SMALL BLUE-GREEN WORLD

Untangling the Wheat from the Chaff in Comparative Anti-Virus Reviews

Version 1.2

David Harley BA CISSP FBCS CITP

http://www.smallblue-greenworld.co.uk/

About the Author

David Harley BA CISSP FBCS CITP launched Small Blue-Green World in April 2006. when he left the UK's National Health Service in order to focus on writing, publishing and consultancy.

He has worked in IT since 1986 - in application and network support, system administration and since 1989 in IT security management. While specializing in virus and email abuse management, he also holds qualifications in general security management, service management (ITIL), security auditing and medical informatics. He was formerly manager of the National Health Service's Threat Assessment Centre.

David is a prolific author, and an experienced technical and general editor, reviewer and proofreader. He is the author and editor of the "AVIEN Malware Defense Guide" 2007: and joint author of "Viruses Revealed" 2001, and has contributed security-related chapters to about a dozen other books to date. He joined the Research team at anti-malware vendor ESET LLC in January 2008, and became Director of Malware Intelligence there in July 2008.

Document Revision Information

Version 1.1: minor editing corrections, plus textual changes taking into account feedback from one of the developers of ClamAV

Version 1.2: updated prior to submission to the Anti-Malware Testing Standards Organization (AMTSO) web site, so as to clarify the current affiliations of the author. The references and resources sections have also been checked and updated where necessary (including references to AMTSO, which didn't exist when the last version of the document was written): otherwise, the content remains unchanged apart from some reformatting to improve readability.

Introduction

One of the banes of the anti-virus researcher's life is the widespread confusion about “correct” comparative testing of anti-virus and anti-malware products. I've just been looking at some reports describing an "AntiVirus Fightclub" test [1, 2] carried out recently at LinuxWorld.

Forget the fact that this looks at first glance like a test of Linux products (actually, it wasn’t [3]): there are questions to be raised about testing methodology that need to be considered on any platform. Discussions on other blogs, mailing lists and other resources [4] within a day or two of the event illustrate how easy it is to take the competence of a public test for granted, and it's because of the widespread misunderstanding of the implications of this test that I'm taking it to a wider audience.

I'm not trying to give you a full-blown guide here to conducting your own comparative tests, but hopefully I can give you an idea or three about what to look for in someone else's test, and which indicators should set off the loudest alarm bells. And while I'm not going for anyone's throat here, Untangle's test gives us several of those indicators.

Of course, I'm not suggesting deliberate malpractice or duplicity on the part of the tester or the testing organization: in fact, the tester's openness about testing methods and his expressed willingness to improve his methodology according to advice from others is commendable. In fact, he’ll see a copy of this paper and be given the opportunity to comment. On the other hand, you should be aware that the testing organization markets a gateway product which incorporates the use of ClamAV which did far better in this test than it has in more stringent formal testing [5].

A couple of mainstream products also did well, but the fact that there’s such a disparity between their results and those of other mainstream products in itself suggests major problems, as does the fact that other testers within the industry have got different results using the same test set.

This article isn’t intended to “get at” ClamAV (clamav.net), by the way. I don’t happen to believe that it can offer all the facilities of a full-blown commercial product suite, and there are issues such as contractual and support requirements that make it unsuitable for many organizations, but it is, in some contexts, able to offer comparable performance, and I agree that it’s a pity that there aren’t more publicly available test results for tests in which the product was a participant. There are people on the project and in the research community who would like to see better communication and cooperation between both parties, but there seem to have been some misunderstandings and slippages of communication. I think, though, that it’s wrong to suggest that the product’s absence from such tests is due to some sort of conspiracy by the AV establishment.

