Global Information Assurance Certification Paper

Wrapping Malicious Code in Windows Shell Scrap Objects

Justin Ennis

GSEC Practical Assignment Version 1.2b

April 5, 2001

Introduction

An important aspect of Windows email-borne viruses is the appearance of the actual executable file itself. It is important to the virus writer for the file to appear as innocuous and familiar as possible to the user so that the chances of the virus being executed are increased as much as possible. One of the current techniques in use for achieving these ends involves the use of the Microsoft Shell Scrap Object. This paper will examine the most threatening aspects of the Shell Scrap Object, how to create Shell Scrap Objects and the advantages and disadvantages of current defensive measures that have been used against email deployment of viruses wrapped in Shell Scrap Objects.

The Threat

Viruses that are passed as attached emails are partially reliant upon the appearance of the attached file. Surely the impact of the virus will be greatly reduced if the file is named "Destructive virus.exe" and by the same token, the impact can be greatly increased if the file is carefully named to appeal to a broad audience. The impact can be increased even more if the file is named in such a way as to appear innocuous to the untrained eye. One of the best ways of accomplishing this is to modify the extension of the file name so that the attached file appears to be a text file or a JPEG or something else that many users will not associate with a virus. The Shell Scrap Object has two attributes that make it ideal for this task:

1. The file type is one of a handful that are permanently hidden by default on the Windows platform. Even if the folder options are set to view file extensions, the file extensions for these Scrap Objects will not be displayed. Because of this, the file can be set to have a faked extension that will display, but will be in front of the actual file extension that does not display.
2. The file type's icon looks somewhat like a text file icon. This likeness is not close enough to fool anyone familiar with a text file icon, but it can give the impression that the file type is similar to a text file and therefore may share some of the text file's attributes. Most importantly, the plain text file is widely considered safe for execution and therefore any likeness to it imparts some feeling of safety to the user that is unfamiliar with Shell Scrap Objects. This is paramount to the design of email-borne viruses since the safer the user feels with the attachment, the more likely they are to execute it and spread the virus.

As an example of these attributes Figure 1 is an image of a folder with a Shell Scrap Object and a text file in it. The attributes on this folder have been set to show the

extensions for the files, however, notice how the Shell Scrap Object does not have any extension associated with it. Notice also how the Shell Scrap Object icon slightly resembles a text file icon.

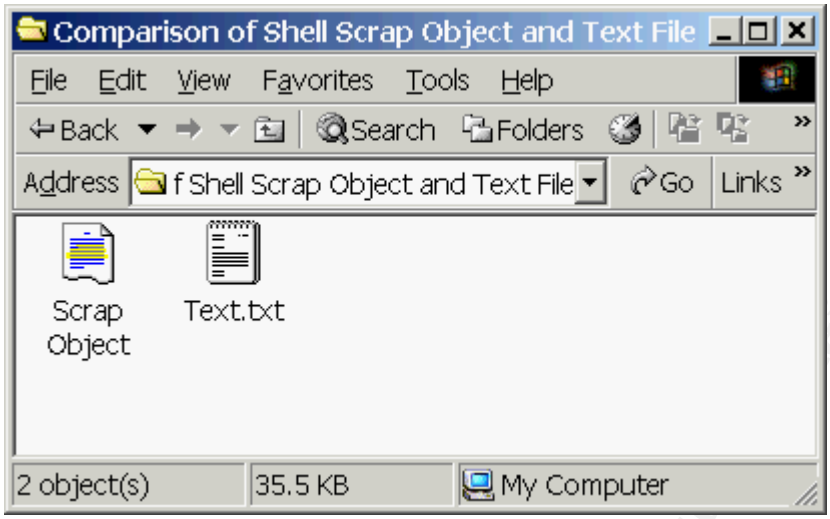

Figure 1

Figure 2 is the same folder, only the name of the Shell Scrap Object file has been modified to appear to have a .txt extension.

