



Testing testing

Anti-Malware Evaluation for the Enterprise

David Harley
Research Author
ESET LLC

Andrew Lee
Chief Research Officer
ESET LLC

This paper was written for and presented at the 10th Annual AVAR
(Association of anti-Virus Asia Researchers) International Conference in Seoul 2007.

About the Authors

David Harley CISSP, ESET Research Author, is an experienced and well-respected anti-virus researcher, and also holds qualifications in security audit, ITIL service management, and medical informatics. Until 2006 he worked in the UK's National Health Service, where he specialized in the management of malicious software and all forms of email abuse, and managed the Threat Assessment Centre. He has worked as an independent author and consultant to the anti-virus and security industries, and is Chief Operating Officer of AVIEN (Anti-Virus Information Exchange Network).

He was co-author of "Viruses Revealed" and has contributed chapters to many other books on security and education for major publishers, as well as a multitude of articles and conference papers. He was technical editor and lead author of "The AVIEN Malware Defense Guide for the Enterprise", also for Syngress.

Andrew Lee CISSP is Chief Research Officer of ESET LLC. He was a founding member of AVIEN and its sister group AVIEWS (Anti-Virus Information & Early Warning System), is a member of AVAR (Association of anti Virus Asia Researchers) and a reporter for the WildList organization. He previously worked at the coalface of malware defense as a systems administrator in a large government organization.

Andrew was a major contributor to the "AVIEN Guide", and is also author of numerous articles on malware issues. He is a frequent speaker at conferences and events including ISC2 Seminars, AVAR, Virus Bulletin and EICAR.

Table of Contents

	Page
About the Authors	1
Abstract	3
Introduction	4
Testing Our Patience	5
Reading Between the Lines of Comparative Reviews	6
Red Flags & Red Teams	7
Anti-Malware Companies against the World	10
Last Scan Standing	12
The Ethics of AV Testing	13
Technical aspects	14
Garbage In, Garbage Out: Verifying Samples	14
Call of the WildList	15
Spot the Fallacy	17
How Practical is DIY Testing?	18
Viruses Aren't the Whole Problem	19
What do you Need for Sound Testing?	21
Who are the trusted testers?	22
It's Not My Default	23
Conclusion	24
References	25
Additional Resources	28

Abstract

Anti-malware software remains an essential defensive component for most enterprises, understandably anxious to get the right balance of affordability and effectiveness. Unfortunately, journalists, consumer groups and security amateurs keep finding ever more creative and inappropriate ways to test detection focused software. In this paper, we attempt to address a number of core issues:

1. Reading between the lines of comparative reviews
2. Anti-virus/malware against the world
 - The ethics of product testing
 - Trust and competence
 - Fact and fiction in the public view of the anti-malware industry
 - What the rest of the security industry doesn't understand
3. Technical aspects of testing:
 - Garbage In, Garbage Out: sample verification
 - Testing with replicative malware
 - Proactive (retrospective) testing and heuristics
 - Time to Update (TtU) testing
 - In the Wild testing
 - Non-replicative malware
 - Realtime versus on-demand testing
 - False positive testing
4. Evaluating the evaluators: sound versus unsound resources
 - Testing and certification
 - Specialist reviewers
 - Outsourced testing
 - The persistent droning of the security amateur and instant expert
5. The pros and cons of DIY testing: how practical is it?

Introduction

Like malware nomenclature and the alleged total reliance of anti-virus software on signature detection, detection performance testing of anti-malware programs, especially comparative testing, comes round time and again as a cause of furious debate. It is, yet again, a major issue at the time of writing because of a controversial recent comparative test carried out by Untangled.com at LinuxWorld [1; 2]. Quite rightly, this aspect of product evaluation is seen as being of major importance:

- Measures to control the encroachment and impact of malware, whether as conventional anti-virus or as part of an overall intrusion prevention system, are an essential defensive component in the enterprise
- They're not "all the same": while the range of known viruses detected by mainstream anti-virus scanners is fairly consistent, the range of other functionalities and other types of malware detection varies very widely. This is particularly so in detection of new (previously unseen) threats.
- There is a need for most enterprises and individuals to achieve a balance between affordability and effectiveness

Testing Our Patience

Unfortunately, some journalists, consumer groups, and security amateurs keep finding ever more creative and inappropriate ways to test detection-focused software. Some of the trouble spots we have noted over the years in comparative reviews include:

- Test sets that include non-viruses and non-viable malware such as intendeds, garbage files, innocuous files, and harmless test files.
- Simulated malware. This can lead to many complications, such as the essential paradox that a scanner may be “rewarded” for incorrectly diagnosing a simulation as the malware it impersonates (bear in mind that the purpose of a malware detector is to detect malware, and only malware).
- Kit malware, which frequently results in the generation of unviable samples [2, 3]
- Contextually inappropriate malware or non-malware: for instance, testing web scanners with HTML samples that only ever appeared in the wild as SMTP transmissions [4], or the use of the EICAR test file incorrectly embedded into a Word document [5]
- Unvalidated samples presumed to be malware (usually because something has identified it as a specific threat, as a recognizably generic detection, or as “suspicious.”) [6]
- “Circular” validation (malware is “validated” by testing against one of the products under test) [6, 7]
- Apples vs. Oranges: comparative tests where products of significantly differing functionality, levels of configuration, and so on, are tested with the same test set and essential methodology. For instance, testing scanners using different operating systems, or irrespective of whether they’re designed for desktop, LAN server or perimeter placement, or the range of protected services. [1, 6]
- Fuzzy test targets: for example, making no clear distinction between tests of heuristic detection, generic filtering, near-exact identification, and so on. [6, 7, 8]

It has been pointed out to us that you don’t need to be a cook to know if something tastes good. We contend, though, that you do need to know something about nutrition to know whether something that tastes good is actually good for you...

