September 2010

Feature Article: Can’t teach an old worm new tricks?

Table of Contents

Feature Article: Can’t teach an old worm new tricks?

Urban Schrott, IT Security & Cybercrime Analyst, ESET Ireland

Just when we thought the most basic social engineering worms were a thing of the past, and that both users and service providers were beyond significant problems with such pests, a good old-fashioned mail massmailer gave us reason to pause in just a few days of early September. I am, of course, talking about the recent occurrence of the variants of so called “Here you have” virus, known as Win32/Visal to ESET.

The virus was propagated as a standard social engineering e-mail which requires the user to initialize the infection, having been tricked into cooperating with a promise of naughty movies or an attached pdf file. Unlike older approaches, when the files themselves carried the malicious code, this time round these files and links download malware onto the computer instead, if you click on them. The virus installs itself to the Windows directory as CSRSS.EXE (mimicking the legit Windows System CSRSS.EXE) and immediately tries to disable many types of security services and antivirus protections before sending itself to all the contacts in the user's address book. ESET's heuristics picked it up immediately before it was even recognised as a threat. Overall, a very clichéd threat, with no innovative approaches, yet it still spread rapidly and massively.

ESET's researcher David Harley commented: *“Who'd have thought that such hackneyed social engineering hooks would still work? Well, to be honest, it doesn't surprise me at all. My experience suggests that tried and tested hooks often work where a sophisticated, innovative approach isn't particularly effective. Kind of like 419s. The code wasn't particularly innovative either. Its resemblance to other threatcode, its disabling of security processes, and the use of common vectors such as mass-mailing, Autorun and network shares to spread tend to raise suspicions pretty quickly. And of course, the use of filetypes like .SCR (or, more blatantly, .PDF.SCR) is in itself a danger flag that may have tripped generic filtering on secondary infection, even though the initial binary wasn't mailborne.”*

So why did it make such an impact? Well, one reason may, perhaps, simply be its visibility. Hundreds of thousands mailboxes filling with these mails in a short time do attract attention. In the US in particular, several large organisations quickly responded, having encountered it in large volumes. NASA, AIG, Disney, Procter & Gamble and Wells Fargo were all reported as struggling to contain an outbreak of the worm. Another reason for infection is, once more, that many computer users are still confused about what it is or isn’t safe to click on. In spite of years of attempting to educate the general public, it appears the message still hasn't gotten through entirely.

In a slightly unusual twist to the story, however, in the days following the outbreak, the media reported that a hacker or VXer has come forward saying the worm was his work (http://www.theregister.co.uk/2010/09/13/hacker_claims_credit_for_here_you_have_worm/ and also http://uk.ibtimes.com/articles/61545/20100913/here-you-have-e-mail-worm-maker-says-he-s-not-bad.htm). There was speculation as to whether this was just a promotional exercise for a hacking group, as the hacker appears to have claimed he deliberately made the worm rather benign, when he could, he claimed have included a much deadlier payload. However, as no actual ties between the hacker and the worm were proven, and a group called Iraq Resistance also received some mentions in connection with Visal, suspicions arose as to how valid *any* of these claims were: could it all be media manipulation and propaganda spinning?

In any case, the worm was detected quickly and the servers propagating the malware were shut off promptly and efficiently. Microsoft's Malware Protection Center reported within days of discovering the virus that *"In any case, after the worm was discovered, the URL was rendered unreachable. Therefore, although the malware can still send spam, the malicious links are inactive, preventing the worm from spreading further using the spam vector. Although mailboxes can continue to fill up due to unprotected machines executing the malware, those emails will no longer be able to find any malware at the target URL."* (http://blogs.technet.com/b/mmpc/archive/2010/09/10/update-on-the-here-you-have-worm-visal-b.aspx)

Though, in spite of ESET's heuristic detection of it and the prompt shutting down of the malicious servers, the moral of the story is still that the human factor remains the weakest link in malware protection. And since the emails primarily targeted corporate environments, the education of employees who were happily clicking away at the dangerous links is something that should be addressed more seriously.

Nothing Succeeds Like Stuxnet

And it's still succeeding: if nothing else, in generating intense debate and speculation.