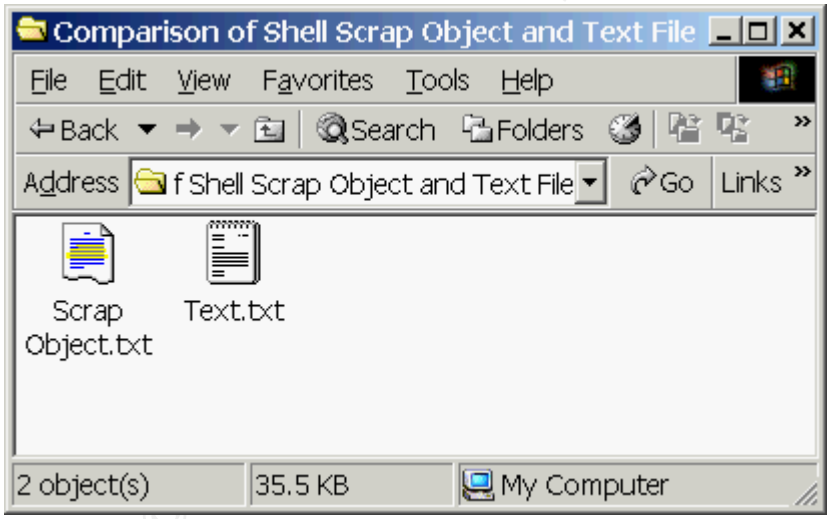

Figure 2

From the preceding figures it can be seen that the attributes of the Shell Scrap Object file do not seem to imply that the file can be executed, and this leads to the most important attribute of all, that anything that the Windows operating system can execute (.exe, .com, .bat, etc.) can be wrapped in a Shell Scrap Object and it will be executed by the operating system when the user selects to open the file. This provides a ready-made mechanism for further hiding the intent of the attached file.

How to Create Shell Scrap Objects

Wrapping executables with Shell Scrap Objects is a surprisingly simple process:

1. On a Windows system, open the WordPad application (Start->Programs->Accessories->WordPad).

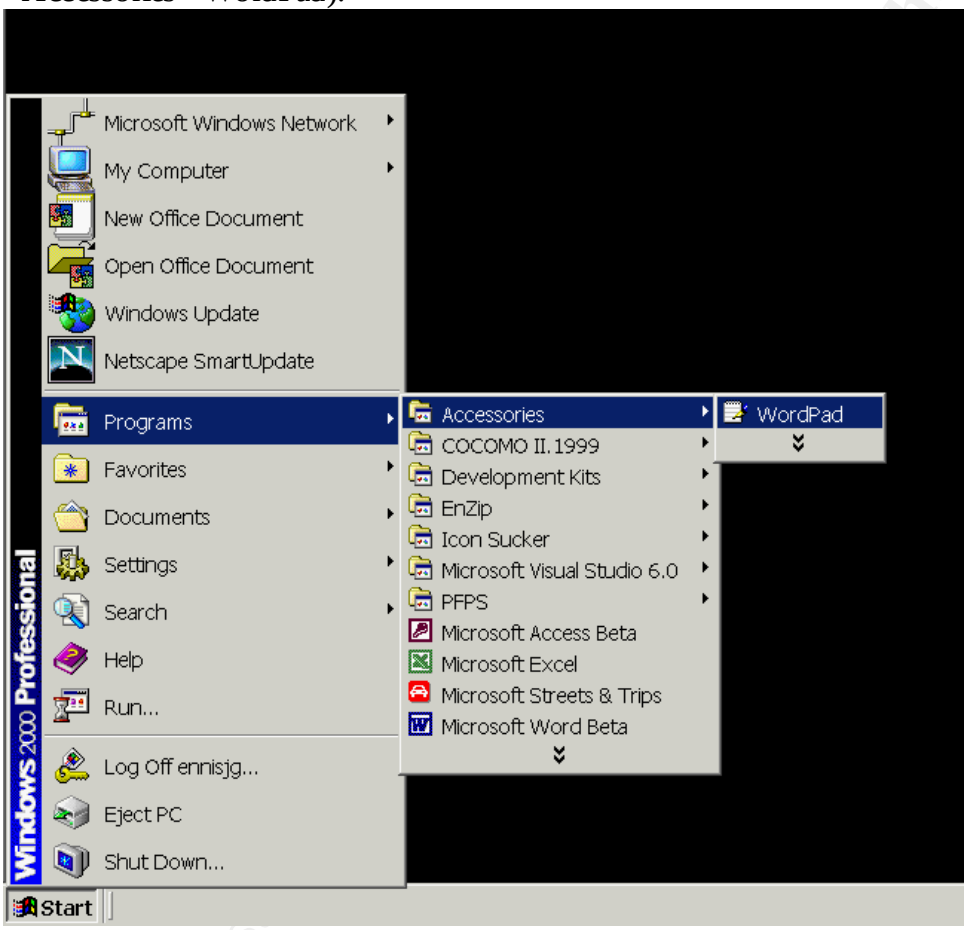

Figure 3

2. Drag an executable (.bat, .com, .exe, .vbs, etc.) file onto the WordPad document editing space.

3. Right-click on the resulting icon in WordPad and select Package Object -> Edit Package.

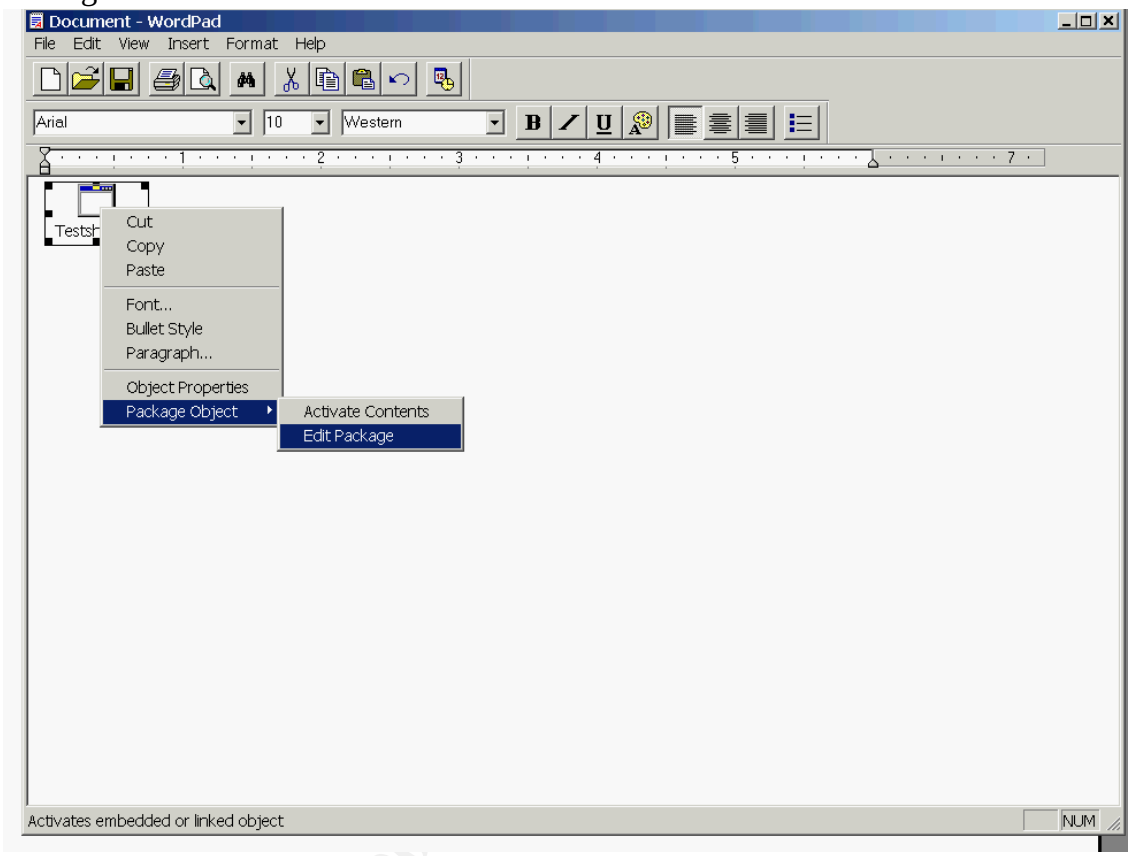

Figure 4

4. From the Object Packager window that pops up, select Edit->Copy Package