Reviewing the Test

OK. To business. Firstly, the test under discussion uses a test set of just 35 [3] samples of (presumed) viruses. Or 18, according to the tester’s blog (there are several duplications, but I haven’t verified the exact number of individual virus variants: life’s too short, and no-one is paying me to do Untangle’s verification for them) Believe it or not, this isn't the smallest test set I've seen. And at least the tester tried to use real viruses, even if he didn't succeed, or validate the ones he did use – I'll talk at length about validation in a moment. I've lost count of the number of tests I've seen where the tester used simulated viruses, virus fragments, or even "virus-like" programs, whatever that means. Clearly, though, this number pales into insignificance besides the total number of known malicious programs.

I'd love to tell you exactly what that total *is*, but it would be pointless to try. Because of the huge tangled mess of malware families, variants and subvariants, it depends entirely on how you count them, and everyone does it differently. At the moment of writing, for instance, one particular AV program's web site claims to detect 73,622 [6] items of malware, while another claims 430,888 [7]. The difference isn't hype: it's just a different way of counting.

What I can tell you is how many viruses are "In the Wild" (ItW) at the moment (August 2007), as defined [8] by the WildList Organization (www.wildlist.org) in its latest WildList: five hundred and twenty-five. Or one thousand, nine hundred and fifty-eight, if you include the supplementary list. And most sound testing (or what is generally accepted in the anti-virus industry as sound) is based, at least in part, on the WildList collection. (I'll tell you why in a minute.) Even if we assume (for the sake of argument, rather than from conviction) that these 18 samples represent 18 of those 525 validated viruses, what does this tell us about the comparative capabilities of each product in terms of ItW detection? Virtually nothing. A product that detects all 18 may not detect any of the others, and a product that only detects one or two may detect all the others. And between those two scenarios, there are many, many intermediate possibilities.

But should we assume that this sample set is a subset of the viruses categorized as ItW by the WildList? Unfortunately, we can't:

- Several of these “viruses” aren't viruses at all. They're instances of the EICAR test file (www.eicar.org/anti_virus_test_file.htm), a tiny program which is recognized by most anti-virus programs and demonstrates that they are working as expected, but which says nothing about their malware detection capabilities.
- Some of these samples were supplied by the audience at the Linuxworld event, and the tester admits that he doesn't really know what they are. He assumes that they're either "zero-day" viruses, by which he presumably means that they're too recent to be identified as specific variants, or "custom" viruses. I don't know what he means by that: samples modified to make them harder to detect? Malware written specifically for the occasion? Malware targeting specific individuals or organizations that hasn't been seen anywhere else? Whatever, this gives the lie to the assertions elsewhere that the test set consisted of “common” viruses.
- Some commentators have assumed they were supplied by one of the vendors whose products were testing, which would obviously skew the results [9]. (We can safely assume that he doesn't have access to the WildList collection: such access is only given to trusted, bona fide members of the AV research community, and no-one in that community would publish a test based on an insignificantly sized sample set). Subsequent discussions indicated that they were simply acquired from the tester's own mailbox, as on previous occasions [10]. Either way, we have no evidence of true validation. I presume the testing organization uses its own gateway product, which uses ClamAV, which suggests that the tester committed the cardinal sin of “validating” using one of the products under test.

Sample Validation & WildCore

Sample validation is a critical factor in sound detection testing [11], and is technically very demanding. You cannot just point one or more virus scanners at a sample, and, if it is identified by one of them as virus X, assume that it is indeed virus X. Especially if the scanner you use to "validate" is one of the scanners that you're testing: this introduces an enormous bias into the test, relying on the competence of the scanner provider -- advantageous if you're conducting a marketing exercise, unacceptable in a genuinely impartial test. Real validation requires, among other things, that you prove that you're working with a viable replicative sample (assuming we're talking about viruses, of course - other types of malware present other problems...) and that it is correctly identified as a specific malicious program/variant/subvariant. This is difficult and time-consuming to do correctly, which may be why amateur testers hardly ever do it. That's also why well-founded comparative tests are expensive to mount and therefore not necessarily made available to non-subscribers. It's also

why WildList testing [12] is still important [13], even though the entrants on a specific WildList represent only a small proportion of all known malware, and even of malware that is known to have been ItW at some point. (A great deal of malware, notably those viruses we sometimes call zoo viruses, never gets into the wild at all.)

