VIRUS BULLETIN 2010 – A RETROSPECTIVE

Steve R White

IBM TJ Watson Research Centre, PO Box 704, Yorktown Heights, NY 10598, USA
Tel +1 914 784 7368 • Fax +1 914 784 6054 • Email srwhite@us.ibm.com

ABSTRACT

We look back on the years 2000–2010 from the perspective of the anti-virus industry. Four technology trends were responsible for substantial changes in the computing environment, which formed a backdrop for the virus problem.

(1) Pervasive computing devices are now the dominant way that people interact with the digital world, far outnumbering traditional PCs, and the shift in architecture was responsible for both new problems and new protections.

(2) The decline of Moore's Law resulted in dramatically falling prices for commercial computing systems, resulting both in their commoditization and much more widespread use throughout the world.

(3) Broadband access to the Internet from most of the developed world put much of the earth's population on-line all the time.

(4) The rise of e-commerce has affected every sector of the economy; the digital economy now rivals its material counterpart.

We review the most significant viral disasters in the past ten years, showing how they could have been predicted from these technology trends, and usually avoided. To the contrary, we show how the anti-virus industry actually responded, often after the fact. While anti-virus technology has evolved significantly since the year 2000, with several technological marvels to its credit, perhaps the most surprising change is that few users are even aware that it exists any more.

INTRODUCTION

Welcome to *Virus Bulletin* 2010. I'm Steve White. Some of you may recall that I was involved in the anti-virus industry back all those years ago, when it first began. On this twentieth anniversary of the first *Virus Bulletin* Conference, the nice people at *VB* asked me to come back to say a few words. Figuring that you might all be tired of hearing the same old predictions of what the future will be like, I'm going to use this opportunity to look back at the last ten years in the anti-virus industry: the years 2000 through 2010.

2000: THE NEW MILLENNIUM

In many ways, the beginning of the New Millennium was surprisingly boring. The 'Y2K Virus'✻, as it was called all too often, did not cause widespread havoc. Nor did it fry all of the computers in the world, bankrupt entire countries or lead to the end of the world. As the wave of midnight spread over the South Pacific and then on to the rest of the world, the biggest surprise was that almost nothing happened.

The news media, which had spent the previous year focusing on worst-case scenarios, worked hard to find anything at all, in any country whatsoever, that happened as a result of the Y2K problem. Sure, there were a few minor problems, but fewer than happened on any normal day due to normal computer problems. There were, of course, computer virus problems that year, and a couple of them were quite significant at the time. They largely centered on the fact that the most popular applications – the mail and document applications from *Microsoft* – were themselves programmable, and were the medium in which viruses spread.✻ It was the first year in which a self-mailing virus really hit big, becoming the most rapidly spreading, most widespread virus up to then.✻ And even so, it took 24 hours to spread all around the world (since it still required people to arrive at work, open their mail and look at what was called an 'attachment'). I doubt that more than a few techies even remember the name of that virus.[1]

What I remember is that anti-virus companies still were not prepared to handle it. It still took them many hours to make a solution to the new virus widely available. It took some of them days just to get it right. Imagine – days – when the virus itself sprinted around the world with the speed of the rising sun! And this was after several self-mailing viruses the previous year made it more than apparent that the problem had reached a critical point.✻

There were two other important events that year, though I do not think many people realized how important they were at the time. The first thing that happened was that several anti-virus companies teamed up with mail and other providers to integrate anti-virus products into some of the infrastructure of the Internet.✻ Some people at the time viewed this as a great marketing move, capitalizing as it did on the publicity that self-mailing viruses had gotten, but perhaps conferring only a small increment in real virus protection. Those of you who have followed this technology realize how wrong they were! (More about that later.)

The second thing that happened is that some very basic immune system technology was deployed for the first time. In fact, I was part of the team that developed it at *IBM Research*, in a joint effort with *Symantec*.✻ We were pretty proud of it at the time. It really did find new viruses, analyse them, and distribute cures for them, and it did it fairly quickly compared to what other

companies could do then. Still, looking back on it, it seems rather primitive. And there were a lot of sceptics who thought it would not work as well as what they were already doing, or would not work at all.

