FEATURE

2007: THE YEAR OF THE SOCIAL ENGINEER?

Martin Overton
Independent researcher, UK

Last year Martin Overton described how phishers had borrowed techniques from malware authors to try to cover their tracks [1]. In this article he looks at the flip side, as malware authors have started to borrow techniques from the phishers.

On Friday 19 January last year hundreds of emails started to arrive in my inbox, all claiming to be news items. Originally the messages related to the storms that were raging through parts of Europe at the time, and it was this initial wave of emails that inspired the name that was subsequently given to the gang behind them: the Storm Worm gang.

The next wave of emails offered news about the start of World War III, the launch of nuclear missiles, the shooting down of satellites, and so on. After a while the effectiveness of these started to wane, so the gang moved on to using fake e-card notifications instead. They then started to target specific holidays and events, especially in the US.

What all of these attacks had in common, apart from all being part of an attempt to build a very large botnet, was that they relied almost exclusively on getting the recipient to click on a link or attachment, thus getting them to infect their own computer. No need for the Storm Worm gang to waste time writing infection and propagation routines, they just relied on end-user curiosity, naivety, fear, greed, altruism, etc. – in other words, good old social engineering.

This article is not about the Storm Worm gang, it is about the fact that 2007 seems to have been the year that social engineering became the mainstay of the malware author's infection routine, and when malware authors borrowed techniques from phishers.

The article will cover a couple of interesting cases which illustrate clearly that malware authors are borrowing techniques from phishers.

In [2] I posited that social engineering in malware was just coming up to the teenage stage – to continue that analogy, we are now seeing the teenager turning into a young adult who is ready to take on the world; full of enthusiasm and brimming with confidence.

During November 2007 I received several very well crafted emails that claimed to have come from *YouTube* and *Microsoft* respectively. These emails appeared very professional and links within them led to phishing-quality fake websites on which malware was hosted.

This was different from anything we had seen so far from the Storm Worm gang – although their emails had been very successful, these new ones were quality pieces of work which borrowed extensively from the phishers' bible.

Let us now have a look at two of the best examples I have seen of these professional-quality malware spam runs and their associated 'phishing-quality' payload-hosting websites.

CASE 1: DO YOU YOUTUBE?

Figure 1 shows a screenshot of an email I received one morning in November. It claims to have been sent by 'YouTube Service', a.k.a. 'service@youtube.com', on behalf of a friend who wants to share a video.

This nicely formatted email that claims to have come from a friend contains lots of links to click on. All the links shown on the right-hand side of the email really do take you to *YouTube* or *Google* pages, as they claim to. However, clicking on any of the links on the left-hand side of the email will take you to the site shown in Figure 2.

The site is very convincing. It almost looks like the real *YouTube* site. In order to view the video mentioned in the email the user is prompted to download an updated version of *Adobe*'s *Flash Player*. Unfortunately, however, this isn't the real *YouTube* site at all. To make matters worse, anyone downloading the 'latest *Flash Player*' from the site would have downloaded a malicious file instead, leaving them with an infected computer.

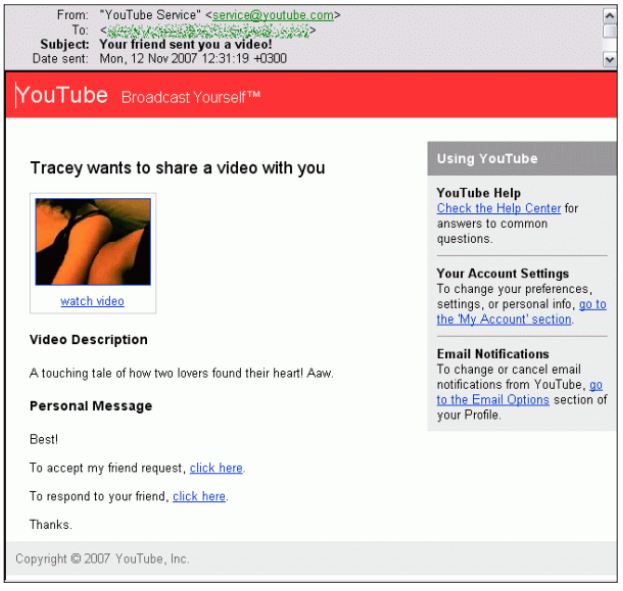

Figure 1: 'YouTube' email.

