The case of the strange NT-style path that was discovered by SearchPath

devblogs.microsoft.com/oldnewthing/20210722-00

July 22, 2021

Raymond Chen

A customer reported that the `SearchPath` function was returning an NT-style path, which is not something their program (an IIS CGI handler) could deal with. Why are they getting an NT-style path?

The problem was not reproducible when run under a debugger, so we had to resort to tracing and logging.

One theory was that IIS was putting an NT-style path in the `PATH`, and that's where it was coming from. But logging the value of the `PATH` environment variable showed no weird `\\?\` directories.

I recalled that when you have eliminated the impossible, whatever remains, however improbable, must be the truth. The `SearchPath` function looks in the directories specified in the `PATH` environment variable, and if the item can't be found in any of those directories, then it looks in the current directory.

We have ruled out the `PATH` environment variable.

Therefore, the current directory must be the one with the NT-style path.

It turns out that, for security reasons, IIS runs CGI programs via their NT path, and also uses an NT path as the current directory. It so happened that the file was found in the current directory, so that's what `SearchPath` returned.

We can even run a test program to check the theory.

```
#include <windows.h>
#include <stdio.h>

int __cdecl main()
{
  SetCurrentDirectoryW(L"\\\\?\\C:\\Users");

  wchar_t buffer[256];
  wchar_t* file;
  if (SearchPath(nullptr, L"Public", nullptr,
     256, buffer, &file)) {
     printf("%ls\n", buffer);
  }
}
```

Run this program and it prints

```
\\?\C:\Users\Public
```

Okay, that explains the problem. But how do they fix it?

You could try to convert the NT path to a Win32-style path before processing it. But really, since this is a CGI program, you probably shouldn't be grabbing files from the current directory in the first place. You aren't really in control of your current directory, and the fact that you're looking up things in the current directory means that an attacker might be able to pass paths like `..\..\wwwroot\config\jackpot.xml`, and steal files from anywhere on your system.

So the real issue is that their program is somehow reliant upon the current directory, and they should fix it to remove that dependency.

Raymond Chen

Follow