WildCore, the collection on which sound ItW testing is based, has already been through a validation process, though testers given access to it are still expected to generate and validate their own samples rather than simply throw WildCore samples at a scanner. This offers a good baseline for comparative testing, perhaps the best that we have in the current climate, partial and imperfect though it is. Also rather importantly, starting from a good baseline reduces the risk of false positives (innocent objects misdiagnosed as malicious). This is important because otherwise, a competent product may actually be penalized for being right, because the tester incorrectly assumes that it failed to detect malware. Since this test simply assumes that all samples are valid, the FP (false positive) issue hasn't been addressed.

As it happens, on this occasion the sample set was made available to all and sundry on the tester/vendor web site. This isn't altogether a good thing -- in fact, it's something no professional tester would normally do, for ethical reasons. There are other problems, too: this puts the site to all intents and purposes into the position of a malware distributor, which could compromise it legally and might also be a breach of web and internet service provider requirements. However, it does show good faith in terms of being honest and open and allowing others to reproduce the results of the test. Unfortunately, though, most people do not have the knowledge or resources to verify the samples any better than the original tester did, so will tend to reproduce not only the results, but the flawed methodology. At best, they'll try the samples against sites like VirusTotal (www.virustotal.com), which test uploaded files against a battery of scanners. However, this is poor practice, and sites like this are usually eager to disassociate themselves from any suggestion that they're suitable for that purpose. Such a submission may produce rough identification of samples, but it doesn’t constitute the sort of stringent validation that’s necessary for competent formal testing.

Apples & Oranges

Another indicator of imperfect testing methodology is the use of "apples and oranges" comparative tests, where applications with disparate functionalities are lumped in together as if they were . In this test, appliances, gateway scanners, and desktop scanners for more than one platform were all tested together, and command-line and GUI interfaces were used indiscriminately. This is actually a special case of a more general methodological weakness: a failure to recognize the importance of establishing a level playing field in terms of configuration and capabilities (level of heuristics set, archive scanning, and so on). By the tester's own admission [1] that the original setup disadvantaged the Sophos product, it looks as though the products were tested pretty much "out of the box" without considering whether the conditions of the test would disadvantage specific default configurations. There's a particular problem in this case. WatchGuard, which uses the ClamAV engine, nevertheless failed to detect anything but the EICAR test file, in stark contrast to Clamav's own apparently stellar performance. This, to me and many others [1], clearly suggests a problem with configuration. The wide variation in detection rates compared to the comparatively narrow ranges in most “professional” tests may also reflect configurational inconsistency, as well as an unreliable sample set.

Muddled Methodology

Yet another indicator of poor methodology is a lack of clear purpose. In this instance, the "Wild" test set was further invalidated by the inclusion of the EICAR test file (www.eicar.org). The user sample set, already compromised by lack of validation, was presumed to include "zero-day" and "custom" malware. It's unreasonable to expect a scanner to identify specific

malware that isn't yet known, so this apparently simple test now includes at least four types of test.

- Recognition of the EICAR test file (based on the incorrect assumption that recognizing EICAR proves correct configuration [14])
- Recognition of presumed malware, presumed known to be "In the Wild" (or meeting the testing organization's view of what ItW means, more to the point)
- Recognition of presumed malware not presumed to be ItW but might be known to the scanner ("zoo" testing)
- Recognition of presumed malware not expected to be known to the scanner (heuristic testing).

Any of these might be considered a valid test target, I guess, but in the absence of proper explanation, separation or quantification of the targets, effectively useless. Not to mention misleading: bloggers and commentators [4] have already been misled into thinking that this tested “In the Wild” detection.

