# THE WILDLIST IS DEAD, LONG LIVE THE WILDLIST!

*Andreas Marx, Frank Dessmann*
AV-Test GmbH, Klewitzstr. 6, 39112 Magdeburg, Germany

Tel +49 391 6075460
Email {amarx, fdessmann}@av-test.de

## ABSTRACT

For a very long time, the WildList was the accepted standard for all kinds of anti-malware software test. However, today's real challenges – like targeted attacks and zero-day exploits, as well as adware and spyware – are not covered by the WildList. Traditionally, the WildList has only focused on self-replicating malware such as viruses and worms, but in today's world these malware types have almost died out and been replaced by Trojan horses with keyloggers and options to steal PIN and TAN codes for online banking. (The malware world has gone commercial and some of the bad guys are making more money than traditional AV companies!) Besides this, the WildList is usually published two to three months after the reporting period, so it is outdated when released.

This paper will focus on the current problems with the WildList and suggest methods to increase its usefulness again – to ensure not only that all of today's malware types are covered, but also that the WildList will always be current when published on a more regular basis. This includes an analysis of all required processes, better reporting methods and automation techniques which must be used to avoid delays in publication.

## INTRODUCTION

Several years ago, Vesselin Bontchev gave a presentation entitled 'The WildList – still useful?' at the Virus Bulletin Conference 1999 [1]. The presentation was so controversial, that the published proceedings only included some slides, while the actual paper was only made available at his personal website [2]. Some of the problems discussed in the paper were academic in nature, but many valid points were made which required attention. These included, but were not limited to, reporting problems, classification and naming problems, as well as testing issues. Unfortunately, many of the points have still not been addressed almost a decade later.

At the beginning of 2007, we submitted an abstract regarding the WildList problems for this year's Virus Bulletin Conference. Just a few weeks after it was accepted, we attended the International Antivirus Testing Workshop in Reykjavik, Iceland [3], where many leading researchers from almost all AV companies met with a few testers for the discussion of various aspects of what to test, how to test, what is stupid, and so on.

Based on the feedback received from the audience of the workshop, *Eset*'s Randy Abrams' conclusion 'Agreement was virtually unanimous that the WildList is no longer useful as a metric of the ability of a product to protect users', was highlighted in the June 2007 issue of *Virus Bulletin* [4]. A month later, *About.com*'s Mary Landesman responded with a comment about the 'The wild WildList', with the highlighted citation 'The WildList is more pertinent than ever – particular given today's threat landscape' [5].

The topic appears to be very hot and controversial. The goal of this paper is to look at the facts behind the WildList and propose some possible solutions to the problems.

## PROBLEM 1: THE CHANGING THREAT LANDSCAPE

While boot and file-infecting viruses were a problem in the first few years of the AV industry (up to about 1998), macro viruses soon started to become a problem (1996–2000), later followed by executable email worms and malware written in Visual Basic Script (VBS). Today, we have to deal with a wide variety of different types of attack and attack vectors. It's not only email communication which could be dangerous, but also standard web surfing, downloads, P2P networks, USB sticks and much more raises concern.

Malicious software can, for example, be classified based on whether or not it is able to self-replicate (which viruses or worms can do), which means that distribution systems are used to spread such samples. The latter category includes Trojan horses, backdoors, exploit codes which use vulnerabilities in certain software products, rootkits, adware and spyware as well as diallers and jokes. Some special categories exist, like bots (zombies), where some samples have the ability to self-replicate while others don't.

A classification can also be made based on whether we need to deal with clearly intentionally malicious software (e.g. viruses, worms, bots, exploit codes, Trojan horses), or not. Some 'grey areas' exist where it's not always easy to say if the program in question is intentionally malicious or just potentially unwanted (e.g. adware and spyware, diallers, jokes).

The WildList only contains intentionally malicious software which is able to self-replicate by infecting other files (viruses) and PCs in a network environment (worms). After some internal discussions, the WildList coverage was slightly extended to include some known bots, but only those that are able to spread by themselves, excluding the ones that fall more into the backdoor category.

Even more limiting, the WildList only includes PC viruses, so other non *IBM*-compatible hardware platforms, including mobile devices such as *Symbian*-based phones, are ignored. The same applies to operating systems which are not developed by *Microsoft*, as only viruses and worms for the MS-DOS and *Windows* platforms are included on the WildList. Other platforms like *MacOS*, *Linux* or *Solaris* are excluded. Memory-based worms like Code Red, SQL Slammer or MS Blaster are not included either.

One question would be whether limiting the WildList to just *Microsoft*-OS-based viruses, worms and (some) bots is a real limitation.

Many AV companies publish regular statistics of the malware they are seeing or which has been reported to them, so they could be a good source of information to answer the question.

*Panda Software* has recently released some details about the radical change in the 'malware industry' [6]. According to them, 83% of the newly discovered malware in the first half of 2007 was made up of Trojan horses. Worms accounted for only 8% of the newly discovered malware. Adware and spyware were responsible for 7% of all 'infections', while the

figures indicated that diallers are not relevant any more. The report concludes that, for malware authors, worms don't pay off as they mainly only cause damage, without a 'return on investment', while trojans that steal online banking data or passwords do pay off, as a lot of money can be made from the information gained. It is not only online banking accounts that hold interest for such criminals – identify theft in general and industrial espionage are lucrative fields, too.

Almost all AV companies publish regular 'top ten'-like lists of widespread malware. However, they tend to do so without providing any details as to how the data was gathered (e.g. what was counted: infected email attachments vs. the number of infected files on a system vs. infected systems only) and over what time span the information was collected, where the incidents occurred (e.g. was it a local phenomenon in a specific region or country, or a global outbreak for a specific continent or worldwide?) and without telling the reader the absolute number of infiltrations, just giving a ranking with or without percentages. This already violates many of the rules of basic statistics, so while the numbers might be good for PR purposes or press releases, they are irrelevant as an indicator of what's really 'in the wild'.

*Microsoft* recently launched its Malware Protection Center: Threat Research and Response website [7]. While they do not show any numbers or percentages in their rankings, they do differentiate what has been found in different environments.

Figure 1 shows the entries included in *Microsoft*'s list as of 18 July 2007.

The 'top desktop threats' are the samples that have been stopped by users that have one of *Microsoft*'s desktop protection tools – *OneCare* or *ForeFront* – installed on their systems. This includes website downloads, email attachments and the like. Interestingly, only one of the top ten threats could ever make it onto the WildList, as it was mainly non self-replicating malware that was blocked, with Win32/IRCbot.OP as the only exception. The remaining samples are exploits and various kinds of trojan and downloaders for further trojan components.