Reading Between the Lines of Comparative Reviews

Poor testing is rarely questioned except by the anti-malware industry, which is perceived as having sinister motives (even by testers) for not wanting any testing, or at any rate no testing it doesn't in some sense control. This isn't altogether without foundation: the industry generally acquires the most complete collections and routinely validates and classifies suspicious programs as part of their processing, and doesn't share them easily. At least some of the reasons for this reluctance are entirely honourable, but it's all too easy for the public (or the "thought leaders" who influence the public perception) to interpret it as self-protective and self-serving [9].

There's a curious duality here: firms in this industry sector are often assumed to pursue competitive advantage by keeping samples to themselves, creating their own malware, and so on, yet are also believed to close ranks and conspire together for the good of the industry and to the detriment of the common good. [10] We can't swear that no two (or more) AV people have ever conspired together in sinister, monopolistic, cabalistic ways, but we rarely have cause to suspect such a conspiracy. We do sometimes find it strangely difficult to convince outsiders of how much cooperation there is between researchers across corporate perimeters, especially in the context of sharing samples between trusted individuals.

"...irrespective of its technical advances, the anti-virus industry continues to fail to win hearts and minds. On the contrary, we are mistrusted by our customers, by the media, and especially by other sectors of the security industry. We are, apparently, incompetent, elitist, cabalist, money-grabbing, publicity-greedy, and generally ethically challenged. But we have our bad points, too." [9]

To the informed eye, the testing trouble spots referred to in the previous section can flag an incompetent test as surely as the stars of CSI can pinpoint a murderer. However, where test reports are based on these assumptions and stereotypes, such a bias also suggests questionable competence and general lack of knowledge of the field. Certainly, there are well recognized and accepted testing organisations, which would belie such arguments. Such organizations and testers manage to maintain their independence from the industry, while being largely embraced by it. Surprising though it may seem, most anti-virus researchers want to see good testing, for a number of reasons.

Firstly – a good product will shine in a good test, and may do badly in a bad test. This is a huge frustration for vendors who know that the quality of their product is not being correctly reflected by a test.

Secondly – a good test can legitimately reveal flaws and areas of weakness in anti-malware products, which it is in the best interest of the vendors (and their customers) to know about, and be able to rectify.

Thirdly – it's good marketing to achieve a good result in a well respected test. Again, it may surprise some to find that some vendors will not use the results of certain tests in marketing (even if the result was 'good') because the reputation and quality of the product would not be upheld by achieving a high score in a poor test (and surely it would be used against that vendor by its competitors).

Fourthly – good testers, and good tests keep the vendors honest. If vendors were simply to use the results from poor tests, it would be largely unnecessary for them to actually make a decent product, rather just fitting their products out to pass tests. A real test of the capabilities of a product are of benefit to everyone. Conversely, a poor test is a disservice to the customers of the vendors in the test, and actually to the wider community of anti-malware users, as it misrepresents (either favourably or otherwise) the real capabilities of the products. Anti-malware products are some of the most complex and advanced technologies in modern software, and it requires no little effort to test and evaluate them correctly

It may be possible to conduct useful tests without being a world-class expert on anti-virus, but it probably isn't possible to do without a reasonable idea of how malware and anti-malware technologies work, and dubious conspiracy theories, however well they work as movie plots, are a poorer basis for rigorous testing than scientific principles.

There are other indicators of poor practice, though.

Red Flags & Red Teams

Third party providers of anti-malware services (outsourced services, re-badged engines, multi-engined products) may be considered as part of the anti-malware industry, but their knowledge of threat and counter-threat technology is often surprisingly basic [11]. Not infrequently, the anti-malware component of the service is essentially a black box containing an unidentified engine with a proprietary wrapper around it. In extreme cases, neither the service provider nor the customer is able to carry out significant customization or configuration of the core functionality, either because they lack the knowledge or because the application wrapper wasn't designed to give them sufficient access. So when a 3rd-party provider publishes their own test, it might be naïve to trust their competence, let alone their own partiality.

But, of course, vendor sponsorship/testing may well suggest a conflict of interest. In one recent case [1] a provider of services including malware filtering ran some tests on a number of products including the one that they were already incorporating into their service. While we don't suggest deliberate malpractice, there is a risk in such a case that the tester may overrate their own competence [12] and be biased in favour of a program that they already use (especially when that program happens to be free): after all, results that indicate that the product concerned doesn't meet or exceed the same standards as other products may have a negative marketing impact on the wrapper service. Clearly, producers of commercial (core) anti-malware products are usually well capable of running competent comparative tests and often do so routinely in order to check their own performance against the competition. However, they usually avoid these dilemmas of ethical uncertainty and risks of negative marketing by not making their results public and maintaining a discreet distance from reputable testing organizations even while cooperating with them.

Unsuitable or unspecified validation and testing methodologies constitute a major red flag. While it would often be impractical to go into enormous descriptive detail for each test, statements such as "we took viruses off blackhat web sites and ran them against the tested products" or complete silence as to how a test was conducted should be regarded with extreme prejudice. [13]

Tiny test sets rarely have a place in competent testing, even if the samples concerned have been correctly validated. There can be exceptions to this [14], but the onus in such a case is on the tester to make unequivocally clear the limitations of this approach and why it is appropriate in this particular case. There is a point of view that "if you test with 100 viruses, of which only three are actually in the wild, it's only those three that I'm interested in." This view is by no means altogether invalid, but it misses important points:

- If you accept this viewpoint, you have to be sure that what you're testing with is in fact In the Wild (ItW). This isn't nearly as simple as it appears to the testing neophyte.
- Anti-virus/anti-malware products cannot only detect what is ItW: they have to detect not only what used to be ItW (because it might pop up on an obsolete system, resurrected media and so on), but also malware that is known to exist but has never been ItW (zoo viruses, for example) in case it does suddenly "get lucky" and appear in the real world.