2001: THE ZUZU VIRUS

The Zuzu virus* was a heinous and costly event, of course, but in retrospect it was probably inevitable. It started on 22 March, 2001 with a trickle, then a flood of panicky messages posted on various Internet newsgroups from what appeared to be hundreds of companies around the world. They said that some new, terrible virus had hit their company, that it was wreaking havoc, and that they were unable to cope with it.

At the same time, anti-virus companies got copies of a very large, very complex virus that contained the string 'What is Zuzu?' Given the obvious urgency of the situation, anti-virus gurus started analysing it right away. It was easy to see that the virus had all sorts of code related to mail, network-based spread, password cracking, etc. However, its size and complexity meant that no one would understand what it did for quite a while.

The news media picked up the story.* Managers of anti-virus groups were interviewed, saying that this was the most complex virus ever seen and it could be capable of just about anything. Security experts were interviewed, saying that this was just the kind of thing they had feared for years. Could this be the killer virus, the media asked, the virus that really does bring down the Internet? Maybe, they all agreed, maybe it is.

Security teams at hundreds, perhaps thousands of companies, responded quickly. Not wanting to get hit themselves, they did what they had done in previous epidemics – they shut down their mail systems, since this was the primary way that epidemics spread at the time. However, because it was not understood how the virus was propagating or what effects it had, they did more – they shut down their Internet connections altogether and disabled their internal networks as well. They were prepared to wait out the epidemic.

Then things started getting odd. News reporters, eager to do follow-up stories on the initial warnings, sought interviews with officials from severely affected companies. They found a lot of companies that had shut down their networks, and these made very effective stories. They found a few small companies that claimed their computers had crashed because of the virus. But, and this was the odd part, they found very few companies that would say they had been hit. And none of them could actually produce a copy of the virus.

It is a measure of the naiveté of the time that it was not until the next day that the underlying story emerged. It turned out that the newsgroup postings were almost all forged.[2] There was a Zuzu virus – the one that anti-virus companies had – but it was not spreading wildly around the world. In fact, it was not spreading at all. In fact, it had never spread anywhere.

The whole thing was a publicity stunt gone horribly wrong. A public relations firm named *MacIntyer Knox Oldsen & Urquhart* had the clever idea of generating buzz for their customer *Zuzu Industries*, an Internet security start-up, by attaching its name to a virus scare. Unfortunately for everyone, this worked far too well. Damages to businesses from cutting off their network access were estimated at over a billion dollars.

The resulting liability suits sank *Zuzu Industries* almost immediately.* It was followed into bankruptcy soon thereafter by the ill-considered *MacIntyer Knox Oldsen & Urquhart.* The news media concluded that the anti-virus industry had blown the problem out of all proportion.* There was a trendy term in use around that time: 'viral marketing'. I do not think this is what it meant, but it certainly fell out of use very quickly after the Zuzu incident.

2002: A LITTLE TEA PARTY

Now I *know* you remember the Tea Party of 2002! Even those of you from outside the US remember *this*. It was the early evening of 15 April, the day on which income taxes must be reported to the US government. The Internal Revenue Service (the 'IRS'), which collects federal taxes in the US, had made a big and rather successful push to get people to file their taxes electronically. So, millions upon millions of people were typing on their PCs, finishing their electronic tax forms, and submitting them over the Internet.

At the same time, a new virus had been released and was spreading rapidly via email. It came as an 'attachment' to mail with the subject line 'IRS announces 10% tax break for electronic filing'. It first appeared on the east coast of the US, and was subsequently thought to have originated in Reston, Virginia. With a subject line like that, and on the very day when taxes were due, perhaps it is not surprising that it spread like wildfire across the US, infecting a large but still unknown number of home computers.

