Analyzing Security Costs

Quantification tools, if applied prudently, can assist in the anticipation, budgeting, and control of direct and indirect computer security costs.

Costs related to computer security are often difficult to assess, in part because accurate metrics have been inherently unrealistic. Of those costs that can be measured, the largest in terms of monetary value typically involve theft of proprietary information or financial fraud. Others that are more difficult to quantify but have resulted in severe loss of use or productivity include viruses and malware, Web server denial-of-service attacks, abuse of access privileges, and equipment vandalism or outright theft. We see the results of surveys of organizations providing estimates as to breach incidents (supposedly affecting 90% of large corporations and government agencies in 2002, according to the Computer Security Institute), security expenditures (projected at more than $3 billion in 2004 by International Data Corp.), malicious code (worldwide loss estimates by Computer Economics exceeded $13 billion in 2001 alone), and so on, with numbers continuing to reflect dramatic growth each year. However, lacking any way to translate such statistics into expenditures and losses per organization, per computer, or per user, the true impact of these figures remains uncertain.

Industry traditionally has seemed willing to write off some level of computation service downtime and loss of access to (or even misuse of) data and equipment as a "matter of course," but as such services and information have become increasingly critical to business and everyday operations, this casual attitude may soon be unacceptable. Yet, in the absence of methodologies for calculating actual values, many insurers have begun to exclude certain computer loss coverages or have made such policies prohibitively expensive, especially for smaller businesses and individuals. Hence, the establishment and use of appropriate quantification tools can allow those who rely on computers to better anticipate, budget, and control their direct and indirect security costs. These metrics might be derived, said Dan Geer at the 2003 DIMACS Workshop on Software Security, from "traditional accounting models, quality assurance literature, public health reporting structures, portfolio management processes, accelerated failure testing, and insurance projections." Geer, the CEO of @stake, a digital-security consulting firm, asserts that "even a poor estimator can expose trends" that may be helpful.

One of the earliest used estimators in the computer industry was Annual Loss Expectancy (ALE), a quantitative method for performing risk analysis. It was described in a 1979 FIPS publication (#65) by the National Institute of Standards and Technologies as appro-

SERGE BLOCH

Even when security protection mechanisms are provided for free (as in the case of automatically downloaded software patches), there are other costs that may be incurred.

priate for use by large data centers. Calculation of a risk estimate was produced by multiplying the estimated frequency of occurrence of attacks by the possible loss amount for each data file, and then summing these results. The method was criticized because of the "lack of empirical data on frequency of occurrence of impacts and the related consequences" thus producing an interpretation of "results as having more precision than they actually had" [6]. Nevertheless, the ALE figures may still provide some useful information.

More recently, Cost-Benefit Analysis (CBA) techniques have become the most popular metrics applied to the assessment of computer-related risks. CBA is well established in microeconomic and management accounting theory, and can be used to determine estimated levels of expenditures appropriate to the values of assets requiring protection. Victor Hazlewood of the San Diego Supercomputer Center finds it particularly convincing since "most managers and directors know little about computers and computer security, but they do understand risk and cost-benefit analysis." Roger Clarke, at Australian National University, explained that CBA is application independent, involves identification and measurement of all related costs and benefits, should include "lost opportunity" costs from diversion of resources, needs to account for shared costs and uses, and must consider and address risks, uncertainties, qualitative factors and assumptions [5]. It is important to note, though, that where appropriate care is not taken, results may be grossly over- or underestimated.

The National Institutes of Health (NIH) has a useful guidance document for preparing CBAs as required by the U.S. Federal government to support IT management decisions [8]. This thorough guide addresses appropriate allocations of cost for preparation of the CBA itself, procedures to be followed for performing the CBA, determination of life cycles, collection of cost data, estimation of tangible and intangible benefits, evaluation of alternatives, and other salient aspects. Although the NIH document does not specifically pertain to security, many of the IT topics and examples discussed are highly relevant, so it is worth a close look.