What is really being tested? There is a class of misconceived comparative review often referred to as "Oranges and Apples" (or vice versa) testing, because it involves treating very different objects as if they were identical in form and function, and therefore assuming that identical methodology is appropriate to testing. We won't go into detail on the finer

points of different types of performance testing to avoid undue duplication of material in another presentation at this conference [15]. (For similar reasons, we will not attempt an exhaustive definition of testing strategies and solutions [16] and recommend the other papers in this strand to you.) However, we cannot leave this point without stressing the need to understand the differences between different kinds of performance testing and the dangers of mixing test types without such understanding. To take a recent example already cited [1], the test concerned attempted to hit several targets with the same arrow [6]:

- It included appliances, gateway scanners, mail scanners, and desktop scanners, irrespective of platform and interface (GUI or command-line) all in the same test.
- It also attempted (knowingly or otherwise) to combine several kinds of test in a single sweep:
 - Recognition of the EICAR test file (apparently based on the incorrect assumption that recognizing EICAR.COM proves correct configuration) [1, 6, 13]
 - Recognition of presumed ItW malware (since it's unlikely that the tester had access to WildList (<http://www.wildlist.org>) samples, we assume that the malware was "validated" by identifying it with one or more scanners as malware named on the WildList or a similar resource – of course, this doesn't meet a professionally acceptable standard of validation, identification, or collection maintenance [17; 18])
 - Recognition of presumed malware not thought to be ItW, but which "ought" to be known to the scanners under test ("zoo" testing)
 - Recognition of presumed malware not expected to be known to the scanner (in this case, "zero-day" and "custom" malware submitted by members of the audience). In this case, there appears to have been no validation of the samples as malicious or replicative, and the tester admitted that he didn't really know what they were. By zero-day, he may well have meant samples of malware too recent to be identified as specific variants. Custom samples, we suppose, refer to modified or custom written samples. This could be categorized as an attempt to test heuristics – basically, the effectiveness of a scanner at detecting currently unknown malware by recognizing characteristics that indicate malicious behaviour. However, the complete absence of validation in this case means that what it really does is test the ability of a scanner to second guess other scanners: a tested scanner "wins" in this case by identifying objects as malicious (or at least as suspicious) that at least one other scanner will also detect, irrespective of whether it is really viral or malicious. Unfortunately, this is an extreme example of a common testing mis-methodology, bringing to

mind Rhine's parapsychological research at Duke University [19;] rather than anti-malware testing at Hamburg [20] or Magdeburg [21].

Unfortunately, there is some evidence that such poor testing (not to mention increasingly unmanageable quantities of new samples) has led to a phenomenon for which we have coined the term "cascading copycat detection", where a given object is detected (whether false alarm or not) by an anti-malware scanner, and is then, on that basis, added to the detections of numerous other scanners. This seems in many cases to be a largely automated process, where unfortunately certain vendors are simply using the scanners of other vendors to determine maliciousness – a method of categorization for which they in turn chastise bad testers. This is a slightly tangential discussion that we may revisit in another paper, with a fuller investigation of this phenomenon. Suffice to say that this sort of behaviour is certainly not helping the situation, and in fact, has led instead to some rather embarrassing replication of false positives across anti-malware scanners.

Anti-Malware Companies against the World

We have long been fascinated by the phenomenon of public ambivalence towards the anti-virus/anti-malware industry [9]. On one hand, there's the common belief that practically anyone knows more or is more truthful on the subject of malware and malware management than the anti-malware industry, including hackers (in the pejorative sense) and malware authors, security amateurs and vulnerability bounty hunters, and practically any security professional outside the anti-malware industries. On the other, it is assumed that they exercise a sinister and self-serving influence over testing agencies and other hopefully impartial groups.

There is a whole series of popular myths and (hopefully) misconceptions disseminated among all those groups about the products and the people who create and maintain them:

- That they are only concerned with detecting viruses, not malware in general. (Mind you, we have known providers attempt to wriggle out of penalty clauses by asserting that malware they missed was a worm, not a virus, and that they were therefore not contractually obliged to detect it. [22].)
- That they cannot be trusted because they write all the viruses (still!)
- Vendors are greedy because they insist on charging for their products when everyone

“knows” (and poor tests “prove” or are misinterpreted as proving [23]) that free AV is better (!)

- Notwithstanding the considerable advances in heuristic technology since the 1990s, the industry retains a reputation for being obsessed with signature detection and the protection of their revenue stream.
- Or they're simply incompetent, since they are unable to stop all malware (and bring about world peace in their spare time)

Over the years, a number of instances of “what everyone knows” have circulated relating specifically to testing:

- That the testing “establishment” is in thrall to and essentially inseparable from the vendor establishment.
- That established testers are reliant for their income on fees from the vendor establishment and that this introduces a bias against small vendors, open source vendors and so on. For example: “I’m left to assume that the testing labs are biased in their testing, probably because they get their funding from the commercial vendors that pay them for testing. Their customers surely wouldn’t be happy if the testing labs claimed a free and open source solution was better.” [1]
- That they are wilfully obscure about their methodology.
- That they concentrate on tests that perpetuate unrealistic or obsolete approaches that favours the interests of the industry rather than benefit to the customer.

Clearly, there is more truth to some of these “mythconceptions” than others. For example, some of the complaints about poor establishment methodologies may be related to early mistrust of early NCSA testing protocols [24; 25] Other complaints are related directly to concerns about the adequacy of the WildList/WildCore (at least in their present form) as a mainspring detection testing resource [26; 27; 28].