The Tea Party virus*, as it became known, did three things. Like any self-mailing virus of the time, it sent itself to everyone in the victim's address book. That was typically a few dozen to a few hundred people. Then it looked for files from a few common tax return programs and, if it found them, changed them so that more money was owed to the government. The amount was not so large as to be obvious, but was large enough to be particularly annoying to the people said to owe it. The third thing the virus did was to delete itself, and all evidence of itself except for the changes to the tax files themselves. This turned out to be the most important characteristic of the virus.

Of course, anti-virus companies got samples of this virus within minutes of its first appearance. But it was highly polymorphic, using techniques that had been widely discussed in anti-virus circles but not incorporated into automated defences at that time. So, it sat in queues at various companies until human virus analysts got around to looking at it. By then, it was much too late. The Tea Party virus had infected thousands, perhaps millions, of home computers, and corrupted as many tax returns – returns that had already been submitted to the IRS.

By the next morning, news of the virus was all over the media*. It was the major story of the week. The IRS, seeking to calm worried taxpayers, announced that they had put their best people on it, that they were working closely with anti-virus companies, and that the situation was well in hand. As you might recall, it was not. There was no way of telling which tax returns had been corrupted. Indeed, if anti-virus software had not caught the virus with heuristic detection in the first place and the virus had actually activated and had a chance to cover its tracks, there was no way of knowing if the virus had ever been on a user's system.

And that was the trick. The IRS did not know which returns had been corrupted. The users did not know if the virus had been on their system. There was no way to tell who owed what the

government claimed, and who owed less. Not unless everyone went through their records in great detail and did their tax calculations again. After weeks of agonizing denials that they had an intractable problem, the IRS finally conceded that they could not determine which tax returns were correct and which were corrupted. Their only recourse was to ask all taxpayers who had submitted electronic forms (and there were lots!) to file their tax returns again – on paper.✻

It was the right thing to do, but it caused serious delays in settling how much money taxpayers owed, and delayed many tax refunds by months. The resulting public outcry prompted the US Congress to pass legislation requiring all federal agencies to install and run anti-virus software on all of their systems, and to filter incoming and outgoing traffic for viruses.✻ They also banned electronic filing of tax returns, which is why, to this day, we still submit them on paper every year.✻

Now, the clever among you will have noticed that none of the things the Congress did would have actually prevented the Tea Party virus, or in fact diminished it in any way. But the Congress felt they had to do *something*.

Later that same week in April, a man named Martin Fennig was arrested and charged with writing the Tea Party virus. He was subsequently convicted in what appeared to be a fairly straightforward case. The conviction was overturned on appeal for procedural errors by the investigators and Fennig was freed.✻ While there are various theories about who might have written the Tea Party virus, no one else was ever charged.

2003: PERVASIVE PERVASIVENESS

In 2003, another seminal event occurred though, again, few people realized how seminal it would be for the anti-virus industry. For the first time since the early 1980s, the PC was no longer the most prevalent computing platform in the world.✻ It had been overtaken, as widely predicted, by what were then called 'pervasive computing devices'.[3] These were Personal Digital Assistants, Web Phones, and most devices running the low-end operating systems from *The Windows Company*.

These devices were aimed at people who used them as special-purpose artifacts – designed to do a few tasks and nothing else. These people were not interested in general-purpose computing, and were certainly not interested in becoming system administrators for a half-dozen such devices. So the manufacturers did the obvious thing. They relieved their customers of the responsibility of system administration by doing it for them.

Almost all of these devices had subscription features that allowed the manufacturer to update the device automatically with bug fixes and feature enhancements. As you know, this strategy was very successful.

Anti-virus companies were working on protection for these devices, but these were not usually high-priority projects. Sure, viruses had been written for virtually all of the environments used by these devices, but no viruses were actually spreading in any significant way. Anti-virus efforts were not, therefore, fabulously aggressive. There was basic technology to scan some objects, or to prototype a simple heuristic or two but, with few exceptions, there was not a concerted effort to protect these platforms against future threats.