Win32/Stuxnet is, perhaps 2010's Code Red, Melissa, or Blaster, in terms of media impact, though I'm not going to promise that something even bigger won't happen along before Santa hangs up his boots for another twelve months before sipping a large single malt with Rudolf as midnight brings in the New Year. Well, it's sometimes difficult to know what makes a particular item of malware a media sensation. If it appears to affect a lot of people, that sells copies, of course. It may be that there's a particular "hook" or quirk that sells it to journalists, like some association with a currently popular celebrity, or a particular event.

(Oddly enough, that's also the sort of hook that gangs use to trick people into accessing malicious sites, running malcode and so on. Well, maybe not so odd. Everyone with something to sell has to learn something about psychology, whether it's newspapers, or security software, real or fake.)

On this occasion, though, there's more than hype behind all the attention, even though the speculation about nuclear meltdowns and cyberwarfare between Iran and Israel has generated more red-herrings than the European fishing industry ever landed, inspiring an uncharacteristically exasperated blog post from David Harley at http://blog.eset.com/2010/09/25/cyberwar-cyberhisteria. It's true that Iran seems to have experienced more infections than anyone else (this is shown in ESET's white paper "Stuxnet Under the Microscope": see below).

However, distribution doesn't tell us anything about targeting, because while the payload is clearly targeted (though we don't know at time of writing which specific installations or even type of installation may have been in the crosshairs), the distribution isn't.

It may have been an error of judgement to spread Stuxnet using self-replicative mechanisms, ensuring that it spread far enough to attract the attention of security researchers, though it may also be that the team behind it knew their target well enough not to care about that. But it's certainly a cut above the usual workmanlike "only-as-good-as-it-needs-to-be" code that malware gangs churn out hour after hour, day after day. In fact, one of our friends at Kaspersky claims it's so sophisticated that he had it in his lab for a week before he noticed that it was malware, though we suspect that he was hamming it up there for comic effect.

However, it *is* a sophisticated chunk of code that is still revealing some surprises, even for those of us who've been looking at it for longer than a week. ;-) Certainly it's more sophisticated than some of the malware that took a leaf out of its book and made use of the LNK exploit that we might still not have known about, had Stuxnet made use of it. (David Harley wrote about that, by the way, in a Malware Analysis article "Chim Chymine: a lucky sweep?" for the September edition of Virus Bulletin (http://www.virusbtn.com/virusbulletin/archive/2010/09/vb201009-chymine), though it's only available to subscribers at the moment. It will probably be published eventually on the ESET white papers page, though, if Virus Bulletin kindly give us permission as they have done on other occasions. This considers how other malware, old (Win32/Sality) and new (Win32/Chymine) jumped on the LNK vulnerability exploited by Stuxnet. David also talked about the LNK vulnerability in his monthly article for Security Week on "Shortcuts to Insecurity: .LNK Exploits" (see http://securityweek.com/shortcuts-insecurity-lnk-exploits).

However, there's a lot more to Stuxnet than a single (now patched) LNK vulnerability (http://www.microsoft.com/technet/security/bulletin/MS10-046.mspx), or the much earlier MS08-67 patch (http://www.microsoft.com/technet/security/bulletin/ms08-067.mspx) for an RPC-related vulnerability also used by Conficker, though Stuxnet did manage to find an interesting further twist to that one, using sophisticated shellcode employing advanced techniques such as ROP (return oriented programming)that have recently become popular and attracted much attention in the security community. There's also the more recently patched MS10-061, which addresses a critical vulnerability in the print spooler service, which results in privilege escalation. Essentially, it allows a remote user using a Guest account to write into the %SYSTEM% directory of the target machine. Of course, it shouldn't be able to do this, especially when it takes the opportunity to write (malicious) binary files into %SYSTEM%.

But that's not all, either. There are a couple of 0-day vulnerabilities, one using a specially-crafted keyboard file, that we can't discuss yet because Microsoft are working on mitigation. And we're not even going to talk about all those interesting SQL strings and code signing certificates. Yet.

