Title: Designing User Incentives for Cybersecurity

Authors: Terrence August[1], Robert August[2], Hyoduk Shin[3]

Affiliations:

[1]Assistant Professor, Rady School of Management, University of California, San Diego, La Jolla, CA 92093; Visiting IlJin Professor, Korea University Business School, Seoul, Korea, 136-701.

[2]Associate Professor Emeritus, School of Business and Leadership, Our Lady of the Lake University, San Antonio, TX 78207.

[3]Assistant Professor, Rady School of Management, University of California, San Diego, La Jolla, CA 92093.

Main Text: The traditional "patching" approach to managing software vulnerabilities and cybersecurity risk has been less effective than desired. In theory, once a vulnerability is discovered, software patches should be quickly developed and released by producers and then expeditiously applied by users. Successful completion of this process would help to maintain secure systems. However, what has been consistently observed in practice is that this process instead breaks down (*1*). Of particular concern is the failure of the current approach to adequately address the economic incentives that underlie users' decisions to patch their systems. We propose a simple adaptation to software producer offerings ("versions") involving users' patching rights and argue why this change would make a patching approach more effective.

In particular, we advocate that users should be charged for the *right* to control patching of their own individual copies of software. That is, a user's ability to choose whether to install patches should no longer be an implicit right. Rather, a user who prefers to retain the ability to choose whether and when to patch would need to pay a certain price; one can think of this fee as the price to cause additional security risk (perhaps temporarily) which has negative consequences on other users and the software producer. A user who chooses to forgo this right, and not pay the premium, has his or her system automatically updated with security patches as soon as they are released. Framed differently, a user can choose whether to purchase a "discounted", default version that is automatically updated or a "premium" version that includes the right to patch at one's convenience or even not at all.

Although on the surface this prescription is simple, the design of an incentives-based approach to improve cybersecurity is a difficult task because the level of risk that realizes on a given system or network is a complex outcome of the behaviors of many stakeholders: government, critical infrastructure providers, technology producers, malicious ("black hat") hackers, and users. With regard to users, how has the current approach failed? For more than forty years, malware has posed a significant challenge to cybersecurity. The commercialization of the Internet in the 1990's exacerbated this situation tremendously as millions of users began connecting vulnerable devices to networks. Today, the situation is even more problematic. It is common to find an end user with a desktop computer, laptop, tablet, and smart phone running both system software and

applications that are not secure. The problem is not limited to companies like Microsoft, who has released more than 80 security updates to its Windows 7 operating system. There are already one billion smartphones in use globally and this number is expected to double by 2015 (*2*). Many of these devices are running the Android operating system which is being modified by the carriers. These devices are so insecure that the American Civil Liberties Union (ACLU) has filed a complaint with the Federal Trade Commission (FTC) requesting action be taken to require vendors to provide more frequent and timely updates to their systems (*3*).

But a patching approach to mitigate cybersecurity risk will only work if users actually apply the patches. Historically many users have lacked sufficient incentives to properly maintain computing systems by applying these patches. As an example, Code Red is notorious malware that attacked instances of Microsoft's Internet Information Services (IIS) web server software by exploiting a particular vulnerability. Microsoft had developed a security patch for this vulnerability and released it roughly three weeks before Code Red erupted. Yet, even with the technical fix being made publicly available, most users failed to patch their installations in time, and nearly 360,000 servers were struck by the worm (*1*). Code Red is not an exception. Similar patching windows existed in the cases of other high-profile attacks by SQL Slammer, Blaster, and Sasser. To effectively address the security problem, one must understand how to incentivize improved user behavior.

Users are heterogeneous in the total value they derive from deploying and protecting a system running a given software product. For example, one can consider the differences between two users of enterprise application software: an organization that highly values a system enabled by the software and an organization that values the software just enough to purchase it. A prudent approach to patching involves costly activities including testing, staging, and a controlled roll-out of patches to production systems. Because of the higher value the former organization derives from the software (e.g., perhaps the software supports an e-commerce website), it also possesses stronger economic incentives to incur these patching costs and protect its systems. In contrast, the organization that derives less value has much weaker incentives. Not only would such an organization likely not allocate resources to follow the extensive patching process described above, even potential inconvenience costs associated with a patch failure may sway it to not deploy patches in a cursory manner either. Although the example with two types of users described above is informative, in reality, there is a continuum of users varying in their patching preferences. But, the point remains: some users tend to have less incentive to patch while other users who utilize software for mission-critical purposes have stronger incentives to patch and protect their systems at the expense of time and resources.

