FEATURE

OUTBREAK RESPONSE TIMES: PUTTING AV TO THE TEST

Andreas Marx
AV-Test.org, Germany

Most comparative reviews of anti-virus programs focus on the products' worm and virus detection rates. But an important aspect of anti-virus products is often overlooked: it is not merely 'malware detection' that is offered by the developer, but rather a service that promises to keep your PC virus-free. This service includes responding quickly in case of local or global malware outbreaks.

In May 2003 we set out to measure the reaction times of anti-virus companies in the case of new malware outbreaks. The results for the Win32/Fizzer.A (http://www.pcwelt.de/news/viren_bugs/31094/) and Win32/Bugbear.B (http://www.pcwelt.de/news/viren_bugs/31671/) outbreaks were reported in German computing magazine *PC-Welt*.

THE FIRST ATTEMPTS

At first, we waited until we heard about a new malware outbreak, then installed all AV products in our lab as quickly as possible and checked for product updates and detection of the malware. However, this process proved to be both rather hectic for the lab staff and fairly inaccurate.

Next we tried using *VMware GSX Server*, which allows a number of virtual PCs to be run at once (providing there is sufficient processor power and especially RAM). We were able to install all GUI anti-virus products on these virtual PCs (all running *Windows 98 SE* due to RAM limitations), but this system did not work very well. After 24 hours, the consoles of one or two products would usually have crashed, so we would have to restart the virtual PCs. Furthermore, it was not always easy to grab the downloaded and installed updates, and the *Windows* patch management did not work as it should. Of course, it was not only the host PC that needed to be updated with all available (security) patches, but the virtual PCs had to be updated, too, and restarted.

THE SOLUTION: SCRIPTS WITH WGET

Finally, we decided only to check for updates, download them and store them in an archive. We did not want to install and test them automatically, nor did we want to use the GUI or command-line products; we wanted to rely only on our own scripts.

The first prototype of a shell script running on *Debian Linux* with a CVS version of wget was ready at the end of October 2003. At this time, we implemented automatic checking of the updates of only about ten anti-virus products, but we encountered several problems straight away.

The script checks every five minutes for changes in the anti-virus update and program files on FTP and HTTP servers. As an example, the updates for *H+BEDV AntiVir* are always stored at http://www.antivir.de/dateien/antivir/fuse/fuse.zip. Using wget, we simply check the length of the file, its date and time stamp every five minutes. As soon as the file has changed we download the update, write information about the update to a log file and store a copy of this file in our archive. Unfortunately, fewer than half of the products are updated so simply.

Today, most products have incremental updates, but for testing purposes it is much easier to use the full definition files. This also makes the update download process much simpler, because there is no need to use complex scripts or additional programs to recreate the full definition files, only a simple wget.

Another issue is that some companies, for example *Symantec*, publish 'intelligent EXE updater files' almost daily (which are easy to monitor, because they are stored on a public FTP server), but updates for LiveUpdate (which is built into every *Symantec* tool) are published only once or twice a week. Therefore, we had to implement additional checks to make sure that we monitored the presence of new LiveUpdates as well.

TROUBLE MAKERS

Some companies, such as *Symantec* and *Computer Associates* use more than one FTP server to store update files. We found that these servers were not always synchronised, so it was possible that we would see different updates ('old' and 'new' ones) when connecting to different IP addresses. Unfortunately, work-arounds for this issue could lead to greater problems, so we were forced to live with these discrepancies (and sort them out later).

Sometimes we found that the date/time stamp of files had changed (for whatever reason) on the server, but the file itself remained the same length. This was particularly common with the regular definition files from *CA* and the beta definition files from *Panda* and *Symantec*. When this situation arose, we downloaded the file and made a check using 'cmp' (which is similar to file compare ('fc') on *Windows*-based systems) to determine whether the file had been changed (an MD5sum for both files would be an alternative way to check for differences in the files). If the files were different, the 'new' file was processed like a standard update, but if they were identical it was ignored.