Let us, as it’s a convenient point for such an aside, put paid to the (logically fallacious, but still irritatingly persistent) myth that the vendors create malware. This is by far the most frequent question we are asked by ‘normal’ members of the public on discovering our area of specialty. Apart from the wearied smile and swift rebuttal, usually followed by a sighed response about how much we’d appreciate having more time to lay about on beaches rather than being buried in malware analysis, a more reasoned rebuttal is obvious. On the one hand, the public expect anti-virus programs to detect 100% of all malware, all the time, however, in their experience, and proven in numerous test from multiple sources, this is not the case.

Indeed, a frequent complaint to vendor support departments is 'your product missed the xxx virus on my system'. Surely, if the 'virus writing department' existed, it would behoove vendors to provide detection well before they released the malware, so that the customer would actually experience the benefit of completely comprehensive protection. Of course, not only is it utterly ridiculous to suggest that vendors create malware, it is also obvious that it would be commercial suicide for them to do so. Nonetheless, such conspiracy theories persist, and despite the enormous expenditure and logistics that would be required to cover up such activities (even greater in scale than NASA having to pretend that they landed three men on the moon in 1969), they are unlikely to go away any time soon. One thing is certain though: more testers of anti-virus have openly created 'new' viruses than ever has been the case in the anti-malware vendor community.

Unfortunately, this is a lose-lose situation: it's often suggested to us that it's a symptom of the industry's incompetence that it doesn't test its own products with its own new viruses. How do we know they don't? Actually, we're fairly sure that some industry researchers do try "proof of concept" attacks, but under conditions strictly controlled by people who are very well aware of both the ethical and practical risks. What they don't do is publish the results of those tests as a publicity exercise, as to do so would clearly be misleading. Of course, it is practically impossible to convince some people that the AV industry doesn't directly control the testing industry.

Last Scan Standing

There is also an issue here in that the anti-malware industry enjoys fairly poor standing with the rest of the security industry, which consistently fails [2] to understand that:

- Malware is not an easy specialty, and those who work in different specialties are not necessarily blessed with a deep, instinctive understanding of malware/anti-malware technology, management issues and culture
- Its 20-year-old assumptions about detection technology being entirely signature based are misfounded, again based on a poor grasp of the realities of modern malware and anti-malware technologies.
- Ultracrepidarianism and False Authority Syndrome [29] are alive and well and living at SANS, among other places, where Alan Paller commended a poorly implemented test by Consumer Reports for "helping to do important product improvement research" by 'proving' that "antivirus vendors don't find and block viruses quickly" [30; 2]

- The culture clash between the full disclosure model favoured by most of the security industry, where AV is historically secretive [31], continues to affect the relationship anti-malware specialists and other sectors of the industry, not to mention those groups influenced by those sectors (press, public, security wannabes).

The Ethics of AV Testing

The anti-malware industry frequently complains bitterly about poor tests, but does a poor job of explaining what its objections are. Outside this specific industry sector, few people understand the ethical objections raised by the industry to the writing of replicative software for testing purposes [32]: this is translated as “They say it’s unethical to test because they don’t want us to know what rubbish they are.” By all means, let’s make it clear what the ethical issues really are, but perhaps there are points that need to be made even more strongly:

- While ethical and safety objections, though by no means trivial, often fail to convince either the security mainstream [33, 34] or the security wannabes who respond to AV blog entries [35], our experience is that it’s sometimes easier to convince on a technical level. After all, while not everyone (even in the anti-malware industry) believes that it’s never justifiable to create replicative malware for test or research purposes, even under controlled conditions, it’s harder to argue that there are no moral or ethical difficulties in misleading the press and public [36], deliberately or unintentionally, by using inappropriate and poorly conceived methodology.
- The industry will not win hearts and minds by fostering the impression that all attempts to test anti-malware detection will be dismissed out of hand. It has always been next to impossible for anyone outside the charmed circle of a few trusted independent researchers to test some features of anti-malware products to a standard that the industry itself finds acceptable. In general, the historical reasons for this are honourable, but the world finds it odd that the industry can use this “elitism” to cry foul at practically any test that shows unexpected results.

Technical aspects

Let's move on to an overview of some of the more technical aspects of testing. We don't subscribe to the view that only an elite group of professionals can say anything useful about AV performance. We do contend that you can't produce a fair test on the basis of misconception, muddled thinking and false authority syndrome. If you don't know anything about testing techniques -or- malware, the odds are pretty much against your producing a valid test. Even if you have the knowledge, you can't apply that knowledge correctly without the requisite time and resources.

One of the authors was privately taken to task recently for criticizing the Untangled test methodology [1, 6] without having tested the sample set for himself. (This sample set was, somewhat problematically, made freely available on the Untangled web site.) In fact, this was based on a misunderstanding: the sample set was examined in enough detail to identify some problem samples (zero-byte files, for example). The miscreant author argued in his defense that:

- A fundamental of good testing is that you know what you're testing with – i.e. you must validate samples. However, no-one was paying him to do the tester's validation for him: validation is time- and resource-intensive when done properly, and there was no incentive or useful purpose to be served by that expenditure retrospectively. (
- Reproducing a faulty test serves no purpose except to verify that the results were as reported: it doesn't validate the methodology by which they were obtained.
- In this instance, even if all the samples had checked out, it would have had no material effect on the many methodological flaws present, such as the tiny sample set, bias towards an included scanner, and inconsistent configuration, choice of platform, and testing targets.

The important point here, though, is that sound testing has a lot to do with knowing which test types are feasible and useful, given your available resources. And of course, knowing not to run samples against inappropriate resources such as VirusTotal (which was never intended for that purpose).

Garbage In, Garbage Out: Verifying Samples

Sample validation has always [37] been seen in the industry as a critical factor in sound detection testing [38], and is technically very demanding. The industry holds fast to the belief that you cannot just point one or more virus scanners at a sample, and, if it is identified by

one of them as a particular virus, accept that “detection” as validation. More so if the scanner used for this purpose is one of the scanners under test.

