TAREA COLABORATIVA - TÉCNICAS DE INFECCIÓN, EVASIÓN Y PERSISTENCIA ASOCIADAS A BOOTKITS DE WINDOWS

PatchGuard

Índice

1. Que es Kernel Patch Protection (KPP) o PatchGuard

PatchGuard es una tecnología desarrollada por Microsoft para proteger la integridad del kernel en las versiones de 64 bits (x64) de Windows. Se introdujo por primera vez en 2005 con las ediciones x64 de Windows XP y Windows Server 2003 Service Pack 1 y su objetivo principal es prevenir que el malware modifique las estructuras del kernel del sistema operativo.[1]

2. Cómo funciona PatchGuard

Patchguard se ejecuta a intervalos regulares y verifica que las estructuras protegidas del sistema en el kernel no se hayan modificado. Aunque es posible que un malware modifique una estructura y la vuelva a dejar como estaba antes de que PatchGuard vuelva a realizar el escaneo, lo que provocaría que no se detectara la modificación.

Algunas de las áreas que PatchGuard protege incluyen las tablas de descriptores de interrupción (IDT), las tablas de descriptores globales (GDT), las tablas de descriptores de llamadas al sistema (SSDT), y otras estructuras críticas del kernel.

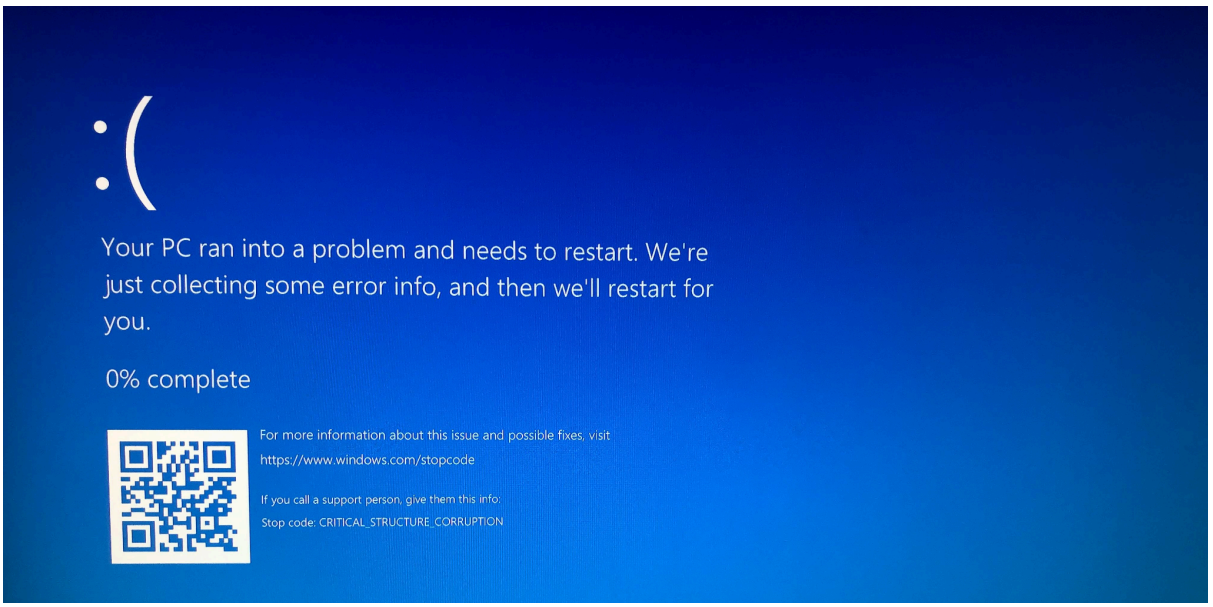
PatchGuard utiliza varias técnicas de ofuscación avanzadas para proteger su propia estructura y los datos críticos del kernel para evitar descifrar como se ha desarrollado.

Una de estas técnicas es el uso de variables globales, como **KiWaitNever** y **KiWaitAlways**, que contienen valores aleatorios generados durante el inicio del sistema. Estas variables son utilizadas por **KiInitPatchGuardContext**[2] para cifrar el contexto de PatchGuard, lo que significa que cualquier atacante que intente acceder a esta estructura necesitaría conocer la ubicación y los valores de estas variables globales. Además, PatchGuard se asegura de que estas variables y sus valores no sean fáciles de predecir ni de localizar, aumentando así la dificultad para los atacantes.