A special situation was caused by *McAfee*'s beta definition files called DailyDats which are refreshed usually every hour and available as a ZIP file. The standard definitions like scan.dat are stored inside this file. Several times a day the time stamp of the files inside the ZIP will change, but the definitions remain unchanged – a result of the fact that they are freshly uploaded, even if the files themselves are unmodified. We did not want to store such files in our archives, therefore we implemented a quick check: if the size of the ZIP file was unchanged and only the time field of the included files had altered, we would ignore the update.

In the case of *Ikarus* or *Sophos* it was difficult to use wget to download all updates, due to the fact that the names of the files we needed to download were displayed inside an HTML file only and changed often. In this case, we used 'curl', combined with 'sed', which is a little more complex, but it worked well.

Not all servers have the same bandwidth. Some are quite slow, others are very fast. After observing that the script sometimes hung on servers for quite some time, we implemented a 'quick skip' in case a server was unreachable. Ten seconds did not work very well and 20 seconds was usually too long; we found that 15 seconds worked best as a time-out value.

Currently the system runs on a Pentium IV 2.4 GHz PC with 256 MB memory and 500 GB HDD space. This should be enough to store at least all the updates released over the course of one year – currently we are collecting about 500 to 750 MB updates every day. We do not have a backup system, but we are using two DSL lines from different providers.

COOPERATION WITH AV COMPANIES

We invited all the anti-virus companies we knew of to participate in this project (which is free of charge). We needed a login to password-protected websites with the virus definitions or program updates (which would not be shared with third parties, of course). Additionally, we needed licence keys or registrations for the programs so we could test them.

At the time of writing we check the updates of 20 anti-virus companies with 21 different engines and four beta definition files. Additionally, we check for updates of A2 (an anti-Trojan scanner developed by Andreas Haak) and *RAV* (*Reliable AV*), but since the *RAV* product is no longer sold we will no longer publish test results for this tool.

The invitation emails sent to several other anti-virus companies, including *Ahnlab* (*V3*), *Cybersoft* (*VFind*), *Eset* (*Nod32*), *Hauri* (*ViRobot*) and *Proland* (*Protector Plus*), went unanswered, but we hope to be able to welcome these companies to the list of participants in the near future.

MEASURED OUTBREAK: SOBER.C

Shortly before Christmas the Win32/Sober.C worm was discovered in Germany. Like Win32/Sober.A its distribution was mainly limited to German-speaking countries, where it was widespread – possibly due to the fact that it applies very good social engineering tricks and it uses the German language (http://www.pcwelt.de/news/viren_bugs/36527/). However, after this worm was discovered (at around 03:00 h CET on 20 December 2003), it rose quickly in the statistics of several email security providers. For example, in the *MessageLabs* virus statistics (http://www.messagelabs.com/viruseye/threats/) it jumped to the sixth position very quickly and (at the time of writing) remains high in the chart with more than 4000 copies stopped every day. The *Frisk AVES* homepage (http://aves.f-prot.com/) showed it at the number one position for several days – with about 2 per cent of all scanned mails infected by Sober.C.

We felt that this local outbreak was significant enough to test the response times of all anti-virus companies that were on our watch list (see Table 1). This time, *BitDefender*, *Kaspersky*, *F-Prot* and *F-Secure* were first to release updates, but there is no guarantee that they will win the race next time.

It came as a surprise that big companies like *CA* or the German company *G Data* (which relies on the *Kaspersky* and *BitDefender* engines) seemed to have missed this outbreak completely and provided signatures at a time when

Table 1. Response times of AV companies (CET) to the outbreak of W32/Sober.C (worm discovered 2003-12-20 at 03:00 h).