The 'top MSRT detections' in Figure 1a include only malware that was found installed on users' PCs during a scan of the *Microsoft Malicious Software Removal Tool* (*MSRT*), which usually runs monthly during the installation of security patches for *Windows* and *Office* products via *Windows Update*. It should be noted that *MSRT* can only detect a subset of the malware a typical AV product knows about, so the detection might be limited in some ways. No information is available as to whether or not the user has an AV application installed and running on his PC. The list only includes three items of malware which could ever make it onto the WildList, namely, Win32/RBot, Win32/Jeefo.A and Win32/Parite.B. None of the other malicious files – many of which are used to steal banking account data and passwords – would make it onto the WildList.

| Top desktop threats | Top MSRT detections |
|---|---|
| Exploit:HTML/IframeRef.gen | TrojanDownloader:Win32/Zlob.gen |
| TrojanDownloader:JS/Agent.FA | TrojanDownloader:Win32/Zlob.gen!A |
| Trojan:Win32/Fotomoto.A | TrojanDownloader:Win32/Zlob |
| Exploit:Win32/Anicmoo.A | Backdoor:Win32/Rbot |
| TrojanSpy:Win32/VBStat.E | Virus:Win32/Jeefo.A |
| Backdoor:Win32/IRCbot.OP | Virus:Win32/Parite.B |
| TrojanDownloader:Win32/Agent!D529 | Backdoor:Win32/Hupigon |
| TrojanDownloader:Win32/Zlob | TrojanSpy:Win32/Banker |
| TrojanDownloader:Win32/Small.NCK | TrojanDownloader:Win32/Renos.gen!A |
| TrojanDownloader:Win32/Small!AA7A | TrojanDropper:Win32/Hupigon.gen!A |

*Figure 1a: Microsoft malware prevalence statistics, part 1.*

| Most active email threats | Top adware/spyware |
|---|---|
| Worm:Win32/Netsky.P@mm | Adware:Win32/WhenU.SaveNow |
| Worm:Win32/Bagle.ZD@mm | Adware:Win32/Hotbar |
| Worm:Win32/Netsky.Z@mm | Adware:Win32/ClickSpring.PuritySCAN |
| TrojanDropper:Win32/Stration.gen!F | Program:Win32/Starware |
| TrojanDropper:Win32/Stration!ZIP | Trojan:Win32/Fotomoto.A |
| Virus:Win32/Virut.A | Spyware:Win32/CnsMin |
| Worm:Win32/Netsky.C@mm | Program:Win32/Winfixer |
| Worm:Win32/Netsky.CZ@mm | Trojan:Win32/C2Lop.C |
| Worm:Win32/Netsky.N@mm | Adware:Win32/ZangoSearchAssistant |
| Worm:Win32/Mytob.Q@mm | RemoteAccess:Win32/RealVNC |

*Figure 1b: Microsoft malware prevalence statistics, part 2.*

The world of statistics changes completely if we look at the 'most active email threats' section in Figure 1b – these are files which are usually detected and blocked by the email scanner at the gateway level. The two listed Win32/Stration entries are the only entries that are unsuitable for the WildList statistics… but almost all entries listed there are well-known, old viruses and worms, some of which first appeared many years ago. With current or even outdated AV protection in place, it's unlikely that anyone would get infected by the old Win32/Netsky and Win32/Bagle variants listed there. Of course, such 'old' malware needs to stay on the WildList to ensure that all products are tested against these aged critters and are able to protect against them.

However, systems that are already infected might never been cleaned up properly, because the user won't notice an infection or doesn't care about it, so infected systems continue to send out tens of thousands of email messages using a single DSL line connected to an infected PC. Therefore, AV companies and end-users might see very high numbers for this attack vector only. But when all this 'old stuff' – which comes in high numbers – is properly blocked at the email gateway/guard layer, the question remains as to which are the current infiltration vectors. The same question can be asked when we look at the files which are blocked and reported as 'top desktop threats' (Figure 1a) and already installed malware listed in the 'top MSRT detections' (Figure 1a).

The question is easy to answer: we're no longer talking about email only, but also about HTTP, downloads (with or without fake content), exploits in common web browsers like *IE*, *Mozilla* or *Opera*, as well as exploits found in operating systems like *Windows* [8]. Website advertisements delivered from hacked servers can be used to infect PCs, too [9]. Products in use here not only block malicious software using AV technology, but also use URL filtering [10] and web reputation services [11]. Other infection vectors are P2P networks, local networks, USB devices (including portable music players), DVDs and CDs, and maybe even floppy disks.

This is similar to what AV company *Trend Micro* reported: the days of the big global outbreaks are over; the current threats are targeted, regional, web-based and financially motivated [12].

The last section of Figure 1b, the 'top adware/spyware', won't be covered by the WildList at all, for definition reasons, so WildList-based tests will ignore this category entirely as being irrelevant.

'Password stealers targeting games are growing more than ever' is the title of a statistic created by *McAfee* [13]. While it only covers a special kind of trojan, one of the conclusions made is 'by and large, when June ended, malware classified in that category came close to 35,000. If the trend goes on, we will reach 45,000 items at the dawn of the next year.' When we look not at individual samples, but at an entire family of password-stealing malware, the statistics of the most common malware changes rapidly. PWS.Banker would make it to the number one position in the top five, followed by PWS.Lineage, PWS.Legmir, PWS.Mmorpg, PWS.Gamania, PWS.WoW or PWS.Ldpinch, depending on which quarter of the year 2007 was reviewed.

There are a lot more statistics available from various websites (some more can be found at the link section at [14]), but we can already easily see that restricting the WildList to just viruses, worms and some bots is a real limitation, as almost all of the currently distributed (not self-spreading) malware samples are missed. Just a small percentage of self-replicating malware can ever make it onto the WildList, so today's real problems – like the never-ending stream of trojans and (zero-day) exploits – are completely ignored.

Users might think that a review which is based on WildList will also include tests like the detection of the infamous and various ANI, DOC, JPG or WMF exploit samples, which never was, nor is the case today. There is no question that these cases are relevant and many of the exploits have been used for massive or targeted attacks [15, 16].

While there are often reports in the media about malware for non *Microsoft* platforms, nobody knows for sure what's 'in the wild' and what isn't. So one company might declare such a piece of malware to be widespread while a competitor denies that this specific variant has ever been seen anywhere in the world.

## PROBLEM 2: THE INCREASE IN THE NUMBER OF MALWARE SAMPLES

One of the problems we're faced with today is the high number of new malware samples which are seen spreading every day. The graph in Figure 2 shows the number of new unique samples we received from various sources (such as honeypot networks, AV companies, infected companies and organizations as well as other testers).