The last indicator of unsound methodology that I’m going to consider here is the enormous disparity between the results here and the general trend in established, reputable tests. Maybe everyone else has got it wrong and Untangle have it right, but given the other indicators of poor methodology, I doubt it.

So, did Untangle do anything right? Technically, no. In his own reply to some of the comments his blog has stimulated, the tester makes the point that his test is more “valid” than other tests because it’s more “relevant” to real world detection, missing two vital points:

- Commercial AV *has* to try to catch all known ItW and zoo viruses because it can’t predict all the contexts in which it may be installed and used. Because they’re contractually responsible to their customers, they have to try to cover all bases. It’s obviously not only valid but desirable to test their ability to do this.
- Professional testers don’t just run “600,000” viruses against a scanner, contrary to the tester’s assertions. They conduct many different kinds of test, including ItW testing, zoo testing, Time to Update testing (contentious though that is [11]), retrospective testing of heuristics, and even some forms of usability testing.

But I do commend his willingness to discuss and improve his own methodology, and he may have done us all a favour by pointing out again the security community’s failure to communicate the difficulties and complexities of competent testing, and persuade the wider community that “professional” testing organizations, however imperfect, offer very real advantages over informal, unconsidered tests.

Lessons to Learn

Finally, what can we learn from the above and apply to other tests?

1) Small sample sets tell you how a given scanner performed against a small set of presumed malware, in a specific "snapshot" context:

- According to the testing conditions
- According to the way the scanner was configured for the test

They tell you nothing about how the scanner will perform against any other sample set. If you want to test detection of ItW viruses meaningfully, you have to use a full and fully validated test set that meets an acceptable definition of “In the Wild,” not a few objects that may be viruses and may be ItW. This is why the WildList Organization, with all its problems [15], remains important to the research community.

2) Unvalidated samples invalidate a test that uses them. A collection needs care and maintenance [16], as well as a significant test corpus.

3) You get what you pay for. Well, often you get less than you pay for, or thought you were paying for, and sometimes you do get more than you pay for. However, while a voluntary community resource can make a significant contribution to the common weal, even in security (ClamAV, Snort, and so on), it can't match a full strength industrial solution in all respects. Watch out for the halo effect: when people find a positive attribute in an object, such as a $0 price tag, they're tempted to overestimate its other positive attributes and capabilities. That's very human and understandable, but it has no place in a rigorous testing program, and to be less than rigorous when you're making recommendations that affect the security and well-being of others is reprehensible.

4) Another concept you should be aware of is ultracrepidarianism [17], which can be informally defined as a tendency for those with a platform to speak from to overestimate their own competence in subjects in which they have no specialist expertise. Sadly, the anti-virus field in general and amateur testing in particular is over-endowed with "instant experts." [18] When looking at a test, you are advised to take into account the expertise and experience of the individual conducting the test. The widespread popular distrust of the anti-virus community [19] extends not only to attributing malicious behaviour to AV vendors ("they write the viruses") but to assuming their essential incompetence. Scepticism is healthy, but apply it to people outside that community, not just those within it!

5) In the Wild is a pretty fluid concept. In fact, it's not altogether meaningful at all in these days when worms that spread fast and far are in decline, and malware is distributed in short bursts of many, many variants. Come to that, viruses (and worms) are much less of an issue they were. Anti-virus isn't restricted to that arena (though you may have been told otherwise by instant experts), but it can't be as effective in all areas as it was when viruses were public enemy number one. On the other hand, *no* solution is totally effective in all areas. The AV research community is (slowly and painfully) coming to terms [20] with the fact that the test landscape has to change. Mistrust any test that doesn't even recognize that the problems exist.