BitDefender	2003-12-20 at 13:20 h
Kaspersky	2003-12-20 at 14:45 h
F-Prot (Frisk)	2003-12-20 at 15:25 h
F-Secure	2003-12-20 at 15:45 h
Norman	2003-12-20 at 18:25 h
eSafe (Aladdin)	2003-12-20 at 18:35 h
Trend Micro	2003-12-20 at 19:50 h
AVG (Grisoft)	2003-12-20 at 20:15 h
AntiVir (H+BEDV)	2003-12-20 at 22:20 h
Symantec	2003-12-21 at 04:05 h
Avast! (Alwil)	2003-12-21 at 09:55 h
Sophos	2003-12-21 at 14:35 h
Panda AV	2003-12-21 at 17:05 h
McAfee/NAI	2003-12-22 at 04:10 h
Ikarus	2003-12-22 at 10:35 h
eTrust (CA)	2003-12-22 at 17:50 h
AVK (G Data)	2003-12-23 at 23:50 h

it was already much too late to prevent the spread of the worm. The *G Data* case is especially interesting: standard customers receive updates only once a week, but after a few discussions they change the update interval to twice a week.

It should be noted that we tested only the virus definitions which were available to all customers – we did not include beta definitions which had to be applied manually. For example, *McAfee* had DailyDats available which included a detection routine for Sober.C as well as extra.dats which were available by request only.

For us, the test process for the updates was very simple: we installed the anti-virus products and tested them against the updates we had saved in our archive. To make sure that the worm had not been detected generically or heuristically, we tested the products' detection using older definitions as well as using the most current updates. However, none of the products we tested was able to catch this worm without updates.

UPDATE RELEASE CYCLES

The archived updates we have collected could be used for a number of other tests. For example they could be used to measure the actual release cycle of updates. Many companies claim that they update their signatures daily or every few hours, but after sorting out all the definitions released over a three-month period, and after duplicate updates had been removed, the reality looked a little different (see Table 2).

Table 2. Standard regular update release intervals

Product	Number of updates per week
AntiVir (H+BEDV)	5 to 6
Avast! (Alwil)	2
AVG (Grisoft)	2
BitDefender	3 to 4
Command	2
Dr.Web	6
eSafe (Aladdin)	5
eTrust (CA)	4 to 5
F-Prot (Frisk)	4 to 5
F-Secure	6 to 7
Ikarus	4
Kaspersky	about 20
McAfee/NAI	1
Norman	2
Panda	7
Quickheal	4
Sophos	4 to 5
Symantec	1 to 2
Trend Micro	2 to 3
VirusBuster	4 to 5

It is good to know that most anti-virus companies update their scanners more or less on a daily basis. They act like real security service providers, protecting against new threats proactively. Regardless of whether a malware threat has the ability to spread widely, it will be stopped by an updated product, so the chances of the virus spreading are lowered significantly. Using current pattern-based anti-virus technology, this is the only opportunity we have to stop malware – especially mass-mailer worms – quickly. It is true that providing more regular updates will result in higher costs for testing and QA, but that is what today's market expects and wants – and it is what the customers are paying for.

As an addendum to Table 2, it should be noted that *Network Associates* (*McAfee*) plans to release daily DAT updates starting early in the second quarter of 2004. Let's hope that other companies follow suit soon, because update releases only once or twice a week are simply too infrequent today.

We also have access to beta virus definitions from four anti-virus companies and these are often updated at least every few hours (see Table 3). However, according to the anti-virus companies these updates are usually only 'minimally tested' and could cause false positives or non-detections for existing viruses, so these patterns should be used only in emergencies.

Table 3. Beta update release intervals

Product	Number of updates per day
McAfee/NAI	5 to 12
Panda	40 to 50
Symantec	14 to 18
Trend Micro	6 to 7

It should be noted that there might be no correlation between the update frequency of (beta) definitions and outbreak response times. For example, *Panda* released 40 to 50 beta updates a day, yet it took more than 38 hours for an update with Sober.C detection routines to be made available. Let's hope that we will see more of a correlation in the future.

CONCLUSION

We hope to start a new interest in 'real-world' anti-virus tests. As well as testing outbreak response times, this project enables us to test the heuristics of products using retrospective test methodologies, count the number of updates released and we are even able to test the quality of these updates without any time pressure, because they are collected automatically. At a later stage we hope to make all the information available on a webpage which will be updated at regular (five-minute) intervals so that anyone can check the current update status of their anti-virus products.