By the end of 2004, we usually had to deal with around 10,000 samples a month. By the end of 2006, the number had exploded to

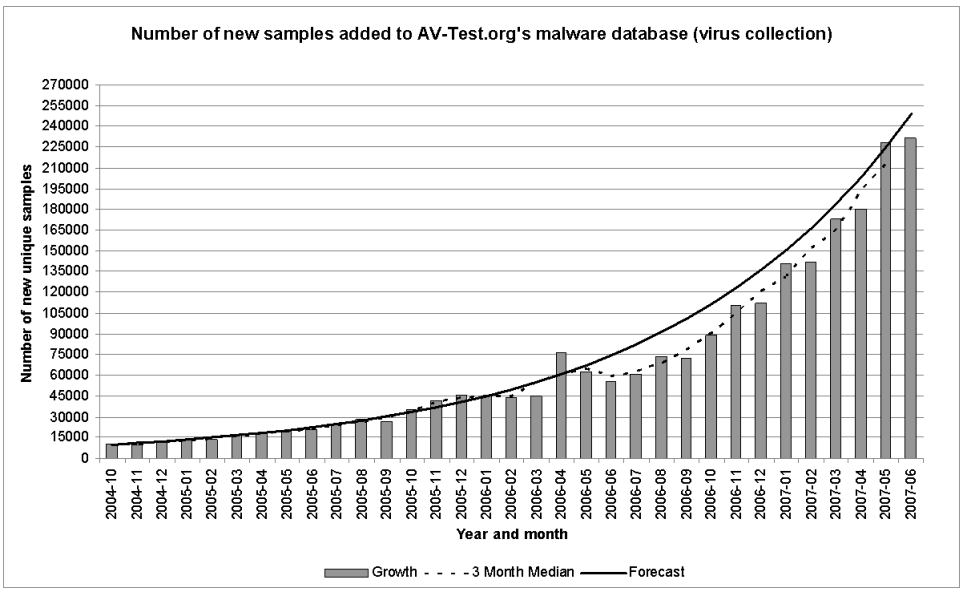


*Figure 2: AV-Test's malware collection growth.*

100,000 new samples a month and at the time of writing this paper, the number has more than doubled again to around 225,000 new malware files per month, or around 7,500 samples for a single day!

At the time of writing this paper, *VirusTotal*, a free online multi-scanner service where any PC user can submit suspicious samples for a scan with various AV products, shows that more than 11,000 (non-unique) infected samples have been found by the systems within the last 24 hours. That's quite an impressive number as well [17].

Of course, it's important to mention that the number of unique samples has nothing to do with the number of malware families, which is a lot lower. While *Trend Micro*'s virus map (which is based on data gathered by their online virus scanner *HouseCall*) shows a worldwide number of 561,626 malicious files found in the case of the top threat EXPL_ANICMOO.GEN [18] (an exploit which uses the MS07-017 ANI file vulnerability [19]), the number of different infected computers was a lot lower at 44,182 [20].

However, the dramatic change in the case of virus samples and the number of malware variants is not visible when we look at the WildList reports from October 1995 to May 2007. On average, only 13 new samples were added to the WildList per month, with a minimum of no new samples added and a maximum of 69 new files. Surprisingly, this maximum was not reached recently, but in mid-2005, and after this, the number of newly reported malware files decreased consistently (see Appendix A). When we look at the average number of samples which made it onto the WildList in a specific year, one might think that the malware problem is getting smaller (see Figure 3), but in fact the opposite is true (see Figure 2).

It's definitely true that targeted attacks are becoming more and more common (so nobody sees a real 'outbreak' which will get a lot of media attention and which will most likely be reported to the WildList sooner or later), but even so, the low number of new additions to the WildList is a big surprise. Based on the significant increase in the number of new unique malware samples, one might expect some 100 to 1,000 new malware entries to make it onto the WildList.

However, local 'outbreaks' might be under-reported or never be reported at all. This especially applies to all these 'Rechnungen'-like Trojan horses with fake invoices from various companies which are widely spammed to many inboxes in the German-speaking region every few days [21]. Some changes in the tactics are appearing every few weeks, so one of the last big Trojan horse seedings was for a fake one-time password generator for *PayPal* [22, 23]. A different example is the Gromozon case which was relevant in Italy, but not in the rest of the world [24].

If we look at *Trend Micro*'s virus map again [20], we can see that in North America WORM_ANIG.A was the top threat during the last 30 days (when the statistic was sorted by the number of infected computers), in South America the top problem was SPYW_RASERV.A, in Europe WORM_NETSKY.P, in Africa PE_SALITY.AS, in Asia EXPL_ANICMOO.GEN and in Australia PE_PARITE.A-O. So despite the fact that we live in a widely connected world, every region has to deal with different problems.

There must be something wrong here. For example, there is no clear definition as to when a WildList reporter should start to report something to the WildList or how many different sources need to be affected. In addition to this, only malware which is reported by at least two independent parties will be added to the top part of the WildList.

Furthermore, the malware must have the ability to self-replicate, so it must be a virus, a worm or a bot, as anything else won't make it onto the WildList. If one keeps this in mind, the statements made in the first section of this paper – that Trojan horses and other financially motivated malware are responsible for much more than 90% of today's malware problem – we can see that the WildList isn't that bad, as it does reflect the decline of the traditional self-replicating malware.

## PROBLEM 3: NOBODY WANTS TO REPORT ANYTHING

The WildList started with a very few reporters from specific AV companies, and was then extended to a small number of independent researchers as well as companies which are using AV products. The WildList from October 1995 lists a total of 33 WildList reporters (see Appendix A for the full details). However, only nine of them were actively reporting malware during this month, and just three were working actively on the November 1995 revision.

The WildList from May 2007 shows an impressive number of 84 reporters – but only 11 of them had something to report. Even worse, during February and April 2007, only nine WildList reporters were active – that's exactly the same number as we had back in October 1995, even though the number of listed WildList reporters had almost tripled during this period (see Figure 4).

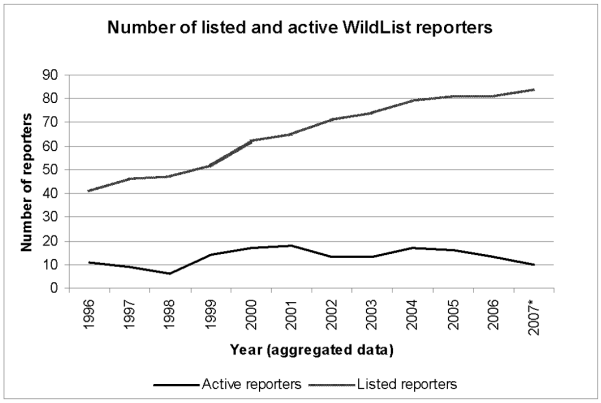


*Figure 4: Comparison of listed and really active reporters. (*Up to WildList 2007-05 only.)*

It seems that, even though the list of 'supposed' reporters gets longer every few months, only a very small subset – about

| Year | 1996 | 1997 | 1998 | 1999 | 2000 | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 | 2007* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Added samples | 8 | 8 | 4 | 9 | 12 | 8 | 6 | 10 | 21 | 36 | 26 | 20 |

*Figure 3: Number of newly added WildList samples. (*Up to WildList 2007-05 only.)*

10% – of the listed reporters are still sending reports to the WildList. There are many problems which come together and which are described in detail in Vesselin Bontchev's paper [1, 2].

