Why can't I put hotlinks in notification icon balloon tips?

devblogs.microsoft.com/oldnewthing/20040225-00

February 25, 2004

Raymond Chen

The short answer: “Because there is no NIF_PARSELINKS flag.”

The long answer:

When balloon tips were first developed, there was no ability to embed links. Consequently, programs were free to put insecure text in balloon tips, since there was no risk that they would become “live”. So, for example, a virus scanner might say “The document ‘XYZ’ has been scanned and found to be free of viruses.”

Now suppose hotlinks were supported in balloon tips. Look at how this can be exploited: I can write a web page that goes

```
<TITLE>&lt;A HREF="file:C:\Windows\system32\format.com?C:"&gt;
Party plans&lt;/A&gt;</TITLE>
```

I then rename the file to “Party plans.html”, attach it to some email, and send it to you.

You download the message and since you are a cautious person, you ask your virus scanner to check it out. The balloon appears:

Virus scan complete ×

The document ‘Party plans‘ has been scanned and found to be free of known viruses.

“Oh, how convenient,” you say to yourself. “The virus scanner even included a hotlink to the document so I can read it.”

And then you click on it and your hard drive gets reformatted.

“So why don’t you add a NIF_PARSELINKS flag, so people who want to enable hotlinks in their balloon tips can do so, and still remain compatible with people who wrote to the old API?”

(I’ve heard of one person trying to pass a TTF_PARSELINKS flag in the NOTIFYICONDATA.uFlags member and wondering why it wasn’t working. I hope it’s obvious to everybody why this had no chance of working.)

Because that would just be passing the buck. Anybody who used this proposed flag would then have to be extra-careful not to put untrusted links in their balloon tips. Most people would just say, “Wow! A new flag! That’s awesome!” and start using it without considering the serious security implications. Then somebody can trick the program into putting untrusted text into a balloon tip and thereby exploit the security hole.

“Aw, come on, who would be so stupid as to write code without considering all the security implications?”

I hope that was a joke question.

The best way to make sure things are secure is to make it impossible to be insecure.

Raymond Chen

Follow