A number of last-minute presentations at Virus Bulletin's 2010 conference at the end of September address some of these issues, though no-one is going to cover the whole thing in a 30-minute presentation. ESET researchers don't claim to know all the answers yet, either: however, on the 23rd September, 2010, ESET released a lengthy report that covers quite a lot of that ground: "Stuxnet under the microscope", by Aleksandr Matrosov, Eugene Rodionov, David Harley and Juraj Malcho is available at http://www.eset.com/resources/white-papers/Stuxnet_Under_the_Microscope.pdf, as blogged at http://blog.eset.com/2010/09/23/eset-stuxnet-paper.

ESET NOD32 Antivirus for Mac OS

In response to demand, ESET launched ESET NOD32 Antivirus 4 Business Edition for Mac systems in September 2010. Since December 2009, when ESET NOD32 Antivirus 4 (Beta) for Mac was released for public testing, ESET has seen more than 135-thousand installation packages of this solution dowloaded.

About the launch of a security solution for the Macintosh platform, Richard Marko, ESET's Chief Technology Officer said: "*Engineered with our award winning ThreatSense® scanning engine, it ensures industry-leading protection against emerging*

threats".

The key features of ESET NOD32 Antivirus 4 Business Edition for Mac are:

- Market-proven scanning engine now offering enterprise-class protection for Macs in corporate networks.
- Proactive protection from cross platform threats to prevent Macs from becoming carriers and spread malicious code across the network.
- Centralized management of ESET products from a single console.
- Universal installer package for remote deployment using existing tools.
- Removable media control that blocks USB, Firewire, CD/DVD devices.

It also has **flexible scheduling tasks** to plan when tasks like virus signature updates and on-demand computer scanning will run on the systems, **advanced scanning settings**, **Full Screen Mode** to avoid disruptions from alerts and information windows when running full screen and **security settings control** that allows to define users who can alter security features.

More information:
http://www.eset.com/business/medium/nod32-antivirus-mac

FAQ about Twitter incident

On September 21th, Twitter was affected by a public vulnerability that inspired enormous interest among the media, not surprisingly given how dependent many journalists are on this service nowadays. However, we've also noticed quite a few misunderstandings and misconceptions about the incident. Here is a FAQ (frequent asked questions) about the attack, published on ESET Latin America's Laboratory Blog:

How does the attack work?

It was a classic Cross Site Scripting attack (better known as XSS) on the Twitter web site. It made it possible for an attacker to be able to run JavaScript code on victim systems by the publication of tweets specially designed to exploit the vulnerability.

What was the impact of the attack?

When the user passed the mouse pointer over the tweets, it was possible to run types of program code to launch a pop up, create a new tweet, or redirect the victim to another website. The last one was particularly dangerous, as the user could be redirected to malicious web-pages: clearly, though, the other payloads also have malicious possibilities.

How could users protect themselves?

While the vulnerability was active, the only solution was to avoid the use of the twitter.com website, and use the service only through desktop or mobile applications. Or, of course, to avoid it altogether.

Has the incident been dealt with?

Yes, Twitter solved the problem five hours after it was published, and made the final adjustments a couple of hours later. This means that the lifetime of the attack was short, and there is no need to worry.

Does Twitter already knew about the attack?

Yes, the company confirmed that they received a report about the issue last month and fixed it, but recent site changes caused

it to resurface.

Many users were affected?

Although thousands of malicious messages were spread, most of them did not have significant impact on innocent users. No cases of malcode execution or massive attacks were detected or reported.

Is it a worm?

No, although the attack has been incorrectly described as a worm or virus. It is an XSS attack that has the feature of automatically self-replicate with the publication of tweets, without needing any action or permission from the user. It has the capabitlity to do this because it uses the *onmouseover* function that allows it to be "retweeted" simply by the reader's passing the mouse cursor over the tweet. Due to this feature of automatic propagation, some people have linked this attack to a worm, but there is no malicious binary involved, so it's misleading to describe the attack as malware.

Source (in Spanish): http://blogs.eset-la.com/laboratorio/2010/09/22/faq-sobre-la-vulnerabilidad-en-twitter/

Autumnal AMTSO

The Anti-Malware Testing Standards Organization (AMTSO) was founded in May 2008 as an international non-profit association that focuses on addressing the global need for improvement in the objectivity, quality and relevance of anti-malware testing methodologies. AMTSO membership is open to academics, reviewers, publications, testers and vendors, subject to guidelines determined by AMTSO.

