DOI:10.1145/1787234.1787246

Eugene H. Spafford

Privacy and Security
Remembrances of Things Pest

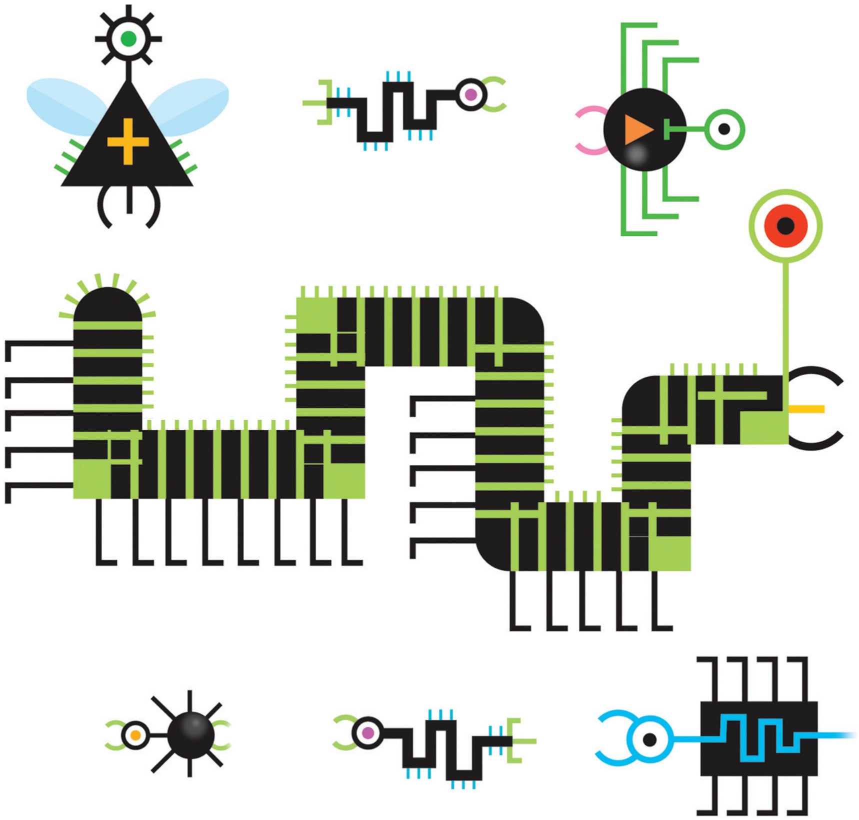
Recalling malware milestones.

ANNIVERSARIES PRESENT AN opportunity to reflect. Sometimes we celebrate anniversaries (birthdays, graduations, some marriages), and sometimes we grieve (deaths, disasters, and some marriages). There are also anniversaries when we compare what might have been against what has actually happened (all the above, and more).

Consider:

▸ The first use of the term "computer virus," occurred 40 years ago in *Venture Magazine*, in a science fiction story by Gregory Benford, involving computer code and a corresponding vaccine program.[a] Benford's friend, David Gerrold,[b] later incorporated these ideas into his novel, *When HARLIE Was One*.

▸ Next year is the 25th anniversary of the first widespread PC (MS-DOS) computer virus (known as Brain, Lahore, or Pakistani). By 2000 there were 40,000 families (code variations) of viruses for Microsoft-based operating systems; a few score viruses existed for other systems, including the Macintosh, Amiga, and Atari systems.[c]



▸ Last November was the 21st anniversary of the Internet Worm[d] that brought malicious software (malware) to the fore after it spread in part of the early Internet over several days. Next year is the 10th anniversary of its conceptual descendent, Nimda, which affected hundreds of thousands of Windows systems worldwide in less than 30 minutes.

By 2005, malware existed that spread by Web pages, email, and other network services. "Blended" threats were common, including components spread by inadvertent user activation. Malware developers quickly overcame new defenses as they were devised, deploying alteration of OS functions, code to disable security mechanisms and antivirus programs, and self-modification to foil pattern-based detection. Some malware applied vendor software patches to prevent other malware

[a] Personnel communication, later confirmed in a letter to the editor of the *New York Times*, published in 1994.

[b] David Gerrold is perhaps best known to many as the author of *The Trouble With Tribbles* story that was made into a much-beloved "Star Trek" episode.