Cuando PatchGuard detecta una modificación, inmediatamente comprobará errores en el sistema (KeBugCheck[3] (0x109)[4]), lo que dará como resultado la conocida pantalla azul de la muerte (BSOD) con el mensaje: "CRITICAL_STRUCTURE_CORRUPTION".

---

[1] Protección contra revisiones del núcleo - Wikipedia, la enciclopedia libre
[2] Posts :: Demystifying PatchGuard: An In-Depth Analysis Through Practical Engineering (archive.org)
[3] KeBugCheck function (ntddk.h) - Windows drivers | Microsoft Learn
[4] Bug Check Code Reference - Windows drivers | Microsoft Learn


:(
Your PC ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.
0% complete
For more information about this issue and possible fixes, visit https://www.windows.com/stopcode
If you call a support person, give them this info:
Stop code: CRITICAL_STRUCTURE_CORRUPTION

3. Técnicas de evasión de PatchGuard

Evitar PatchGuard es una tarea muy complicada y técnicamente desafiante debido a la robustez con la que Microsoft ha diseñado esta característica de seguridad y a la poca información que existe al respecto. Sin embargo, a lo largo de los años, algunos investigadores y desarrolladores de malware han encontrado formas de evitarlo.[5]

- GhostHook (2017)[6] [7]

GhostHook explota una característica de Intel Processor Trace (Intel PT), la cual es una tecnología que permite el rastreo detallado y preciso de la ejecución de programas en la CPU. Utilizando esta técnica, un atacante puede interceptar llamadas de función en el kernel y modificar el comportamiento del sistema, eludiendo así las protecciones de seguridad establecidas por PatchGuard.

- ByePg (2019)[8] [9] [10]

Byepg es una técnica que sirve para eludir la protección de PatchGuard mediante el uso específico de hooking de excepciones.

Esta técnica modifica el comportamiento de las excepciones generadas por el kernel mediante técnicas avanzadas de hooking. Al interceptar y modificar estas excepciones, los atacantes pueden alterar las operaciones normales de PatchGuard, lo que potencialmente permite ejecutar código no autorizado en el espacio del kernel.

También existen algunos repositorios de GitHub con herramientas para evadir PatchGuard que se han ido desarrollando durante los años.

- DisPG (2015)[11]
- UPGDSED (2017)[12]
- InfinityHook (2019)[13]
- Shark (2021)[14]
- NoPatchGuardCallback (2021)[15]
- PatchGuardBypass (2023)[16]

---

[5] Uninformed - vol 3 article 3 (archive.org)
[6] GhostHook – Bypassing PatchGuard with Processor Trace Based Hooking (cyberark.com)
[7] GhostHook Attack Bypasses Windows 10 PatchGuard | Threatpost
[8] ByePg: Defeating Patchguard using Exception-hooking – Can.ac
[9] New bypass disclosed in Microsoft PatchGuard (KPP) | ZDNET
[10] GitHub - can1357/ByePg: Defeating Patchguard universally for Windows 8, Windows 8.1 and all versions of Windows 10 regardless of HVCI.
[11] Releases · tandasat/PgResarch (github.com)
[12] GitHub - hfiref0x/UPGDSED: Universal PatchGuard and Driver Signature Enforcement Disable
[13] GitHub - everdox/InfinityHook: Hook system calls, context switches, page faults and more.
[14] GitHub - 9176324/Shark: Turn off PatchGuard in real time for win7 (7600) ~ later
[15] GitHub - kkent030315/NoPatchGuardCallback: x64 Windows PatchGuard bypass, register process-creation callbacks from unsigned code
[16] GitHub - AdamOron/PatchGuardBypass: Bypassing PatchGuard on modern x64 systems

- EfiGuard (2023)[17]

4. Conclusión

PatchGuard es una función esencial en Windows x64, desempeña un papel fundamental al proteger el kernel del sistema operativo contra modificaciones no autorizadas, incluyendo potenciales ataques de bootkits o rootkits. Estos tipos de malware son conocidos por su capacidad para infiltrarse en el sistema operativo, aprovechando vulnerabilidades en el kernel para evadir detección y controlar el sistema.

[17] GitHub - Mattiwatti/EfiGuard: Disable PatchGuard and Driver Signature Enforcement at boot time