



Brick by brick: Platforms, Viruses, Doorstops

infectionvectors.com

October 2005

Overview

The mortar that holds much of the virus/anti-virus market together is its mythology. The idea that a virus could sneak into your system and wipe out every critical file, erase the entire hard disk, set-up a rogue web server, etc. is pretty scary stuff to the average Internet user. That's not to imply that it's all fantasy, all of the tactics mentioned have been included as the payload for various worms and viruses. However, the fear associated with malware is often well out of scope with the likelihood that one will be infected.

This report briefly examines the "bricking" Trojan, a piece of malware that attempts to render a platform unusable and what elements are responsible for the success or death of these types of malware.

Legend of Virii

Malware is often publicly known because of a payload, not a propagation mechanism. The majority of virus hoaxes are based around the "terrible consequences" of becoming the victim of the latest piece of malware (often, "erase your entire hard disk" is somewhere in the warning). Mass chaos and destruction are effects of early generation malware on particular platforms. The attention that "professional" malware creators are receiving carries with it the idea that the business worm is a more evolved creature than the Chernobyl virus.¹ Chernobyl (CIH) was one of the most destructive pieces of malware in history, erasing disk after disk during its 1999 run. Although Korean officials identified the writer, there is no indication that he received any financial gain from creating the program. Chernobyl was exceptionally successful, however, at increasing the amount of public fear with respect to computer attacks.

Without entering into a debate over the evolutionary status of any particular application or author, it is important to note the introduction of various payloads into the lifecycle of any platform's viral universe. Specifically, actions such as "write," or "erase," are basic, necessary functions of any file system. It is of little mystery that they are the payloads of first generation viruses. More complex routines are possible once a writer has both the understanding of the system and a compelling drive (reason) to invest the time into coding them. This has been seen with early malware like the above-mentioned CIH and Mypics² worm from 1999, but is not seen as much in the year 2005 (but, rest assured, plenty of writers continue to attach these types of functions to their code).

The recent Sony Playstation Portable and Nintendo DS malware, heretofore known as PSPBrick and DSBrick,⁴ fit this profile as well. They are the first Trojans for the respective game consoles; both also make the hardware unusable. The two files take the same path onto a system: they must be downloaded and executed by the system user. Why would someone do that you ask? The same reason PC users do, the files are disguised as cracked software.

PSPBrick, once executed, deletes four system files (two of which are loadcore.prx and loadexec.prx, which, even without experience with the Sony platform, sound pretty important). DSBrick erases critical sections of the platform's memory, producing the same results, an unusable piece of gaming hardware.

Magnet

Money is the most likely force attracting malware coders towards preserving the clients they infect. There is no money without exploiting the host, and no way to exploit a host that is incapable of booting. There is not a lot of profit potential by writing "bricking" Trojans for game platforms (unless, of course, you are commissioned to hit a competitor's product, but that's another story altogether). In the future, however, there may be money in stealing accounts, passwords, or digital items from gamers.⁵

The reaction of the respective game console makers is critical to the life of these Trojans. Mobile phone worms have been around for over a year now (Caribe was released in the summer of 2004) and are just beginning to take characteristics beyond simply damaging telephones (although that is still the most used and successful function of the applications). Telephone manufacturers are working on education and defensive software (as are companies such as F-Secure) – it is their response that will prove to be the key element in the life of telephone worms as well.⁶

Although the realm of vandals, bricking Trojans are also the first step in producing much more complicated pieces of malware; they are the first measure of control over a system.

References

1. CIH/Chernobyl

<http://www.symantec.com/avcenter/venc/data/cih.html>

2. Mypics

<http://www.symantec.com/avcenter/venc/data/w32.mypics.worm.html>

Guts to say Jesus Hoax

<http://www.symantec.com/avcenter/venc/data/jesus-hoax.html>

3. PSPBrick

<http://securityresponse.symantec.com/avcenter/venc/data/trojan.pspbrick.html>

4. DSBrick

<http://securityresponse.symantec.com/avcenter/venc/data/trojan.dsbrick.a.html>

Interesting aside on the malware: CNET.com thread on worm potential for Nintendo DS

<http://news.com.com/5208-1043-0.html?forumID=1&threadID=2548&messageID=13849&start=-196>

and the first prediction I found for Nintendo DS malware (from late 2004):

http://kaleem.blogware.com/blog/_archives/2004/10/7/165530.html

5. Such as the SIMS auctions that involve real money

http://www.gamespot.com/news/2005/02/08/news_6118160.html

6. One thing that has developed along with the worms is the increased amount of spam is hitting the cell phones:

<http://news.softpedia.com/news/More-spam-now-it-039-s-reaching-for-your-cellphone-145.shtml>