Una vulnerabilidad descubierta en el subsistema de control de tráfico (*traffic control*) del kernel de Linux puede permitir que un usuario local sin privilegios obtenga acceso como *root* en los sistemas afectados.

La falla, identificada como CVE-2026-46331 y conocida como «pedit COW», corresponde a una escritura fuera de límites (*out-of-bounds write*) en la acción de edición de paquetes (*act_pedit*), lo que provoca la corrupción de memoria compartida perteneciente a la caché de páginas (*page cache*). Apenas un día después de la asignación del CVE, el 16 de junio, se publicó una prueba de concepto (PoC) completamente funcional. Red Hat clasificó esta vulnerabilidad con un nivel de severidad Importante.

El exploit no modifica el archivo almacenado en disco. En su lugar, altera únicamente la copia en memoria de un binario *setuid root* (`/bin/su`), inserta una pequeña carga útil (*payload*) y ejecuta esa versión modificada con privilegios de *root*. Como resultado, las verificaciones de integridad del archivo continúan indicando que el binario permanece intacto, aun cuando el atacante ya dispone de una consola privilegiada.

Para que la explotación sea posible deben cumplirse dos condiciones: que el módulo act_pedit pueda cargarse y que los espacios de nombres de usuario sin privilegios (*unprivileged user namespaces*) estén habilitados, ya que estos proporcionan la capacidad CAP_NET_ADMIN dentro del espacio de nombres, requisito indispensable para activar la vulnerabilidad.

En las pruebas realizadas sobre sistemas RHEL y Debian, ambas condiciones estaban presentes.

Funcionamiento de la vulnerabilidad

La herramienta tc de Linux permite modificar encabezados de paquetes durante su tránsito mediante una acción denominada pedit. La función del kernel encargada de esta operación, tcf_pedit_act(), debería generar una copia privada de los datos antes de aplicar cualquier modificación, siguiendo el mecanismo conocido como *copy-on-write* (COW).

No obstante, la función únicamente validaba el rango de escritura una vez, antes de conocer los desplazamientos (*offsets*) definitivos. Algunas claves de edición determinan dichos desplazamientos únicamente durante la ejecución. Cuando esto ocurre, la escritura se realiza fuera de la región previamente copiada, provocando que el kernel modifique una página compartida de la caché en lugar de una copia privada. Si esa página corresponde a un archivo almacenado en caché, la representación del archivo en memoria queda corrompida.

Este patrón de vulnerabilidad no es nuevo. Casos como Dirty Pipe, Copy Fail, DirtyClone y Dirty Frag presentan una característica común: una ruta rápida (*fast path*) del kernel termina escribiendo sobre una página de memoria que no posee de forma exclusiva, afectando directamente a la *page cache*.

La principal diferencia en esta ocasión radica en el punto de entrada. Un usuario sin privilegios puede configurar acciones tc desde el interior de un espacio de nombres de usuario, obteniendo así la capacidad CAP_NET_ADMIN necesaria para explotar la vulnerabilidad.

Sistemas afectados

El autor de la prueba de concepto informó que la escalada de privilegios desde un usuario sin privilegios hasta *root* fue exitosa en RHEL 10 y Debian 13 (Trixie), distribuciones donde los espacios de nombres de usuario sin privilegios se encuentran habilitados por defecto.

En Ubuntu 24.04, la explotación fue posible únicamente mediante perfiles de AppArmor que todavía permiten el uso de estos espacios de nombres. En contraste, Ubuntu 26.04 bloquea este método de forma predeterminada gracias a perfiles de AppArmor más restrictivos, aunque el kernel subyacente continúa siendo vulnerable.

Las actualizaciones de seguridad varían según cada proveedor:

- Debian corrigió la vulnerabilidad en Debian 13 (Trixie) mediante su canal oficial de seguridad. Las versiones Debian 11 y Debian 12 continúan apareciendo como

vulnerables.
- Ubuntu mantiene como afectadas todas las versiones con soporte comprendidas entre 18.04 y 26.04, según la información publicada hasta el 25 de junio.
- Red Hat confirmó la afectación en RHEL 8, RHEL 9 y RHEL 10, mientras que RHEL 7 no figura dentro del boletín oficial.

Recomendaciones

La medida más efectiva consiste en instalar la versión corregida del kernel y reiniciar el sistema. Esta actualización debe priorizarse especialmente en entornos donde un «usuario local» no implica necesariamente un usuario de confianza, como servidores multiusuario, ejecutores de CI/CD, nodos de Kubernetes, servidores de compilación o equipos compartidos de investigación y laboratorio.

Si la actualización inmediata no es posible, existen dos medidas de mitigación capaces de interrumpir la cadena de explotación.

En aquellos sistemas que no requieran reglas tc pedit, se recomienda verificar previamente si el módulo está cargado (`lsmod | grep act_pedit`) y, posteriormente, impedir su carga mediante:

```
echo 'install act_pedit /bin/true' | sudo tee
/etc/modprobe.d/disable-act_pedit.conf
```

Como alternativa, puede deshabilitarse el uso de espacios de nombres de usuario sin privilegios estableciendo user.max_user_namespaces=0 en RHEL o kernel.unprivileged_userns_clone=0 en Debian y Ubuntu. Esta configuración elimina la capacidad CAP_NET_ADMIN utilizada por el exploit, aunque también afecta el funcionamiento

de contenedores *rootless*, algunos entornos de aislamiento utilizados por plataformas CI/CD y determinados navegadores que emplean mecanismos de *sandbox*. Por ello, se recomienda validar previamente el impacto operativo.

Debido a que la corrupción ocurre únicamente sobre la copia en memoria del archivo, las herramientas tradicionales de verificación de integridad pueden no detectar la manipulación. Limpiar la caché de páginas mediante:

```
echo 3 > /proc/sys/vm/drop_caches
```

elimina la copia alterada almacenada en memoria, pero no revierte el acceso privilegiado que el atacante pudiera haber obtenido previamente. En consecuencia, cualquier sistema comprometido mediante esta vulnerabilidad debe considerarse totalmente comprometido y responderse conforme a los procedimientos de gestión de incidentes.

La corrección del problema fue publicada inicialmente a finales de mayo en la lista de correo netdev, presentada únicamente como una corrección rutinaria de corrupción de datos. Durante varias semanas, los detalles explotables permanecieron disponibles públicamente sin estar asociados a un identificador CVE ni acompañados de una alerta de seguridad. El identificador CVE-2026-46331 fue asignado únicamente cuando la corrección se integró oficialmente el 16 de junio, mientras que la prueba de concepto funcional apareció apenas un día después. Este incidente demuestra que, frente a vulnerabilidades relacionadas con la corrupción de la *page cache* del kernel, esperar a que las herramientas de escaneo incorporen reglas de detección puede resultar insuficiente.