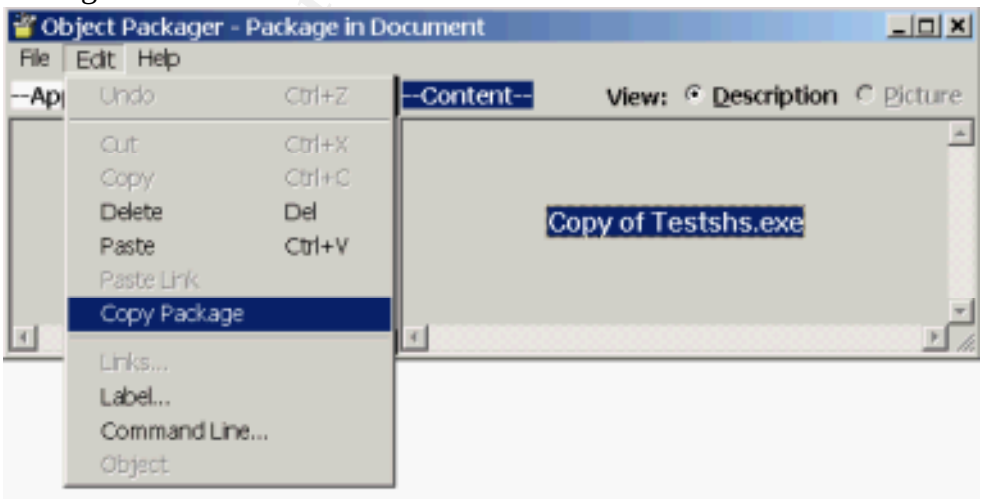

Figure 5

5. Open a folder in Windows Explorer, right-click on the background, and select Paste (see Figure 6).

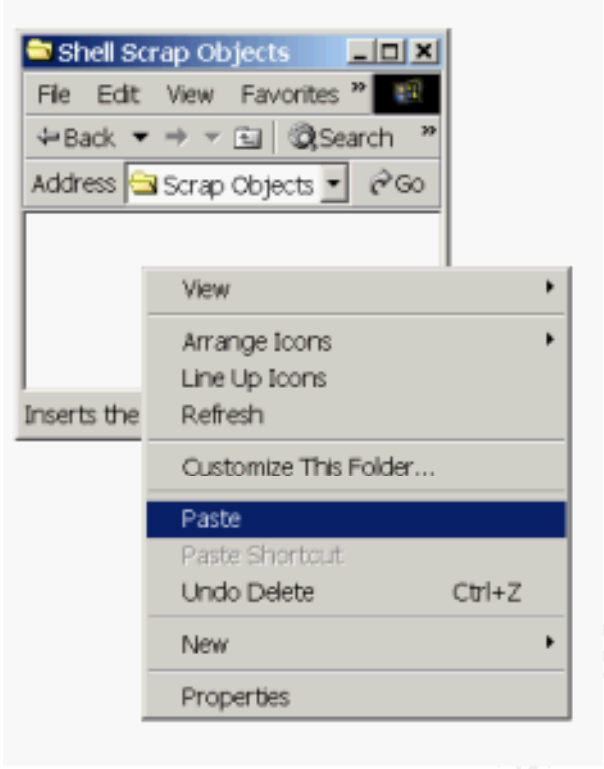

Figure 6

The Shell Scrap Object will paste into the selected folder.

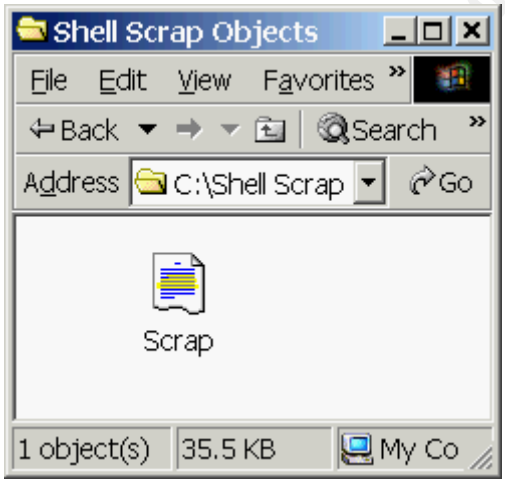

Figure 7

After this, the Shell Scrap Object can be opened like any other Windows executable and the wrapped executable will run.

