

본 컬럼에 대한 모든 저작권은 DevGuru에 있습니다.
컬럼을 타 사이트 등에 기재 및 링크 또는 컬럼 내용을 인용 시 반드시 출처를 밝히셔야 합니다.
컬럼 들을 CD나 기타 매체로 배포하고자 할 경우 DevGuru에 동의를 얻으셔야 합니다.

© DevGuru Corporation. All rights reserved

기타 자세한 질문 사항들은 웹 게시판이나 support@devguru.co.kr으로 문의하기 바랍니다.

Application 안에 Driver 숨기기

© 2003 Devguru (Device driver Guru), Inc.

Driver 목적 중에 하나는 User Mode에서 할수 없는 일들을 Application들에게 서비스를 하기 위해서 작성을 한다. 이런 경우 Driver는 단독적인 File로 존재해 왔다. 나는 사용자에게 여러 Utilitizes들이 배포할 때 최대한 눈에 보이는 파일 개수를 줄이려고 한다.(관리 차원 에서 볼때도 적으면 적을수록 좋을것이다. -필자생각-)

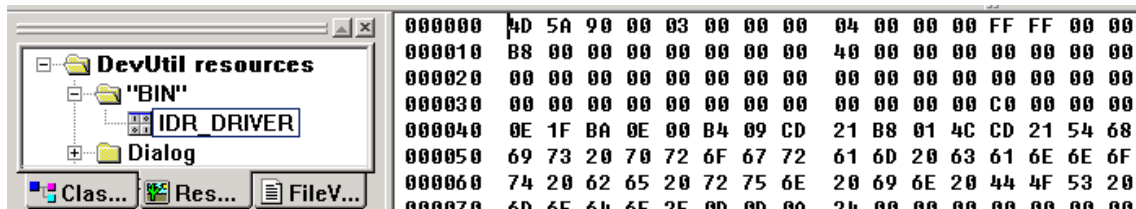
그래서 이번 컬럼에서는 Driver파일을 Application Resource에 포함 시켜 사용하는 테크닉을 설명 할 것이며, 아주 유용하게 사용 할수 있을것이다.

이 테크닉은 간단하고 손 쉬운 방법이므로 구현 하는데 무리 없이 할수 있을것이다.

VC로 작성한 Application을 중심 적으로 설명하게 될것이다.

▶ Driver를 Application 안에 숨기는 방법은 다음과 같다.

- 먼저 Driver File을 (.Sys) Application Resource에 Binary 형태로 추가를 시킨다.



- 그리고 나서 Resource를 이용하여 임시 파일을 만든다.

```

hRsrc = FindResource(hInst,MAKEINTRESOURCE(IDR_DRIVER),"Bin");
hDriverResource = LoadResource(hInst, hRsrc);
dwDriverSize = SizeofResource(hInst, hRsrc);
lpvDriver = LockResource(hDriverResource);

hfTempFile = _lcreat("PHYSACE.sys",0);

```

```
_hwrite(hfTempFile, (const char*)IpvDriver, dwDriverSize);
_close(hfTempFile);
```

- 만들어진 파일을 SCM(Service Control Manager) 함수를 이용하여 Driver를 로딩시킨다.

```
LoadDeviceDriver( SYS_NAME, Path, &m_hDriver, &error );
```

이 함수 안에서 SCM 함수들을 사용한다]

- Driver로딩 완료후 임시 파일을 만들었던 Driver를 삭제한다

```
OpenFile("PHYSACE.sys",&of,OF_DELETE);
```

지금까지 짧지만 유용한テクニック에 대하여 살펴 보았다.

본 column에 대한 문의 사항은 저희 홈페이지 QnA을 이용하시면 됩니다.

>>>>>>>>>> 실제 코드 <<<<<<<<<<<<

```
BOOL CDevUtilDlg::OpenUtilDriver()
{
    //////////////////////////////////////
    // Resource에서 드라이버 찾기..
    static OSVERSIONINFO os={0};
    os.dwOSVersionInfoSize = sizeof(OSVERSIONINFO);
    GetVersionEx(&os);

    HINSTANCE      hInst;
    HRSRC          hRsrc;
    HGLOBAL        hDriverResource;
    DWORD          dwDriverSize;
    LPVOID         IpvDriver;
```

```
HFILE      hfTempFile;
OFSTRUCT   of;

hInst = AfxGetInstanceHandle();

// If we're under NT, load the .SYS resource...
if(os.dwPlatformId == VER_PLATFORM_WIN32_NT)
    hRsrc = FindResource(hInst,MAKEINTRESOURCE(IDR_DRIVER),"Bin");
else
{
    return FALSE;
}

hDriverResource = LoadResource(hInst, hRsrc);
dwDriverSize = SizeofResource(hInst, hRsrc);
lpvDriver = LockResource(hDriverResource);

// Dump the resource out to a file
if(os.dwPlatformId == VER_PLATFORM_WIN32_NT)
    hfTempFile = _lcreat("PHYSACE.sys",0);
else
    return FALSE;

_hwrite(hfTempFile, (const char*)lpvDriver, dwDriverSize);
_lclose(hfTempFile);

char Path[256];

// Driver Loading
```

```
GetCurrentDirectory( 255, Path );
Istrcat( Path, "WWPHYSACE.sys" );
DWORD error;

if( !LoadDeviceDriver( SYS_NAME, Path, &m_hDriver, &error ) )
{
    OpenFile("devutil.sys",&of,OF_DELETE);
    return FALSE;
}
/* m_hDriver = CreateFile( "WWW.WWDEVUTIL", GENERIC_READ|GENERIC_WRITE
                        , 0, 0, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL
                        , 0 );

if( m_hDriver == INVALID_HANDLE_VALUE )
{
    OpenFile("devutil.sys",&of,OF_DELETE);
    return FALSE;
}*/

OpenFile("PHYSACE.sys",&of,OF_DELETE);
return TRUE;
}
```