Quantitatively Differentiating System Security

Stuart Schechter
Harvard University
stuart@post.harvard.edu

May 17, 2002

1 Introduction

Security is not considered a priority by developers of shrink-wrap systems because without a means to accurately and understandably measure it, security fails to provide a competitive advantage. I assert that the cost to break into a system is an effective metric, that this metric can be measured from the start of testing until product retirement, and that using this metric to differentiate products will provide developers with the competitive advantage needed to lead the industry to more secure systems.

2 Willingness to Pay for Security: Why Metrics are Key

Investments in securing shrink-wrap systems have been hard for developers to justify as they have failed to provide the necessary return. Claims that security failures are externalities[1, 2] have shifted the blame to consumers' willingness to pay for security. However, strong sales of security products, such as firewalls, show that consumers are willing to pay for security when they understand the benefits.

Though consumers often comprehend the benefit of having a security product in place when compared to not having one at all, they have been unable to differentiate the value provided by the competing security properties of complex systems. Even if consumers are willing to pay for more secure systems, choosing a system based on its security properties is difficult. This is not a failing of the consumer, as even industry experts rarely have little more than crude heuristics available to them to compare the security of competing products. Reliable and understandable metrics for system security are needed.

Nor is the absence of metrics or the lack of differentiation of security the result of collusion. The software market is dominated by a single player with competitors who are desperately looking for means of differentiating their products.

Attempts have been made to use the number of security features, such as strong encryption, as a metric for differentiating security. These prove futile as all products quickly copy these features and all products continue to be promoted as being the most secure. The resulting feature escalation may even result in lower security as time that should be spent shoring up security is instead spent adding inadequately tested security features to make products appear to be more secure.

Differentiating without a metric can backfire, as Oracle learned after heavily promoting its products as ”unbreakable” only to see new vulnerabilities reported.[3]

3 The Metric: Cost To Break (CTB)

The *Cost To Break* (CTB) of a system is the lowest expected cost for anyone to discover and exploit a vulnerability in that system.

This metric is not new. To evaluate the strength of a cryptosystem, cryptographers approximate the costs that would be incurred by an adversary to break the system by buying the latest (cheapest) tools and using the best known techniques. The result is meaningful because it allows us replicate the cost/benefit analysis of the adversary. From such an analysis one might posit that a system that can't be broken for less than $1000 is strong enough to protect $100 from an adversary seeking financial gain.[1]

RSA is responsible for one of the earliest attempts to measure CTB by economic means. It offers rewards of up to $200,000 for breaking the security of the keys used by its cryptosystem.[3] This reward placed a reassuring lower bound on the CTB of the cryptosystem, possibly playing a role in industry-wide adoption of the standard despite the existence of systems with more rigorous cryptographic assumptions.

Camp and Wolfram[2] also propose to measure CTB by economic means. They describe a means for creating a market for vulnerabilities in both running system installations and shrink-wrapped software to increase the security of systems. They contend that security vulnerabilities are negative externalizes and that government intervention, issuing a new currency in the form of credits for security vulnerabilities, will create incentives to make systems more secure.

While security vulnerabilities are an excellent definition of a good, comparing consequences of vulnerabilities in different types of products is quite difficult, especially before the vulnerabilities are actually found as would be the case when defining rules for a new currency. Camp and Wolfram acknowledge this problem and propose that the consequences of a vulnerability might be measured by the processing power of the machine broken into. Network bandwidth or information inside the system may be more important. Given that the consequences and implications of a vulnerability will always be specific to the attributes of the running system (hardware, software, and data), it may not be possible for an efficient currency to be created. What's more, governmental introduction of

[1] Assuming the adversary can't amortize the cost of the attack over ten or more thefts.

economic penalties to individuals who fail to secure their personal computing devices would likely be deemed excessively intrusive, making such a proposal politically infeasible.

