available at www.sciencedirect.com

ScienceDirect

www.compseconline.com/publications/prodinf.htm

Information Security Technical Report

Making sense of anti-malware comparative testing

*David Harley**

Director of Malware Intelligence, ESET, 610 West Ash Street, Suite 1900, San Diego, CA 92101, USA

Keywords:
Anti-malware comparative testing
Detection testing
Dynamic testing
Static testing

ABSTRACT

If there is a single problem illustrating the gulf between the anti-malware industry and the rest of the online world, it revolves around the difficulties and misunderstandings that plague product testing and evaluation. This article considers these issues and the initiatives taken by the anti-malware and testing sectors to resolve some of them.

1. Introduction

The testing world is, from a certain point of view, divided into a number of groups.

- There are those who consider that malware evaluation is just about detection testing, and that it is easy for someone with a quantity of samples to test the ability of a range of products to detect those samples.
- There are those who have a vested interest in disproving the bona fides of either the entire anti-malware industry or of products that compete with those in which they have an interest.
- There is a huge audience of consumers influenced by bold statements about product competence ranking that may indeed reflect comparative competence, but may also (or alternatively) reflect the testing practices and prejudices of the tester. (We are, of course, talking here of corporate evaluation and procurement teams, not just individual users.) Those who go beyond reading (or performing, or commissioning) a single review to basing their procurement decisions on multiple reviews and tests face a daunting game of ''pin the tail on the donkey'': either hoping to find the One True Review, or trying to synthesize an adequate evaluation from a jumble of conflicting reviews of varying conviction and competence.
- There are providers of security products and services who are perpetually on the defensive, as their marketability fluctuates according to their perceived importance in highly visible, highly variable tests.
- And there are the mainstream testers. In general, these are to some degree allied to the security industry, because it is very difficult to test security software without those alliances to enable the sharing of samples and information. However, they also (usually) stand to some extent outside the vendor community, due to the need to maintain integrity and independence.

For many years, the vendor community has been open to the accusation that while it was eager to protest at what it perceived as incompetent or intentionally biased testing, it was far less ready to provide help or guidance on what it considers to be ''good'' practice.

This paper will, therefore, examine the issues and problems that make testing malware far more difficult than is normally assumed, and evaluate some of the ongoing initiatives aimed at addressing those problems, including the initial work of the Anti-Malware Testing Standards Organization (AMTSO).

2. Testing and evaluation

In educationalist circles, it is sometimes said that ''examinations are intended to find out what you know, not what you don't know.'' Many anti-malware product tests (in the

* Tel.: +1 619 204 6461.
E-mail address: dharley@eset.com

1363-4127/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.istr.2009.03.002

broadest sense) derive from the opposite viewpoint: they are intended to ''trick'' the tested products into failing. It is sometimes argued that this is essentially invalid (Andrew Lee and David Harley, 2007): this may be an overstatement, but such an approach can be seriously misleading if it doesn't reflect current (or likely future) real world malware technology and practice. However, the problems are deeper than this.

We talk about testing in the anti-malware field as if it were much simpler than is actually the case. Firstly, it isn't only about detection testing. Secondly, detection testing is far more complex than most people think it is, including many people who regularly publish test reports (whether it's to the world at large, or to subscribers to a specific publication).

We can't possibly cover all aspects of this complex topic in detail here, so I'll restrict myself to a general overview of the topic as a whole, and then focus on detection testing, which is probably what most readers are really interested in, rightly or wrongly.

2.1. *Evaluation*

Detection is important, of course: it is, after all, one of the primary functions of a malware-specific product or service, which can be categorized as follows.

- Malware detection.
- Prevention of infection or compromise by malware.
- Remediation in the event of a compromise.

However, stellar detection performance is of little use if the product doesn't meet the needs of the customer in other ways, such as:

- usability (both at the systems administration level and at the end-user level),
- configurability,
- adaptability, especially when there's a drastic change in the threat landscape (such as a significant new threat vector),
- responsiveness to changes in the organizational environment or infrastructure (network changes, hardware and software upgrades, changes in policy or strategy framework), and
- responsiveness or adaptability to business needs: for instance, the impact of security software on host hardware and other applications, and therefore on day-to-day business processes.

Some reviewers try to take some of these factors into account. However, organizations that try to take an approach to procurement that balances technical, operational and business requirements rarely find a comparative review that tries to include the same areas of interest, yet is neither too subjective nor too focused on the assumption of a one-size-fits-all view that applies to all types and size of business.

As there is little in the way of formal objective testing that addresses these issues, it's not surprising that reviewers and their audiences tend to fall back on detection testing as the main criterion for comparative evaluation. It is, after all, a core function, and offers a deceptively simple, apparently objective metric.

2.2. *Objective performance data*

Before we consider how objective detection testing really is, let's consider other possible ways to harvest ''hard'' objective data. Perhaps the most obvious possible alternative metrics centre round the impact of a tested product on system performance.

There are, in fact, a number of basic functionalities that can be tested fairly objectively. For instance:

- On-demand scanning speed on a clean machine. (That is, passive scanning of a folder, system, or individual files without actually opening files for execution.) This has the advantage that it doesn't even require a malware collection. However, it has to be done properly, which needs reasonable understanding of the technology. For instance, if product X scans all files by default and product Y scans only files with selected file extensions, it is misleading to present a test that doesn't take this difference into an account as an objective speed test. This is because it doesn't take into account the likely differences in detection performance between the two products. Even though this isn't a detection test, detection is an important consideration, since product X may be penalized for scanning speed performance, without reference to the fact that in some scenarios product Y may miss malware that X won't.

On-access scanning (scanning each file as it's opened for reading or execution) needs a little more setting up, not only to measure a product's performance as it goes through a significant volume of test files, but also to ensure a level playing field between products. While it's generally assumed that on-access scanners scan files as they are accessed (as the term suggests), some products actually try to maintain a speed advantage by scanning only if the file has a filename extension recognized by the scanner as denoting an executable *unless* the file is accessed for execution – that is, just reading the file won't trigger a scan. While it's sometimes argued that it isn't necessary to scan a file until it is executed, this clearly isn't an approach taken by all vendors or the approach expected by all customers.

- Scanning speed on an infected machine is even more of a can of worms (not to mention Trojans, viruses, bots, adware,…). Many of the problems described below that apply to out-and-out detection testing might also apply here (for instance, proper validation and classification of samples), but the issue is complicated further in that the sample set must consist of samples that are known to be detected by all products (otherwise it's a speed-and-detection test, not a speed test). In addition, considerable care has to be taken to ensure that the configuration of all scanners is equivalent, so as to lessen the risk of bias. Commonly, tests are carried out using ''out-of-the-box'' (default) configuration. However, anti-malware scanning can be a trade-off, not only between speed and security, but also between speed and other factors such as