One of the biggest problems might be all of the manual steps that the volunteer reporters are required to go through to get a sample reported: one has to wait for a report form, which, once it arrives, needs to be filled out very quickly or it might not be processed. All reporters must be named individuals (even if some names are not shown on the WildList), as only individuals, but not corporate-wide reports from different persons are allowed. When a WildList reporter is travelling, on holiday, or ill, nobody else can step into his shoes and replace him even temporarily as reporter, so a reporting opportunity will be lost completely.

When looking at a recent WildList, we can also see that the majority of reporters are from AV companies – only a small subset of 10–15% of the listed reporters appear to be independent experts and users of AV software. Of course, an AV company will most likely only report samples they can detect (and clean), so some trouble-makers might never get reported, as being tested against these problem cases could result in bad test results for the AV companies or the loss of a certification. (This point is pure speculation, but theoretically a possible scenario!) So a good number of samples which are considered to be widespread might be lost here, as nobody really wants to report them.

Besides this, the current owner of the WildList is in the business of certifying AV products on detection, cleaning and other aspects, so their interest is that AV tools can easily pass their benchmarks, as only then can the product certifications be sold to interested companies. One could be forgiven for using basic economics to conclude that the easier it is to pass the WildList level, the more products can be certified and the more money can be made.

## PROBLEM 4: THE WILDLIST IS OUTDATED WHEN PUBLISHED

New versions of the WildList are released every month. However, a new WildList is not made available to AV companies and testers immediately after the end of the reporting period, but rather it usually takes around 40 days (or longer). So the WildList does not reflect today's threats, but the malware which was spreading at some time in the past, usually 40 to 70 days ago (see Figure 5, details can be found in Appendix A). For example, the WildList for May 2007 (which should include all reports from May 1, to May 31) was not available until mid-July 2007.

To make the situation worse, many tests and certifications are still performed not with the current WildList, but with a three-month-old version, 'to give AV companies time for adding detection of all samples', to quote a report. So samples might already be five months old before the first products are tested against them. This is not a fault of the WildList, but we hope testers might change their opinion about how to test. Using the most recent WildList for a test should be a must, as the WildList is already a bit antiquated when published.

However, one might even think about the idea to test the scanners' detection with archived updates as part of a retrospective test: a product should be able to find and remediate all malware from a specific WildList at the end of the reporting period (maybe plus a few days). At this time, no official WildList would have been available, but such a test would reflect the 'real world' situation in a better way. A user wants to know if he is protected against all critters which are released during 1–31 May 2007 (and which are included in the May 2007 WildList) at the latest at the beginning of June 2007 (as no indication is given as to on which date a piece of malware was seen first, we have to use the last possible time for additions).

Of course, such a test would imply that the WildList is truly reflective of the 'real world' malware situation. Right now, the results of such testing might not be very useful or could lead to wrong conclusions, due to problems 1 to 3 we listed earlier. But it would be a good addition to future tests, to answer the question 'how well am I really protected?', and not 'how protected am I a few weeks after the WildList (and thus, all samples) became available to AV companies, so detections for missing samples could easily be added to their products?'

## SUGGESTIONS TO MAKE THE WILDLIST MORE USEFUL THAN IT IS TODAY

The first WildList problem – the changing threat landscape – is easy to solve, at least on paper. We only need to change the definition of the WildList, to include all kinds of malicious software and potentially unwanted software. It should not only be the 'traditional' viruses, worms and bots that are listed, but also trojans, backdoors, adware and spyware and other incarnations such as rootkits, diallers and all these file-based exploit codes. Different malware categories and possible unwanted applications can be listed in separate sections of the WildList.

There is no reason to exclude non self-replicating malware from the WildList – of course, these files cannot spread by themselves, but they are actively distributed and used by criminals to gain money. Even worse, a virus or a worm could download and install an independently running Trojan horse or backdoor on a PC, but only the worm would be listed on the WildList, even if both components are equally distributed 'in the wild'. Of course, including non self-replicating malware in the WildList would cause the number of reported samples to increase dramatically and the same would apply to the level of resources required.

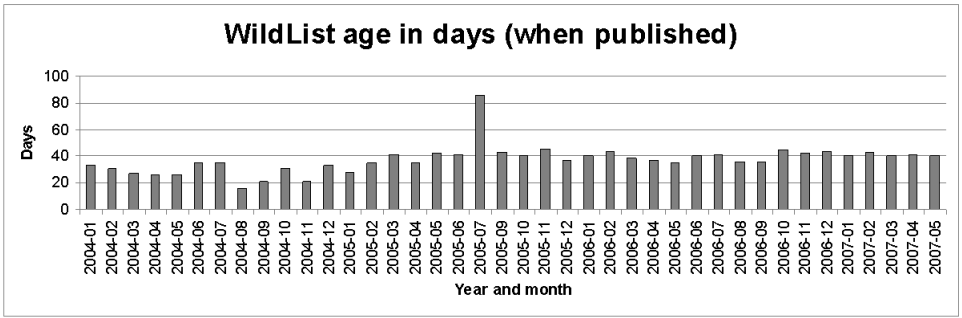


*Figure 5: Age of the WildList in days (since 2004).*

In order to make it as easy as possible for users to submit samples, a system is needed that provides sufficient and appropriate user interfaces. The success of systems like *VirusTotal* (with tens of thousands of samples submitted per week [17]) is in providing an easy-to-use interface. So the short-term task should be to provide a web interface and an email account to which samples can be submitted, because both solutions are very easy to implement.

More active reporters and people who work independently from AV companies are required to create a better view of the malware which is supposed to be 'in the wild'. First, a lot more users of AV software need to report to the WildList, but we have to keep in mind that this group might be very product-centric and only report what is already stopped by their malware guard, ignoring unknown or (rootkit-)hidden infections on their systems. Secondly, CERTs and groups which are dealing with large numbers of malware infections on a daily basis should be added to the reporting group, too.

These groups need to be offered benefits in return for participating and actively contributing data to the systems on a regular basis. This might include early access to important samples and related data (e.g. about the attack or the current AV detections) as well as quick descriptions for new malware. The number of such benefits could be decreased or even declined if the reporter stops reporting or the quality of the reports deteriorates. So the number of listed reporters would reflect the number of really active reporters in a better way than today, when almost none of the reporters which have ever been listed has ever been removed.

It is important that in addition to individual memberships corporate-wide efforts are possible (where the company takes the responsibility to report regularly), so that different (possibly regularly changing) individuals from the same company can send reports to the WildList. So the illness of a named reporter or if he is on holiday (for example, at the same time as many other named reporters during the summer season) should not result in important malware going unreported. While most AV labs work 24/7 (168 hours per week), the individual researchers are usually only available for around 30 to 40 hours per week.

A reporter should be able to send an initial report with a short description of an incident and a sample, and update or add more information at a later time when new details are available – for example, when the number of known infected systems has increased. Such a report might also include additional facts such as where the malware was found (e.g. which regions are affected, whether the samples were found on the mail gateway, web gateway, or if it was on a disk, in memory etc.). Automatic processes can work in the background and escalate files to a human only when problems arise or any other kind of feedback is required.

