FEATURE 2

ANTI-VIRUS VS ANTI-VIRUS: FALSE POSITIVES IN AV SOFTWARE

Andreas Marx, AV-Test.org
University of Magdeburg, Germany

I am sure that almost everyone working in the security business knows that it is not a good idea to install two (or more) anti-virus programs at the same time on the same computer – simply because each on-access guard wants to kill the other one … but that is not the only reason.

While performing a comparative review of anti-virus tools for a German magazine a few months ago, we discovered another interesting side effect of trying to use 'too much AV power' at the same time: false positives.

We found that *H+BEDV*'s *AntiVir* flagged the pavdll.dll file of *Panda Antivirus* as being infected by the W32/Kenston-1895.X virus.

Similarly, *Computer Associates*' *InoculateIT* (with the *CA* engine enabled) found Win32/Funlove.4099 in the file pavcl.exe (*Panda Antivirus* command-line scanner). Meanwhile, *DialogueScience*'s *Dr.Web* found Win32.Benny.6382 in the same file. And finally, *F-Secure*'s product identified a new variant of the Trivial virus inside one of the documentation files of *Kaspersky Anti-Virus*. What a mess!

THE REASON?

After a brief check of the files pavdll.dll and pavcl.exe an explanation for the false positives was identified: *Panda Software* does not fully encrypt its virus signatures and stores a lot of them in plain text, which is as they appear inside infected files. That was the reason why the signature scanning algorithms of *AntiVir*, *InoculateIT* and *Dr. Web* had flagged the files as infected.

Once we had identified the cause of the problem, we asked *Panda Software* if they would agree to fix it, by encrypting all of the virus signatures. However, the response from *Panda* was that, currently, this is not possible and that this is not their problem, because it is easy for an anti-virus program to see that these signatures are not a sign of an

infection. If other anti-virus companies would improve their tools to scan these files properly, the problem would not occur, they said.

We asked the other three anti-virus companies concerned for their opinions. Two of them told us that they could not do anything to avoid this problem and that the problem could only be fixed by *Panda Software* encrypting their signatures.

CA was the only company to try to fix the problem by altering its scanning engine. This was because *CA* had also received a number of customer complaints about false positives in the – guess what – only partly encrypted signature file pav.sig.

For a limited time, the string 'Signature file system (c) Panda Software' (the header of the pav.sig file) was visible in the *CA* engine (avh32dll.dll) to ensure that the scanner would skip this file during a scan and avoid generating this false positive.

The false positive generated by *F-Secure Anti-Virus* (which has two main scan engines, *F-Prot* and *Kaspersky*) was caused by the presence of the eicar.com file inside a short part of the *Kaspersky Anti-Virus* documentation. The *F-Prot* engine found this suspicious.

OTHER EXAMPLES

The examples described above are the result of just one test! In the past we have encountered several other problems like this, and in the majority of cases they were caused by plain, unencrypted signatures.

Examples include a routine in *AntiVir* which was written to clean systems infected with Win32/Qaz. In order to restore the registry, *AntiVir* stored the strings 'StartIE' and 'qazwsx.hsq' in plain text to delete keys created by this worm. This was enough for *Network Associates*' *VirusScan* to flag the anti-virus tool as being a possible new variant of the Win32/Qaz worm.

In this case, however, both companies fixed the problem in their next product releases: *AntiVir* encrypted the text strings and *Network Associates* extended the driver to check for more than just this signature in order to report a new variant of an existing virus.

APPORTIONING THE BLAME

But not every anti-virus company fixes problems like this silently, as illustrated by the following text which was linked from the *F-Prot* website for quite some time but has since been removed (source: http://www.f-prot.com/f-prot/news/noworm.html):

'**The RealTime Protector component is not a worm**

Mcafee's antivirus product, using definition files number 4199 , falsely detects the RealTime Protector component of F-Prot Antivirus as a new worm. This problem with the Mcafee product applies to machines running Windows NT, 2000, and XP with F-Prot Antivirus 3.12.

Needless to say the RealTime Protector component of F-Prot Antivirus is not a worm, neither a new nor an old one. The source of this problem lies solely with Mcafee's apparent lack of quality control.

Mcafee users encountering this false detection of the RealTime Protector component are encouraged to ignore it and upgrade their definition files when newer files become available from Mcafee Inc. when they have fixed this problem. Users of Mcafee can also upgrade to F-Prot Antivirus for Windows here to get a more secure and reliable antivirus protection.'

As I was writing this article, I received a question about *Kaspersky Labs*' clrav.com utility, which cleans PCs infected by worms such as Win32/Opaserv. The download of the utility was blocked by *NAI VirusScan*. The heuristic reported the following: '*Found virus or variant New Worm !!! Please send a copy of the file to Network Associates*'. So I did.

CONCLUSION

Anti-virus tools from one company often have problems co-existing with the tools from another, especially in the area of false positives. Some of these problems could easily be avoided – the developers would only need to store their virus signatures properly encrypted in all parts of the program, the engine and the virus definition files. Not only should the signatures be encrypted to avoid false positives, but also to provide a form of protection against virus writers (who, having access to the easily-visible signatures can create new variants using different patterns) as well as protecting the company's intellectual property.

A simple runtime-compression or encryption of the whole executable file is not a viable option, because many anti-virus tools are able to uncompress or decrypt such programs easily. Therefore they would still find the signatures that caused the false positive.

In addition, the detection routines of a number of anti-virus programs should be fine-tuned so that a single short signature found in a file does not result in a virus alert at all. Last but not least, it is important for anti-virus vendors to have a copy of all competitors' programs (including the most recent updates and special cleaning tools) in a false positive test set which should be scanned before releasing a new definition update.