Clearly, to do this introduces an enormous bias into the test, relying on the competence of the scanner provider and assuming that it is more “correct” than scanners that disagree with it, regardless of the risk of false positives (or indeed of detection of damaged files that may just happen to have enough parts intact to be detected – some vendors deliberately detect such files to reduce the amount of junk the customer will be faced with at, say, the email gateway). This approach has obvious advantages when conducting a marketing exercise, but is unacceptable in a genuinely impartial test, and demonstrates the importance of separating the testing agency from the vendor or anyone else with a vested interest in marketing one of the tested products [39].

To quote Joe Wells [40] “...one critical and often overlooked issue is the tendency to immediately suspect the AV product when a virus sample is missed. Given the historical quality of viruses and anti-virus products, it is preferable that the tester should suspect the virus sample immediately, rather than the product. It is far more likely that the sample is bad, than the product.” However, this reference to problems with a possible false negative clearly doesn’t mean that a single scanner reporting a threat should not be suspected of flagging a false positive. It simply emphasizes the need for the tester to be scrupulous about (1) the quality of the sample – it needs to be a genuinely malicious and/or replicative program, depending on the type of test (2) the provenance of the sample – it needs to be correctly identified as a specific item of malware, and in its correct context (for instance, a single variant/subvariant of unknown “wildness” should not be described or used as if it were a validated, WildCore-originated sample meeting the technical criteria for In the Wild malware [41]

Call of the WildList

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 [6] still has a place [7], even though the entrants on a specific WildList represent only a small proportion of all known malware [41], and even of malware

that is known to have been In the Wild (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 samples at a scanner. This does offer a baseline for comparative testing, perhaps the best that we have in the current climate, partial and imperfect though it is. Starting from a good baseline reduces the risk of false positives (innocent objects misdiagnosed as malicious): otherwise, a competent product can be penalized for being right, because the tester incorrectly assumes that it failed to detect malware. (We may have mentioned this issue before...) Nonetheless, it would be naïve to pretend WildList-based testing is universally admired, even among the AV research community [17, 42]. The problems with the current WildList are well known (not least to the WildList Organization – <http://www.wildlist.org> – which is currently working on addressing them):

- Only replicative malware is listed
- WildCore represents only a tiny proportion of all malware (even replicative malware, though it can be argued that it includes most of the viruses and worms that people are likely to consider most critical). Of course, this can be said of any small sample set, only more so.
- The WildList is always behind the curve: the stringent requirements for validating samples before a variant is added would involve a significant time delay for even the best-resourced organization.
- Testers outside the charmed circle would no doubt also point to the difficulties of being accepted as a WildCore recipient: clearly, this reflects a perceived need to trust the competence and bona fides of a recipient, but it remains a bone of contention.

Nonetheless, WildList testing continues to be a significant component of some of the best current tests [43], probably for the following main reasons:

- WildCore samples can reasonably be assumed to be real (replicative) malware, not junk files
- The samples have already been validated and identified (though there's still a need for further validation – or at least replication – by the tester)
- The above factors minimize the risk of incorrect identification and false positives
- They provide a consistent baseline collection.

The fact that most mainstream vendors might be expected have access to such samples and detect them accordingly is often cited as proof of the inadequacy of the collection as a test criterion: however, the fact that there is often a wide discrepancy between products in a sound test context does suggest that there is something useful to learn from WildList testing.

Even where the WildList is not a suitable base for testing (tests of Trojans or perhaps some more proactive based testing), it is to be expected that the same stringent standards for sample validation should be followed

Spot the Fallacy

SC Magazine reported in September 2007 that “The long-awaited report...from the House of Lords (the United Kingdom’s highest parliamentary body)...[recommends]...increasing the liability of IT security vendors involved in security breaches...Both McAfee and Symantec pointed to the complexity of the IT industry and the potential for users to compromise otherwise secure products.” A spokesman for McAfee, was quoted as saying that “It would be very difficult to hold vendors responsible for breaches, it really comes down to how solutions are deployed. A security vendor supplies businesses with tools, but it is down to the business to use them correctly.”

None of these assertions is incorrect, or, we’re sure, intended to mislead. (for instance, McAfee does state in some advertising material that their software “does not guarantee protection against all possible threats.”) But it leaves the consumer with the already all-too-common idea that if they don’t mess about with their settings, they’ll be protected. And to most people, that means that all malware will be detected. Clearly, this doesn’t happen, and that’s one of the reasons people think the worst of us. Every false negative (let alone every high-profile false positive) is seen as a reprehensible failure to deliver a level of protection that known malware and even heuristic scanning cannot realistically deliver in today’s threat climate. Of course, ethical vendors have never promised 100% protection using predominantly signature-based (virus-specific and heuristic signatures, that is) scanners: what we’re looking at here is wishful thinking. What customers mostly want is automatic pseudo-exact identification of all threats that doesn’t, unlike generic solutions, require them to make any decisions.

We won’t, on this occasion, address the issue that if you actually tweak the characteristically conservative configuration of an out-of-the-box package, you can often enhance your security dramatically.

What does all this have to do with testing? Simply this: if we cannot rely on anti-malware to process and protect us from the whole range of incoming threats, and we can't identify and verify all the threats to be found somewhere in cyberspace at the time of testing, the best that we can hope for is a snapshot of the current threatscape at which we can point the products under test. So it is incumbent upon testers to work their hardest to ensure that the snapshot in question is as close as it can be to the real topology of that threatscape. A random image that highlights one or two rocks – or the bugs beneath them – is not going to reflect that topology well enough to function as a sound basis on which to draw sound conclusions.

How Practical is DIY Testing?

Good detection testing requires, among other things, meticulous procedures, large and carefully maintained collections of both malware and clean files (for false positive testing) without spurious samples. However, this is time- and resource-intensive, technically demanding, and difficult to achieve without industry cooperation where samples are normally only shared between trusted individuals. (The expense entailed means that detailed results may not be readily available to non-subscribers.)