The sample processing work flow must contain several steps: some kind of black/whitelisting check, multi-scanner checks, sandbox analysis, automatic classification approaches and as the final frontier possibly a human check, if there is nothing else left. Maybe some kind of scorecard approach could be introduced, as in automatic spam filters. With every step the sample takes, a score is given – a positive number of points reflects a malware sample, while a negative number reflects the opposite. When all steps have been completed you are left with a positive or negative number of points. Now we have to define the critical value beyond which the sample is categorized as malware. All samples that do not fit into good or bad must be further investigated by hand.

The black/whitelisting check might save a lot of work, as already known malware files (which might have been submitted earlier, or are available in various virus collections, or are known corrupted files) could be filtered out using the blacklist, and a whitelist can be used to see if the file is a known good file. A whitelist will always be incomplete, as the volume of software that is 'out there' is simply too great, but many important cases can be handled this way, e.g. by adding the installers and installed files for the top 100 software products, including all *Windows* and *Office* versions ever released, in different Service Pack levels. Such collections already exist in the labs of AV companies and in most cases, the files themselves are not required (which might cause legal problems, as this could be seen as software piracy), but just the cryptographic checksums and lengths as well as the origin of the file. This will prevent a harmless 'notepad.exe', even if submitted by several parties, from making it onto the WildList.

A multi-scanner system can then be used to give an initial idea as to what kind of malware has been submitted. If no detection is in place (by any scanners), it could be that a file is simply too new and the malware author has spent time fine-tuning the package until it can no longer be detected by an AV product, generically or heuristically. Some AV detections do not necessarily indicate that a file really contains malware, as false positives are not uncommon and there are quite a few AV companies that add detection for files without analysis as long as they are detected by enough competitor scanners [25].

There are a number of sandbox utilities available publicly, which work based on different concepts. These include the *CWSandbox* [26], the *Norman Sandbox* [27], *Anubis* [28] and *PC Tools Threat Expert* [29]. As the technology behind such tools is already a target of malware writers (who are trying to detect virtual machines, for example), the output of many and not just one sandbox should be used and compared. An executable which behaves quite differently on *VMware* when compared with the run on a 'real' PC might already be suspicious.

Of course, neither sandboxes nor multi-scanner systems can guarantee that a file is harmless or dangerous, as malware components could be missing or the malware could be waiting for time- or behaviour-triggered actions. Further manual analysis is required in such situations to avoid possibly harmless files making it onto the WildList.

As part of the sample aggregation process, it is important that similar samples (based on file characteristics, memory dumps and actions/behaviour) are grouped together, as malware writers can easily create variants of existing malware just by changing a few bytes and/or by repacking files using runtime packers and encryption routines. Commercial tools such as those from *Sabre Security* are available for malware binary and behaviour analysis, visualization of code and program structures, as well as for automated malware classification [30].

At the moment, every submitted sample is checked manually by the organization behind the WildList to verify it is a working one and it is real malware. With automation, a lot of time can be saved and this must not decrease the quality, but could even increase it. Even then, it could be that some non malware samples could make it onto the WildList, so a process to report innocent samples is required, so that

samples can be excluded quickly, if required. It should be noted that it has always been a requirement for every AV company and tester to replicate and validate the samples themselves. Given the benefit of samples and related data being made available a lot earlier, the price of the possibility that the 'wrong' files might make it on WildList from time to time shouldn't be too high.

As soon as samples can be processed quickly enough, not only submissions from individuals, organizations and corporations should be accepted, but also those from other sources. This should include sensor networks which are already widely deployed by AV companies and security service providers, which report data directly to the WildList or using an AV company as 'relay'. In most cases, it should be enough to distribute only 'smart' checksums of an incident rather than the sample every time – only when a malware is seen for the first time, is a file required. Such a system might either check if the WildList already has access to a sample (e.g. using a database look-up) or the WildList can send an automatic request to the reporting entity to get a copy of the malware.

To raise the security level, the submission system could support the transmission of encrypted samples via PGP, GPG or S/MIME and the connection to web interface could be secured by SSL. In order to suppress bulk mailings or flooding the system, mechanisms to restrict the number of submissions per day from a mail domain or using a Captcha in the web interface to lock out automatic HTTP submissions should be implemented. In the long term, more fine-grained access control mechanisms should be used. So it might be possible for registered users to submit more samples per day or get access to the samples, statistics and more via an online portal.

To assure the independence of the WildList from marketing money and to put up enough capital to finance the needed sample processing infrastructure – not only the hardware and software, but also the people behind it – a subscription-based access to the system could be established. So while all AV vendors, testers and interested parties can see basic data and access samples free of charge, advanced and more powerful statistics as well as outbreak warnings or test results could be made available for fee-paying subscribers (read: supporters) only. For example, large users of AV software could purchase real-time access to a database whose data wouldn't be made publicly available instantly, but only after some time, or in an aggregated form only. Organizations offering paid-for WildList certifications could be asked to pay (more) for accurate and up-to-date data.

The WildList must also be owned and operated by one or more independent parties. One main point of criticism from third parties is that, right now, it is mainly the AV industry that is providing reports to the WildList (and paying a bit for its maintenance), and the same samples are being used to test the AV industry's products, in tests for which they are also paying. A user organization might not only help to acquire more independent reporters (and thus, samples), but also to increase the standard and protection factor of AV products, as the sample reporting, sample collection and sample testing would then be done by different parties. So the AV industry would no longer be able to define their own testing standards.

Besides the manual and automatic additions made to the WildList, it is important also to remove 'outdated' malware which has not been seen spreading or distributing after some amount of time. Right now, a reporter must explicitly state that a malware has not been seen within the last few weeks or months. Only one auto-deletion feature is implemented – that an item of malware will be removed exactly six months after the initial report, and then no more reminders will be sent. A piece of malware should simply stay on the WildList only as long as it's reported within an amount of time by at least one party, and if no further constant reports are made, a sample will finally be removed. Thus reporters no longer need to remember what they haven't seen. This way, current malware threats like Trojan horses (which are unable to spread by themselves) are also handled in a better way, as they are often used for one-time attacks only, then replaced by a new incarnation.

When an automated malware tracking system is in place, it will be easier to publish the WildList in a more timely manner, without an average delay of 40 days. In the best case, a monthly WildList could be made available as quickly as two to three days after the end of the reporting period. As the monthly interval was introduced more than a decade ago, we have to consider now whether this period is still properly reflecting today's malware landscape where AV updates are not released every quarter, or once a month, but usually every few hours or even hourly.

With enough resources and will, weekly or even daily WildLists could be made available for the benefit of AV companies (to quickly get access to new important samples) and AV users (who are protected quickly against new threats). It should be noted that the virus collections which are shared between the AV industry on a network of trusted parties are usually made available once a month. Some AV companies have already switched to weekly or daily collection updates, at least for the most important samples. These monthly collections (which cover all received non-confidential samples of the entire last month) are usually made available no later than a week after the end of the month. These collections are already a lot more current than the WildList samples, so it's not only theoretically possible, but also practically possible to share samples in a timely manner!

