Los responsables del mantenimiento del lenguaje de programación PHP, publicaron una actualización sobre el incidente de seguridad que salió a la luz a fines marzo, indicando que los piratas informáticos pudieron haberse apoderado de una base de datos de usuarios que contiene sus contraseñas para realizar cambios no autorizados en el repositorio.

> *«Ya no creemos que el servidor git.php.net se ha visto comprometido. Sin embargo, es posible que la base de datos de usuario master.php.net se filtró»*, dijo Nikita Popov el 6 de abril.

El 28 de marzo, actores no identificados utilizaron los nombres de Rasmus Lerdord y Popov para enviar confirmaciones maliciosas al repositorio «psp-src» alojado en el servidor git.php.net que implicaba agregar una puerta trasera al código fuente PHP en una instancia de un ataque a la cadena de suministro de software.

Aunque esto se trató inicialmente como un compromiso del servidor git.php.net, una investigación adicional sobre el incidente reveló que las confirmaciones fueron el resultado de presionarlas mediante HTTPS y autenticación basada en contraseña, lo que los llevó a sospechar una posible filtración del archivo de base de datos master.php.net.

> *«Git.php.net (intencionalmente) soporta empujar cambios no solo a través de SSH (usando la infraestructura de Gitolite y la criptografía de clave pública), sino también a través de HTTPS. Este último no usó Gitolite, sino que usó git-http-backend detrás de la autenticación Apache 2 Digest contra la base de datos de usuario master.php.net»*, dijo Popov.
>
> *«Es notable que el atacante solo hace algunas conjeturas sobre los nombres de usuario y se autentica con éxito una vez que se ha encontrado el nombre de usuario correcto. Si bien no tenemos ninguna evidencia específica para esto, una posible explicación es que la base de datos de usuarios de master.php.net se ha filtrado,*

> *aunque no está claro por qué el atacante necesitaría adivinar los nombres de usuario en ese caso».*

Además, el sistema de autenticación master.php.net está en un sistema operativo muy antiguo, al igual que su versión de PHP, lo que aumenta la posibilidad de que los atacantes también hayan aprovechado una vulnerabilidad en el software para organizar el ataque.

Como consecuencia, los encargados del mantenimiento migraron master.php.net a un nuevo sistema main.php.net con soporte para TLS 1.2, además de restablecer todas las contraseñas existentes y almacenarlas utilizando bcrypt en lugar de un simple hash MD5.