita acceso a los servicios correspondientes y pruebas de comportamiento ante fallas de red.

3. Ejecutar el comprobador del proveedor desde una fuente verificada y aplicar la alternativa adecuada al despliegue. Conservar resultados antes y después, parámetros de JVM y evidencia del reinicio.

4. Probar conexiones TLS salientes y validación de artefactos firmados. No desactivar la validación de certificados para resolver errores posteriores al cambio. Investigar cada cadena rechazada y coordinar su reemplazo con el proveedor del servicio.

3. Node.js: antecedente de seguridad del 21/10

El proyecto publicó diez correcciones, cuatro de severidad alta. Las versiones objetivo son 26.12.1, 24.22.1 y 22.23.4, o posteriores de la misma rama. Los identificadores y el alcance por rama se detallan a continuación.

CVE-2026-	Severidad	Impacto resumido	Ramas
111497	Alta	Brotli: corrupción de memoria y posible RCE	26 / 24 / 22
105482	Alta	DNS: truncamiento por NUL	26 / 24 / 22
105483	Alta	TLS: validación de IPv4 con dígitos no ASCII	26 / 24
106069	Alta	TLS: normalización divergente de IP	26 / 24 / 22
105481	Media	HTTP: aceptación de respuesta anticipada	26 / 24 / 22
106072	Media	TLS: IPv4 no canónica	26 / 24 / 22
106074	Media	Valores aleatorios repetidos en snapshots	26 / 24 / 22
106070	Baja	Contenido ZIP tras validación fallida	26
106071	Baja	Permisos: concesiones por comodines	26
106073	Baja	Permisos: fs.realpathSync()	26 / 24 / 22

Brotli requiere que la aplicación descomprima datos controlados por un atacante mediante node:zlib. Las condiciones difieren entre vulnerabilidades; no todas afectan a todas las ramas. La actualización original y la última actualización de la página son del 21/10, fuera de las 32 horas.

Fuente: Node.js — aviso, créditos, versiones y descargas oficiales

Plan de actualización recomendado por Instel

Inventariar copias embebidas, imágenes base, runners de integración continua, funciones serverless y ejecutables distribuidos. Actualizar el runtime del artefacto que se ejecuta, además del instalado en el host. Reconstruir snapshots y aplicaciones ejecutables; reemplazar los artefactos desplegados y retirar los anteriores del circuito de publicación.

Antes de desplegar, probar descompresión, conexiones TLS, resolución DNS y permisos de archivos utilizados por la aplicación. Mantener activada la validación de certificados. Si se restringe temporalmente el procesamiento de contenido comprimido no confiable, documentar el alcance: esa restricción no corrige los otros errores del lote.

La verificación de cierre debe registrar la versión del proceso real y el digest de la imagen, no sólo el resultado de node --version en una terminal administrativa. Repetir la validación en réplicas, workers y entornos de recuperación.

4. GitLab y Ruby: aplicación y runtime

GitLab: parche crítico original del 21/10

CVE-2026-110264 permite, bajo las condiciones del aviso, ejecución de código sin autenticación al compilar expresiones regulares especialmente preparadas. GitLab asigna CVSS 3.1 de 10,0 y acredita el reporte a joaxcar. El alcance comienza en 7.0 y se divide por rama; las versiones de corrección son las de la tabla.

Rama	Primera versión corregida para CVE-2026-110264
18.11	18.11.13
19.0	19.0.10
19.1	19.1.9
19.2	19.2.9
19.3	19.3.6
19.4	19.4.3
19.5	19.5.1

El lote también corrige CVE-2026-15723, vinculada con enlace de identidades OmniAuth/LDAP, y CVE-2026-112212, relacionada con documentos OAuth/OIDC en caché. Los backports de ramas anteriores no deben interpretarse como corrección de todos los problemas del lote. GitLab.com ya ejecuta la versión parcheada, según el aviso.

Fuente: GitLab — alcance por CVE y ruta oficial de actualización

Ruby: correcciones publicadas el 23/10

Ruby publicó 4.0.9, 3.4.12 y 3.3.13 con todas las correcciones de Onigmo relacionadas con CVE-2026-89078, CVE-2026-93577, CVE-2026-97229 y CVE-2026-110264. El riesgo aparece cuando la aplicación compila expresiones controladas por un atacante. Regexp.timeout y limitar la longitud no ofrecen protección suficiente. Si no puede actualizarse, Ruby recomienda evitar compilar expresiones no confiables. La publicación sólo precisa el día; se incluye como contexto fechado.

Fuente: Ruby — correcciones, requisitos y correspondencia con GitLab

Recomendaciones de Instel

Actualizar GitLab por su procedimiento oficial, con respaldo y comprobación de migraciones. No sustituir manualmente su Ruby empaquetado como atajo. En aplicaciones Ruby propias, inventariar los puntos que compilan patrones de usuarios y probar la actualización del runtime.

Validar inicio de sesión, integración LDAP/SAML, repositorios y pipelines. Conservar registros de fallas de workers y de renderizado para investigación; un error aislado no acredita explotación. Revisar por separado los secretos accesibles a los runners si aparecen indicios de compromiso.

