AV-TEST
Tests of Anti-Virus-Software independent • qualified • fast

Test Strategies & Common Mistakes

International Antivirus Testing Workshop 2007

Andreas Marx, MSc.

CEO, AV-Test GmbH

http://www.av-test.org

Table of Content

- About AV-Test.org
- Tests of Security Software
  - Prerequisites for Evaluation and Testing
  - Evaluation and Testing the Programs
  - Documenting and Editing the Test Results
  - Current project: cross-reference lists (XREF)
- Questions & Answers

AV-TEST
Tests of Anti-Virus-Software independent • qualified • fast

About AV-Test.org

- Founded as company in 1996 and 2004 (GmbH)
- About 15 full-time employees and freelancers
- Working for 45 computer magazines world-wide
- Working for many companies as consultants
- People are involved in AV programming, testing and research since 1991 (as University project)
- Our test lab is equipped with more than 100 PCs
- Large collection of malware and clean files (60 TB)
- Over 2,000 product tests per year

Prerequisites for Evaluation and Testing

- Tester has to be independent from the companies he wants to review (sponsored reviews needs clearify the fact that the test was paid by a specific organization)
- The tester needs to know what he wants to do
  → Detailed test plan is important
- A secure, separated network (which is not connected to any external networks like the internet) is required as test environment → Dedicated test network
- Detailed knowledge about malware is required
  → “Reverse Engineering Skills”
- Every malware file needs to be checked (e.g. replicated and analyzed) if it’s working properly or possibly corrupted before it’s included in any collection used for tests!
- Reminder: Malware is not a toy!

Evaluation and Testing the Programs (I)

- The ‘classic’ criteria: Detection rates
  - Virus scanner should detect viruses...
  - Easiest method: One simply scans a formerly created malware database (log files? how to count? crashes?)
  - Differentiation possible between WildList and Zoo tests (old vs. new files?), intentionally malicious software (e.g. viruses, worms, bots) and potentially unwanted software (e.g. dialer, jokes, ad-/spyware) etc.
  - Often, only the on-demand scanner (because it’s so easy to do?), but not the on-access guard is reviewed
  - Results in many cases meaningless (99.5 vs. 99.7%)
    - Exact CRC/MD5 detections of files by many AV products
  - Malware databases are often badly maintained

Evaluation and Testing the Programs (II)

- The counterpart: False positive tests
  - Less frequently tested, even if scanners with lots of false positives (and possible high malware and heuristic detection rates) can't be used on any production PC
  - A preferably big database of known to be good / harmless files is required (at least, some 100,000)
  - Sources: CDs and DVDs, ftp and http server mirrors
  - Should be sorted after importance / priority (e.g. severity of a false positive: Windows system file vs. Office program vs. 'any' unknown 3rd party tool)
  - Procedures: Scan a system with a high number of applications installed on it vs. scan installer files 'as is'

Evaluation and Testing the Programs (III)

- Today, cleaning is getting more important:
  - Never-ending and increasing malware stream
    - A high number of PCs will get infected sooner or later
  - Malware is using advanced self-protection techniques (including rootkits) which are working better than similar functions implemented in malware scanners
  - Procedure: Infect a system and test the cleaning functions (the scanner might not detect all malware-related pieces, but it should clean everything!)
  - Important: Are all files and the Windows Registry treated properly? Are all programs still working? (Some less important traces might be left behind, e.g. skin files)
  - Very complex and time-consuming test

Evaluation and Testing the Programs (IV)

- Even more important: Prevention
  - What kind of techniques are offered by the products to detect (and prevent) the infection by unknown malware?
  - Keywords: Application Control Mechanisms, Host-based Intrusion Detection & Prevention Solutions (HIDS/HIPS)
  - Procedure: Start a malware and see what will happen
  - Important: The test environment must look very real, simulated internet connection, no virtual machines
  - Compare the number of warning messages during normal operation (including patches which are installed by Windows Update) vs. during malware execution
  - What kind of critical actions are blocked or not?
  - Can malware changes be undone (if so, how well?)

Evaluation and Testing the Programs (V)

- Testing (Outbreak) Response Times
  - Question: At which time was my PC protected?
  - Create an archive with all ever-released AV updates (e.g. signatures, engine and program files)
  - Use a (scripted) multi-scanner system, plus some manual tests
  - Test of all archived updates against the different scanner versions in a given period of time (start date, end date?)
  - Look for heuristic and proactive detections (retrospective tests), reaction times, plus detection and name changes

→ Future development: Application Lifecycle Testing

- Not only a single update is tested, but all available ones
- How did the scanner perform over a period of time in case of reliability of detection, avoiding false positives etc.?
- We want to show how the products are performing in “real-life”

Documenting and Editing the Test Results

- Representation of the results
  - Write what was tested and how so a third party can understand it
    - Tell, what's important and what's not so essential!
  - Summarize all results into manageable tables
    - Not all data will fit into tables
    - Additional comments are essential
  - Give the tested developers some time for proofreading of results and verifying the samples used for the test
    - Remove samples from the test which are questionable or not malicious
  - Publication in readable form
    - Use a clear document style and structure with easily readable fonts
    - HTML pages or PDF files are "universal"
- After publication…
  - Keep contacts to the developers
  - Keep on discussion about current and future test strategies

Creation of cross-reference lists of malware names (code name: XREF) and known bad files which are unsuitable for testing

File Name	AVG	AntiVir	BitDefender
MYTBAB.EXE	I-Worm/Mytob.Z	Worm/Mytob.AB	Win32.Worm.Mytob.S
MYTBAE.EXE	I-Worm/Mytob.BB	Worm/Mytob.BM	Win32.Worm.Mytob.FE
MYTBAH.EXE	I-Worm/Mytob.AE	Worm/Mytob.AH	Win32.Worm.Mytob.X
MYTBAL.EXE	I-Worm/Mytob.AL	Worm/Mytob.BF	Win32.Worm.Mytob.AC
MYTBAM.EXE	I-Worm/Mytob.AC	Worm/Mytob.BF	Win32.Worm.Mytob.V
MYTBAN.EXE	I-Worm/Mytob.AM	Worm/Mytob.BF	Win32.Worm.Mytob.AN
MYTBAR.EXE	I-Worm/Mytob.AP	Worm/Mytob.BA	Win32.Worm.Mytob.AA
MYTBAU.EXE	I-Worm/Mytob.AK	Worm/Mytob.AU	Win32.Worm.Mytob.Y
MYTBAW.EXE	I-Worm/Mytob.AQ	Worm/Mytob.AW	Win32.Worm.Mytob.AB
MYTBAX.EXE	I-Worm/Mytob.AR	Worm/Mytob.AX	Win32.Worm.Mytob.AA
MYTBBB.EXE	I-Worm/Mytob.AU	Worm/Mytob.BG	Win32.Worm.Mytob.AE
MYTBBD.EXE	I-Worm/Mytob.AS	Worm/Mytob.BE	Win32.Worm.Mytob.AB
MYTBBI.EXE	I-Worm/Mytob.FW	Worm/Mytob.ED.1	Win32.Worm.Mytob.BC
MYTBBJ.EXE	I-Worm/Mytob.AI	Worm/Mytob.AS	Win32.Worm.Mytob.T
MYTBBL.EXE	I-Worm/Mytob.BF	Worm/Mytob.BR	Win32.Worm.Mytob.M
MYTBBM.EXE	I-Worm/Mytob.BO	Worm/Mytob.BW	Win32.Worm.Mytob.AF

Questions & Answers

- ???

- Note: Many testing papers can be found at: http://www.av-test.org → Publications → Papers