Ransomware hack: what we can learn about VB6

By John Browne on May, 17, 2017

Friday May 12, 2017 saw the worst cyber attack in years, as a worm (WannaCry, Wanna, or Wcry) spread through computer networks encrypting files and demanding a ransom of anywhere from $300 to $600.

The UK's National Health System was hit hard, causing hospitals in England and Scotland to basically shut down and forcing ambulances to divert to functioning emergency rooms. Telefonica in Spain was hit hard. Some estimates put the number of infected computers at 75k worldwide, with large infections in Ukraine, Russia, and China. XP computers were the most susceptible.

As of Monday, May 15, Ars Technica is reporting researchers believe the worm has North Korean origins. Propagation may have occurred both via phishing emails as well as a known vulnerability in Microsoft's SMB file-sharing protocol. The worm works through your local network, then starts firing off at random IP addresses.

This is scary stuff.

Sadly, the source code for the attack apparently came from a release of archived exploits collected by the US National Security Agency (NSA). Even more sadly, Microsoft already patched their OS versions still in support **back in March** but because people either have bootlegged copies of OS or out of support versions like XP many computers were zapped. These same baddies attacked a SWIFT network in the Middle East earlier this year.

Responding to the severity of the attack, Microsoft took the unusual step of releasing a patch for three operating systems that are out of support: XP, Server 2003, and Vista.

Fortunately, Windows 10 is not vulnerable, nor are Win 7 & 8.x if current with updates. And note: the NHS in Wales was not affected, because they had recently updated their computer systems.

Why this matters

OS security and application security aren't the same thing and I'm not trying to scare you into thinking that Wcry can do something to your VB6 apps (or Silverlight or PowerBuilder ones either).

If you've got a VB6 app running on Windows 10, and you keep your OS up to date with all the Microsoft updates, you should be ok.

If you've got a VB6 app running on Windows XP, 2003, or Vista, you're probably not reading this because you're too busy trying to figure out if you've got backups of all your now-encrypted files.

What matters is that *old stuff is vulnerable*. New stuff less so.

It's really that simple.

Old stuff is vulnerable not only because well, it's old, and that means when it was created people hadn't seen (and thus armed themselves against) all the exploits that have surfaced in the intervening years.

Old stuff is vulnerable because the code is crap and no one really knows if it's buttoned up nice and tight.

Old stuff is vulnerable because it relies on the full stack of other old stuff, some of which can be used against you.

Old stuff is vulnerable because--being old--the bad guys have had more time to play around with it and find weaknesses.

And finally **old stuff is vulnerable** when it uses COM objects about which you really don't know much--maybe they are connecting to servers in Moscow warehouses but it

looks like perfectly innocent traffic if you bothered to run Fiddler all the time but you don't so you can't tell.

New stuff is safer

It just is. .NET is managed code--good luck trying to overflow a buffer. Good luck trying to write into someone else's address space. VB6 is cowboy code compared to .NET. HTML and JavaScript run in the browser--and modern browsers have chopped off support for nasty stuff like the Netscape Plugin API (NPAPI) that enabled Flash and Silverlight to be a vector for malware. C# is **clean**: you can make nice compact classes with it, keeping data private, all perfectly encapsulated. Sure you can do that with VB6 too, but is that really how that 20 year old app was written?

I doubt it.

What you don't know can kill you

The problem with exploits is we don't hear about them **before** they are released. Sometimes the way you learn about one is when you realize something terribly, horribly wrong has happened. To you. To your data. **To your customer's data**.

What's going to happen to the patients in Britain's National Health System (NHS) whose electronic health records are now irretrievable? What's going to happen to whomever is responsible for NHS still running XP long after support ended? Why was this ever an option, for even 1 second, let alone months? Somebody will surely get fired. But real people might die because of this.

Think about it: we're now in a world where ones and zeros can literally kill people.

If that doesn't scare the hell out of you, I can't help you. But it if does, give us a call. We're in the business of making old stuff go away quicker, easier, and for less money than any other reasonable alternative. And we've put everything you need to get off VB6 right here.

Topics: application modernization, VB6, ransomware, Wannacry