In a validated test set spurious samples are carefully weeded out, and supplementary techniques such as large, carefully vetted false positive (FP) test sets are used. (Such false positive sets should be carefully sorted. Consider a self extracting archive (SFX) or packed file: this is not the same as a Portable Executable (PE) file, as it contains multiple objects. Some scanners will scan all the contained objects, others will simply scan the container, the contention being if a malicious file is run then it will be caught at that point. These types of files should be carefully sorted so that consistent and 'apples to apples' false positive testing is achieved. Then, when actually testing, procedures are scrupulously planned, documented, and followed. Sometimes the service is funded by vendors whose products are under test, sometimes to the disadvantage (intentionally or otherwise) of small vendors and community projects, however, the full results should be reproducible and properly recorded, and all data from the test retained and archived in case of later querying of the method.

Many of the recommendations reported in security circles are informal, based on the apparently trouble-free performance of a live installation. Variables such as configuration and the quality of any test set used have to be taken on trust, in the absence of a clearly reported test methodology.

Here, at least, we can raise one moderately hearty cheer for Dirk Morris of Untangled [1],

who did at least make some attempt to explain his methodology, even if he failed to answer direct questions. The hard (if not unfair) corollary to that is that his being commendably (if naively) honest about his methods and his sample set made it easier to criticize the obvious holes in his methodology. The strategy of openness has certainly helped in many cases though, as testing bodies as venerable and well respected as Virus Bulletin have occasionally printed retractions or changes where genuine problems have been found, and in some cases have altered their methodology for the future.

Viruses Aren't the Whole Problem

Not that they aren't a bad thing when they happen to you, but they're only a (shrinking) percentage of the total malware problem. So the selection of an anti-malware solution is affected by a whole range of subsidiary detection issues; that is, how effectively it detects non-viral malware (as opposed to legitimate objects), and increasingly some categories of 'greyware' such as remote administration utilities, not to mention a whole range of other issues such as usability.

While we focus here on performance (and particularly detection and to some extent disinfection, though this is less often addressed in formal testing – perhaps this has something to do with the steeply escalated resource implications), we have defined a number of core evaluation issues [37] that aren't all addressed here (we realize that the ordering could be contentious: it's always going to be a compromise between “best security practice” and “what the CEO demands”):

- Cost
- Performance
- Ease of use
- Functional range
- Configurability
- Support functions

We won't discuss individual test methodologies here in detail, but some of the most common test types are [38]:

- Proactive (retrospective or frozen) testing of heuristic capabilities
- Time to Update testing (sometimes called response testing)
- In the Wild testing

- Zoo testing
- Non-replicative malware
- Realtime
- On-demand testing
- False positive testing

In fact, performance testing can, potentially, entail a huge range of detection (and in some cases disinfection) targets [37], such as:

- ItW
- zoo viruses
- New, high-profile threats
- Unknown threats (heuristic performance)
- Range of threats detected
- System viruses (hardware/firmware/OS-specific)
- Parasitic viruses
- Macro and script viruses
- Multipartite/multipolar threats
- Mail-specific malware (mailers, mass mailers, and so on)
- Web-hosted/borne malware
- Network worms, worm/virus hybrids
- Trojans (destructive, password stealers, backdoors, banking Trojans, and so on)
- Bots
- Latent viruses
- Cross-platform viruses/heterogeneous transmission issues
- Generators
- Intendeds, corruptions, other non-viables
- Jokes
- Spyware
- Adware
- Lots of other stuff we couldn't think of off the tops of our heads.

What do you Need for Sound Testing?

- Appropriate and correctly applied methodology
- Reproducibility
- Independently verifiable results and methods
- Validated and realistic sample sets
- Adherence to safe and ethical practices in handling and testing samples
- Understanding of what the technology you're testing is (and what it's not)

Most amateur testers (many of whom consider themselves to be security professionals) fail to understand the need for issues such as sound testing methodologies (separation of targets, consistent configuration) and the need to understand anti-malware technologies (in particular, detection techniques) – hence, the many reviews that confuse testing for exact or near exact identification, heuristics, and more generic technologies such as whitelisting.

Much testing is based on attempting to “trick” scanners [38], for instance by running them against inappropriately modified or contextualized samples. We regard this as ethically suspect, not least because of the way it can mislead an audience. False positive testing, for instance, requires an appropriate “wild” FP test set (that is, test objects that would really be found on real computers, not bespoke trick samples). ‘Grey’, unusual or very strange and unlikely files will tend to penalize heuristic based products that flag objects that don’t look “normal.”

Poor sample sets containing garbage files and junk simply confuse the issue: the more junk is added to test sets, the more irrelevant objects scanners are required to detect simply to stay in the game.

“Time to Update” testing tends to introduce a statistical bias, where means of more successful products are calculated over less samples [37; 44] and is less suitable for comparative testing than proactive testing. Concentrating on speed of update is surely sending the wrong message to the consumers, giving them the false impression that buying a product that releases a lot of updates very quickly is going to protect them better.

Retrospective (proactive) testing, where updates are frozen for a pre-selected time period, is, properly administered, a better test of heuristics than such strategies as kit viruses, bespoke samples and so on. However, it’s not an easy technique.

Who are the trusted testers?

Tests by certain organizations along fairly uniform lines are generally considered valid by the (notoriously conservative) AV community, and derive from the need to implement a stringent and impartial baseline set of methodologies. This requires considerable time and expertise, and that expense is one of the reasons that many first-class tests (let alone their complete methodology) are not made freely available (that is, are only available, at least in the short term, on a subscription basis).