Figure 2: 'Phishy' YouTube website.

The following are some details about the file that was hosted on the fake *YouTube* site, as well as the level of detection at the time I found it and submitted it to a range of vendors and researchers:

```
FileName: install_flash_player.exe
FileDateTime: 12/11/2007 12:09:43
Filesize: 1228800
MD5: 29a8b08786a6a5bd253df5b2a42e7979
CRC32: E8ED5280
File Type: PE Executable

Scan report of: install_flash_player.exe (source:
AV-Test)

@Proventia-VPS      -
AntiVir             -
Avast!              -
AVG                 -
BitDefender         -
ClamAV              -
Command             -
Dr Web              -
eSafe               -
eTrust-VET          -
eTrust-VET (BETA)   -
Ewido               -
F-Prot              -
F-Secure            -
F-Secure (BETA)     Trojan-Dropper:W32/Agent.CPL
Fortinet            -
Fortinet (BETA)     -
Ikarus              Win32.SuspectCrc
Kaspersky           -
McAfee              -
McAfee (BETA)       -
Microsoft           -
Nod32               -
Norman              -
Panda               -
Panda (BETA)        -
QuickHeal           -
Rising              -
Sophos              -
Sunbelt             -
Symantec            -
Symantec (BETA)     -
Trend Micro         -
Trend Micro (BETA)  -
VBA32               -
VirusBuster         -
WebWasher           -
YY_A-Squared        -
YY_Spybot           -
```

To find out what the file does when executed, I ran it in the *Norman Sandbox*:

```
install_flash_player.exe.1 : W32/Malware (Signature:
NO_VIRUS)
* Compressed: NO
* TLS hooks: NO
* Executable type: Application
* Executable file structure: OK

[ General information ]
* Decompressing UPX3.
* Drops files in %WINSYS% folder.
* File length: 1228800 bytes.
* MD5 hash: 29a8b08786a6a5bd253df5b2a42e7979.

[ Changes to filesystem ]
* Creates file C:\WINDOWS\TEMP\cmd0999.tmp.
* Creates file C:\WINDOWS\TEMP\cmd0999.exe.
* Deletes file C:\WINDOWS\lg32.txt.
* Creates file C:\WINDOWS\TEMP\~1189.tmp.
* Deletes file C:\WINDOWS\ws386.ini.
* Creates file C:\WINDOWS\ws386.ini.
* Deletes file C:\WINDOWS\TEMP\~1189.tmp.
* Creates file C:\WINDOWS\db32.txt.
* Deletes file C:\WINDOWS\system32\aspimgr.exe.
* Creates file C:\WINDOWS\system32\aspimgr.exe.
* Creates file C:\WINDOWS\TEMP\_check32.bat.
* Creates file C:\WINDOWS\s32.txt.

[ Changes to registry ]
* Creates key
"HKLM\System\CurrentControlSet\Services\aspimgr".
* Sets value
"ImagePath"="C:\WINDOWS\system32\aspimgr.exe" in key
"HKLM\System\CurrentControlSet\Services\aspimgr".
* Sets value "DisplayName"="Microsoft ASPI Manager"
in key
"HKLM\System\CurrentControlSet\Services\aspimgr".
* Creates key "HKLM\Software\Microsoft\Sft".
```

```
* Sets value "default"="{00000000-0000-0000-0000-
00003F000F00}" in key "HKLM\Software\Microsoft\Sft".

[ Network services ]
* Connects to "ns.uk2.net" on port 53.
* Connects to "www.yahoo.com" on port 80.
* Connects to "www.web.de" on port 80.
* Connects to "70.86.123.34" on port 80.
* Connects to "70.86.86.210" on port 80.
* Connects to "67.19.9.186" on port 80.

[ Process/window information ]
* Attempts to open C:\WINDOWS\TEMP\cmd0999.exe.
* Creates process "C:\WINDOWS\TEMP\cmd0999.exe".
* Attempts to access service "aspimgr".
* Creates service "aspimgr (Microsoft ASPI Manager)"
as "C:\WINDOWS\system32\aspimgr.exe".
* Creates process "C:\WINDOWS\system32\aspimgr.exe".
* Attempts to open C:\WINDOWS\TEMP\_check32.bat NULL.
* Creates process "C:\CMD.EXE".

[ Signature Scanning ]
* C:\WINDOWS\TEMP\cmd0999.exe (48128 bytes) : no
signature detection.
* C:\WINDOWS\ws386.ini (12 bytes) : no signature
detection.
* C:\WINDOWS\db32.txt (100 bytes) : no signature
detection.
* C:\WINDOWS\system32\aspimgr.exe (65536 bytes) : no
signature detection.
* C:\WINDOWS\TEMP\_check32.bat (93 bytes) : no
signature detection.
* C:\WINDOWS\s32.txt (63 bytes) : no signature
detection.
```

