

Re: [Full-Disclosure] Windows covert channel

Source: <http://www.derkeiler.com/Mailing-Lists/Full-Disclosure/2003-10/1216.html>

From: Kain (kain_at_kain.org)

Date: 10/20/03

To: full-disclosure@lists.netsys.com
Date: Sun, 19 Oct 2003 23:36:26 -0500

On Sun, Oct 19, 2003 at 10:23:37PM -0400, Karl DeBisschop wrote:

> *On Sun, 2003-10-19 at 19:04, James Kelly wrote:*
> > *I seem to remember in the dim reaches of my memory a covert channel in*
> > *the Windows file system where you could paste one file at the end of*
> > *another without it being detectible when you edited the original file.*
> >
> >
> > *can someone aim me at the right "buzz phrase" that describes this so I*
> > *can Google it further?*
>
> *Many people have mentioned data streams. But since you said 'end of*
> *file' I wonder if you are referring to the DOS idea that ^Z is an end of*
> *file marker, and many apps won't look beyond it.*

I don't know enough about NTFS to know if the same concept applies, but in FAT/FAT32, your files are allocated in clusters. Therefore, given a file of size X, and a cluster size of Y, you will have $X \bmod Y$ bytes in the last allocated cluster that won't be visible through the filesystem that you can directly write and hide information in. Of course, it's possible (likely) that processes that truncate, grow, or defragment/reallocate the file on disk will lose the information, so it is spotty at best.

This sort of fun is also not too hard to pursue with other filesystems. For example, ISO9660 (cdromfs) has multiple directory and file tables, and there's nothing stopping you from hiding data all over an ISO that just browsing the filesystem would show. This sort of thing lets you create structures like hybrid ISO9660/HFS/HFS+ images for Macs.

For NTFS, I would suggest starting at <http://linux-ntfs.sourceforge.net> which has plenty of pointers to NTFS utilities and documentation.

If I were desining a covert file storage system for Win*, I would probably write an installable file-system driver that would mangle rarely-modified NTFS files, adding additional streams to them to store data. That would however, produce massive forensic evidence to an informed observer. I would also consider writing a filesystem driver that could use a (read-only/unmounted)

Full-Disclosure: Re: [Full-Disclosure] Windows covert channel

NTFS partition as it's storage, hiding its filesystem in the unallocated space of the disk.)

--

Bryon Roche

Professional {Developer,Guru,Mad Scientist}

<kain@kain.org>

PGP Key Fingerprint: FE0D EC23 6464 726A CD54 48D3 04AD 86FE 6878 ABD5

Success, recognition, and conformity are the bywords of the modern world where everyone seems to crave the anesthetizing security of being identified with the majority...Human salvation lies in the hands of the creatively maladjusted.

-- Martin Luther King, Jr.

Full-Disclosure – We believe in it.

Charter: <http://lists.netsys.com/full-disclosure-charter.html>

- application/pgp-signature attachment: [stored](#)