Se ha revelado una vulnerabilidad de seguridad en el plugin LiteSpeed Cache para WordPress que podría posibilitar a usuarios no autenticados aumentar sus privilegios.

Identificada como CVE-2023-40000, la vulnerabilidad fue corregida en octubre de 2023 con la versión 5.7.0.1.

> *«Este plugin presenta una vulnerabilidad de almacenamiento [cross-site scripting] en todo el sitio sin autenticación y podría permitir que cualquier usuario no autenticado acceda a información sensible, lo que, en este caso, podría resultar en el aumento de privilegios en el sitio de WordPress mediante la ejecución de una sola solicitud HTTP»,* señaló el investigador de Patchstack, Rafie Muhammad.

LiteSpeed Cache, utilizado para mejorar el rendimiento del sitio, cuenta con más de cinco millones de instalaciones. La versión más reciente del plugin es la 6.1, lanzada el 5 de febrero de 2024.

La compañía de seguridad de WordPress indicó que CVE-2023-40000 es el resultado de la falta de saneamiento de la entrada del usuario y de la escape de la salida. La vulnerabilidad tiene su origen en una función denominada update_cdn_status() y puede replicarse en una instalación estándar.

> *«Dado que la carga XSS se coloca como una notificación de administrador y esta notificación puede mostrarse en cualquier punto final de wp-admin, esta vulnerabilidad también podría activarse fácilmente por cualquier usuario con acceso al área de wp-admin»,* agregó Muhammad.

```
            if (isset($result['summary'])) {
                $this->_summary['cdn_dns_summary'] = $result['summary'];
            }
            $this->cls('Cloud')->set_linked();
            $nameservers = esc_html($result['nameservers']);
            $this->cls('Conf')->update_confs(array(self::O_QC_NAMESERVERS => $nameservers, self::O_CDN_QUIC => true));
            Admin_Display::succeed('🚀 ' . __('Congratulations, QUIC.cloud successfully set this domain up for the CDN. Please
update your nameservers to:', 'litespeed-cache') . $nameservers);
        } else if (isset($result['done'])) {
            if (isset($this->_summary['cdn_setup_err'])) {
                unset($this->_summary['cdn_setup_err']);
            }
            if (isset($this->_summary['cdn_verify_msg'])) {
                unset($this->_summary['cdn_verify_msg']);
            }
            $this->_summary['cdn_setup_done_ts'] = time();

            $this->_setup_token = '';
            $this->cls( 'Conf' )->update_confs( array( self::O_QC_TOKEN => '', self::O_QC_NAMESERVERS => '' ) );
        } else if ( isset($result[ '_msg' ] ) ) {
            $notice = $result[ '_msg' ];
            if ( $this->conf( Base::O_QC_NAMESERVERS )) {
                $this->_summary[ 'cdn_verify_msg' ] = $result[ '_msg' ];
                $notice = array('cdn_verify_msg' => $result[ '_msg' ]);
            }
            Admin_Display::succeed( $notice );
            $this->cls('Conf')->update_confs(array(self::O_QC_TOKEN => '', self::O_QC_NAMESERVERS => ''));
        } else if (isset($result['_msg'])) {
            $notice = esc_html($result['_msg']);
            if ($this->conf(Base::O_QC_NAMESERVERS)) {
                $this->_summary['cdn_verify_msg'] = $notice;
                $notice = array('cdn_verify_msg' => $notice);
            }
```

Esta divulgación llega cuatro meses después de que Wordfence revelara otra vulnerabilidad XSS en el mismo plugin (CVE-2023-4372, puntuación CVSS: 6.4) debido a un saneamiento insuficiente de la entrada y a la escape de la salida en atributos suministrados por el usuario. Esta falla fue corregida en la versión 5.7.

> *«Esto posibilita que atacantes autenticados con permisos de contribuidor o superiores inyecten scripts web arbitrarios en páginas que se ejecutarán cada vez que un usuario acceda a una página inyectada»,* detalló István Márton.