Los actores maliciosos están utilizando el directorio «mu-plugins» en sitios de WordPress para ocultar código malicioso con el objetivo de mantener acceso remoto persistente y redirigir a los visitantes a sitios fraudulentos.

Los «mu-plugins» (abreviatura de «must-use plugins») son complementos ubicados en un directorio especial dentro de WordPress («wp-content/mu-plugins») que se ejecutan automáticamente sin necesidad de activarlos manualmente desde el panel de administración. Esta característica hace que este directorio sea un lugar atractivo para alojar malware.

> *«Este enfoque representa una tendencia preocupante, ya que los mu-plugins no aparecen en la interfaz estándar de administración de complementos de WordPress, lo que los hace menos visibles y más fáciles de ignorar durante las verificaciones de seguridad rutinarias»*, explicó Puja Srivastava, investigadora de Sucuri, en un análisis.

Durante las investigaciones realizadas por la empresa de seguridad web, se han identificado tres tipos de código PHP malicioso en este directorio:

- `"wp-content/mu-plugins/redirect.php"`: redirige a los visitantes a un sitio externo malicioso.
- `"wp-content/mu-plugins/index.php"`: proporciona funcionalidades similares a una web shell, permitiendo a los atacantes ejecutar código arbitrario al descargar un script PHP remoto alojado en GitHub.
- `"wp-content/mu-plugins/custom-js-loader.php"`: inyecta spam en el sitio infectado, probablemente con la intención de promover estafas o manipular el posicionamiento SEO. Esto se logra reemplazando todas las imágenes del sitio con contenido explícito y redirigiendo los enlaces salientes a dominios maliciosos.

El archivo `"redirect.php"` se hace pasar por una actualización del navegador para engañar a los usuarios y hacer que instalen malware capaz de robar información o descargar

cargas adicionales de código malicioso.

«El script incluye una función que identifica si el visitante es un bot. Esto permite excluir a los rastreadores de motores de búsqueda y evitar que detecten el comportamiento de redirección», explicó Srivastava.

Este tipo de ataques forman parte de una estrategia más amplia en la que los ciberdelincuentes utilizan sitios de WordPress comprometidos para engañar a los visitantes y hacer que ejecuten comandos maliciosos de PowerShell en sus computadoras con Windows. Estos comandos suelen disfrazarse como una verificación de Google reCAPTCHA o Cloudflare CAPTCHA, una táctica conocida como *ClickFix*, con el objetivo de distribuir el malware *Lumma Stealer*.

Además, los sitios de WordPress comprometidos también están siendo utilizados para inyectar JavaScript malicioso que puede redirigir a los usuarios a dominios de terceros no deseados o actuar como un skimmer para robar información financiera ingresada en páginas de pago.

Aún no se ha determinado con certeza cómo han sido vulnerados estos sitios, pero las causas más probables incluyen el uso de complementos o temas con fallos de seguridad, credenciales de administrador comprometidas y errores en la configuración del servidor.

De acuerdo con un informe reciente de Patchstack, los actores de amenazas han estado explotando activamente cuatro vulnerabilidades de seguridad en plugins de WordPress desde

el inicio del año:

- CVE-2024-27956 (Puntuación CVSS: 9.9) – Vulnerabilidad de ejecución arbitraria de SQL sin autenticación en *WordPress Automatic Plugin – AI content generator and auto poster plugin*.
- CVE-2024-25600 (Puntuación CVSS: 10.0) – Vulnerabilidad de ejecución remota de código sin autenticación en el tema *Bricks*.
- CVE-2024-8353 (Puntuación CVSS: 10.0) – Vulnerabilidad de inyección de objetos PHP para ejecución remota de código en el plugin *GiveWP*.
- CVE-2024-4345 (Puntuación CVSS: 10.0) – Vulnerabilidad de carga arbitraria de archivos sin autenticación en el complemento *Startklar Elementor Addons for WordPress*.

Para mitigar estos riesgos, los propietarios de sitios WordPress deben asegurarse de mantener sus complementos y temas actualizados, realizar auditorías periódicas en busca de malware, usar contraseñas seguras y configurar un firewall de aplicaciones web para bloquear solicitudes maliciosas y prevenir inyecciones de código.