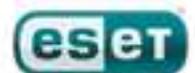# Global threat report

May 2010
Feature Article: AV Blowing its Own Trumpet

Table of Contents

eseT

Feature Article: AV Blowing its Own Trumpet

Jan Vrabec, ESET Security Technology Analyst, in Bratislava, commented in the ESET ThreatBlog (http://www.eset.com/blog/2010/05/27/test-toot-suite-antivirus-vendors-blowing-own-horn) on anti-virus companies using their own, in-house detection tests to "prove" that their products are the best, and here's a shortened version.

Of course, most vendors use in-house testing as a tool for monitoring and improving the capabilities of their own products, and so they should. However, it's also very easy to misuse as a vehicle for showing a company's own AV products in the best possible light, when combined with extravagant marketing claims, and illustrated by carefully assembled graphic and video content. In fact, some vendors are only using in-house testing to showcase their detection.

We can't help but wonder whether this practice stems from insecurity: are they worried about how their product will perform in tests run by independent testing organizations? Are they concerned that an independent test might show up deficiencies in their own methodology? Is the attraction the ease with which results can be manipulated to the advantage of their own products, for example by careful selection of a test set?, resulting in a "home turf" bias? Perhaps newcomers are intimidated by the track records of more established vendors and worry that they may not easily match their detection results. When the tester and the tested are the same party, it's hardly surprising if the company's own products are ranked very highly indeed.

However, if we compare the way in which in-house testing is practiced to the way in which competent independent testers work, several major issues become apparent.

Competent testers are not usually secretive about their methodology, and may allow tested vendors access to samples in order to validate their results. This rarely seems to be the case with in-house testing, however, if the conclusions are used for advertising purposes. This "take it or leave it" approach to testing – "you'll have to take my word for it that we did it properly" – makes it difficult for a thoughtful reader to draw meaningful conclusions about the objectivity and competence of the test, and therefore about the comparative detection performance of the products under test.

What kind of information should you expect? If you have information on how the test was carried out, as well as by whom, you are in a better position to check that it appears to conform to industry-agreed standards, such as those outlined by AMTSO (the Anti-Malware Testing Standards Organization – http://www.amtso.org).

Test results often don't stand up to scrutiny because of poor selection, classification and/or validation of samples.

Seemingly reliable anti-malware testing results often turn out to be invalid because the samples used in the tests were misclassified. Clean files get

For example, what looks like a high volume of false positives might be the result of misidentification of malicious samples as clean (or as AMTSO puts it, "innocent." If innocent files are misclassified as malicious because of inadequate validation (or non-validation), this can be misleadingly reported as poor detection results. A common but less obvious problem occurs when a "grey" sample such as a "Possibly Unwanted" application is classified as malicious:. In this case, a product that

doesn't detect such applications by default may be marked down for non-detection of a sample which it is, in fact, perfectly capable of detecting. When this happens, it effectively means that the test is influenced by design philosophy rather than accurate detection statistics.

Similar problems are caused by corrupted and/or unviable samples, or objects that are only unequivocally malicious in very specific contexts, especially contexts which are at best unlikely to be seen in real life. Proper classification and validation of malware samples in testing are crucial to accurate comparison of the effectiveness of anti-malware solutions.

Principle 5 of AMTSO's fundamental principles of testing document encourages testers to revalidate test samples that appear to have caused false negative or false positive results (http://www.amtso.org/amtso---download---amtso-fundamental-principles-of-testing.html).

When choosing a security product or a suite, a customer should be aware of the sources used to support such claims, such as independent test results and reviews. Industry-recognized independent testing organizations to look out for include:

- AV-Comparatives [http://av-comparatives.org/]
- Virus Bulletin [http://www.virusbtn.com/vb100/index]
- AV-Test.org [http://www.av-test.org]

When you choose your next antimalware product, you may find it helpful to think about all the development and maintenance overheads of producing such a complex and sophisticated application as an antivirus program. It's a good idea to scrutinize the vendors' history closely, including the track record of their products. Awards received from independent testing organizations over time are a good indicator of sound

technology and long-term commitment to keeping consumers secure.

EICAR & AMTSO

May has been a busy month for the various Research teams. Ján Vrabec and David Harley presented a paper on performance testing at the EICAR conference in Paris. Another paper on Apple security issues was presented by David Harley, Pierre-Marc Bureau and Andrew Lee. Both papers are available through links on the white papers page at http://www.eset.com/documentation/white-papers, in the ESET conference papers section. "Real Performance?" is also available from the AMTSO resources page at http://www.amtso.org/related-resources.html, which is currently being developed as a major, product-agnostic resource of testing related material. At the AMTSO workshop in Helsinki, two more guidelines papers, "AMTSO Performance testing guidelines" and "Whole Product Protection Testing Guidelines" were approved, and will be available shortly from the AMTSO documents page at http://www.amtso.org/documents.html.

There was also an interesting if somewhat confused post at Offensive Computing (http://www.offensivecomputing.net/?q=node/1569) regarding the AMTSO paper "Issues involved in the "creation" of samples for testing". Danny Quist claimed that the paper demonstrated that AMTSO was missing the point because

"Malware authors are using every single one of these techniques with spectacular success."

