Global Information Assurance Certification Paper

Virus Generators and Their Implications
Bryan Fansler

Introduction

On Monday, February 12, 2001, warnings flashed up all over the internet about a new virus that was very similar to the 2000 Love Bug. "Alex Shipp, senior anti-virus technologist at MessageLabs, which scans customers email for malicious code, said the VBS/SST virus is 'spreading twice as fast as the Love Bug' "[1]. This virus only affected Windows machines running Outlook "that have not installed the patch Microsoft provided after the outbreak of the Love Bug, which used very similar propagation techniques"[1]. Fortunately, this virus did not have near the impact of the Love Bug, but it still created quite a stir. According to McAfee, a part of Network Associates and a major player in the antivirus market, "the virus has been found in 50 enterprise size companies including Fortune 500 firms"[2]. Much like the Love Bug, this virus mixed malicious code with social engineering (in the form of an attachment users would be willing to open). Rather than promising a love letter to the unfortunate recipient, however, this virus purported to be a picture of tennis star and sex symbol Anna Kournikova. Titled Anna Kournikova.jpg.vbs, this virus fooled enough people into opening it that it created quite a stir.

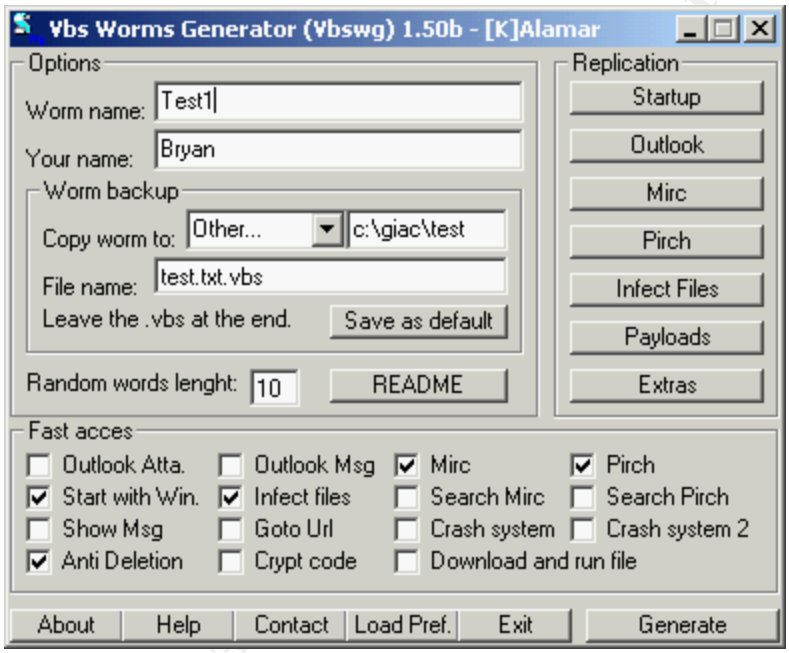
Unfortunately, viruses of this type are, in all likelihood, going to be more prevalent as time goes on. Support for this statement can be found in one of the most unlikely places of all: a confession written by the author of the virus. In an online confession written by the virus' author who is currently only identified by his online handle "OnTheFly": "I have made this virus with a Visual Basic Worm Generator, written by [K]Alamar. K. is NOT involved with this worm! I have been using this programm [sic] because I don't know any programming languages" [3]. This is an interesting and scary proposition. If someone who openly confesses not having any programming knowledge can infect "50 enterprise companies including Fortune 500 firms", information security professionals need to look into these virus generators.

Accordingly, I have done a search for the same generator used to create this worm and tested it against McAfee's VirusScan NT. The test system used was a Windows 2000 machine running VirusScan NT 4.03. The scan engine was version 4.070 and the virus definition file was version 4.0.4120, which was created on February 7, 2001. I chose that definition file as it was released 5 days prior to the VBS/SST virus discussed above. The name of this generator is VBS Worm Generator 1.50b and its author calls himself [K]Alamar [4]. It is probably necessary to provide a common sense public service announcement here: anybody planning on duplicating these tests would be well advised to disconnect their computer from their network after downloading, but before running, these programs.