[c] Early history of computer viruses can be found in many references, including "Virus" by E.H. Spafford in *Internet Besieged: Countering Cyberspace Scofflaws*; D. Denning and P. Denning, Eds., Addison-Wesley, 1997.

[d] See *Communications of the ACM 19*, 1 (Jan. 1989) for several analyses and views.

ILLUSTRATION BY GORDON STUDER

from displacing it: ironically, that malware performed better at maintaining systems than their owners!

Malware now includes "social engineering" components to entice the careless, unprotected, and unwary. Phishing, botnets, cross-site scripting and SQL injection have become commonly known terms. There have been many notorious uses of malware, including political action in Estonia, supporting military actions against the country of Georgia, and spying on human rights activists and the Dali Lama.

Early malware was developed for bragging rights or out of curiosity; today's malware is often written by criminals—including organized crime—to commit fraud, distribute spam email, obtain identity and account data, and steal proprietary commercial information. Malware-generation tools have proliferated, including some posted online for anyone to use. Globally, annual losses from malware may total in the tens of billions of dollars (or more)—and how do we put a price on the loss of national defense information, or the safety of activists opposed to oppressive regimes?

Tens of thousands of new instances of malware appear daily,[e] although it is impossible to get a precise count because of their often-polymorphic nature: a "new" version is created each time a download occurs. Of those, only a fraction is detected because of built-in stealth techniques and poor security practices by the victims. Current malware may remain without detection indefinitely (the APT or Advanced Persistent Threat; see http://www.wired.com/threatlevel/2010/02/apt-hacks/), and some botnets whose origins cannot be traced may include millions of compromised hosts (for example, Conficker).

Early malware was developed for bragging rights or out of curiosity; today's malware is often written by criminals.

The science fiction story of 40 years ago is now a scourge causing huge global losses and evolving as a new tool of organized aggression. The public is beginning to realize what specialists have known for years: these problems are getting worse. How did this happen? And what can we do about it?

Factors

In no particular order, some of the most notable factors contributing to the proliferation of malware have included:

- Software is usually produced using error-prone tools and methods, including inadequate testing. Well-established security principles are ignored—if the developers even knew about them. Too many people believe narrow "secure programming"[f] approaches are the solution, and equate penetration efforts with security assurance;
- The market often rewards first-to-sell and lowest cost rather than extra time and care in development;
- Vendors favor producing large, all-in-one products to minimize development and marketing costs, but these have larger attack surfaces and more options to misconfigure and misunderstand;
- Vendors pursue upgrades and new releases as a means of maintaining revenue streams, but backward-compatibility and new features both contribute to new vulnerabilities;
- Customers in industry and government have placed more emphasis on acquisition cost than on total cost of operation, risk, and quality;
- Feature lock-in (product and training compatibility) coupled with a lack of good metrics on security and safety have hindered innovation and competition;
- Insufficient diversity enables "write-once, run everywhere" attacks;
- The end user is burdened with the costs and responsibilities for dealing with malware, leading to a culture of "add-ons" for security and skewed expectations;
- Periodic software patching and production use of beta products are viewed as the norm rather than as unusual exceptions;
- Law enforcement has not been given sufficient resources, support, or prioritization to pursue malware authors and operators;
- Research has been funded mostly to respond to current threats rather than to devise disruptive but safer replacements for current systems; and
- Issues involving pricing and licensing software in a diverse, global marketplace have led to numerous, unauthorized copies that may be ineligible for patches, and whose operators cannot afford security add-ons.

Remediation

We can reduce the malware problem by actions on four major fronts.

Economics. The economics of security need to be changed. This includes increasing our understanding of the long-term risks and cost-effectiveness of security-related choices to enable better choices by system owners and operators; reducing the barriers to competition that might lead to safer products such as by embracing vendor-neutral, open standards to improve portability; and reexamining those parts of regulatory and intellectual property regimes that interfere with research and (re)use of sound security features. Judicious use of rewards and penalties for product quality might help. Changes to liability protections for vendors, ISPs, and end users could also encourage more proactive actions by all involved.

Milieu. The public needs basic education about good security and privacy practices to make better-informed choices. Where private owners cannot afford necessary upgrades or services to "disinfect" and reconfigure their systems, public "computing health" organizations should be created: contaminated clients are a threat to the community as a whole. Although not without their own problems, some uses of virtualization and software as a service (SaaS) present opportunities for migration of end users away from poorly maintained systems.

[e] Personal communications from Vesselin Bontchev, John Thompson, and John Viega.