2004: WITH SUGAR, PLEASE

In July of 2004, the Java 4 standard was announced.⁜ It included a new security model called 'Sugar'.⁜ The *Java Group* had focused on security since Java began. In this release, they focused on enterprise-wide, and even global, administration of security. Behaviour Control Lists (BCLs), which were introduced in Java 3, were greatly extended so that developers and administrators could enforce very fine-grained restrictions on the operation of a Java applet or application.

Developers could specify the behaviours that needed to be allowed for the program to run. Administrators could specify policies for what behaviours were allowed globally, for each software developer, or for each program.

Having extended BCLs, the *Java Group* also put into place a clever hierarchical management scheme for it. An enterprise could establish and enforce a global BCL policy, and each division within the enterprise could add its own local BCLs. BCLs and their management structure were set up to be dynamic. They could be modified or updated relatively quickly. A change in the global BCL policy could be reflected across an enterprise in an hour or so.

Anti-virus companies viewed this as an opportunity to expand their existing services of examining programs and declaring them to be either viruses or Trojan horses. They offered services in which they would certify that the BCLs associated with a given app were correct, that is, both needed by the program and not generally dangerous if used. Subscribers to the service could get updates to the BCL certifications and deploy them very quickly to every Java installation they had. The anti-virus companies offered to certify programs developed by others, initially for a fee and then, when that proved unpopular with the development community, for free. Curiously, this was not a commercial success. It seems that developers felt they could do this better themselves, and companies did not want to rely on anti-virus vendors to certify the software they used.

Ominously, the Sugar architecture was not adopted by *The Windows Company*, which continued to pursue its strategy of promoting a competing active content language that did not have a similar security architecture.⁜

2005: THE DIGITAL ECONOMY

As quickly as the Web had become a major social force during the 1990s, this last decade saw the dramatic rise of the global digital economy. First seeking broader markets for their services and more competition among their suppliers, companies started finding, contracting with, and doing business with other companies over the Net. It was clear that these first few sparks would burst into a bonfire as soon as the number of these businesses reached critical mass. It was clear that it would transform the global economy. What surprised everyone, as they had been surprised by the Web a decade earlier, was how quickly it happened.

By 2005, there was no longer any doubt that the world was in the midst of an economic revolution that would be bigger than the Industrial Revolution. Company after company rushed to solidify their presence in the digital economy, eventually automating much of their routine business processes and supplier relationships. Opportunities for new companies that facilitated business in this new world were at an all-time high.

This was the revolution that carried *Lixxuid*[4] into global prominence. What started as a small Australian-based Internet bank in late 2000 grew explosively to become the twenty-fifth largest bank in the world by 2005 – an event unprecedented among financial institutions – by facilitating financial transactions for businesses in this digital economy.✳

Then, on 9 August, 2005, *Lixxuid's* luck ran out.✳ It was mid-morning in Melbourne, and usage of their primary transaction gateways went through the roof. Almost simultaneously, their phones filled with customers reporting that their transactions were not being processed. It took over an hour for worried administrators to confirm what they feared: they were under attack.

At first, they thought it was hackers, since it looked initially like a common kind of denial of service attack. But, each time they thought they had a handle on the problem, it grew worse. By the end of the first day, they were under attack by more computers than they could count.

The attackers turned out not to be hackers, but viruses, using a variant of the VDP (Viral Distributed Ping) attack.[5] The number of attackers kept increasing because the number of infected systems kept increasing in those first few hours.

You may not remember, but anti-virus companies did pretty well during this incident. They got copies of the virus right away, and had solutions for the virus available well before the sun set in Melbourne. (Some companies had a solution much faster than others, for reasons that modesty forbids me to mention.)

What did not go well was actually eliminating the virus. While almost everyone had the capability of automatically updating their virus definitions and cleaning any new viruses off of their systems, very few people had this feature turned on. Indeed, most corporations still required manual approval to distribute definitions, either because they had extensive in-house testing procedures or because they didn't want to be the first ones to distribute definitions that might cause internal problems.✳

That was probably a good and conservative choice for their own companies. But it meant that the VDP-XX virus could gain an early and firm foothold in hundreds of companies, and tens of thousands of households, worldwide. And all of them were aimed at *Lixxuid*.

