The buffer size parameter to GetFileVersionInfo is the size of your buffer, no really

devblogs.microsoft.com/oldnewthing/20070329-00

March 29, 2007

Raymond Chen

The `GetFileVersionInfo` function takes a pointer to a buffer (`lpData`) and a size (`dwLen`), and that size is the size of the buffer, in bytes.

No really, that's what it is.

The application compatibility folks found one popular game which wasn't quite sure what that `dwLen` parameter meant. The programmers must have thought it meant "The size of the version resources you want to load" and called it like this (paraphrased):

```
void CheckFileVersion(LPCTSTR pszFile)
{
 BYTE buffer[1024];
 DWORD dwHandle;
 DWORD dwLen = GetFileVersionInfoSize(pszFile, &dwHandle);
 if (GetFileVersionInfo(pszFile, dwHandle, dwLen, buffer)) {
  ...
 }
}
```

"Gosh, the `GetFileVersionInfo` function wants to know how big the version info is, so we need to call `GetFileVersionInfoSize` to find out!" they must have thought.

This code worked great... for a while. It was checking the file version of the video driver. (My guess is that they were trying to detect specific video drivers so they could work around bugs in them or take advantage of driver-specific features.) But if you had a video driver whose version resource needed more than 1024 bytes of space, the program crashed with stack corruption.

I don't know whether the Windows Vista application compatibility folks decided that it was worth fixing this program's bug, since it occurred even on Windows XP. If so decided, the fix would have been fairly straightforward. Once the program was detected, we would just have had to take the value the program passed as `dwLen` and modify it according to the simple formula

```
dwLen = min(dwLen, 1024);
```

before doing the real work of loading the version information.

For those playing along at home, by the way, the correct code would go something like this:

```
void CheckFileVersion(LPCTSTR pszFile)
{
 DWORD dwHandle;
 DWORD dwLen = GetFileVersionInfoSize(pszFile, &dwHandle);
 if (dwLen) {
  BYTE *pBuffer = (BYTE*)malloc(dwLen);
  if (pBuffer) {
   if (GetFileVersionInfo(pszFile, dwHandle, dwLen, pBuffer)) {
    ...
   }
   free(pBuffer);
  }
 }
}
```

(Use your favorite memory allocation technique instead of `malloc` and `free`.)

Raymond Chen

Follow