6) There are a whole raft of problems connected with the types of object used for testing: non-viral test files, garbage files from poorly maintained collections, unvalidated samples from malware generators, simulated viruses and virus fragments, and so on. I won't go into further detail at this time, but the points to take away are:

- If you don't know anything about the test set, assume it's rubbish.
- If you don't know where it came from, mistrust it.
- If you don't know how or if it was validated, mistrust it.
- If you suspect that it came from one of the vendors under test, mistrust it.
- If you don't know what the samples were or how many, mistrust them.
- If you're offered the chance to test the same set for yourself, be aware that unless you're an expert on malware and testing, or have reliable contacts in the community who can do it for you, you'll probably reproduce faulty methodology, so the results will be useless.

7) Sites like VirusTotal are not intended to conduct any sort of comparative testing, they're for trying to identify a possibly malicious object at a given moment in time. Unless you know exactly what you're doing, any results you get from such a site is useless for testing purposes.

8) The EICAR test file is not a virus, and doesn't prove that you've configured your test-bed apps properly.

9) Your test-bed apps have to be similar in functionality, and should be configured carefully so that no single product has an unfair advantage. One particularly memorable example some years ago was a test that reviewed several known virus scanners and a single generic

application. The latter was given the “editor’s choice” accolade because it stopped the entire test set from executing. This sounds fair enough unless you realize that many people and organizations still prefer virus-specific detection because generic products can’t distinguish between real threats and “innocent” objects: this usually means that either all objects are blocked (executable email attachments, for instance) or the end user has to make the decision about whether an object is malicious, which, for most people, defeats the object. By failing to acknowledge this issue, the tester invalidated his conclusions by imposing his own preferences on what should have been an impartial decision. In other words, apples and oranges look nice in the same fruit bowl, but you need to know the difference between them before you choose one to eat.

10) There are reasons why some tests are generally considered valid by the AV community. Some of them may be self-serving – the vendor community is notoriously conservative [21] – but they do derive from a very real need to implement a stringent and impartial baseline set of methodologies. Unfortunately, to do so requires considerable investment of time and expertise, and that’s expensive: that’s one of the reasons that many first-class tests are not available to all-comers. (To the gentleman who pointed out to me that you don’t need to be a cook to know if something tastes good, I have to point out that you *do* need to know something about nutrition to know whether something that tastes good is actually good for you…)

To understand what makes a test valid, look at the sites listed below, find out how they conduct tests and learn from it. You don’t have to accept everything they say – I don’t – but you’ll be in a better position to assess comparative reviews in the future. I’ve also included some extra reading resources, to the same end.

None of the following sites has the universal, unquestioning approbation of the entire anti-virus research community, but they are taken seriously:

- Virus Bulletin http://www.virusbtn.com
- ICSA Labs http://www.icsalabs.com
- West Coast Labs http://westcoastlabs.org
- AV-Test.org http://www.av-test.org
- AV Comparatives http://www.av-comparatives.org

[Added 4th November 2008] Since the initial version of this document was released, the Anti-Malware Standards Organization (AMTSO [22]) has come into being, with members including most mainstream anti-malware vendors, major testing organizations, reviewers and publishers. After several face-to-face meetings and much internal and public discussion, the membership unanimously approved the final versions of two major documents: “The Fundamental Principles of Testing” and “Best Practices for Dynamic Testing” [23]. These are essential reading for anyone interested in testing issues, whether as a consumer, tester, or security professional. There are lots of other resources in the pipeline, and we strongly recommend that everyone with the faintest interest in testing issues should follow the organization’s progress closely.