5. LiteLLM: respuesta a compromiso confirmado

Novedad dentro de la ventana

LiteLLM publicó el análisis definitivo el 23/10 a las 21:00 UTC, 18:00 de Argentina. Corrige conclusiones preliminares: una clave expuesta en mayo no fue revocada, permitió obtener un token de propietario de GitHub y facilitó la extracción de secretos y alteración de la publicación. El incidente de distribución ocurrió el 21/10.

Fuente: LiteLLM — análisis definitivo y cronología

Están afectados los siete wheels de litellm==1.107.0 y las imágenes asociadas a v1.107.0, latest, main-latest y main-stable durante el intervalo del incidente. El proveedor retiró o redirigió los artefactos y recomendó volver a 1.106.2 y reconstruir entornos. Una etiqueta actual no identifica necesariamente el contenido ejecutado antes.

Fuente: LiteLLM — artefactos, ventanas de distribución y recuperación

Investigación de Microsoft del 23/10

Microsoft confirmó que iniciar Python podía ejecutar el código malicioso sin importar LiteLLM. Analizó un recolector de credenciales, un cargador y un módulo que intentaba leer credenciales y configuración de modelos en la base del proxy. Advierte no iniciar Python ni pip en un entorno sospechoso para comprobar la instalación. Sus observaciones en Windows y macOS coinciden con comportamiento de recolección, cifrado y conexiones salientes.

El estudio de Microsoft documenta una amenaza de cadena de suministro. No constituye un aviso de vulnerabilidad ni un parche de Windows, Active Directory o SQL Server.

Fuente: Microsoft Security Research — mecanismos y respuesta

Secuencia recomendada por Instel

1. Contener los entornos afectados de forma coordinada y preservar evidencia antes de reconstruir. Examinar archivos, inventarios de paquetes e imágenes desde un entorno confiable, sin activar el intérprete sospechoso.

2. Correlacionar registros de descarga y ejecución, etiquetas y digests. Incluir estaciones de desarrollo, runners, cachés, contenedores y repositorios internos de paquetes.

3. Reconstruir desde una base confiable. Revocar y renovar desde un equipo limpio los secretos alcanzables por la identidad comprometida, incluyendo nube, repositorios, proveedores de IA y bases de datos según el alcance comprobado.

4. Revisar el uso posterior de esas identidades, procesos sobrevivientes y persistencia. Documentar qué evidencia respalda el cierre. Desinstalar el paquete o cambiar de versión, por sí solos, no revierten el robo de credenciales.

6. Perímetro y servidor web

Palo Alto: aviso original 21/10, actualización 24/10

El historial del aviso incorpora nuevas firmas y versiones de contenido el 24/10. El alcance en PAN-OS y Panorama es la gestión HTTPS y XML API; GlobalProtect portal, gateway y Clientless VPN no están afectados por este vector. La evaluación específica es CVSS 4.0 de 8,7, con impacto en disponibilidad. El fabricante declara no conocer explotación maliciosa contra sus productos.

Las correcciones de las ramas 10.2, 11.1, 11.2, 12.1 y 12.2 figuran como futuras. El aviso ofrece fechas estimadas por hotfix: deben confirmarse antes de planificar el cambio. Las firmas se distribuyen en Applications and Threats 9162, 9163, 9164 y posteriores. La protección requiere encaminar gestión por el plano de datos, configurar certificado, descifrar el tráfico y aplicar prevención.

Fuente: Palo Alto PSIRT — estado, hotfixes, firmas e historial

Instel recomienda restringir gestión a redes confiables o bastión, comprobar desde el exterior el bloqueo y preservar una vía autorizada de administración. Verificar la política aplicada al flujo real. Preparar respaldo y recuperación; instalar firmware oficial cuando esté disponible. No instalar paquetes PHP independientes dentro del appliance.

NGINX: antecedente del 14/10

El changelog fecha NGINX 1.31.7 el 14/10. El listado oficial marca 1.31.7+ y 1.30.6+ como no vulnerables a los siguientes cinco problemas de severidad media. NGINX 1.31.8 fue publicado el 21/10; no debe presentarse como un parche surgido en estas 32 horas.

CVE	Condición o componente
CVE-2026-15176	alias combinado con rewrite
CVE-2026-93479	Evaluación de scripts
CVE-2026-95101	Inyección de cuerpo en HTTP/3
CVE-2026-97629	Falsificación de sesión con tickets TLS
CVE-2026-97205	Uso después de liberación con AIO

Fuente: NGINX — avisos y rangos afectados por caso

Fuente: NGINX — fechas originales de las versiones

Instel recomienda cotejar módulos y configuración efectiva con cada aviso, actualizar por la distribución mantenida y validar el paquete corregido. Probar configuración antes de recargar y comprobar el reemplazo de workers. Verificar proxy inverso, TLS, sesiones y rutas de aplicaciones después del cambio.