An example of early CBA use in computer security is in the I-CAMP (Incident Cost Analysis Modeling Project) model developed by the Big Ten Universities during the 1990s. Factored together are the time, wages, overhead, and direct costs related to the resolution of individual security incidents. Person-hours are logged, typically for incident investigation, system administration, and recovery efforts, and then salary-weighted sums (including benefits) are computed. Necessary direct expenditures (such as for replacement hardware, software, and analysis tools) are also added. The I-CAMP model is appropriate for situations where the related usage losses are considered to be modest or ignored entirely. At universities, for example, student research time is generally deemed expendable, customer service is usually irrelevant, and deadlines are typically flexible. For smaller colleges, I-CAMP may not be as effective, since it can be difficult even to ascertain the per-incident costs, especially where sysadmin allocation is performed on an ad hoc basis.

In the business arena, CBA equations are necessarily more complex. Here, the models incorporate the use of risk-adjusted cash flows in order to examine internal rate of return (IRR) and maximum net present value (NPV) figured as a percentage of information security expenditures. Gordon and Loeb [7] explain that a simple return on investment (ROI) calculation that divides income by asset value is insufficient because it is based on historical rather than future valuations as affected by

breach incidents. They use weighted annual expected loss estimates derived by multiplying the dollar value associated with potential breaches by the probability of occurrence for each breach. But they note that even the IRR and NPV metrics may be deficient because these compare the actual cost savings from the security investment to the anticipated cost savings, which "is difficult because the benefits of specific investments aren't easily separated from other activities within a company. This is particularly relevant to security investments, the more successful the project, the less likely you are to see breaches."

Recalling Y2K is illustrative here, since the computer industry was later criticized for fear-mongering to run up costs when nothing catastrophic happened at the turn of the century, rather than being complimented for averting the many problems that would likely have occurred had timely corrections not been put into place. Gordon and Loeb also observe that "as security investments increase, there's a strong reason to believe the net benefits from preventing breaches may initially increase but will eventually decline." They suggest that "firms should invest substantially less in information security than the expected loss from security breaches" because of this ROI drop-off. By using breach probabilities "the notion of deriving an optimal level of information security investment" can be pursued, but as observed with the ALE technique, such probabilities may be hard to determine.

Even when security protection mechanisms are provided for free (as in the case of automatically downloaded software patches), there are other costs that may be incurred. Researchers on a DARPA-funded project [3] developed "a mathematical model of the potential costs involved in patching and not patching at a given time." They observed that the risk of loss of functionality from applying a bad patch decreases in time, while the risk of loss due to penetration while the patch is not applied increases with time. They hypothesized that the optimal time to apply the patch is when these curves cross, and developed a mathematical model (similar to the weighted ROI) that took into account various cost and probability factors. Using data collected from a study involving 136 patches, they were able to determine that at 10 and 30 days following a patch release, application is optimal. Of course, these intervals rely on some folks applying the (potentially bad or even bogus) patches sooner and reporting the defects they experienced—if everyone waits for the patches to be fixed, the time would be shifted forward, thus increasing early penetration risks.

One potential misuse of CBA is in its application to public-key cryptography in order to derive appropriate key sizes and expirations. Robert Silverman, of RSA Laboratories, asserts that a financial model, rather than a purely computational one, should be used to assess cryptographic vulnerabilities. He says that "it makes no sense for an adversary to spend (say) $10 million breaking a key if recovering the key will only net (say) $10 thousand" [9]. To demonstrate this, Silverman analyzed the time and hardware requirements used by the General Number Field Sieve method that was successful in cracking RSA-130, RSA-140, and RSA-512. Projecting these numbers against Moore's Law produced an estimate of feasibility, in other words, the point where it becomes cost-effective to break an n-bit key. Based on these calculations, Silverman believes 1024-bit RSA or Diffie-Hellman keys will be secure for at least 20 years, although SRI's Peter Neumann points out this analysis is flawed since it is based only on key attacks whereas inappropriate embedding is more frequently the weak link in cryptosystems.

The cost-benefit model has also been effective in assessing network intrusion detection systems. Here, it can be used "to periodically review the effectiveness of planned and implemented security controls to determine if they are doing what they are supposed to do, rather than creating additional vulnerabilities" [10]. Such a scheme involves first performing a risk analysis that produces a cost matrix for the assets under attack, and then independently calculating damage, response, and operation costs for those assets. Resources to counter the attack can be classified as low, medium, or high, in terms of price, and weighted by amounts

of use where appropriate, to obtain total expenditures. Probabilistic models also include false negative and false positive costs, since these may have an impact on losses.