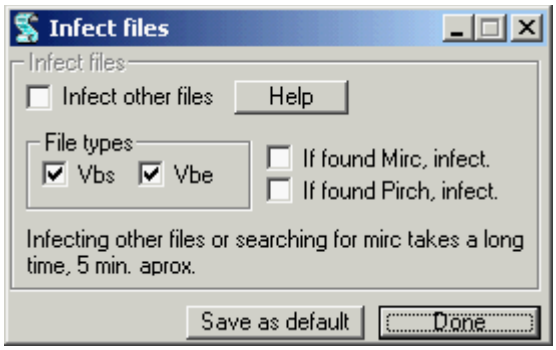
VBS Worm Generator

This is the same application used to create the VBS/SST worm discussed in the introduction. Out of curiosity, I scanned the executable itself first and found that the kit itself was classified as a virus:

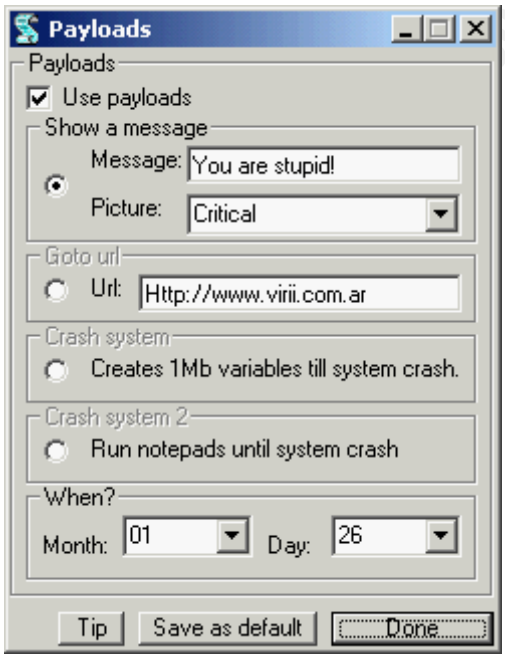
My next task was to see how easy, or difficult, it was to create a virus like the AnnaKournikova.jpg.vbs virus. This generator has a main graphical interface from which the virus can be named and all options necessary could be set:



Some of these fields are pretty self-explanatory. However, there are several features included in this generator that are downright powerful. The Outlook and mirc replication features are obviously effective methods for spreading this worm. Additionally, with the "infect files" option, the user can specify that the worm searches all local and (mapped) network drives for files with a specified infection and replace them with copies of the worm. Currently, the only files that can be replaced are .vbs and .vbe files. However, according to [K] Alamar's readme file, plans are apparently in the works for adding other file types to this screen:

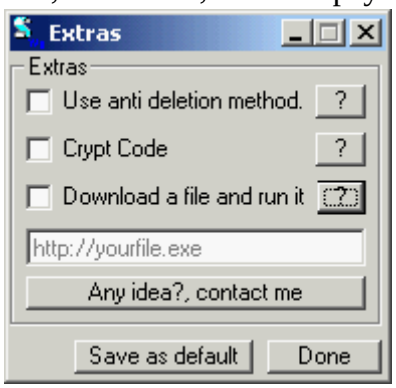


In the Payload section of this generator, the user has the option to display a custom message box with whatever message they desire (it defaults to a reassuring "You are stupid!" message). In the same way as the AnnaKoumikove.jpg.vbs virus did, the user could specify a URL to visit. Additionally, they get the options to crash the system on a scheduled basis by either generating one megabyte variables until the system runs out of memory or by opening instances of notepad.exe until the system again runs out of memory and crashes. The user gets the option to schedule when they would like the message, URL visit, or system crash to happen, and even gets a helpful tip from the creator of this program (spelling/grammar errors are the fault of the application's author): "Try to don't use crash system every day of every month if you're going to use the antideletion method, cause the antideletion method will never work." The payload screen is shown below:

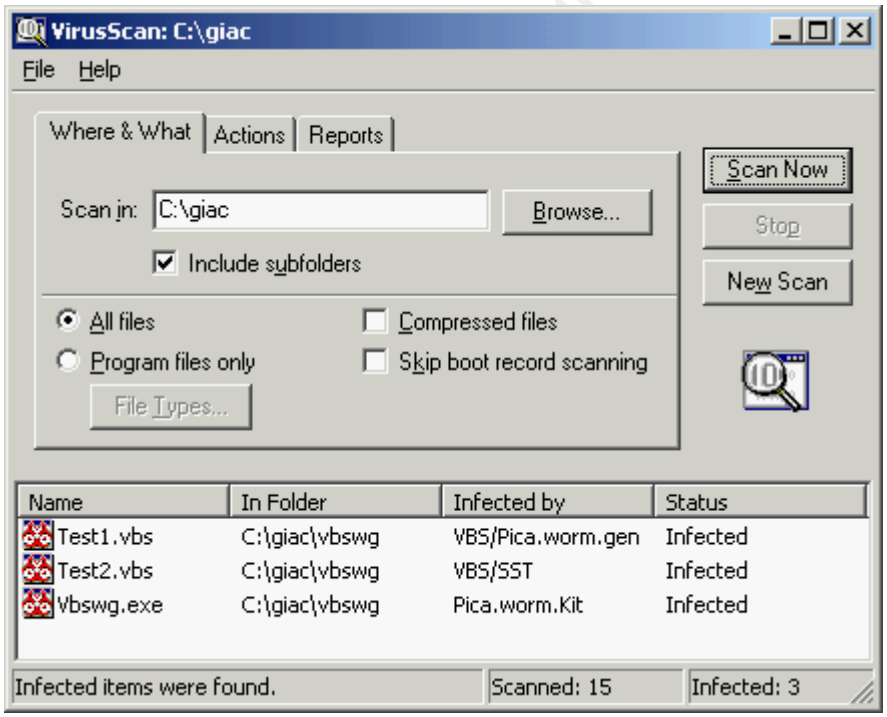


There are also various features that [K]Alamar calls "extras". These include the ability to protect against deletion by running the virus in memory and recreating the files if they become deleted. His second feature is to encrypt the code, preventing people from editing or changing it. [K]Alamar admits that the encryption scheme is fairly weak, as it

only uses the hex value of the characters' ASCII values, but this feature still adds an additional layer of complexity to any worm generated using this option. The final feature in the extras section he added is the ability to download and run a file from a given web site, IP address, etc. The payload screen is as follows:



For the purposes of this test, I created a viruses that was just set to display the default message of "You are Stupid", replace all .vbs files with itself, and replicate itself via Outlook. This file was named it test1.vbs. I also created another one, encrypted it, and named it test2.vbs. I then scanned the directory these files were stored in for viruses. As demonstrated in the following screen shot, the files were successfully detected as infected files:



Interestingly enough, Test1 and Test2 were detected as separate viruses, which can probably be attributed to the fact that Test2 was encrypted, which changed its signature.

Conclusion

As this paper has demonstrated, it can be extremely easy to create a virus that while "considered an unsophisticated and easily preventable virus by most experts… still caught many enterprises off guard"[5]. While this virus' creator, appalled at the problems he caused, admitted to the crime and turned himself in, information security professionals should not view this as an abnormal incident. According to self professed cracker "Taltos": "He lit a fire, it raged out of control, and he ended up burning himself badly. He was obviously unprepared for the furor he unleashed. If he had intended to do something truly evil, he wouldn't have been so unprepared for the consequences"[6]. One can easily infer from this calm statement that creating and unleashing a virus with the potential to cause quite a bit of havoc only takes a little bit of preparation for the firestorm to follow.

Outbreaks of automatically created viruses need to be considered, and plans need to be in place to prevent, or at least mitigate, the potential problems associated with these. As always, security administrators need to makes sure they do at least the following three things:

- Educate users about attachments and their potential dangers in order to create an atmosphere of awareness
- Maintain up to date vendor patches. If the patch Microsoft had released following the Love Bug had been put into place at a given organization, this virus would have had no effect at that organization. However, "many enterprises have been hesitant to install the patch because they were unsure of its effectiveness and because it's a bulky 26 MB download" [5].
- Keep antivirus software installed and updated. As shown earlier in this paper, a definition file created before the virus' creation date still caught the worm.

While there are a plethora of virus generators available, much like the one demonstrated above, a security administrator following these three steps can go a long way toward ensuring outbreaks, such as the AnnaKournikova.jpg.vbs outbreak, are non-events at their sites.

Endnotes

1. Leyden, John. "Anna Kournikova virus spreading like wildfire". *The Register*. http://www.theregister.co.uk/content/8/16846.html

2. Leyden John. "Anna Kournikova bug drops harmlessly onto the Net". *The Register*. http://www.theregister.co.uk/content/8/16878.html

3. http://members.tripodnet.nl/on_the_fly/index.html.

4. http://vx.netlux.org/dat/tv07.shtml

5. Drucker, David. "Anna Virus Catches IT Shops Off Guard." *Internet Week*. http://www.internetweek.com/story/INW20010216S0004

6. Dellio, Michelle. "Why Worm Writer Surrendered". *Wired News*. http://www.wired.com/news/culture/0,1284,41809,00.html