Lixxuid issued a hasty press release, suggesting that they would take legal action against companies and individuals who did not take rapid measures to prevent their systems from attacking their bank. The media picked this up and made it part of almost every story they ran.✳ This got the attention of lots of people, especially in the more litigious countries, and lots of people and companies made sure they updated their anti-virus software to eliminate the virus.

The viral population peaked early in the second day. *Lixxuid* system administrators worked around the clock, and had achieved some reduction in the incoming flood of traffic, but not nearly enough to control the attack. It took another day and a half before a combination of anti-virus software, media warnings, and hastily-crafted network filters brought the attacking traffic down enough that *Lixxuid* could once again process transactions, and even then only slowly.

But the damage had been done. *Lixxuid's* doors had been closed for just over three days, and the world does not appreciate a bank that closes its doors. On the first day that *Lixxuid* reopened for business, they bled to death from customers withdrawing their money and closing their accounts. It is, by now, the most analysed bank failure in history.✳

Police and investigatory agencies from around the world joined in the search for the perpetrator or perpetrators of this crime. The search went on for many months. Whether it was because those responsible were crafty or just very, very quiet, no one was ever arrested. To this day, theories abound.＊ In the following year, over 50 copycat attacks were stopped before they started by anti-virus protection that was already in place, and several authors of the copycat viruses were arrested and ultimately convicted.＊ Whether any of these copycat authors was the author of the original VDP-XX virus is not known.

2006: MOORE'S WALL

2006 brought a worrisome realization. For decades, Moore's Law＊ was the foundation on which progress in computing was built – the nearly unshakable belief that advances in silicon technology would lead to chips whose performance doubled every 18 months.

A prescient article by an *Intel* engineer in 1999＊ suggested that, in the following decade, the chip industry faced a series of very difficult obstacles. The oxide layer, which allows transistors to be switched on and off, would become so thin – just a few atoms thick – that it would no longer be an effective insulator for the switching current. Dopants, which create free electrons for the switch's current, would become so sparse that the transistors themselves would be unreliable. Solutions to these problems, the article pointed out, were not obvious.

Still, people in the chip industry, and throughout the computing industry, were unfazed. Moore's Law would continue its inexorable climb one way or another, they assured each other. It had always been thus, they reasoned, and thus it would always be. Unfortunately, their optimism was not borne out.

New ways of building transistors on chips to avoid these problems turned out to be difficult to manufacture. New chip architectures to deal with the inherent unreliability of the transistors turned out to be more elusive than hoped.

For a few years, everyone watched the performance curve deviate ever so slightly from Moore's Law. That had happened before, they said, and it always gotten back on track. They were sure that someone, somewhere, would find a solution to these problems.

But by 2006 the trend was clear. Chip performance was not increasing as rapidly as predicted. Despite tremendous efforts, the problems had not been solved. We now refer to this as 'Moore's Wall'＊ – the wall into which the chip industry ran, headlong, and with dramatic consequences.

The optimists are with us always. Now they tell us that new technologies are just around the corner: 3D devices＊, molecular computing＊, quantum computing＊. They assure us that we will soon return to those halcyon days, that Moore's Law will rise again in a new realm as it has before, that Moore's Wall will be known to our children only as Moore's Lapse.＊ And I think they are right. What is not clear is how long it will take to perfect these new technologies, to make them manufacturable, to make them reliable, to make them affordable. What is not clear is how long Moore's Lapse will be.

In 2006, *Intel*, then the world's pre-eminent maker of microprocessor chips, introduced the *Intel Googlium*™ microprocessor and, at the same time, announced that Moore's Law was at an end,

that decades of easy performance increases were over, at least for the time being.[※] The *Intel Googlium*™ was to represent the last significant silicon-related performance enhancement of the decade.

2007: THE UNWIRING OF INDIA

Moore's Wall had an interesting effect. As it became harder to compete on raw chip perform-ance, basic chips became cheaper. And this happened at the same time as the world piled into the digital economy. Several progressive countries made big bets on these two trends. India was probably in the forefront.