However irritated vendors and researchers may be by the need for and (especially) implementation of (most) comparative testing, they will usually reluctantly admit the need for customers to have some comparative information. 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>)

By comparison, reviews in general computing magazines and other non-specialist resources are a hit and miss way of evaluating the effectiveness of anti-malware products. Few non-specialist journalists are technically adept in the field (in terms of understanding both the attack technologies and the countermeasures), or the booby traps in detection testing. Testing methodology is rarely described, especially if detection testing is outsourced. (Outsourcing can be a very responsible way of handling detection testing, but only if the testing organization is competent. [2])

Reviews may focus on more subjective aspects such as usability, impact on system resources, perceived speed, and so on. This approach can be problematical [45]: the issues that concern systems administrators or security managers and directors may not be obvious to a non-practitioner, or anyone thinking in terms of single machines in the home or small office. Nonetheless, these are issues that are:

- more susceptible to non-expert testing
- less liable to lead to serious consequences when performed incompetently

Sadly, there are still such tests where the editor's choice is suspected of being unduly

influenced by the advertiser roster. In an article by Dr. Alan Solomon [24], ways are discussed in which comparative reviews have inadvertently or deliberately reflected the bias or commercial agenda of the tester. Alas, despite the age of the article, the general principles and some of the specifics are as relevant today as they were in the 1990s.

Specialist magazines such as Virus Bulletin and reputable testing organizations such as the testing facilities at Magdeburg tend to offer more reliable information, but these facilities tend to focus mostly on detection (including variations such as false positive testing), rather than a full range of features. Such issues as usability are very important, and it's an area professional testers rarely address in detail, not least because detection is actually conceptually and practically easier to test than usability, if you have the resources and knowledge to do it properly.

It's Not My Default

We'd like to address one more point that is often used to justify methodologies where no attempt is made to level the playing field: in effect, all detection testing in such tests is based on default, out-of-the-box configuration.

An end user is not necessarily going to use a configuration that will catch all samples, and most default configurations prioritize speed over deep scanning. So there is indeed a distinction between default detection and overall detection capability (there's an issue here with setting heuristic levels, too.) If a product is capable of detecting 100,000 strains, but not out of the box, and the vendor makes it difficult for the customer to use it to its best advantage, that's a usability and configuration testing issue: it's not pure detection testing. However, there's certainly an argument for testing default detection (though in that case you should, perhaps, also test with maximum security settings. It's harder to test defaults, though, because the number of variables makes it difficult to maintain parity between tested configurations: otherwise, you're not only testing performance, but configurational philosophy. That doesn't mean, though, that it's not worth trying to do.

However, there are many interim tests that are worth considering (different levels of heuristics, on-demand versus on-access, and so on), as well as strictly limited tests such as sensitivity to test files (especially the EICAR test file), macro disinfection, and so on.

Conclusion

You don't, perhaps, have to be an AV researcher to test AV, though testing is a very specific sub-field of AV research. Some of the rules for testing AV at consumer level are the same as for other types of product, but it's more complicated because everyone has an idea of how to use, say, a word processor, and what to expect from it, but most people have a very distorted idea of what AV does and how. (We happen to believe that the AV research community has to some extent brought that unfortunate state of affairs about themselves, but that's yet another debate.)

We don't expect you to take everything we (or the anti-malware industry in general) say as written on tablets of stone. We are all for healthy skepticism. What we do find unhealthy is the tendency to assume that the AV industry is one big fraud, and that anything that doesn't come from that sector is therefore true.

There are many problems with making it easier for people outside the industry to test, and we don't have the answer to all of them. Supplying samples to people you don't/can't trust is an obvious problem area. There are some partial solutions to this: outsourcing the detection part of a comparative to a competent agency, or making use of facilities made available by such an agency (or even by an anti-malware vendor) under tightly controlled conditions (so that samples don't "escape", for example) are possibilities. However, making people more aware of good and bad practice, teaching them what they can and can't effectively do, empowering them to run their own meaningful tests and assess the tests of others is, we hope and believe, a practical step towards better understanding and practice.

We do think that the AV industry has a responsibility to address the whole issue better than it does at present, and in our own small way, hope to address that issue at book length in the near future.

References

- [1] <http://blog.untangle.com/?p=95>; <http://blog.untangle.com/?p=96> (2007)
- [2] David Harley: "AV Testing SANS Virus Creation," Virus Bulletin, October 2006.
- [3] Igor Muttik: "Shall we all write viruses to find the best antivirus?" at <http://www.avertlabs.com/research/blog/?p=71> (2006)
- [4] Igor Muttik: "A Tangled Web", in "AVIEN Malware Defense Guide for the Enterprise," Syngress 2007
- [5] Alex Eckleberry: "More Testing Silliness" at <http://sunbeltblog.blogspot.com/2006/08/more-testing-silliness.html>
- [6] David Harley: "Untangling the Wheat from the Chaff in Comparative Anti-Virus Reviews" at http://www.smallblue-greenworld.co.uk/AV_comparative_guide.pdf
- [7] David Harley: "Insider's Guide to Comparative Anti-Virus Reviews" at http://blogs.technet.com/industry_insiders/pages/insider-s-guide-to-comparative-anti-virus-reviews.aspx (2007).
- [8] Vesselin Bontchev: "About Anti-Virus Testing" at <http://www.f-prot.com/workshop2007/presentations.html>
- [9] David Harley: "I'm OK, You're Not OK" in "Virus Bulletin", November 2006 (see <http://www.virusbtn.com/virusbulletin/archive/2006/11/vb200611-OK.dkb>).
- [10] David Harley et al: "Customer Power & AV Wannabes" in "AVIEN Malware Defense Guide for the Enterprise", Syngress 2007.
- [11] David Harley: "Fact, Fiction and Managed Anti-Malware Services" in "Proceedings of the 13th Virus Bulletin International Conference" (2003)
- [12] Justin Kruger and David Dunning "Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments" in "Journal of Personality and Social Psychology" Volume 77 No. 6 (1999) pages 121-1134
- [13] David Harley, Jimmy Kuo: "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/>
- [14] Martin Overton: "FAT32 – New Problems for Anti-Virus, or Viruses" in "Proceedings of Virus Bulletin Conference, October 1997."
- [15] Andrew Hayter: "Nature of Anti-Malware Testing and Certification Programs Life and times of testing Anti-virus Products" for AVAR 2007.
- [16] Maik Morgenstern & Andreas Marx, AV-Test.org: "Testing of 'Dynamic Detection'" for AVAR 2007.