Unfortunately, cybersecurity risk is characterized by detrimental network externalities. That is, when users behave insecurely on a network, they increase the risk faced by everyone else connected to the network *(4, 5*). Nevertheless, users in the segment who choose not to patch can still be targeted by economic incentives that align their protection decisions more closely with the objective of keeping systems secure. The approach we propose is for software producers to version their products based on users' patching rights. One can logically reason through how this approach would affect different types of users. Organizations and other high value users will opt for premium versions and pay for the right to patch according to their own timeframes, thereby causing additional security risk on others while remaining unpatched. However, because of their inherent incentives for patching, most of these users will indeed patch at the end of their internal, systematic processes. For users who value the software enough to use it but not enough to protect

their systems, they will now be forced to decide whether the right to remain unpatched in a network environment is worth paying the price. Because they may likely belong to the more price-sensitive segment, many of them will prefer not to pay the premium and instead have their systems automatically updated.

Considering that people make daily decisions on whether to install security updates for operating systems, web browsers, productivity tools, antivirus software, and a long list of other application software, the impact of well-crafted incentives that target user behavior can be quite substantial. Will some consumers prefer discounted versions of popular products like Microsoft Windows and Microsoft Office that automatically update, no longer include the right to be unpatched, and remove the hassle of deploying patches? The answer is likely “yes”, and if such a versioning strategy can convert a sizable percentage of unprotected systems from the status quo, then cybersecurity can improve considerably.

Given all of the stakeholders involved, our recommended approach to version based on patching rights also faces counterarguments. Users may contend that they should be endowed with the right to choose whether to patch. By not having this right, they would be forced to incur inconvenience costs associated with loss of control, system rebooting, and even system instability due to poorly designed patches. These concerns are valid and underscored by the not-so-distant memories of Windows XP Service Pack 2. However, we argue that the social concern should tilt in favor of improved security. In fact, similar trade-offs exist in other settings where it is common for “users” of products and services to be required to protect others’ interests. For example, most states require that children be immunized before being permitted to attend school. Similarly, many states have requirements that vehicle owners regularly have their vehicles inspected and possibly corrected to meet emission standards. In our case, we are advocating an even milder approach. Users may always retain the ability to choose whether to patch – mandates are not necessary. What is important is that users internalize the cost of causing greater security risk as is reflected through a higher price for retaining patching rights.

Software producers may push back on our recommended approach because they will either lose customers at the low end due to these inconvenience costs or be forced to cut prices to keep them as paying users. While this may be true, producers may also find versioning on patching rights to be beneficial for several reasons. First, organizations and other high end users will derive greater value from more secure software. If the unpatched population shrinks considerably, these organizations will bear less security risk during the time it takes them to undergo their patching processes. For this lower risk, software vendors can charge a higher price that helps make up for revenue losses associated with the user segment that forgoes patching rights. Second, the often employed argument by software producers that they should not be held liable for security vulnerabilities because they make patches available is somewhat fragile if patches are not being applied (*6*). By providing better economic incentives such that the patching approach they subscribe to actually leads to more secure outcomes, software producers can strengthen their arguments against governmental intervention through means such as liability. There is an opportunity for future research to formally examine these trade-offs using economic models of the decision problem faced by software producers; such research can yield useful and important insights.

One noteworthy risk of our approach concerns the reaction of black hat hackers. Like other economic agents, black hats have typically found it more cost effective to reverse engineer security patches and develop attacks to exploit the vulnerabilities these patches aim to fix. In this

sense, black hats are leveraging the fact that users' lack of incentives to deploy patches leads to large exploitable populations. Our recommended approach would likely force black hats to redirect their efforts toward other endeavors such as finding unknown vulnerabilities and exploiting them with zero-day attacks. What is important to note is that by their revealed attack preferences, these endeavors appear to be costlier for the same economic return on effort. Hence, the extent of their efforts may be partially reduced. Nevertheless, zero-day attacks can cause considerable economic losses. In anticipation of how black hats respond, both organizations and end users will necessarily need to adjust their defense-in-depth strategies such that the economic gains from a world with versioning on patching rights are not overshadowed by losses in the "equilibrium" state that arises. More research studies that endogenize black hat behavior can help to better predict the actual outcomes.