[f] "Secure programming" is writing code without certain features that have been frequently exploited.

There must be a change in the attitude that end users are solely responsible for their systems' security. Customers are not to blame that systems are shipped without appropriate safeguards, nor should they be forced to buy and maintain a large (and growing) set of additional protections to use their systems safely. Additionally, everyone should learn that patching a system is not security, and penetration testing is no substitute for proper design and development.

Technology. As a field, we should reexamine construction of smaller, more protected systems and applications. Known, effective techniques such as putting code in read-only devices, code whitelisting, integrity monitoring, and better separation of privileges could all play a role if used integrally rather than as add-ons. Tools, programming languages, and platforms in use should also be reexamined from the perspective of how to build functional, safe systems cost-effectively rather as instruments perpetuating legacy decisions. Test methods, including some that were previously considered to be too complex to be practical, should be reconsidered given our continually advancing capabilities.[g]

Law. Most malware is a law enforcement issue, not a military one; it is *cybercrime*, not *cyberwar*. Police need tools, trained personnel, authority, and a clear mandate to pursue the authors and operators of malware. This will require a concerted international effort—but the trends are clear that people in every country are at risk if effective actions are not taken. Perhaps, with some creativity, approaches other than traditional criminal statues might be employed, akin to using tax law violations to convict Al Capone. Authors and operators of malware presented with a significant risk of substantial penalties might instead choose to pursue more legitimate professions.

Current and past methods employed against malware have perhaps slowed the growth of the problem but certainly have not stopped it.

Conclusion

It has taken decades for computing to evolve into the current worldwide infrastructure. Malware and automated attacks have also been evolving, and the result is an increasing, usually unnoticed drag on our innovation and economy. We are now at a point where it is becoming an existential issue for some companies and even governments.

Current and past methods employed against malware have perhaps slowed the growth of the problem but certainly have not stopped it. If we simply continue to do more of the same we will continue to be victimized, and the problem will get worse. The longer we wait, hoping that piecemeal and uncoordinated responses will be enough, the more difficult (and expensive) it will be to address the problems when we finally attempt to do so.

Change requires resources, will, and time. We do not need to do everything everywhere at once—but we do need to start. Unfortunately, some of those who are in the best positions to make changes are also under the most pressure to defer change precisely because it requires resources and disruption of the status quo. It is up to all of us to facilitate the changes that are needed—before too many more anniversaries pass us by.

Eugene H. Spafford (spaf@cerias.purdue.edu) is a professor of computer science and the executive director of the Center for Education and Research in Information Assurance and Security (CERIAS) at Purdue University.

[g] This is a special case of what I described in "Rethinking computing insanity, practice and research" available at http://snipurl.com/rethinking.

Calendar of Events

August 16–17
Creativity and Innovation in Design,
Aarhus, Denmark,
Contact: Christensen Bo,
Email: bc.marktg@cbs.dk

August 16–20
Designing Interactive Systems Conference 2010,
Aarhus, Denmark,
Contact: Olav W. Bertelsen,
Email: olavb@cs.au.dk

August 18–20
International Symposium on Low Power Electronics and Design,
Austin, TX,
Sponsored: SIGDA,
Contact: Vojin G. Oklobdzjia,
Email: vojin@ece.ucdavis.edu

August 19–20
International Conference on Intercultural Collaboration 2010,
Copenhagen, Denmark,
Sponsored: SIGCHI,
Contact: Anne-Marie Soederberg,
Email: ams.ikl@cbs.dk

August 25–27
European Conference on Cognitive Ergonomics,
Delft, Netherlands,
Contact: Neerincx Mark,
Email: mark.neerincx@tno.nl

August 30–September 3
ACM SIGCOMM 2010 Conference,
New Delhi,
Sponsored: SIGCOMM,
Contact: Shivkumar Kalyanaraman,
Phone: 518-782-7875,
Email: shivkuma@gmail.com

August 31–September 4
International Conference on Distributed Smart Cameras,
Atlanta, GA,
Sponsored: SIGMM, SIGBED,
Contact: Marilyn Claire Wolfe,
Phone: 404-894-5933,
Email: wolf@ece.gatech.edu

September 1–3
Symposium on Solid and Physical Modeling,
Haifa, Israel,
Contact: Anath Fischer,
Phone: 972-482-93260,
Email: mereanath@technion.ac.il