The next Anti-Malware Testing Standards Organization (AMTSO) Meeting will be on 21 and 22 October 2010 in Munich. Details about the venue, discounted rates for the hotel-room, registration form and the preliminary agenda can be found here: http://www.amtso.org/meetings.html. It's likely to be another lively meeting: a paper on False Positive testing is expected to be considered for approval at the meeting, and there's also likely to be discussion about the possibility of adding some form of affiliate membership for individuals at low cost but with limited voting rights. There are also several testing-related papers due to be presented at the Virus Bulletin Conference at the end of September and at AVAR in November (including papers by ESET researchers) which are likely to be added to the AMTSO resources pages.

New White Papers

We have recently published several more papers related to new attacks, forensics and anti-malware testing. Check them out:

- "**PWN2KILL, EICAR and AV: Scientific and Pragmatic Research**" by David Harley, An article about the implications of the PWN2KILL challenge at iAWACS 2010: is this the new face of AV testing? (Originally published in Virus Bulletin in June.) http://www.eset.com/resources/white-papers/pwn2kill-whitepaper.pdf

- "**Antivirus Testing and AMTSO: Has Anything Changed?**" by David Harley. A conference paper from the CFET2010 forensics conference summarizing how the Anti-Malware Testing Standards Organization has developed in the past few years and the way in which the AV and testing industries have responded to those developments. http://www.eset.com/resources/white-papers/Antivirus-Testing-and-AMTSO.pdf

- "**SODDImy and the Trojan Defence**" by David Harley. This paper, also presented at the CFET2010 conference looks at the implications in the age of the botnet of the "Some Other Dude Did It" and "it must have been a Trojan" defenses against conviction for possession of illegal material, especially pornography. http://www.eset.com/resources/white-papers/SODDImy-and-the-Trojan-Defence.pdf

Following the Virus Bulletin Conference at the end of September, at least two more papers co-authored by ESET researchers will be made available.

The Top Ten Threats

1. INF/Autorun

Previous Ranking: 1
Percentage Detected: 6.62%

This detection label is used to describe a variety of malware using the file autorun.inf as a way of compromising a PC. This file contains information on programs meant to run automatically when removable media (often USB flash drives and similar devices) are accessed by a Windows PC user. ESET security software heuristically identifies malware that installs or modifies autorun.inf files as INF/Autorun unless it is identified as a member of a specific malware family.

Removable devices are useful and very popular: of course, malware authors are well aware of this, as INF/Autorun's frequent return to the number one spot clearly indicates. Here's why it's a problem.

The default Autorun setting in Windows will automatically run a program listed in the autorun.inf file when you access many kinds of removable media. There are many types of malware that copy themselves to removable storage devices: while this isn't always the program's primary distribution mechanism, malware authors are always ready to build in a little extra "value" by including an additional infection technique.

While using this mechanism can make it easy to spot for a scanner that uses this heuristic, it's better, as Randy Abrams has suggested in our blog (http://www.eset.com/threat-center/blog/?p=94; http://www.eset.com/threat-center/blog/?p=828) to disable the Autorun function by default, rather than to rely on antivirus to detect it in every case. You may find Randy's blog at http://www.eset.com/threat-center/blog/2009/08/25/now-you-can-fix-autorun useful, too.

2. Win32/Conficker

Previous Ranking: 2
Percentage Detected: 4.52%

The Win32/Conficker threat is a network worm originally propagated by exploiting a recent vulnerability in the Windows operating system. This vulnerability is present in the RPC sub-system and can be remotely exploited by an attacker without valid user credentials. Depending on the variant, it may also spread via unsecured shared folders and by removable media, making use of the Autorun facility enabled at present by default in Windows (though not in Windows 7).

Win32/Conficker loads a DLL through the svchost process. This threat contacts web servers with pre-computed domain names to download additional malicious components. Fuller descriptions of Conficker variants are available at http://www.eset.eu/buxus/generate_page.php?page_id=279&lng=en.