- [17] Vesselin Bontchev: "Maintaining a Malware Collection" at <http://www.f-prot.com/workshop2007/presentations.html>
- [18] Vesselin Bontchev: "Analysis and Maintenance of a Clean Virus Library", at <http://www.people.frisk-software.com/~bontchev/papers/virlib.html>
- [19] J.B. Rhine, "New Frontiers of the Mind", Farrar and Rhinehart 1937
- [20] <http://www.informatik.uni-hamburg.de/AGN/vtc/>
- [21] <http://www.av-test.org/>
- [22] Henk K. Diemer, David Harley: "Perilous Outsorcery" in "AVIEN Malware Defense Guide for the Enterprise", Syngress 2007
- [23] Tim Wilson: "Antivirus Tools Underperform When Tested in LinuxWorld 'Fight Club'" http://www.darkreading.com/document.asp?doc_id=131246&WT.svl=news1_5
- [24] 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
- [25] Pamela Kane: "The Dangers of Experts" in "PC Security and Virus Protection Handbook", M&T Books, 1994
- [26] Andreas Marx and Frank Dessmann: "The WildList is Dead, Long Live the WildList" in "Proceedings of the 17th Virus Bulletin Conference" 2007
- [27] Randy Abrams: "AV Industry Comments on Anti-Malware Testing" in Virus Bulletin, June 2007
- [28] Mary Landesman: "The Wild WildList" in Virus Bulletin, July 2007
- [29] Rob Rosenberger: "False Authority Syndrome", at <http://www.cknow.com/vtutor/FalseAuthoritySyndrome.html>
- [30] <http://www.sans.org/newsletters/newsbites/newsbites.php?vol=8&issue=65>
- [31] Sarah Gordon & Richard Ford: "When Worlds Collide: Information Sharing for the Security and Anti-Virus Communities", in "Proceedings of the Virus Bulletin Conference" 1999.
- [32] Responses to an article by John Leyden http://www.theregister.co.uk/2007/09/28/nsa_hacker_malware_defense_project/comments/#c_68630
- [33] Bruce Schneier: "Teaching Viruses" at <http://www.schneier.com/crypto-gram-0706.html#5>
- [34] John Aycock & Alana Maurushat: "Future Threats" in "Proceedings of the 17th Virus Bulletin International Conference" 2007.
- [35] Hiep Dang: "What a Tangled Web" at <http://www.avertlabs.com/research/blog/index.php/2007/08/12/what-a-tangled-web/>

- [36] Tim Wilson: "Antivirus Tools Underperform When Tested in LinuxWorld 'Fight Club'" http://www.darkreading.com/document.asp?doc_id=131246&WT.svl=news1_5
- [37] David Harley & Robert Slade: "Product Evaluation and Testing" in "Viruses Revealed" (Harley, Slade, Gattiker), Osborne 2001.
- [38] David Harley & Andrew Lee: "Antimalware Evaluation and Testing", in "AVIEN Malware Defense Guide for the Enterprise" Syngress 2007;
- [39] Dirk Morris: "Selling Dead Donkeys" at <http://blog.untangle.com/?p=20>
- [40] Joe Wells: "Pragmatic Anti-Virus Testing" in Virus Bulletin, September 2001. Also available at <http://www.sunbelt-software.com/ihs/alex/Pragmaticantivirustesting.pdf>.
- [41] Sarah Gordon: "What is Wild?" at <http://csrc.nist.gov/nissc/1997/proceedings/177.pdf>
- [42] Vesselin Bontchev: "About Anti-Virus Testing" at <http://www.f-prot.com/workshop2007/presentations.html>
- [43] <http://www.virusbtn.com/vb100/about/100procedure.xml>; [http://www.icsalabs.com/icsa/topic.php?tid=453f\\$2571e0c1-26134a78\\$461e-02308865](http://www.icsalabs.com/icsa/topic.php?tid=453f$2571e0c1-26134a78$461e-02308865)
- [44] Andrew Lee: "Testing Heuristics" at <http://www.f-prot.com/workshop2007/presentations.html>
- [45] Igor Muttik: "Comparing the Comparatives" at http://www.mcafee.com/us/local_content/white_papers/threat_center/wp_imuttik_vb_conf_2001.pdf

Additional 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.
- http://www.av-comparatives.org/seiten/ergebnisse_2007_02.php
- <http://www.av-comparatives.org/seiten/ergebnisse/2ndgrouptest.pdf>
- Randy Abrams: “AV Industry Comments on Anti-Malware Testing” in Virus Bulletin, June 2007.
- Igor Muttik: “Antivirus Testing Workshop in Reykjavik” at <http://www.avertlabs.com/research/blog/index.php/2007/05/29/antivirus-testing-workshop-in-reykjavik/>
- Richard Ford, Attila Ondi: “Testing Times Ahead?”, Virus Bulletin, April 2007
- 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.



www.eset.com

610 West Ash Street • Suite 1900 • San Diego • California 92101 • U.S.A.
866-343-ESET

© 2008 ESET, LLC. All rights reserved. Trademarks used herein are trademarks or registered trademarks of ESET, LLC.
All other names and brands are registered trademarks of their respective companies.