Daniel Pérez Bastante **Máster Hacking Ético**

¿Cómo funcionan los Sistemas Anti-Cheat a nivel de Kernel?

Los sistemas Anticheat a nivel de Kernel, como por ejemplo **Battleye** (PUBG, Destiny, DayZ, etc), **Easy Anti-Cheat** (Fortnite), **Riot Vanguard** (Valorant, LoL) o **Ricochet** (CoD) funcionan de forma similar a los antivirus EDR, es decir, **operan en el Kernel del sistema operativo**, obteniendo así acceso total al hardware y controlando las operaciones a nivel bajo del sistema operativo, como los procesos, la memoria, los drivers, la lectura y escritura de archivos, las inyecciones de código, etc. Por este motivo se consideran más "invasivos" que otros sistemas antitrampas, lo cual no se considera positivo ni negativo persé, pero sí que acarrea ciertos riesgos y desventajas como por ejemplo:

-**Invasión de la privacidad del usuario**, al poder acceder a cualquier información del equipo (por ejemplo, si tienes el Fortnite abierto pero estás viendo vídeos en YouTube o haciendo un PowerPoint para la universidad, el sistema de Easy Anti-Cheat detecta lo que estás haciendo, aunque las empresas detrás de estos sistemas aseguren no ver nada ajeno al juego).

-**Posible inestabilidad en el sistema**, ya que una mala instalación/actualización de los drivers a nivel de Kernel puede provocar crasheos o la temible **Pantalla Azul de la muerte de Windows**, o incluso provocar el malfuncionamiento de otros programas legítimos, y mostraremos un caso real en la siguiente página.

-Y el más peligroso de todos, el cual es que estos sistemas antitrampas, al funcionar en una capa tan baja del sistema operativo, si en algún momento sufriese una **vulnerabilidad** y **un atacante la explotase**, **podría tener control absoluto sobre nuestro sistema y causar bastantes daños**.

Al operar a un nivel tan bajo, estos sistemas antitrampas son capaces de identificar procesos que son invisibles a nivel de usuario, buscando actividad sospechosa que indique que el usuario podría estar haciendo trampas. Pero no todo son puntos negativos, ya que por ejemplo, uno de los puntos positivos de estos sistemas, es su alta eficacia y rapidez a la hora de encontrar trampas *(punto positivo para los jugadores legítimos, y quizá negativo para los jugadores tramposos)*.

¿Qué ocurrió con el fallo de Crowdsrike en el 2024?

A mediados de Julio del 2024, la empresa Crowdstrike lanzó una actualización para su antimalware **EDR,** Falcon, muy utilizado en equipos de Windows de aerolíneas, hospitales, bancos, servicios de emergencia, medios de comunicación, gobiernos, entre otros, la cual debido a un fallo, causó muchísimos problemas a **escala mundial**, estimándose que **más de 8 millones de ordenadores con Windows se vieron afectados por este fallo** que dejó muchos servicios inoperativos y pérdidas de información en muchísimas empresas.

Tal y como hemos introducido antes, los EDR son sistemas antimalware que operan **en el Kernel del Sistema operativo** (al igual que los sistemas antitrampas de la exposición anterior), por lo que un fallo de seguridad, un error ("bug") o cualquier fallo sobre un EDR, puede ser muy perjudicial para el sistema operativo.

Y lo que sucedió exactamente fue que un archivo de configuración incluido con la actualización, llamado **"Channel File 291"**, tenía un error con las reglas de filtrado, que no se habían verificado correctamente, causando que el EDR hiciese una lectura de memoria fuera de los límites (intentando leer entradas inexistentes), causando **crasheos y bucles en el modo recuperación** de los cuales era muy complicado salir y arrancar Windows de manera normal.

¿Cómo podría haberse evitado este incidente?

Al haberse instalado esta actualización de forma automática y masiva en todos los equipos que tuviesen Falcon Sensor operando sobre ellos *(salvándose sólo aquellos que no estuviesen conectados a Internet en ese momento)*, la mejor manera por parte de Crowdstrike para evitar este problema, hubiese sido lanzar la actualización de una forma más gradual, es decir, **poco a poco**, aparte de por supuesto haber comprobado detenidamente la actualización antes de lanzarla al público general, por ejemplo con un programa de beta testers, recibiendo feedback de unos pocos usuarios antes de lanzarla a nivel global.

Fuentes:

Sistemas antitrampas a nivel de Kernel (Inglés)

Los principales detalles del incidente de Crowdstrike (Castellano)

Channel-File-291-Incident-Root-Cause-Analysis-08.06.2024.pdf (Inglés)