Many AV companies also publish statistics of the most commonly seen malware in the last month. (However, see the first section for reasons why this data might not necessarily be trustworthy!) These publications are also made available on the first two to three days of a new month – a long time before the new WildList becomes available. Of course, 'online' statistics provide information which is always up to date (but might only reflect already detectable malware).

## CONCLUSION

The WildList in its current state has a lot of problems, which are known, but have not been addressed for many years. As a result, the WildList has been getting less useful over the years, though it is still being hyped as 'everything someone needs to take care about' [31].

From a marketing point of view, the WildList and tests based on it are still seen as 'state of the art', even today. Technically, the WildList does not reflect today's threats, the number of new malware reported to the WildList is too low, nobody really cares about reporting new malware and the WildList is completely outdated when published.

Important changes must be made. Soon there won't be much left to report as viruses and worms are on the decline, having been replaced by Trojan horses and backdoors. The current (strict) definition of what can be considered to be 'in the wild' must be changed, to really cover all kinds of malware, rather than just a small part of it.

Automatic systems and processes – controlled and backed up by humans – are required in order to allow a greater number of samples to be processed and to enable WildLists to be released more quickly in the future. Last but not least, it should be as easy as possible for a person (not only current WildList reporters) to report new malware, as soon as it is discovered, and not weeks or months later when the WildList report form finally arrives in their inbox.

Nobody can predict how many samples will be on the WildList if important categories are not ignored and samples are reported from many independent entities. If the number of malware samples proves too high, one can still adjust the settings a bit, to allow an AV company to pass a test if it detects all of the top 1,000 malware threats and misses a very few of the rest.

However, before we can think about such actions, we need to change the WildList! A lot of time has already passed without any action, so it's important to act now.

## REFERENCES


[1] Bontchev, V. The WildList – still useful? Proceedings of the 9th International Virus Bulletin Conference 1999, pp.281–288.

[2] Bontchev, V. http://www.people.frisk-software.com/~bontchev/.

[3] International Antivirus Testing Workshop. http://www.f-prot.com/workshop2007/.

[4] Abrams, R. AV industry comments on anti-malware testing. Virus Bulletin June 2007, p.2. http://www.virustbn.com/virusbulletin/archive/2007/02/vb200702-comment.

[5] Landesman, M. The wild WildList. Virus Bulletin July 2007, p.2. http://www.virusbtn.com/virusbulletin/archive/2007/07/vb200707-comment.

[6] Ziemann, F. Malware-Trends: Trojanische Pferde weiter auf dem Vormarsch. http://www.pcwelt.de/start/sicherheit/sonstiges/news/87510/.

[7] Malware Protection Center: Threat Research and Response. http://www.microsoft.com/security/portal/.

[8] http://www.malwareradar.com/malware_today/.

[9] Steel, S. Hackers can now deliver viruses via web ads. http://online.wsj.com/article/SB118480608500871051.html?mod=todays_us_marketplace.

[10] Trend Micro warns of fast-moving web threat spreading from thousands of compromised web domains and URLs in Italy and around the world. http://us.trendmicro.com/us/about/news/pr/article/20070618185040.html.

[11] Trend Micro extends web reputation technology to Internet gateway. http://us.trendmicro.com/us/about/news/pr/article/20070628230427.html.

[12] Trend Micro Virenreport für das 1. Halbjahr 2007: Web-basierte Angriffe greifen um sich. http://de.trendmicro-europe.com/enterprise/about_us/spresse.php?&id=314.

[13] Paget, F. Password stealers targeting games are growing more than ever. http://www.avertlabs.com/research/blog/index.php/2007/07/16/password-stealers-targeting-games-are-growing-more-than-ever/.

[14] http://www.av-test.org/index.php?lang=0&menue=5.

[15] Ziemann, F. Anti-Phishing Working Group: Immer mehr Websites beherbergen Malware. http://www.pcwelt.de/start/sicherheit/sonstiges/news/87910/.

[16] Phishing activity trends: report for the month of May, 2007. http://www.antiphishing.org/reports/apwg_report_may_2007.pdf.

[17] http://www.virustotal.com/estadisticas.html.

[18] http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=EXPL_ANICMOO.GEN.

[19] Microsoft Security Bulletin MS07-017: Vulnerabilities in GDI could allow remote code execution (925902). http://www.microsoft.com/technet/security/bulletin/MS07-017.mspx.

[20] http://www.trendmicro.com/map/.

[21] Ziemann, F. Malware: Vorgebliche Rechnungs-Mails von Otto.de und T-Mobile. http://www.pcwelt.de/start/sicherheit/virenticker/news/87939/.

[22] Schmidt, J. Achtung bei angeblichem E-TAN-Generator für Paypal. http://www.heise.de/newsticker/meldung/92282.

[23] Ziemann, F. Malware-Spam: vorgeblicher E-TAN-Generator für Paypal. http://www.pcwelt.de/start/sicherheit/virenticker/news/86489/.

[24] Florio, E. Gromozon is 'Live' (just in Italy). http://www.symantec.com/enterprise/security_response/weblog/2007/03/gromozon_is_live_just_in_italy.html.

[25] Canja, V. Exploiting the testing system. http://www.f-prot.com/workshop2007/presentations.html.

[26] http://www.cwsandbox.org/.

[27] http://sandbox.norman.no/live_4.html.

[28] http://analysis.seclab.tuwien.ac.at/index.php.

[29] http://www.pctools.com/threat-expert/.

[30] http://www.sabre-security.com/.

[31] Kuo, J. Hello World. http://blogs.technet.com/antimalware/archive/2007/03/15/hello-world.aspx.

## APPENDIX A: SOME STATISTICS ABOUT THE WILDLIST

The following table includes some aggregated details for every WildList which has been released since October 1995. The first column shows the issue of the WildList the report is based on, in the *YYYY-MM* format. All WildLists can be found at http://www.wildlist.org/WildList/ and in order to access a specific version of the WildList, one only needs to visit the website http://www.wildlist.org/WildList/YYYYMM.htm (replacing *YYYY* and *MM* with the proper id.)

The 'Age in days' shows the time difference from the end of the reporting period to the release date when a specific WildList was made available. For example, the May 2007 WildList was available 40 days after the end of May 2007. In June 1998, a different system was in use (the WildList of a specific month was a report of the malware seen a month ago), so the WildList for June 1998 was already published five days before the month's end, as it covered the malware that was widespread in May 1998. The 'Age in days' is unavailable for WildLists published before 1998-06. Between 1998-06 and 2003-01, the file creation time of the uploaded WildList has been used for this calculation. Starting at 2003-02, the calculations are based on the announcements made on the WildList mailing list.

The number of 'Newly added samples' shows how many new malware samples made it to the upper part of the WildList where samples are shared with the AV industry and which is the 'main list' which is used to test and certify AV products. The number of 'Active WildList reporters' shows how many reporters were actually reporting something in a specific month. The 'Total count of WildList reporters' indicates how many reporters were listed in the directory of participants, regardless of whether they are active or not.