The government would likely be a strong advocate of the approach we outline. Several federal agencies including the National Science Foundation (NSF) have recognized that innovative policies are needed to help reduce security risks currently faced by the United States (*7*, *8*). Along with President Obama's executive order to develop a cybersecurity framework, the Department of Commerce was also directed to determine what types of economic incentives will cost-efficiently help facilitate adoption of the framework and whether additional legislation may be required (*9*). The government seems to implicitly favor a *voluntary* approach toward improving cybersecurity. For example, whether software producers should be held liable for the economic losses incurred by their users due to poor security has been heavily debated over the last decade, but with little legislative action taken by the government (*10*, *11*). In spirit, the idea is that holding a company like Microsoft liable will ultimately hurt its bottom line and thus finally provide incentives for greater investments in making its products more secure. This outcome may indeed be the case. But, other undesirable outcomes can certainly arise instead. In particular, Microsoft may make strategic choices to limit its liability. One way to do so is to serve fewer users because a smaller network of users corresponds to reduced security risk due to the network externality. Specifically, all software users benefit in terms of security when there are fewer users exhibiting insecure behaviors, e.g., not protecting their individual systems. Under a liability policy, Microsoft would, in turn, benefit by not paying out as much to cover users' losses. If this latter effect of a liability policy is strong, Microsoft may in fact reduce its investments and/or raise its prices to achieve a smaller user population (*12*).

Instead, an approach where software producers begin versioning their products based on patching rights seems to strike a balance across the interests of government, software producers, and users. Unlike government-imposed liability, this approach is more consistent with how the government has thus far attempted to nudge stakeholders toward better cybersecurity outcomes. Furthermore, targeting user incentives to protect their machines can be a more direct and effective approach in comparison to liability schemes that software producers would prefer to avoid. In fact, if producers are able to charge higher prices from users who appreciate the increased security and it thereby leads to increased producer profitability, there is the potential for win/win outcomes that also substantially improve the economic value associated with software to society.

References and Notes:

1. D. Moore, C. Shannon, J. Brown, Code-Red: a case study on the spread and victims of an Internet worm. ACM SIGCOMM/USENIX Internet Measurement Workshop, 273-284 (2002).

2. J. Yang, Smartphones in Use Surpass 1 Billion, Will Double by 2015. Bloomberg (2012).

3. R. Satter, ACLU: Slow smartphone updates are privacy threat. Associated Press (2013).

4. R. Anderson, T. Moore, The Economics of Information Security. *Science*. **314**, 610-613 (2006).

5. T. August, T. Tunca, Network Software Security and User Incentives. *Management Science*. **52**, 1703-1720 (2006).

6. T. Espiner, EC wants software makers held liable for code. ZDNet (2009).

7. Department of Defense, *Department of Defense Strategy for Operating in Cyberspace* (2011; www.defense.gov/news/d20110714cyber.pdf).

8. National Science Foundation, *Secure and Trustworthy Cyberspace (SaTC) Program Solicitation NSF 12-596*, (2012; www.nsf.gov/pubs/2012/nsf12596/nsf12596.pdf).

9. B. Obama, "Executive Order - Improving Critical Infrastructure Cybersecurity" (The White House, Office of the Press Secretary, Washington, D.C., 2013).

10. D. Ryan, Two Views on Security Software Liability: Let the Legal System Decide. *IEEE Security & Privacy*. **1**, 70-72 (2003).

11. C. Heckman, Two Views on Security Software Liability: Using the Right Legal Tools. *IEEE Security & Privacy*. **1**, 73-75 (2003).

12. T. August, T. Tunca, Who Should be Responsible for Software Security? A Comparative Analysis of Liability Policies in Network Environments. *Management Science*. **57**, 934-959 (2011).

Acknowledgments: This material is based upon work supported by the National Science Foundation under Grant No. CNS-0954234.