The 'unwiring' of India, started in 2003, was declared complete in 2007.[※] High-bandwidth wired access was available in all major urban areas along with moderate-bandwidth wireless access. This accelerated massive buying of now-cheap network devices in India, contributing further to their dramatic worldwide price decline. In the space of a few years, this snowball effect spread devices throughout the developed world and much of the developing world. Today, the people at this conference are constantly connected to the Internet through the half-dozen devices that we carry or wear all the time. This was a big change.

The ubiquity of these new devices was not missed by the virus writers.[※] Device viruses became the dominant virus problem. Anti-virus companies scrambled to update their device technologies to handle the plethora of new viral carriers, and hook them into their automated defences.

2008: NOTHING HAPPENS

In 2008, nothing happened. Well, nothing directly relevant to the anti-virus industry, anyhow. I suppose that people in the US would regard the election of President Clinton as significant.

2009: A SOLUTION EMERGES

Each decade seems to have brought with it a standard architecture to address the virus problem of the time, and this decade is no different. As in previous decades, the solution addresses the new problems that have emerged:

Internet-based spread: Virtually all viruses today spread primarily via the Internet. Naturally, there are viruses that spread by other means, and the anti-virus industry is always issuing breathless press releases about some tricky new way a virus spreads. But nothing even comes close to Internet spread in terms of pervasiveness and speed. So, most of the virus incidents seen by real people are spread via the Internet.[※]

Fortunately, the Internet is an important part of the solution. Ten years ago, the idea of integrat-ing anti-virus software with commercial mail services was new. Now, no one in their right mind would subscribe to a mail service that did not filter out viruses. (There are people who do, and while they seem to have a kind of 'herd immunity' because almost everyone else has filters, they

do get more virus infections than the rest of us.[h]) As active content became a part of standard XML business transactions between companies, and after viruses showed up there as well[h], nearly every business-to-business transaction facility now includes integrated virus filters.[h] Similarly, device hubs quietly watch for viruses in transmissions to and from the many devices we now carry with us or wear.[h] At the endpoints – the devices we all carry around – manufacturers nearly universally integrate anti-virus software into these devices before we purchase them.

Administrative overhead: As the demands of anti-virus updates on system administrators, and particularly users, became more severe, the industry took that burden upon itself. Just as other kinds of software are updated automatically by the company that develops them – correcting bugs or adding features – anti-virus software is largely updated automatically. Anti-virus software was one of the first kinds of software to need continuous updates, and anti-virus companies were among the first to pioneer the subscription models that have become common throughout the software industry. Coupled with network-based virus filtering and nearly universal integration of anti-virus software into devices before they are purchased, automated updates mean that most users are blissfully unaware that they even have anti-virus software. It has become part of the firmament of nature in cyberspace.

Rapid epidemics: When I first got involved in the anti-virus field back in the late 1980s, when personal computer viruses were just beginning, those viruses spread on floppy diskettes, that is, on physical media that one person would hand to another person. This was really slow! It took a typical virus months or even years to become prevalent around the world, if it ever did.[h] These days, viruses sweep around the word in hours or minutes. The anti-virus industry has responded with technology for rapid, network-based response to epidemics. The goal of this technology is the same as for the early immune system in 2000 – find new viruses, craft cures for them, distribute and install the cure everywhere, and do this faster than the viruses themselves can spread. But I must admit that the solution that has evolved is quite a bit faster and more comprehensive than what we put together in 2000! It would have been hard to imagine back then.

Complex viruses: The virus writers did not go to sleep during the last decade – unfortunately! They have continued to develop techniques that tax even our current, very impressive, anti-virus technology. A decade ago, industry pundits predicted that scanning – looking for strings within a file that would indicate a virus – would fall by the wayside, to be replaced by <insert pundit's favorite alternative technology here>.[h] That did not happen, but viral defence did evolve to blunt the tactics of the virus writers. One virus-writing tactic that emerged – at first by accident and later, I think, on purpose – was Lurking[h]: making it hard to find a virus via simple scanning technology that performed only a very simple examination of certain parts of certain objects in the system. The anti-virus industry was forced to move to more comprehensive scanning – scanning all parts of all objects, and doing some fairly sophisticated analysis during the scan.

