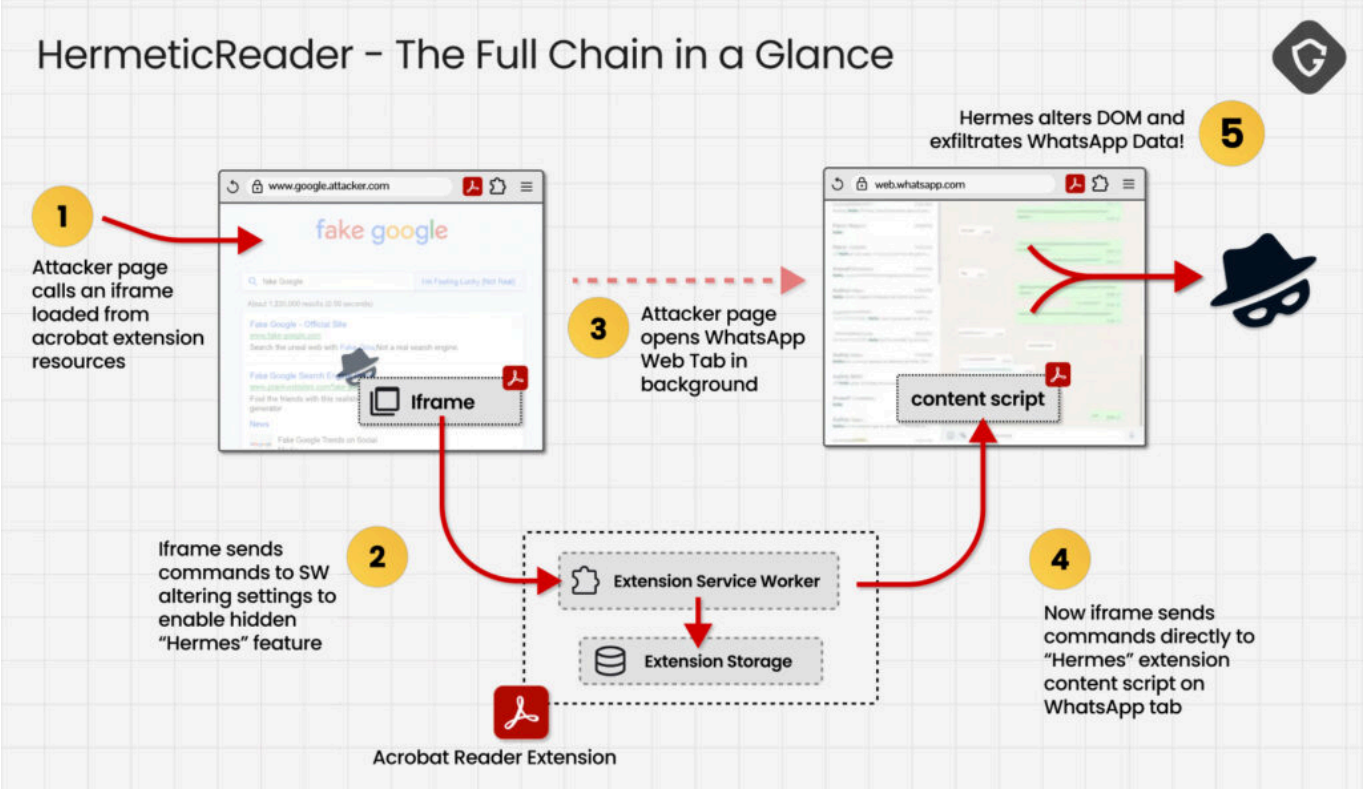
Investigadores de ciberseguridad han revelado detalles sobre una cadena de vulnerabilidades, ya corregida, presente en la extensión de Adobe Acrobat para Chrome, utilizada por más de 314 millones de personas. De haber sido explotada, esta falla habría permitido el secuestro silencioso de la información de WhatsApp de los usuarios.

La vulnerabilidad fue denominada HermeticReader por Guardio Labs y está registrada oficialmente como CVE-2026-48294, con una puntuación CVSS de 7.4. Se clasifica como una vulnerabilidad de divulgación de información entre distintos orígenes (cross-origin) perteneciente a la categoría Universal Cross-Site Scripting (UXSS). El problema afecta a todas las versiones de la extensión (ID: efaidnbmnnnibpcajpcglclefindmkaj) hasta la versión 26.5.2.2, incluida.