This piece of malware is very busy, creating a number of files, deleting others, creating registry keys and connecting to a number of sites – almost certainly to download other components, update itself, and so on. Like most malware today this one is packed, in this case using UPX.

CASE 2: ONE MS UPDATE YOU DON'T WANT!

If you, or anyone you know, installs all *Microsoft* updates religiously, then this case is something you *really* need to be aware of.

Figure 3 is a screenshot of another email I received one evening in November. It claims to have been sent by 'Microsoft Corp' regarding a critical update.

This nicely formatted email states: '*Microsoft* recommends that customers apply the update immediately following the links below corresponding to your system.' There then follow three links. Clicking on any of the links in the email takes you to the site shown in Figure 4.

How many of you would have believed that this is a screenshot of the real *Microsoft Update* site and might have proceeded to download the patch offered? It's very convincing.

Unfortunately it isn't the real *Microsoft Update* site (or any other *Microsoft* site). To make matters worse for anyone that believed it was the real site and downloaded the supposed patch, not only did they not download and install 'MS07-055', but they would now have an infected computer.

Here are some details about the file that was hosted on the fake *Microsoft* site, as well as the level of detection at the time I found it and submitted it to various vendors and researchers:

```
Scan report of: WindowsXP-KB923810-x86-ENU.exe
(source: AV-Test)

@Proventia-VPS      -
AntiVir             -
Avast!              -
AVG                 -
BitDefender         -
ClamAV              -
Command             -
Dr Web              -
eSafe               Trojan/Worm [101] (suspicious)
eTrust-VET          -
eTrust-VET (BETA)   -
Ewido               -
F-Prot              -
F-Secure            -
F-Secure (BETA)     -
Fortinet            -
Fortinet (BETA)     -
Ikarus              Trojan.Win32.VB.azd
Kaspersky           -
McAfee              -
McAfee (BETA)       -
Microsoft           -
Nod32               -
Norman              -
Panda               -
Panda (BETA)        -
QuickHeal           -
Rising              -
Sophos              -
Sunbelt             -
Symantec            -
Symantec (BETA)     -
Trend Micro         -
Trend Micro (BETA)  -
VBA32               -
VirusBuster         -
WebWasher           Win32.ModifiedUPX.gen!84
                    (suspicious)
YY_A-Squared        -
YY_Spybot           Smitfraud-C.,,Executable
```

To find out what the file does when executed, I ran it in the *Norman Sandbox*:

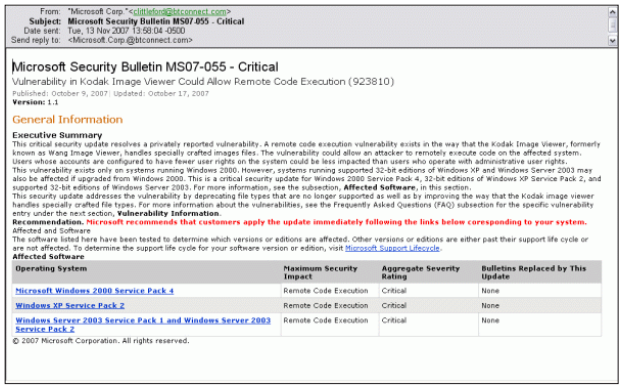

Figure 3: 'Microsoft Corp' email.