This all took time, and a naïve implementation would have been very, very slow. The anti-virus industry came up with a clever solution – use various heuristics, long relegated to second-class status as virus detectors – as filters *in front of* scanners.[*] That is, heuristics are now used as a first check for whether an object might be infected. The front-line heuristics are very fast, and eliminate most objects as not being infected. Any remaining objects are passed to second-line heuristics that are a bit slower and a bit more precise. And down the line until the object is passed to scanners, and then verifiers, to determine with great precision and certainty that it is infected, and with which virus it is infected.[6] Among these front-line heuristics are the nearly abandoned change detectors of twenty years ago[*], which can tell quickly if an object has

changed since it was last checked for viruses; if it has not, if it was not previously infected, and if the virus detector has not been updated, it is not necessary to check it again.

Small devices: Earlier in the decade, it was widely believed that devices – the computing devices we carry and wear – would require radically different anti-virus technology, at least to protect their internal environment. They were, it was argued, so *small* – with hardly any memory at all – so small that it would not be possible to fit the ever-growing PC-based anti-virus products into them. Interestingly, this turned out to be right – and wrong.

It was right in that the monolithic, standalone applications that were typical of anti-virus protection then would not fit. Nor would the ever-growing scanner-based virus definition files – certainly not with as many viruses as we have cataloged today. But it was also wrong; it was not necessary to stuff old programs into new devices.

In retrospect, the solution was obvious. The heuristic hierarchy that solved the speed problem for complex viruses is the first half of the solution. Most of the time, it's not necessary to have anything running in the devices except the first-level, or maybe the second-level, heuristics. And those are typically small and fast.

The second half of the solution is the Internet. If it's ever necessary to actually scan an object inside a device, it's not necessary to scan it for all 500,000 known viruses. Intermediate heuristics can easily cut the search down to a few hundred viruses at most.❋

These devices can easily cache virus definitions for the viruses you're actually likely to see. For all the rest, the definitions can simply be paged in from the next level up in the network. In fact, the networks of anti-virus vendors are now all hierarchical, caching the least information possible in the customer devices and systems, staging the less-used information in intermediate servers and gateways, and connecting them to the automated analysis facilities and human analysts that are at the pinnacle of the pyramid.❋ The Internet makes it all one global system.

2010: HELLO? HELLO?

Here it is 2010. The anti-virus industry has been working on the virus problem for over twenty years. All in all, things seem to be going pretty well this year. There have been no major virus incidents, no overblown virus hoaxes. The nearly seven billion residents of the planet have gone about their daily routines – shopping, gossiping, composing symphonies, and waging war – all without thinking very much about computer viruses. And that is how it should be.

There is one thing that is just a little bit odd recently. In the past few days, the phones[7] have been acting up. It seemed to happen at the same time as an automated update of the operating system from *The Windows Company* for the phone component of devices.

At first, I thought my glasses had stopped working, but everything other than the audio channel was fine. Then I noticed it in my earring too, and then my sketchpad. You may have had the same experience. There was a news alert a few minutes later.❋ This has never happened before, at least not this widespread. It is still not clear what is going on. The media is saying it is a virus.❋ We are not sure yet. Hopefully we will know more during VB2010 itself and I will be able to tell a more complete story.

THREE DECADES

It is instructive to see where we have come over the last three decades.

In 1990, virus incidents were called urban myths, 'like rumors of alligators in the sewers of New York'.* In 2000, it was so clear that viruses were real, and presented such an immediate problem, that businesses would close their network connections when they heard rumors of viruses. In 2010, the problem may be under control, at least for the time being.