References

[1] http://blog.untangle.com/?p=95

[2] http://blog.untangle.com/?p=96

[3] Hiep Dang: “What a Tangled Web” at
http://www.avertlabs.com/research/blog/index.php/2007/08/12/what-a-tangled-web/

[4] Tim Wilson: “Antivirus Tools Underperform When Tested in LinuxWorld 'Fight Club'”
http://www.darkreading.com/document.asp?doc_id=131246&WT.svl=news1_5

[5] Larry Seltzer: “AV-Test.org Reports Stats from Antivirus Roundup” at
http://www.pcmag.com/article2/0,1759,2135092,00.asp

[6] http://www.symantec.com/enterprise/security_response/index.jsp

[7] http://www.f-prot.com/virusinfo/index.html/

[8]Sarah Gordon: "What is Wild?" at http://csrc.nist.gov/nissc/1997/proceedings/177.pdf

[9] Dr. Alan Solomon: “A Reader’s Guide to Reviews” (originally published in “Virus News International” and credited to Sarah Tanner), at
www.softpanorama.org/Malware/Reprints/virus_reviews.html

[10] http://blog.untangle.com/?p=20

[11] David Harley and Andrew Lee: "Antimalware Evaluation and Testing", in "AVIEN Malware Defense Guide for the Enterprise" Syngress 2007

[12] http://www.virusbtn.com/vb100/about/100procedure.xml

[13] Mary Landesman: “The Wild WildList” in Virus Bulletin, July 2007.

[14] David Harley: “Can I get a virus to test my antivirus with?” in alt.comp.virus FAQ,
http://www.faqs.org/faqs/computer-virus/alt-faq/part4/

[15] Randy Abrams: “AV Industry Comments on Anti-Malware Testing” in Virus Bulletin, June 2007.

[16] Vesselin Bontchev: “Analysis and Maintenance of a Clean Virus Library”, at
http://www.people.frisk-software.com/~bontchev/papers/virlib.html

[17] Rob Rosenberger: “False Authority Syndrome”, at
http://www.cknow.com/vtutor/FalseAuthoritySyndrome.html.

[18] David Harley, Robert Slade, Urs Gattiker: “Viruses Revealed”, Osborne 2001.

[19] David Harley: “I’m OK, You’re Not OK” in “Virus Bulletin, November 2006.

[20] Igor Muttik: “Antivirus Testing Workshop in Reykjavik” at http://www.avertlabs.com/research/blog/index.php/2007/05/29/antivirus-testing-workshop-in-reykjavik/

[21] Richard Ford, Attila Ondi: “Testing Times Ahead?”, Virus Bulletin, April 2007

[22] http://www.amtso.org/

[23] http://www.amtso.org/documents/cat_view/13-amtso-principles-and-guidelines.html

Resources

Sarah Gordon and Richard Ford:“Real World Anti-Virus Product Reviews And Evaluations – The Current State Of Affairs” at http://csrc.nist.gov/nissc/1996/papers/NISSC96/paper019/final.PDF

Adam J. O’Donnell: “Real-World Testing of Email Anti-Virus Solutions”, in Virus Bulletin, March 2007.

Joe Wells: “Pragmatic Anti-Virus Testing” at http://www.sunbelt-software.com/ihs/alex/Pragmaticantivirustesting.pdf

Alex Eckelberry: “More testing silliness” at http://sunbeltblog.blogspot.com/2006/08/more-testing-silliness.html

Randy Abrams: “Doesn’t the EICAR test file look spiffy?” at http://www.eset.com/threat-center/blog/?p=15

Randy Abrams: “Giving the EICAR Test File Some Teeth” in the proceedings of the “Ninth International Virus Bulletin Conference and Exhibition”, 1999.

Igor Muttik: “Comparing the Comparatives” at: http://www.mcafee.com/us/local_content/white_papers/threat_center/wp_imuttik_vb_conf_2001.pdf

AV-Comparatives Resources: http://www.av-comparatives.org/seiten/ergebnisse_2007_02.php; http://www.av-comparatives.org/seiten/ergebnisse/2ndgrouptest.pdf; http://www.av-comparatives.org/seiten/ergebnisse/methodology.pdf

AV-Test Resources: http://www.av-test.org/index.php?menue=1&lang=0

The EICAR test file: http://www.eicar.org/anti_virus_test_file.htm

The Anti-Malware Testing Standards Organization (AMTSO): http://www.amtso.org