# patch 문제 write-up

강승민

Patch.exe파일을 하나 주는데 실행해보면


이러한 그림이 그려지고 클릭할 때마다 새로 그려지는 모습을 볼 수 있다. 아마도 리버싱을 통해 바이너리 패치를 해서 검은색 부분 밑에 숨겨진 글자를 보는 문제인 것 같다.

```
int __stdcall WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nShowCmd)
{
  HCURSOR CursorW; // rax
  HWND Window; // rax
  HWND v8; // rbx
  HACCEL AcceleratorsW; // rbx
  WNDCLASSEXW v11; // [rsp+60h] [rbp-A8h] BYREF
  struct tagMSG Msg; // [rsp+B0h] [rbp-58h] BYREF

  LoadStringW(hInstance, 0x67u, &WindowName, 100);
  LoadStringW(hInstance, 0x6Du, &ClassName, 100);
  v11.cbSize = 80;
  v11.style = 3;
  v11.lpfnWndProc = (WNDPROC)sub_1400032F0;
  *(_QWORD *)&v11.cbClsExtra = 0i64;
  v11.hInstance = hInstance;
  v11.hIcon = LoadIconW(hInstance, (LPCWSTR)0x6B);
  CursorW = LoadCursorW(0i64, (LPCWSTR)0x7F00);
  *(__m128i *)&v11.hbrBackground = _mm_load_si128((const __m128i *)&xmmword_1400053F0);
  v11.hCursor = CursorW;
  v11.lpszClassName = &ClassName;
  v11.hIconSm = LoadIconW(hInstance, (LPCWSTR)0x6C);
  RegisterClassExW(&v11);
  qword_140007880 = (__int64)&unk_1400078A0;
  Window = CreateWindowExW(0, &ClassName, &WindowName, 0xC80000u, 0x80000000, 0, 600, 200, 0i64, 0i64, hInstance, 0i64);
  v8 = Window;
  if ( Window )
  {
    hWnd = Window;
    dword_140007920 = 600;
    dword_140007924 = 200;
    GdiplusStartup(&unk_1400078A0, &dword_1400078A8, 0i64);
    ShowWindow(v8, nShowCmd);
    UpdateWindow(v8);
    AcceleratorsW = LoadAcceleratorsW(hInstance, (LPCWSTR)0x6D);
    while ( GetMessageW(&Msg, 0i64, 0, 0) )
    {
      if ( !TranslateAcceleratorW(Msg.hwnd, AcceleratorsW, &Msg) )
      {
        TranslateMessage(&Msg);
        DispatchMessageW(&Msg);
      }
    }
    LODWORD(Window) = Msg.wParam;
```

IDA를 이용해 열어보면 그림과 같은데 Gdiplus라는 걸로 화면을 그리는걸 알 수 있다.

```
LRESULT __fastcall sub_1400032F0(HWND a1, UINT a2, WPARAM a3, LPARAM a4)
{
  _QWORD *v5; // rbx
  __int64 v6; // rbx
  __int64 v7; // [rsp+20h] [rbp-18h] BYREF

  switch ( a2 )
  {
    case 2u:
      PostQuitMessage(0);
      return 0i64;
    case 0xFu:
      qword_140007910 = (__int64)BeginPaint(hWnd, &Paint);
      v5 = (_QWORD *)GdipAlloc(16i64);
      if ( v5 )
      {
        *v5 = 0i64;
        v5[1] = 0i64;
        v7 = 0i64;
        *((_DWORD *)v5 + 2) = GdipCreateFromHDC(qword_140007910, &v7);
        *v5 = v7;
      }
      else
      {
        v5 = 0i64;
      }
      qword_140007918 = (__int64)v5;
      sub_140002C40();
      v6 = qword_140007918;
      if ( qword_140007918 )
      {
        GdipDeleteGraphics(*(_QWORD *)qword_140007918);
        GdipFree(v6);
      }
      EndPaint(hWnd, &Paint);
      return 0i64;
    case 0x202u:
      InvalidateRect(hWnd, 0i64, 1);
      UpdateWindow(hWnd);
      return 0i64;
    default:
      return DefWindowProcW(a1, a2, a3, a4);
  }
}
```