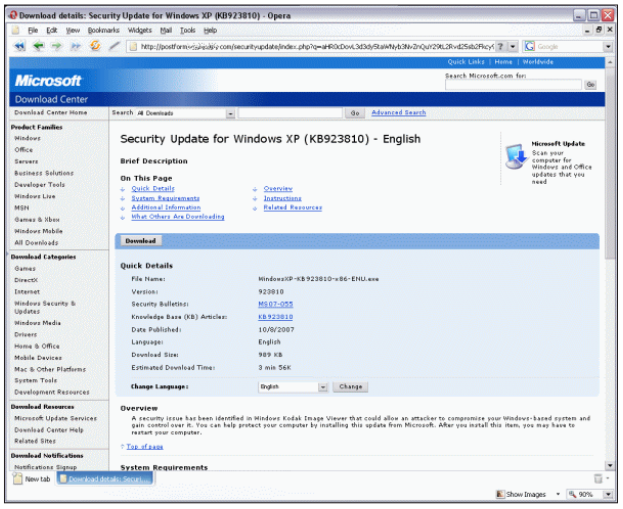

Figure 4: 'Phishy' Microsoft website.

```
WindowsXP-KB923810-x86-ENU.exe : Not detected by
Sandbox (Signature: NO_VIRUS)

* Compressed: YES
* TLS hooks: NO
* Executable type: Application
* Executable file structure: OK

[ General information ]
* Decompressing UPX3.
* Applications uses MSVBVM60.DLL (Visual Basic 6).
* File length:      1057651 bytes.
* MD5 hash: b59d788bc907d9aecb15375abe09c606.

[ Process/window information ]
* Creates a COM object with CLSID {FCFB3D23-A0FA-
1068-A738-08002B3371B5} : VBRuntime.
* Creates a COM object with CLSID {E93AD7C1-C347-
11D1-A3E2-00A0C90AEA82} : VBRuntime6.
```

This piece of malware requires the VB6 runtime files to be present on the PC for it to work, as it has been created using Visual Basic 6 for *Windows*. Like most malware today this one is also packed, in this case using UPX.

CONCLUSIONS

Unlike the scenario described in [1], where phishers had borrowed techniques from malware authors to try and cover their tracks after stealing victims' *PayPal* credentials, name, address, social security number and credit card data, here it looks like the malware authors have been taking lessons from the phishers. Both of the cases described in this article are very believable. The fake *YouTube* email and site are almost perfect. The fake *Microsoft* email is not so convincing, but the fake *Microsoft Update* site is very believable.

Unfortunately, with the malware authors using this level of social engineering, it is very likely that more people will fall victim to their creations. This means that more victims will infect their computers without the malware authors having to code complex infection and propagation routines. If the malware offered via a fake *<insert company name here>* site is a bot or dropper then the infected computer could very soon be sending out lots of spam, taking part in a DDoS attack or worse.

So what is the solution? I'd like to say that technology will solve the problem, but you can't easily patch the exploits in an average computer user.

Technology still has its part to play in the overall solution, but it is clear that we need to try something else too. As much as it pains me to say so, I think that the answer to the problem is user education. This won't be inexpensive or happen overnight (and it won't be popular with some people), but if it is done properly I believe it will make the phishers', scammers' and malware authors' jobs harder, which is all we can realistically hope for.

It is time that the weakest link finally became the strongest link. Are you up to the challenge?

REFERENCES

[1] Overton, M. A phish with a sting in the tail. Virus Bulletin, March 2007, p.S1.

[2] Overton, M. You are the weakest link, goodbye! Malware social engineering comes of age. Virus Bulletin, March 2002, p.14.

STOP PRESS

Just as this article was being finished news came in of several US government labs having been targeted by emails containing links or attachments to malware infected files, just like the methods described in this article. For more details see: http://www.eweek.com/article2/0,1895,2230086,00.asp.