In addition to this, the 'New malware with the highest number of reporters' is included, so one can see what was the biggest challenge during a specific month. The 'Maximum number of reporters' shows how many reporters have seen this malware and reported it to the WildList for this month. If there were more samples with the same number of reporters, only the first entry is shown.

| WildList | Age in days | Newly added samples | Active WildList reporters | Total count of WildList reporters | New malware with the highest number of reporters | Maximum no. of reporters |
|---|---|---|---|---|---|---|
| 1995-10 | n/a | 5 | 9 | 33 | Unashamed | 3 |
| 1995-11 | n/a | 1 | 3 | 33 | Parity_Boot.A | 3 |
| 1996-01 | n/a | 5 | 9 | 37 | Diablo_Boot | 4 |
| 1996-02 | n/a | 2 | 4 | 37 | Bye | 3 |
| 1996-03 | n/a | 6 | 9 | 37 | Burglar | 4 |
| 1996-05 | n/a | 6 | 10 | 39 | Dark_Avenger.2100.SI.A | 3 |
| 1996-06 | n/a | 9 | 10 | 38 | Werewolf.1500.B | 6 |
| 1996-07 | n/a | 6 | 8 | 42 | Hare.7610 | 5 |
| 1996-09 | n/a | 14 | 13 | 42 | Tentacle.10634 | 4 |
| 1996-10 | n/a | 12 | 16 | 44 | WM.Npad | 7 |
| 1996-12 | n/a | 12 | 17 | 45 | WM.Wazzu.C | 6 |
| 1997-02 | n/a | 6 | 9 | 46 | WM.MDMA.D | 4 |
| 1997-03 | n/a | 2 | 4 | 46 | WM.Hybrid.A | 4 |
| 1997-05 | n/a | 9 | 11 | 46 | WM.Appder.A | 5 |
| 1997-07 | n/a | 26 | 16 | 45 | X97M.Laroux.A | 7 |
| 1997-08 | n/a | 2 | 5 | 45 | WM.Alien.A | 4 |
| 1997-09 | n/a | 5 | 7 | 46 | XM.Laroux.D | 3 |
| 1997-10 | n/a | 2 | 3 | 46 | WM.Schumann.C | 2 |
| 1997-11 | n/a | 8 | 8 | 46 | Baboon | 3 |
| 1997-12 | n/a | 9 | 10 | 46 | WM.MDMA.D | 4 |
| 1998-01 | n/a | 2 | 4 | 46 | Invisible_Man.2926.A | 2 |
| 1998-02 | n/a | 4 | 6 | 47 | Lilith | 2 |
| 1998-03 | n/a | 2 | 3 | 47 | Bachkhoa.3999 | 2 |
| 1998-04 | n/a | 3 | 6 | 47 | Nilz.1000.B | 2 |
| 1998-05 | n/a | 0 | 0 | 47 | - (none) | 0 |
| 1998-06 | -5 | 2 | 4 | 47 | Desperado.2403.A | 2 |
| 1998-07 | -10 | 7 | 8 | 46 | W97M/Groov.A | 3 |
| 1998-08 | -9 | 3 | 5 | 46 | W97M/Groov.B | 2 |
| 1998-10 | -16 | 1 | 2 | 46 | X97M/Extras.B | 2 |
| 1998-12 | -1 | 16 | 19 | 46 | W97M/Class.D | 7 |
| 1999-01 | -10 | 1 | 2 | 45 | W97M/Brenda.A | 2 |

| WildList | Age in days | Newly added samples | Active WildList reporters | Total count of WildList reporters | New malware with the highest number of reporters | Maximum no. of reporters |
|---|---|---|---|---|---|---|
| 1999-02 | 1 | 6 | 10 | 46 | W97M/Ethan.A | 3 |
| 1999-03 | -16 | 3 | 15 | 46 | W32/Ska.A | 12 |
| 1999-04 | -13 | 7 | 23 | 48 | W97M/Melissa.A | 19 |
| 1999-05 | -12 | 10 | 13 | 50 | W95/Kenston.1895 | 3 |
| 1999-06 | -8 | 8 | 21 | 50 | W32/ExploreZip | 16 |
| 1999-07 | -12 | 4 | 8 | 51 | W97M/Melissa.I | 3 |
| 1999-08 | -16 | 5 | 10 | 55 | W97M/VMPCK1.BY | 4 |
| 1999-09 | -10 | 15 | 14 | 56 | HLLP.Toadie.7800 | 5 |
| 1999-10 | -13 | 12 | 13 | 56 | VBS/Freelink | 10 |
| 1999-11 | -9 | 12 | 15 | 57 | W97M/Ethan.AW | 4 |
| 1999-12 | -15 | 15 | 23 | 59 | W32/ExploreZip.pak | 13 |
| 2000-01 | -12 | 7 | 10 | 59 | W32/NewApt.A | 4 |
| 2000-02 | -7 | 11 | 15 | 61 | W32/Fix2001.worm | 11 |
| 2000-04 | -6 | 21 | 20 | 61 | VBS/Netlog.A | 4 |
| 2000-05 | -9 | 11 | 24 | 61 | VBS/LoveLetter.A | 21 |
| 2000-06 | -4 | 12 | 14 | 62 | W97M/Bridge.A | 5 |
| 2000-07 | -14 | 13 | 17 | 62 | VBS/Stages.A | 11 |
| 2000-08 | -10 | 6 | 10 | 62 | W97M/Bobo.B | 4 |
| 2000-09 | -9 | 10 | 15 | 63 | W32/Qaz-m | 8 |
| 2000-10 | -1 | 5 | 10 | 63 | VBS/LoveLetter.C | 7 |
| 2000-11 | -4 | 12 | 23 | 61 | W32/Navidad-m | 16 |
| 2000-12 | -9 | 18 | 22 | 61 | W32/Hybris.B-m | 9 |
| 2001-01 | -12 | 5 | 11 | 63 | W32/Hybris.A-m | 3 |
| 2001-02 | -7 | 6 | 22 | 62 | VBS/VBSWG.J-mm | 19 |
| 2001-03 | -7 | 9 | 9 | 62 | VBS/Sorry.C | 3 |
| 2001-04 | -10 | 9 | 16 | 63 | W32/Magistr.A-mm | 11 |
| 2001-05 | -7 | 5 | 28 | 63 | VBS/VBSWG.X-mm | 21 |
| 2001-06 | -2 | 4 | 13 | 64 | VBS/VBSWG.Z-mm | 9 |
| 2001-07 | -5 | 8 | 14 | 64 | W32/Choke.A | 8 |
| 2001-08 | -12 | 9 | 9 | 64 | W32/Bady.C | 4 |
| 2001-10 | -13 | 7 | 19 | 70 | W32/Nimda.A-mm | 15 |
| 2001-11 | -2 | 10 | 32 | 69 | W32/BadTrans.B-mm | 23 |
| 2001-12 | 1 | 10 | 24 | 70 | W32/Goner.A-mm | 15 |
| 2002-01 | -15 | 4 | 11 | 70 | W32/Zoher.A-mm | 9 |
| 2002-02 | -7 | 7 | 12 | 70 | W32/MyParty.A-mm | 7 |
| 2002-03 | -4 | 7 | 9 | 70 | W32/Maldal.C-mm | 5 |
| 2002-04 | -13 | 6 | 18 | 70 | W32/FBound.C-mm | 15 |
| 2002-05 | -7 | 6 | 22 | 70 | W32/Klez.H-mm | 20 |
| 2002-06 | -11 | 6 | 9 | 70 | W32/Benjamin.A-mm | 5 |
| 2002-07 | 4 | 6 | 12 | 70 | W32/Yaha.G-mm | 9 |
| 2002-08 | -4 | 1 | 2 | 70 | W32/Duni.A | 2 |
| 2002-09 | -5 | 7 | 13 | 71 | W32/Datom.A | 7 |
| 2002-10 | 0 | 6 | 22 | 70 | W32/BugBear-mm | 16 |
| 2002-11 | 3 | 3 | 6 | 70 | W32/Opaserv.E | 3 |
| 2002-12 | 14 | 7 | 16 | 73 | W32/Winevar.A-mm | 11 |
| 2003-01 | 6 | 19 | 24 | 73 | W32/Sobig-mm | 13 |
| 2003-02 | 2 | 3 | 5 | 73 | W32/Yaha.L-mm | 4 |
| 2003-03 | 2 | 7 | 15 | 73 | W32/Lovgate.C-mm | 6 |

