Making the Business Case for Software Assurance

Nancy Mead

February 2007

ABSTRACT: It is essential to be able to make a cost/benefit argument in order to justify investment in software assurance during the software development process. Although we are making some strides in identifying costs, quantifying the benefits can be much more elusive. In this article we give an overview of the Business Case content area.

THE STATUS QUO

As software developers and software managers, we all know that when we want to introduce new approaches in our development processes, we have to make a cost/benefit argument to our executive management to convince them that there is a business or strategic return on investment. Executives are not interested in investing in new technical approaches simply because they are innovative or exciting. For profit-making organizations, we need to make a case that demonstrates we will improve market share, profit, or other business elements. For other types of organizations we need to show that we will improve our software in a way that is important—in a way that adds to the organization's prestige, that ensures the safety of troops in the battlefield, and so on.

In the area of software assurance, particularly security, we have started to see some evidence of successful ROI or economic arguments for security administrative operations, such as maintaining current levels of patches, establishing organization entities such as CSIRTs to support security investment, and so on [Blum 2006, Gordon 2006, Huang 2006, Nagaratnam 2005]. Initially there were only a few studies [Soo Hoo 2001, Berinato 2002, Jaquith 2002] that presented evidence to support the idea that investment during software development in software security will result in commensurate benefits across the entire life cycle. This picture has improved, however. As we expected early on, Microsoft has published data reflecting the results of using their Security Development Lifecycle [Howard 2006]. Microsoft is using the level of vulnerabilities and therefore the level of patches needed as a measure of improved cost/benefit [Microsoft 2012a]. The reduced level of patches/vulnerabilities in recent Microsoft product releases is remarkable. In addition, Microsoft has recently published a white pa-

Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-2612

Phone: 412-268-5800
Toll-free: 1-888-201-4479

www.sei.cmu.edu

per describing a method of calculating ROI for investments in security during the development life cycle [Microsoft 2009].

We would also refer readers to the Business Context discussion in Chapter 2 and the Business Climate discussion in Chapter 10 of McGraw's recent book [McGraw 2006] for ideas. There has been some work on a security-oriented version of COCOMO called COSECMO; however, the focus has been more on cost estimation than on return on investment. Reifer is also working in this area on a model called CONIPMO, which is aimed at systems engineers [Reifer 2006]. Data presented by Fortify indicates that the cost of correction of security flaws at the requirements level is up to 100 times less than the cost of correction of security flaws in fielded software. COCOMO data suggests that the cost of fixing errors of all types at requirements time is about 20 times less than the cost of fixing errors in fielded software. Regardless of which statistic is used, there would seem to be a substantial cost savings for fixing security flaws at during requirements development rather than fixing security flaws after software is fielded. For vendors, the cost is magnified by the expense of developing and releasing patches. However, it seems clear that cost savings exist even in the case of custom software when security flaws are corrected early in the development process.

At this time there is little agreement on the right kinds of models to be used for this purpose, and although there is now case study data that supports the ROI argument for investment in software security early in software development, there is still very little published data.

On the other hand, Bruce Schneier believes that it is not feasible to accurately calculate the benefit that is derived from improved security. He points out that there is very little actual data on the cost of a break-in and that predicting the cost of a rare but damaging event is fraught with peril. In this article he summarizes his position on calculating security ROI as follows: "It's a good idea in theory, but it's mostly bunk in practice."

Our belief is that even though they may not constitute a traditional ROI argument, the methods being used to calculate cost/benefit, whether they be reduced levels of patching in the field or reduced cost of fixing security flaws when they are found early in the lifecycle, are convincing.

WHAT WE CAN OFFER

Given this state of affairs, we are unable to recommend a single model for calculating cost/benefit for early investment in software security during software de-

velopment. Instead, we find that we are able to describe a variety of models that may be useful for calculating software assurance valuation. In the article by Shoemaker et al., Models for Assessing the Cost and Value of Software Assurance, we present models that could be considered by organizations who are thinking of investing in software assurance. Another article by Shoemaker et al., A Common Sense Way to Make the Business Case for Software Assurance, provides a practical approach for arriving at a cost/benefit argument. The article by Arora et al.,Estimating Benefits from Investing in Secure Software Development, specifically addresses a way of estimating cost and benefits associated with improved security.

DEVELOPMENTS

In 2008 we conducted a Workshop on Business Case and all indications are that at present there is no single common model that is widely accepted. Many issues and approaches were presented at the workshop, which had more than 70 attendees internationally from industry, government, and academe.

As noted above, Microsoft is using the level of vulnerabilities and patches needed as a measure of improved cost/benefit, and data presented by Fortify is using the comparative cost of correction of security flaws at various points in the software life cycle.

In 2009 we published a guide for "Making the Business Case for Software Assurance." Although there is no single model that can be recommended for making the cost/benefit argument, there are promising models and methods that can be used individually and collectively for this purpose, as well as some convincing case study data that supports the value of building software assurance into newly developed software. These are described in the guide.

REFERENCES

[Berinato 2002] Berinato, S. "Finally, a Real Return on Security Spending." *CIO Magazine (Australia)*, August 4, 2002.

[Blum 2006] Blum, D. *Making Business Sense of Information Security, Security and Risk Management Strategies*. Burton Group, Version 1.0, February 10, 2006.

[Gordon 2006] Gordon, L. A. & Loeb, M. P. "Budgeting Process for Information Security Expenditures." *Communications of the ACM 49*, 1 (January 2006): 121-125.

[Howard 2006] Howard, Michael & Lipner, Steve. *The Security Development Lifecycle*. Redmond, WA: Microsoft Press, 2006.

[Huang 2006] Huang, C.D.; Hu, Q.; & Behara, R. S. "Economics of Information Security Investment in the Case of Simultaneous Attacks." *The Fifth Workshop on the Economics of Information Security (WEIS 2006)*. University of Cambridge, England, June 26-28, 2006.

[Jaquith 2002] Jaquith, Andrew. *The Security of Applications: Not All Are Created Equal* (@atstake Security Research Report) (2002).

[McGraw 2006] McGraw, Gary. *Software Security: Building Security In. Upper Saddle River*, NJ : Addison-Wesley, 2006.

[Microsoft 2009] Microsoft Corp. *Microsoft SDL: Return on Investment*. September 15, 2009.

[Microsoft 2012a] Microsoft. The Microsoft Security Development Lifecycle (SDL): How SDL Builds More Secure Software, 2012.

[Microsoft 2012b] Microsoft. The Microsoft Security Development Lifecycle (SDL): SDL Helps Reduce the Total Cost of Development, 2012.

[Nagaratnam 2005] Nagaratnam, N.; Nadalin, A.; Hondo, M.; McIntosh, M.; & Austel, P. "Business-driven application security: From modeling to managing secure applications." *IBM Systems Journal 44*, 4 (2005): 847-867.

[Reifer 2006] Reifer, Donald. "CONIPMO Workshop." Practical Software Measurement Conference, Vail, CO, July 24-28, 2006.

[SooHoo 2001] Soo Hoo, K.; Sudbury, A. W.; & Jaquith, A. R. "Tangible ROI through Secure Software Engineering." *Secure Business Quarterly 1*, 2 (Fourth Quarter 2001).

This material is based upon work funded and supported by Department of Homeland Security under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center sponsored by the United States Department of Defense.

DM-0001120