Defensive Measures

On June 7, 2000, Microsoft released the Outlook Email Security Update which stops Outlook from sending files with a selected list of extensions that includes Shell Scrap Objects (extension .shs). This patch will allow the user to attach Shell Scrap Object files, but the file is blocked at the recipient's system so that they are not able to see the attachment, but just the text of the message. A notice will be given to the recipient that the attachment was blocked because it was potentially unsafe. Note that this patch differs from previous patches that were released by Microsoft that did not address Shell Scrap Objects (the Outlook Email Attachment Security Update, for example). This is an effective solution in many cases (and greatly reduces the risk of widespread dissemination of directly executable Shell Scrap Object attachments), but once again demonstrates the need for users to act as responsible system administrators for their systems and install the available security patches, or there needs to be an automated server-based patch system that verifies that users are compliant with the latest series of patches. Another issue is that there are users that will be irritated by the loss of functionality that is implied by the Outlook Email Security Update and these users may look for a way around the update (some possible methods have been posted to the Slipstick Systems web site posted below as a Reference). The other possibility is that the virus author sends the file as a zip archive, which will not be stopped by the Outlook Email Security Update, and the user can then save it locally, unzip it and execute the Shell Scrap Object. These drawbacks point to the fact that user education is always one of the most important aspects of any security plan and cannot be ignored. Although the Security Update disables the capability to send raw .shs files as attachments it does not completely close the security hold and therefore is not a complete solution to the problem. This is not surprising since all security issues have a human element to them and the technical solution that can stand alone (without human intervention of some sort) does not exist.

The problem of the file extension not being displayed (even when the folder options are set to show extensions) can be remedied by deleting the NeverShowExt value under the HKEY_CLASSES_ROOT\ShellScrap key in the Windows Registry. (WARNING: modifying the Registry can make your system unbootable. Always back up before modifying the Registry.) Logging off and back on (for a Windows 2000 system) will result in the user being able to see the .shs extension on Shell Scrap Object files. This leads to another education issue however, in that users must be trained to delete without opening files with "dual extensions" (like Life_stages.txt.shs for example, which will appear as Life_stages.txt unless the Registry value mentioned above is deleted). An instance where a legitimate Shell Scrap Object file would need to be sent with a "faked" extension is difficult to imagine.

Conclusion

The capability to wrap Windows executables in the Shell Scrap Object has been examined almost entirely from the aspect of email-borne viruses (which is how the author of the "Life Stages" virus used it). However, this technique could also be used to wrap

old viruses in an attempt to evade anti-viral detection. And it is not difficult to do this. The steps to create a Shell Scrap Object are remarkably simple and lead to a file whose appearance clouds its purpose, which provides an environment that can increase the probability of the user executing the malicious code. Currently the best way to defend against an email-borne attack using these types of files is the best way to defend against any email-borne virus and that is to make sure that users install the latest security patches (the Outlook Email Security Update in this case) and educate them to open email attachments as safely as possible by first verifying the sender of the message (many of the people who received the ILOVEYOU virus could have prevented further spreading it themselves by checking the name of the sender and not opening attachments from someone that they did not know who was sending them love notes), then, if there is any doubt about the validity of the message, verifying that the sender meant to send it (many email viruses automatically send themselves, unbeknownst to the sender, so a simple phone call verifying the message contents can prevent further spread and also alert the sender if they do have a virus), and finally, saving the attachment to the local hard drive and scanning it with anti-viral software before opening it.

References

Little, Keith. "Scrap Files Can Tear You Up." 9 July 2000.
URL: http://www.pc-help.org/security/scrap.htm (6 April 2000).

Vamosi, Robert. "ZDNET: Help and How-To: The Windows Scrap Object (.SHS) Explained." URL: http://www.zdnet.com/zdhelp/stories/main/0,5594,2590847,00.html (9 April 2000).

Ewell, Brian. "SARC Write-up – VBS.Stages.A." 16 June 2000.
URL: http://www.symantec.com/avcenter/cgi-bin/virauto.cgi?vid=18550 (9 April 2000).

Slipstick Systems. "Opening .exe Attachments with the Microsoft Outlook E-mail Security Patch." URL: http://www.slipstick.com/outlook/esecup/getexe.htm (6 April 2000).

Microsoft Corporation. "Q262631 – OL2000: Information About the Outlook E-mail Security Update."
URL: http://support.microsoft.com/support/kb/articles/Q262/6/31.ASP?LN=EN-US&SD=gn&FR=0&qry=shell%20scrap&rnk=2&src=DHCS_MSPSS_gn_SRCH&SPR=MSALL
(6 April 2000).

Last Updated: July 25th, 2017

Upcoming Training

SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
SANS Phoenix-Mesa 2017	Mesa, AZ	Oct 09, 2017 - Oct 14, 2017	Live Event