In 1990, there were around 50 viruses. In 2000 there were around 50,000. In 2010 there are nearly 500,000.*

In 1990, virus defences consisted of scanning tools that were often unreliable and hard to use. Anti-virus companies typically took a month or more to react to a new virus, which was fine because it took the viruses even longer to spread around the world.* In 2000, virus defences had matured to suites of products on multiple platforms that were deployed around the world. Customers had simple Internet connections to anti-virus vendors to submit suspicious objects and receive virus definition updates.

Anti-virus companies typically reacted to a new virus in days – sometimes less if it appeared to be a major customer problem.* In 2010, virus defence consists of global distributed systems, with components in nearly every endpoint device and Internet way station in the world. Anti-virus companies typically react to a new virus in minutes, and it is a good thing too, as that is how fast viruses spread around the world.

We have come a long way.

LESSONS LEARNED

In the last decade, there have been a few dramatic virus incidents that, in some way, affected millions of people. There have been spectacular hoaxes, after which everyone blamed everyone else for not figuring them out earlier. Viruses moved to new parts of the computing ecology, almost always festering in these new niches before anti-virus technology was available to cope with them. Somehow, the world muddled through it all. In short the last decade was, for the virus problem and the anti-virus industry, much like the previous one.

The anti-virus industry had a tough job in keeping up with the changing virus problem and the many new niches for viral mischief. In general, they did a great job. We can breathe the same sigh of relief that we did in 2000 when the Y2K bug did not destroy the world: through all the virus problems, the vast majority were handled quickly and efficiently and we are, after all, still here. In the process, the anti-virus industry created several technological marvels, pioneering vendor-maintained endpoint software and creating global automated defences. Anti-virus technology has become like air: ubiquitous, vital for our survival, and almost completely invisible.

Nevertheless, some people say that the anti-virus industry is still more reactive than proactive, waiting for problems to occur in a new viral niche before creating a solution for them. They say that the self-mailing viral epidemics of a decade ago went on far too long before there was an effective solution, that the Tea Party Virus could have been done years before but the industry

still was not ready for it, that the virus that sank *Lixxuid* could have been prevented. Perhaps they are right. To be fair, it is difficult to anticipate exactly which niche will become populated with viruses, and users do not often change their behavior in the absence of a clear and present danger. Still, the stakes are increasing, and it is becoming more and more problematic to be behind in protecting new areas of the computing environment.

I wonder what will happen in the next ten years?

FOOTNOTES

[1]   It was the 'Love Letter' virus.

[2]   Curiously, a few postings turned out to be genuine, in the sense that real people thought they had the virus, panicked, and posted hysterical pleas for help.

[3]   It is amusing to think back on this, really. Would Henry Ford have referred to 'pervasive transportation devices', or Thomas Edison to 'pervasive illumination devices'? Yet, just a few years ago, the idea that computing elements would be embedded in everything seemed so surprising that people invented a specific term for it.

[4]   The company's name was pronounced 'Liquid'.

[5]   I hope you do not mind if I avoid going into detail about this attack. While it was an obvious attack years before it happened, and is nearly as easy today, there have been mercifully few events like it since then. I would be happy if that trend continued.

[6]   If a heuristic says that an object might be infected, but a lower-level heuristic (or the scanner/verifier) says it is not, it is a candidate to be forwarded and analysed via immune system technology.

[7]   It is an interesting lesson in the spread of technology that today, in most developed countries, you no longer buy a device that would have been called a 'phone' ten years ago. The falling price of components and the ubiquity of devices led to an audio facility being built into just about everything. It was too cheap not to do.

TOUCH REFERENCES ✳

[*The Editors of the* VB 2010 *conference proceedings apologize for the temporary unavailability of the touch-references* (✳) *in this paper. As soon as this week's widespread device problems are straightened out, we are sure the touch-references will work just fine.*]

DISCLAIMER

This paper attempts to take a humorous look at what might happen in the next ten years of the anti-virus field. The author is not actually from the future, and does not actually know how things will turn out. The events described in this paper are not actually historical facts, have not yet happened, and might never happen. Except for *IBM*, *Microsoft*, *Intel*, *Symantec*, the Love Letter virus and me, the names of companies, viruses and people are entirely fictional. Sheesh.