David Harley responded in the AMTSO blog that:

Of course they are. Which is why making more malware is, at the very least, redundant. Furthermore, if you're attempting to create your own malware because you can't get samples from other sources, the chances are that you don't have the knowledge to create samples that represent real-world threats. A malware author's lack of ethical principles is beside the point. A good tester does have ethical principles, and feels that he owes it to audience to be as accurate as possible in his testing. And he also knows that artificial samples are not "the real thing"…

… The ethical issue … around giving unrestricted access to samples to people outside our web of trust isn't just about people whose honesty or good intentions are in doubt, or whose competence to handle samples safety is unproven … There are many ways of introducing a bias that makes one product look good, and many of them have been used… But let's say that I give you a balanced set of sound samples which any product "should" detect. How easy do you think it is to test antivirus fairly and accurately? If your immediate response is "how difficult can it be?" I promise you that you have a lot to learn."

You can read the full blog at http://amtso.wordpress.com/2010/05/18/standards-and-relevance/.

The Top Ten Threats

It probably comes as no surprise that Conficker is once again the top-ranking threat, though perhaps it should, given the age of the extant versions. INF/Autorun continues to be prevalent, even though it's now fairly easy to disable the default setting that makes this attack possible. While it's not reflected in the top ten figures, there's a notable spike in detections of the EICAR test file, which suggests that someone is doing an astonishing amount of testing…

1. Win32/Conficker

Previous Ranking: 1
Percentage Detected: 9.12%

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.

What does this mean for the End User?

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-

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. However, the Conficker Working Group estimates that there are still over 6 million infected machines out there.

2. INF/Autorun

Previous Ranking: 2
Percentage Detected: 8.06%

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.

What does this mean for the End User?

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.

3. Win32/PSW.OnLineGames

Previous Ranking: 4
Percentage Detected: 4.29%

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.

What does this mean for the End User?

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. Win32/Agent

Previous Ranking: 3
Percentage Detected: 3.25%

ESET NOD32 describes this detection of malicious code as generic, as it describes members of a broad malware family capable of stealing user information from infected PCs.

To achieve this, the malware usually copies itself into temporary locations and adds keys to the registry which refers to this file or similar ones created randomly in other operating system's folders, which will let the process run at every system startup.

What does this mean for the End User?

This label covers such a range of threats, using a wide range of infection vectors that it's not really possible to prescribe a single approach to avoiding the malware it includes. Use good anti-malware (we can suggest a good product ☺), good patching practice, disable Autorun, and think before you click.

5. INF/Conficker

Previous Ranking: 5
Percentage Detected: 1.61%

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.

What does this mean for the End User?

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.

6. Win32/Sality

Previous Ranking: 7
Percentage Detected: 1.36%

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 to specific signature:

http://www.eset.eu/encyclopaedia/sality_nar_virus__sality_aa_sality_am_sality_ah

What does this mean for the End User?

This is a classic example of malware that uses a range of techniques (file infection, autorun infection, polymorphism, terminating known security software, drive enumeration) to give itself the best possible chance of infecting and surviving once it gets a foothold. It pays to ensure that your security software is still operational, as many malicious programs try to disable AV processes, and Sality's continued prevalence after

several years in the wild indicates that these strategies are pretty successful.

7. Win32/Packed.FlyStudio.O.Gen

Previous Ranking: 6
Percentage Detected: 1.05%

Flystudio O.Gen are detections for obfuscated FlyStudio executables. They are separated from generic Win32/Packed.Flystudio (without any letter) as that detection covers legitimate Flystudio code as well.

What does this mean for the End User?

Obfuscated executables are not always malicious: sometimes obfuscation is used as a means of legitimate digital rights management (DRM) by hampering attempts at malicious reverse engineering. However the use of packers and obfuscators has been a fairly reliable indicator of malicious intent for some years now, and some vendors detect almost any obfuscated code as malicious or potentially malicious.

8. Win32/Tifaut

Previous Ranking: 9
Percentage Detected: 0.94%

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.

What does this mean for the End User?

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

9. Win32/Pacex.Gen

Previous Ranking: 8
Percentage Detected: 0.83%

The Pacex.Gen label designates a wide range of malicious files that use a specific obfuscation layer. The .Gen suffix means "generic": that is, the label covers a number of known variants and may also detect unknown variants with similar characteristics.

What does this mean for the End User?

The obfuscation layer flagged by this detection has mostly been seen in password- stealing Trojans. However, as more malware families appear that don't necessarily use the same base code but do share the same obfuscation technique, some of these threats are being detected as Pacex.

However, the increased protection offered by multiple proactive detection algorithms more than makes up for this slight masking of a statistical trend: as we discussed in a recent conference paper, it's more important to detect malware proactively than to identify it exactly. ("The Name of the Dose": Pierre-Marc Bureau and David Harley, Proceedings of the 18th Virus Bulletin International Conference, 2008 - http://www.eset.com/download/whitepapers/Harley-Bureau-VB2008.pdf; "The Game of the Name: Malware Naming, Shape Shifters and Sympathetic Magic" by David Harley - http://www.eset.com/download/whitepapers/cfet2009naming.pdf)

10. Win32/Spy.Ursnif.A

Previous Ranking: 10

Percentage Detected: 0.80%

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

What does this mean for the End User?

While there may be a number of clues to the presence of Win32/Spy.Ursnif.A on a system if you're well-acquainted with the esoterica of Windows registry settings, its presence will probably not be noticed by the average user, who will not be able to see that the new account has been created. In any case it's likely that the detail of settings used by the malware will change over its lifetime. Apart from making sure that security software (including a firewall and, of course, anti-virus software) is installed, active and kept up-to-date, users' best defense is, as ever, to be cautious and proactive in patching, and in avoiding unexpected file downloads/transfers and attachments.

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 9.12% of the total, was scored by the Win32/Conficker class of threat.