La explotación exitosa de esta vulnerabilidad permite evadir la política del mismo origen (*same-origin policy*) del navegador y acceder a datos asociados con la sesión activa de la víctima en diferentes dominios. Sin embargo, requiere interacción por parte del usuario, quien debe ser persuadido para visitar una URL especialmente diseñada por el atacante o interactuar con una página web comprometida que active la ruta vulnerable del complemento.

En términos prácticos, un atacante puede aprovechar esta debilidad para obtener acceso de lectura a información vinculada a sesiones autenticadas en diferentes orígenes. Entre los datos comprometidos puede encontrarse contenido protegido perteneciente a aplicaciones web de terceros abiertas en el navegador de la víctima.

«El escenario es sorprendentemente común: una página controlada por el atacante, diseñada para parecer un sitio al que un usuario llegaría mediante resultados de búsqueda, correos de marketing u otros enlaces habituales», señaló el investigador de Guardio Labs, Shaked Biner, en un informe compartido con *The Hacker News*. *«El usuario, que ya tiene instalada la extensión de Adobe Acrobat, simplemente accede a esa página.»*



«La página activa un componente inactivo dentro de la extensión y obtiene acceso directo a WhatsApp Web. En cuestión de segundos, la vista renderizada de WhatsApp Web —la lista de chats, nombres de contactos, mensajes, nombre del perfil y el contenido de la conversación abierta— queda completamente expuesta al atacante.»

Uno de los aspectos más relevantes de esta vulnerabilidad es que no exige que el atacante instale malware, robe credenciales mediante phishing o capture las cookies de sesión del usuario. Basta con que la víctima visite una página web especialmente preparada para desencadenar el ataque.

La secuencia completa del ataque se desarrolla de la siguiente manera:

- Una página controlada por el atacante carga un elemento iframe proveniente de los

recursos de la extensión.
- El iframe envía instrucciones para modificar la configuración y activar el motor Hermes, responsable de la integración con WhatsApp en la extensión cuando está habilitada la marca de función «floodgate-add».
- La página maliciosa abre WhatsApp Web en una pestaña del navegador que permanece en segundo plano.
- Posteriormente, el iframe obtiene el identificador numérico de esa pestaña y envía comandos directamente al motor Hermes para interactuar con ella.
- Finalmente, el motor modifica WhatsApp Web al inyectar un formulario POST en su DOM, con el objetivo de extraer la información mostrada en la aplicación.

«¿Por qué el envío de un formulario permite sacar el contenido de los chats fuera del origen de WhatsApp? La explicación se encuentra en dos características definidas en las especificaciones HTML: un elemento option sin el atributo value envía como valor el contenido de texto que contiene, y dicho contenido corresponde a la concatenación de todo el texto renderizado dentro del nodo. Al mover el cuerpo activo del documento, el valor enviado por el elemento option termina siendo todo el texto renderizado de la página», explicó Biner.

«El segundo factor es que la Política de Seguridad de Contenido (CSP) de WhatsApp Web no define la directiva form-action. Según la especificación, esa ausencia permite que un formulario de nivel superior envíe información hacia cualquier origen. En consecuencia, es el propio WhatsApp quien realiza la navegación y envía mediante un POST el contenido renderizado de su DOM al servidor controlado por el atacante, para después mostrar la respuesta que este devuelva.»

Como resultado, un actor malicioso puede aprovechar HermeticReader para obtener la lista de conversaciones visibles, los nombres de los contactos, las vistas previas de los mensajes, el nombre del perfil y el contenido de la conversación abierta en el momento del ataque.

«La industria suele concentrar su atención en las clases de explotación más llamativas y da por sentado que nadie examinará en profundidad los componentes fundamentales del

sistema», concluyó Guardio. *«La verdadera amenaza está en la composición: vulnerabilidades presentes en los niveles más básicos de la infraestructura pueden combinarse hasta provocar el colapso de todo el sistema. Cuanto mayor sea la base de usuarios de un producto, más tiempo puede permanecer este problema sin ser detectado.»*