분석을 진행한 결과 표기된 부분에 있는 함수에 들어가보면

```
  v2 = qword_140007880;
  sub_140002B80(qword_140007880, a2, 30, 470, 80, -16777216);
  sub_140002B80(v2, v3, 35, 470, 75, -16777216);
  sub_140002B80(v2, v4, 40, 470, 70, -16777216);
  sub_140002B80(v2, v5, 45, 470, 65, -16777216);
  sub_140002B80(v2, v6, 50, 470, 60, -16777216);
  sub_140002B80(v2, v7, 55, 470, 55, -16777216);
  sub_140002B80(v2, v8, 60, 470, 50, -16777216);
  sub_140002B80(v2, v9, 65, 470, 45, -16777216);
  sub_140002B80(v2, v10, 70, 470, 40, -16777216);
  sub_140002B80(v2, v11, 75, 470, 75, -16777216);
  sub_140002B80(v2, v12, 80, 400, 60, -16777216);
  sub_140002B80(v2, v13, 30, 470, 90, -16777216);
  sub_140002B80(v2, v14, 35, 470, 30, -16777216);
  sub_140002B80(v2, v15, 40, 470, 35, -16777216);
  sub_140002B80(v2, v16, 45, 470, 50, -16777216);
  sub_140002B80(v2, v17, 50, 470, 40, -16777216);
  sub_140002B80(v2, v18, 55, 400, 90, -16777216);
  sub_140002B80(v2, v19, 60, 470, 60, -16777216);
  sub_140002B80(v2, v20, 65, 470, 30, -16777216);
  sub_140002B80(v2, v21, 70, 470, 80, -16777216);
  sub_140002B80(v2, v22, 75, 470, 70, -16777216);
  sub_140002B80(v2, v23, 80, 470, 60, -16777216);
  sub_140002B80(v2, v24, 80, 470, 80, -16777216);
  sub_140002B80(v2, v25, 80, 470, 70, -16777216);
  sub_140002B80(v2, v26, 90, 470, 90, -16777216);
  v27 = qword_140007880;
  sub_1400017A0(qword_140007880, 40i64, v28, 4278190080i64);
  sub_140001C80(v27, 80i64, v29, 4278190080i64);
  sub_140002640(v27, v30, v31, 4278190080i64);
  sub_1400020F0(v27, v32, v33, 4278190080i64);
  sub_140002390(v27, v34, v35, 4278190080i64);
  sub_140001240(v27, v36, v37, 4278190080i64);
  sub_140001F20(v27, v38, v39, 4278190080i64);
  sub_140001560(v27, v40, v41, 4278190080i64);
  sub_140001C80(v27, 360i64, v42, 4278190080i64);
  sub_1400019D0(v27, v43, v44, 4278190080i64);
  sub_1400017A0(v27, 440i64, v45, 4278190080i64);
  sub_140002870(v27, v46, v47, 4278190080i64);
  return 0;
```

이러한 부분이 나오는데 이 부분들이 화면에 그림을 그려주는 부분이다.

이중 표시된 부분이 검은 부분을 그리는 부분이고 아래 부분이 그 뒤 숨겨진 글을 그리는 부분이다.

```
00002020  FF 15 3A 26 00 00 48 8B 4C 24 40 48 33 CC E8 1D  ÿ.:&..H‹L$@H3Ìè.
00002030  08 00 00 48 8B 5C 24 78 48 83 C4 50 5F 5E 5D C3  ...H‹\$xHƒÄP_^]Ã
00002040  40 53 48 83 EC 30 90 90 90 90 90 90 90 90 90 90  @SHƒì0..........
00002050  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002060  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002070  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002080  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002090  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
000020A0  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
000020B0  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
000020C0  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
000020D0  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
000020E0  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
000020F0  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002100  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002110  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002120  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002130  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002140  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002150  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002160  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002170  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
00002180  90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  ................
```

바이너리 패치를 진행하는데 오프셋을 유지하기위해 그림과 같이 아무것도 아닌 어셈블리 코드인 0x90으로 수정하였다.


이와 같이 숨겨진 글을 찾을 수 있고 이를 flag로 제출하면 문제를 해결할 수 있다.