Government intervention shouldn't be necessary to create markets for security vulnerabilities. The observation that will bootstrap the use of CTB as a metric is that it can be measured without the consent of the system's manufacturer. A firm can determine the CTB of both its own system and of a competitor's system. Once it has done so, the competitor must either improve the security of its system or surrender the security battle.

4 How To Differentiate Using Cost To Break

An upper bound can be placed on a competing product's CTB by offering a reward for the first report of a vulnerability with an exploit. Assuming that if a tester is willing to find and report an exploit in the competitor's system to you at the reward price, a criminal could have purchased the exploit from that tester for the same price, it follows that the competing system's CTB can be no higher than the amount you paid. It is likely even lower.

A lower bound on the CTB of your system can be placed by offering a reward for each and every unique security vulnerability reported, and repairing each one until the reports cease. Fixing price and demand for exploits ensures that anyone who believes they can supply (find) a vulnerability at a cost less than the reward price will do so. It also ensures that the discoverer of a vulnerability will make more money by reporting it to you than selling it to a criminal who is offering a lower price. As long as the developer has fixed all known vulnerabilities, it can claim that the cost to break its system is at least the reward amount.[2]

By proving that the CTB upper bound of the competing product is less than the lower bound of its own, a firm shows its product is more secure. However, this claim only holds so long as the competitor has yet to fix the exploit that was found in its system.

If your competitor always has access to the exploits you purchase and fixes them, you'll be paying your competitor's costs to secure its system to compete with yours. To prevent this from happening I propose the introduction of trusted third parties. These parties must be trusted by you not to reveal the defect to your competitor and be trusted by the consumer you wish to sell your system to. The role of the third party is to verify that you indeed purchased a exploit that will penetrate the competitor's system and that you paid the price you claim. In addition, you need to enter a confidentiality agreement with the seller to ensure he doesn't resell the exploit.

[2] One might worry that a tester would report the exploit to multiple buyers at a price lower than your reward price. In this case, an arbitrage opportunity exists in which a buyer can purchase the exploit and report it to you. In order to make this possible, you must allow exploits to be reported without revealing the identity of the reporter.

5 Paying Less to Find Vulnerabilities

When placing a lower bound on CTB, offering a reward above the true CTB will cause the firm to pay too high a price for a potentially large number of vulnerabilities. Using a fixed reward is not a cost effective means of finding an initial estimate of CTB.

An innovation is to estimate CTB by raising the reward over time and inviting the public to test the product. Not only can CTB be estimated by the price paid to find the most recent vulnerabilities, but the software developer need not overpay for finding these vulnerabilities. The tester with the lowest cost of finding a vulnerability can not wait until the reward is greater than the cost for another tester to find the vulnerability, lest that tester would step in and do so. A detailed description and analysis of this technique will be provided in concurrent and future work.

6 Conclusion

Omnipotent adversaries don't break into systems, people do. The proper way to measure the security of a system is to find out how hard it is for real people to break. Encouraging anyone with the talent to break your system to do so is the only way to reliably measure its security. Using this measure, products can be differentiated by the quality of their security, developers will finally have an incentive to build more secure systems, and they'll have a new tool to help them do so cost effectively.

7 Acknowledgements

This paper could not have been completed without the advice, comments, and suggestions from Adam Brandenburger, L. Jean Camp, David Molnar, Edward Naim, David Parkes, Michael D. Smith, and Jason Woodard.

References

[1] Ross Anderson, "Why Information Security is Hard, An Economic Perspective", *17th Annual Computer Security Applications Conference*, (December 2001).

[2] L. Jean Camp & Catherine Wolfram, "Pricing Security" *Proceedings of the CERT Information Survivability Workshop*, Boston, MA Oct. 24-26, 2000, pp. 31-39.

[3] David Litchfield, "Hackproofing Oracle Application Server", http://www.nextgensss.com/papers/hpoas.pdf

[3] http://www.rsasecurity.com/rsalabs/challenges/factoring/numbers.html