7. Fortinet: antecedente de actualización coordinada

Fortinet anunció el 21/10 actualizaciones para FortiOS, FortiManager y FortiAnalyzer previstas para el 22/10, junto con versiones de FortiMail y FortiAuthenticator. El fabricante describe correcciones críticas y altas e insta a priorizar los equipos expuestos a Internet. Este antecedente conserva su fecha original.

Versiones objetivo publicadas por el fabricante

Producto	Ramas y versiones de actualización
FortiOS	8.0 → 8.0.2; 7.6 → 7.6.8; 7.4 → 7.4.13; 7.2 → 7.2.14
FortiManager	8.0 → 8.0.2; 7.6 → 7.6.8; 7.4 → 7.4.12; 7.2 → 7.2.13
FortiAnalyzer	8.0 → 8.0.2; 7.6 → 7.6.8; 7.4 → 7.4.12; 7.2 → 7.2.13
FortiMail	8.0 → 8.0.2; 7.6 → 7.6.7; 7.4 → 7.4.9; 7.2 → migrar a 7.4 o posterior
FortiAuthenticator	8.0 → 8.0.4; 6.6 → 6.6.11

Aplicar esas versiones o posteriores según la rama y la ruta admitida. Fortinet anunció los avisos públicos para el 10/11/2026. El lote no permite atribuir individualmente un CVE, una puntuación o explotación a todas sus correcciones.

Fuente: Fortinet — anuncio, tabla y recomendaciones oficiales

Procedimiento recomendado por Instel

1. Identificar modelo, producto, rama, versión y función de cada equipo. Verificar compatibilidad con administradores centrales, analizadores, clientes y componentes de alta disponibilidad. No extrapolar la situación de una rama a otra que no aparece en la tabla.

2. Respaldar configuración y comprobar recuperación. Definir la ventana de cambio considerando el impacto en conectividad, autenticación y correo. Validar la secuencia de actualización indicada por Fortinet para el equipo concreto.

3. Mientras se completa el cambio, restringir gestión HTTPS/SSH a direcciones autorizadas, retirar acceso administrativo externo innecesario y verificar MFA de administradores. Estas medidas reducen exposición; no acreditan corregir cada vulnerabilidad del lote.

4. Después del cambio, registrar firmware y estado de todos los nodos, verificar sincronización y servicios, comprobar accesos autorizados y revisar cambios de configuración o autenticaciones inusuales.

5. Incorporar los avisos individuales del PSIRT cuando se publiquen. Mantener la trazabilidad entre el activo, el firmware aplicado y cada vulnerabilidad que el fabricante confirme, sin trasladar conclusiones de explotación entre productos.

8. Plan institucional de remediación

La prioridad operativa siguiente es una recomendación de Instel & Seguridad. Debe adaptarse a la exposición y a la continuidad de cada servicio.

Situación	Acción prioritaria	Evidencia de cierre
Ejecución de LiteLLM comprometido	Activar respuesta a incidentes	Reconstrucción, revocación y revisión de uso posterior
PHP o GitLab vulnerable expuesto	Corregir y validar; contener mientras tanto	Versión efectiva, pruebas y revisión de registros
Gestión PAN-OS expuesta	Restringir acceso; comprobar prevención	Prueba externa y política aplicada
Runtime Java con confianza afectada	Aplicar guía eMudhra y reiniciar	Comprobación de certificados y pruebas TLS
Node.js / Ruby / NGINX pendientes	Actualizar y redesplegar	Artefactos corregidos en todos los nodos

Registro mínimo por activo

Registrar identificador, función, responsable, exposición, versión inicial, aviso aplicable, medida temporal, versión final, hora del cambio y pruebas. Incorporar nodos de respaldo, imágenes base y entornos de recuperación para evitar reintroducir versiones vulnerables.

Bases de datos y credenciales

La respuesta a un incidente de aplicación debe incluir las credenciales con las que accede a sus bases de datos. Rotarlas cuando el alcance lo justifique, limitar privilegios y revisar auditoría y accesos anómalos. Esto no equivale a atribuir una falla al motor de base de datos. Lo mismo se aplica a cuentas de dominio y servicios de infraestructura accesibles desde el host comprometido.

Preservación y verificación posterior

Ante evidencia de intrusión, conservar registros y copias relevantes con origen, operador, fecha y zona horaria. Calcular SHA-256 y documentar transferencias. El hash permite comprobar integridad respecto de una referencia; no acredita por sí solo autenticidad de origen.

Cerrar el cambio cuando se comprueben tanto la corrección técnica como el servicio: autenticación, sesiones, integraciones, transacciones críticas, recepción de registros y ausencia de accesos administrativos inesperados. Mantener seguimiento después de restablecer la operación.

Responsabilidad y atribución

Elaboración y análisis: Pedro Matías Cacivio — Perito Informático de la Nación. Instel & Seguridad. Las investigaciones originales corresponden a los fabricantes e investigadores enlazados en cada sección.

Actualización editorial: 24/10/2026. Destino: instelseguridad.com/publicaciones-e-investigaciones/.