| WildList | Age in days | Newly added samples | Active WildList reporters | Total count of WildList reporters | New malware with the highest number of reporters | Maximum no. of reporters |
|---|---|---|---|---|---|---|
| 2003-04 | 6 | 11 | 12 | 73 | W32/Ganda-mm | 7 |
| 2003-05 | 11 | 5 | 10 | 73 | W32/Deloder | 4 |
| 2003-06 | 28 | 20 | 19 | 73 | W32/Sobig.B-mm | 13 |
| 2003-07 | 28 | 8 | 11 | 73 | W32/Sobig.E-mm | 8 |
| 2003-09 | 13 | 14 | 19 | 73 | W32/Mimail-mm | 13 |
| 2003-10 | 25 | 12 | 12 | 75 | W32/Sober-mm | 6 |
| 2003-11 | 50 | 7 | 10 | 75 | W32/Mimail.H-mm | 7 |
| 2003-12 | 34 | 3 | 5 | 75 | W32/Sober.C-mm | 3 |
| 2004-01 | 33 | 18 | 15 | 76 | W32/Mydoom.A-mm | 10 |
| 2004-02 | 31 | 12 | 10 | 76 | W32/Netsky.B-mm | 8 |
| 2004-03 | 27 | 32 | 18 | 77 | W32/Bagle.P-mm | 9 |
| 2004-04 | 26 | 17 | 16 | 77 | W32/Bagle.AA-mm | 8 |
| 2004-05 | 26 | 14 | 17 | 78 | W32/Sasser.B | 8 |
| 2004-06 | 35 | 24 | 22 | 78 | W32/Zafi.B-mm | 19 |
| 2004-07 | 35 | 27 | 19 | 79 | W32/Bagle.AI-mm | 13 |
| 2004-08 | 16 | 19 | 15 | 80 | W32/Evaman.C-mm | 6 |
| 2004-09 | 21 | 8 | 8 | 81 | W32/Bagle.AZ-mm | 2 |
| 2004-10 | 31 | 40 | 18 | 82 | W32/Bagle.BB-mm | 11 |
| 2004-11 | 21 | 10 | 15 | 82 | W32/Sober.I-mm | 11 |
| 2004-12 | 33 | 30 | 20 | 82 | W32/Zafi.D-mm | 16 |
| 2005-01 | 28 | 10 | 13 | 81 | W32/Bagle.BK-mm | 9 |
| 2005-02 | 35 | 24 | 16 | 81 | W32/Sober.K-mm | 7 |
| 2005-03 | 41 | 61 | 14 | 81 | W32/Mytob.J-mm | 5 |
| 2005-04 | 35 | 31 | 12 | 81 | W32/Mytob.X-mm | 4 |
| 2005-05 | 42 | 69 | 18 | 81 | W32/Mytob.DX-mm | 7 |
| 2005-06 | 41 | 68 | 20 | 81 | W32/Mytob.EC-mm | 6 |
| 2005-07 | 86 | 19 | 13 | 81 | W32/Mytob.EK-mm | 3 |
| 2005-09 | 43 | 55 | 20 | 81 | W32/Mytob!ITW#241 | 7 |
| 2005-10 | 40 | 14 | 11 | 81 | W32/Bagle.DX-mm | 3 |
| 2005-11 | 46 | 21 | 16 | 81 | W32/Sober.Z-mm | 9 |
| 2005-12 | 37 | 21 | 14 | 81 | W32/Mytob!ITW#414 | 3 |
| 2006-01 | 40 | 42 | 11 | 82 | W32/Alcra.B | 3 |
| 2006-02 | 44 | 32 | 18 | 80 | W32/Bagle!ITW#82 | 9 |
| 2006-03 | 39 | 15 | 10 | 80 | W32/Bagle!ITW#85 | 4 |
| 2006-04 | 37 | 11 | 13 | 80 | W32/Mytob!ITW#498 | 8 |
| 2006-05 | 35 | 12 | 9 | 80 | W32/Mytob!ITW#297 | 3 |
| 2006-06 | 40 | 22 | 14 | 80 | W32/Bagle!ITW#112 | 6 |
| 2006-07 | 41 | 36 | 14 | 81 | W32/Looked!ITW#3 | 3 |
| 2006-08 | 36 | 29 | 14 | 81 | W32/Stration!ITW#1 | 4 |
| 2006-09 | 36 | 8 | 11 | 81 | W32/Stration!ITW#23 | 3 |
| 2006-10 | 45 | 51 | 15 | 81 | W32/Stration!ITW#20 | 4 |
| 2006-11 | 42 | 12 | 9 | 81 | W32/Stration!ITW#124 | 3 |
| 2006-12 | 44 | 32 | 16 | 81 | W32/Puce!ITW#1 | 4 |
| 2007-01 | 40 | 8 | 8 | 82 | VBS/Areses!ITW#48 | 2 |
| 2007-02 | 43 | 41 | 9 | 82 | W32/Allaple!ITW#1 | 2 |
| 2007-03 | 40 | 6 | 9 | 84 | W32/Sality!ITW#4 | 3 |
| 2007-04 | 41 | 9 | 9 | 84 | W32/Sober.AA | 4 |
| 2007-05 | 40 | 35 | 11 | 84 | W32/Looked!ITW#127 | 3 |