While ESET has effective detection for Conficker, it's important for end users to ensure that their systems are updated with the Microsoft patch, which has been available since the third quarter of 2008, so as to avoid other threats using the same vulnerability. Information on the vulnerability itself is available

at http://www.microsoft.com/technet/security/Bulletin/ms08-067.mspx. While later variants dropped the code for infecting via Autorun, it can't hurt to disable it: this will reduce the impact of the many threats we detect as INF/Autorun. The Research team in San Diego has blogged extensively on Conficker issues: http://www.eset.com/threat-center/blog/?cat=145

It's important to note that it's possible to avoid most Conficker infection risks generically, by practicing "safe hex": keep up-to-date with system patches, disable Autorun, and don't use unsecured shared folders. In view of all the publicity Conficker has received and its extensive use of a vulnerability that's been remediable for so many months, we'd expect Conficker infections to be in decline by now if people were taking these commonsense precautions. While the current ranking looks like a drop in Conficker prevalence, this figure is affected by the changes in naming and statistical measurement mentioned earlier: there's no indication of a significant drop in Conficker infections covering all variants.

3. Win32/PSW.OnLineGames

Previous Ranking: 3
Percentage Detected: 2.86%

This is a family of Trojans used in phishing attacks aimed specifically at game-players: this type of Trojan comes with keylogging and (sometimes) rootkit capabilities which gather information relating to online games and credentials for participating. Characteristically, the information is sent to a remote intruder's PC.

These Trojans are still found in very high volumes, and game players need to remain alert. While there have always been unpleasant people who will steal another gamer's credentials just for the heck of it, trading in virtual cash, treasure, avatars and so on is now a major source of illegal income for cybercriminals. It's also important that participants in MMORPGs (Massively Multi-player Online Role Playing Games) like Lineage and World of Warcraft, as well as "metaverses" like Second Life, continue to be aware of the range of other threats like griefing ranged against them. The ESET Research team considered gaming malware in detail in the ESET 2008 Year End Global Threat Report, which can be found at http://www.eset.com/threat-center/threat_trends/EsetGlobalThreatReport(Jan2009).pdf

4. INF/Conficker

Previous Ranking: 5
Percentage Detected: 1.64%

INF/Conficker is related to the INF/Autorun detection: the detection label is applied to a version of the file autorun.inf used to spread later variants of the Conficker worm.

As far as the end user is concerned, this malware provides one more good reason for disabling the Autorun facility: see the section on INF/Autorun above.

5. Win32/Tifaut.C

Previous Ranking: 4
Percentage Detected: 1.64%

The Tifaut malware is based on the Autoit scripting language. This malware spreads between computers by copying itself to removable storage devices and by creating an Autorun.inf file to start automatically.
The autorun.inf file is generated with junk comments to make it harder to identify by security solutions. This malware was created to steal information from infected computers.

See INF/Autorun above for discussion of the implications of software that spreads using Autorun.inf as a vector.

6. Win32/Sality

Previous Ranking: 20
Percentage Detected: 1.61%

Sality is a polymorphic file infector. When run starts a service and create/delete registry keys related with security activities in the system and to ensure the start of malicious process each reboot of operating system.

It modifies EXE and SCR files and disables services and process related to security solutions.

More information relating to a specific signature: http://www.eset.eu/encyclopaedia/sality_nar_virus__sality_aa_sality_am_sality_ah

7. HTML/ScrInject.B

Previous Ranking: 8
Percentage Detected: 1.17%

Generic detection of HTML web pages containing script obfuscated or iframe tags that that automatically redirect to the malware download.

Malicious scripts and malicious iframes are a major cause of infection, and it's a good idea to disable scripting by default where possible, not only in browsers but in PDF readers. NoScript is a useful open source extension for Firefox that allows selective disabling/enabling of Javascript and other potential attack vectors.

8. JS/TrojanClicker.Agent.NAZ

Previous Ranking: 7
Percentage Detected: 0.70%

This malware is a Trojan horse that does not generate copies of itself, but is usually part of other malware.

It contains a list of web addresses to which to send requests, used to simulate clicking on advertisements for financial gain (click fraud).