Risk analysis can be thwarted by cultural filters through which a view of real and virtual dangers is presented, warns John Adams, a professor of geography at University College of London [1]. Individuals who self-select into security professions may be those biased toward greater risk avoidance than are members of the general population, and thus might be prone to overestimate potential losses or willing to authorize greater prevention expenditures than actually necessary. For example, aviation homeland security measures authorized in the wake of the Sept. 11 terrorist attacks may seem appropriate in terms of managing potential terrorism risks, but might inadvertently result in unsustainable economic losses to the airline industry. The Brookings Institution addresses this issue, saying "if private incentives are not aligned with the public good in a compelling way, the results can be catastrophic" [4]. Hence, interdependencies and perceptions can complicate or obfuscate the establishment of the risk matrix necessary to perform a CBA.

This is why Cambridge University's security economics expert Ross Anderson characterizes computer security as a "hard" problem from a financial perspective. Anderson (whose Web site and paper referenced at [2] are essential resources) observes that current motivations favor self-protection over prevention for the good of the whole. On the Internet, this explains the trend toward proliferation of localized firewalls and spam filters rather than ISP-based controls. Anderson notes that the high costs of new technology creation and technology switching can serve as disincentives to security product deployment. He criticizes the Common Criteria security evaluation process (as I have with the NASED voting system certifications) as being flawed in an economic sense, because the vendor (rather than the buyer) pays for the testing, and this potentially opens the door to fraud and dishonesty. Anderson concludes "the real driving forces behind security system design usually have nothing to do with … altruistic goals. They are much more likely to be the desire to grab a monopoly, to charge different prices to different users for essentially the same service, and to dump risk."

Although business process concerns must be addressed, it is hoped that empiricism can shift the balance in favor of the consumers of computer security products and services. Those "add-ons" and providers that do not demonstrably improve the security cost bottom line should be exposed and no longer supported. New tools and metrics that enable risk and cost-benefit assessments need to be developed and proliferated. Only through such independent quantification can we hope to get a true handle on the financial ramifications of security problems so that we might best direct our efforts toward resolving them. C

References

1. Adams, J. *Cars, Cholera, and Cows: The Management of Risk and Uncertainty*. Cato Institute, March 1999.
2. Anderson, R. *Why Information Security is Hard—An Economic Perspective.* Sept. 2001; www.cl.cam.ac.uk/~rja14/econsec.html
3. Beattie, A. et al. Timing the application of security patches for optimal uptime. In *Proceedings of LISA '02: Sixteenth Systems Administration Conference*, USENIX Association (Nov. 2002).
4. Brookings Institution. *Interdependent Security: Implications for Homeland Security Policy and Other Areas.* Policy Brief #108, Oct. 2002.
5. Clarke, R. Computer matching by government agencies: The failure of cost/benefit analysis as a control mechanism. *Information Infrastructure and Policy 4*, 1 (Mar. 1995); www.anu.edu.au/people/ Roger.Clarke/DV/MatchCBA.html
6. Federal Information Processing Standards. *Guideline for the Analysis of Local Area Network Security.* National Institute of Standards and Technology, FIPS PUB 191, Nov. 1994; www.itl.nist.gov/fipspubs/fip191.htm
7. Gordon, L.A. and Loeb, M.P. Return on information security investments: Myths vs. realities. *Strategic Finance Magazine* (Nov. 2002); www.strategicfinancemag.com/2002/11i.htm
8. Office of the Deputy Chief Information Officer. *Cost-Benefit Analysis Guide for NIH IT Projects.* Center for Information Technology, National Institutes of Health, May 1999; wwwoirm.nih.gov/itmra/cbaguide.html
9. Silverman, R.D. A cost-based security analysis of symmetric and asymmetric key lengths. *RSA Laboratories Bulletin 13* (Apr. 2000).
10. Wei, F. et al. Cost-benefit analysis for network intrusion detection systems. In *Proceedings of the CSI 28th Annual Computer Security Conference*, (Oct. 2001).

Rebecca T. Mercuri (mercuri@acm.org) is a systems architect and the president of Notable Software, Inc., Princeton, NJ.