9. Win32/Spy.Ursnif.A

Previous Ranking: 18
Percentage Detected: 0.67%

This label describes a spyware application that steals information from an infected PC and sends it to a remote location, creating a hidden user account in order to allow communication over Remote Desktop connections. More information about this malware is available at http://www.eset.eu/encyclopaedia/win32-spy-ursnif-a-trojan-win32-inject-kzl-spy-ursnif-gen-h-patch-zgm?lng=en

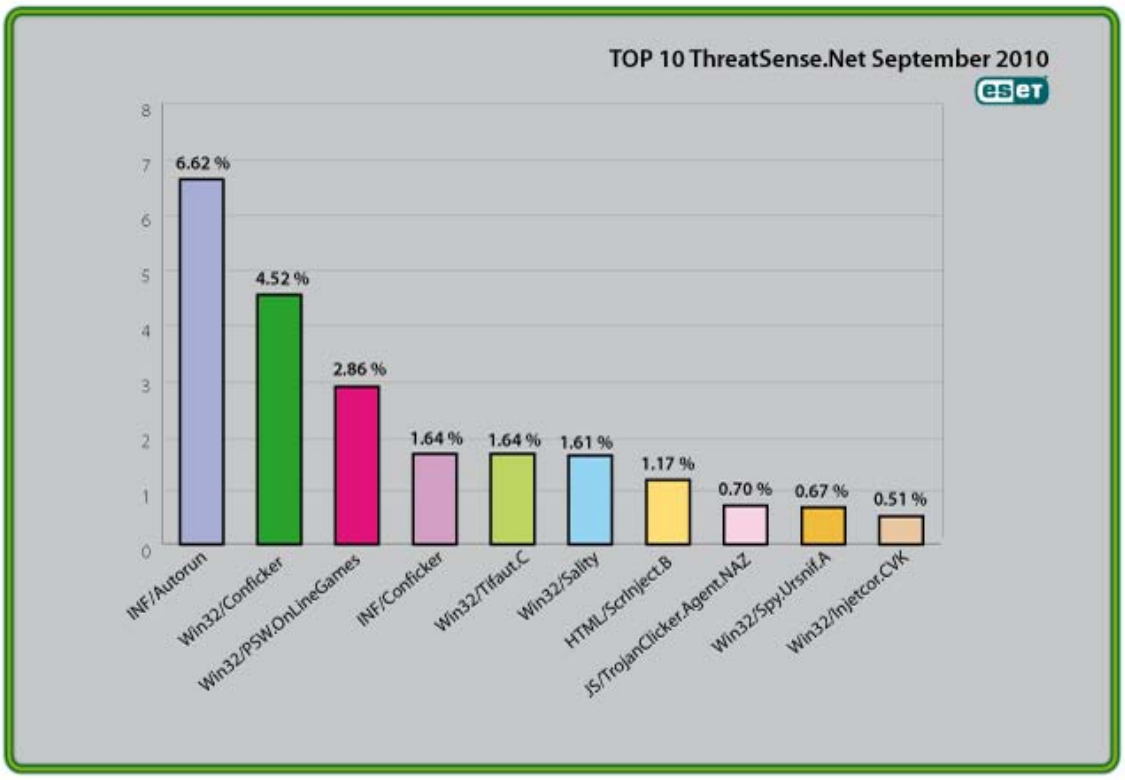
10. Win32/ Injector.CVK

Previous Ranking: n/a
Percentage Detected: 0.51%

Win32/Injector.CVK is the name for generic detection of malware that has capability to create and run a new thread with its own program code within a specific running process.

Top Ten Threats at a Glance (graph)

Analysis of ESET's ThreatSense.Net®, a sophisticated malware reporting and tracking system, shows that the highest number of detections this month, with almost 6.62% of the total, was scored by the INF/Autorun class of threat.



About ESET

ESET is a global provider of security software. The ESET NOD32® Antivirus and ESET Smart Security products are consistently recognized among the most comprehensive and effective security solutions available today.

Additional resources

Keeping your knowledge up to date is as important as keeping your AV updated. For these and other suggested resources please visit the ESET Threat Center to view the latest:

- ESET White Papers
- ESET Blog
- ESET Podcasts
- Independent Benchmark Test Results
- Anti-Malware Testing and Evaluation