The Principle of Distributed Accounting: Misalignment of Security Interests

John McHugh*
CERT/CC® Software Engineering Institute
Carnegie Mellon University
jmchugh@cert.org

29 march 2002

No matter how thin you slice it, it's still baloney – Alfred E. Smith[1]

Efficiency and politics often conflict. In many cases, the centralized provision of a generally useful service attracts unwanted attention to the provider who is forced to justify the provision of the service and its attendant costs. One way that the unwanted attention can be avoided (without actually changing anything) is to distribute the costs among the community that benefits and then subsidize or reimburse that community. This will raise the total costs by, at least, the overhead rates of the beneficiaries and the bookkeeping costs incurred by all players, but it removes a (possibly contentious) line item from the providers budget. While I suspect that this has been going on for most of recorded history (or at least since accountability came into vogue) I first became aware of it when DARPA distributed the cost of ARPANET access to its users, requiring them to pay the service providers directly, and then allowed the cost to be billed, as an expense, on its contracts[2]. Although the cost to DARPA increased, the expenses were distributed over a large number of contracts and became effectively invisible.

I call the principle behind such acts, *"The principle of distributed accounting."* In its purest form, the actual provision of the service in question is diverted so that it is no longer obvious that the primary beneficiary is still (indirectly) providing it and the costs are effectively laundered through the distribution. It is the side effects of this behavior that have security implications.

*This work is supported by the Department of Defense

[1] Governor of New York 1918–1920, 1922–1928, Democratic presidential candidate 1928 (lost to Herbert Hoover).

[2] Ironically, it may be this distribution that led to the Internet as we know it today. While the initial result of the payment distribution was the replacement of a line item in the DARPA budget with a more that equivalent amount of distributed direct costs, the cost eventually became an indirect, or overhead, item as the network was used to support both DARPA and non-DARPA projects. Subsequently, NSF supported the connection of numerous academic institutions to the growing network. The providers of the service became free to sell access to all comers and by the early 1990s numerous commercial users were on the net. With the rapid growth of low cost workstations and personal computers in the 1990s, network access exploded. By this point, DARPA was no longer a significant contributer to the costs of operating the Internet and effectively lost any hope of control.

Centralized payment gives the payer control over the service. When the payment is diluted, the beneficiary loses control and other factors become more important. The Internet as it exists today has no single supporter or group of allied supporters who have sufficient financial interest to shape policy. Although many users, perhaps, even a majority of users, claim that, at some level, security is important to them, none are able to influence it directly, and it is not clear that many would be willing to pay higher costs to improve overall security.

In general, it appears that the costs associated with improving Internet security and the benefits that would accrue are misaligned. For example, a large percentage of the security events reported to CERT are enabled by poor software engineering practices that range from sloppy implementation practices (buffer overflows, etc.) to design omissions (lack of synchronization enabling race condition exploits) to poor human factors work (systems insecurely configured "out of the box" and excessively difficult to configure securely). Just as the automobile industry in the 1950s claimed that "Safety doesn't sell.", software developers, for the most part, do not see a return on investment from efforts to make their software less vulnerable[3]. In other areas of commerce, product liability law exerts a stabilizing influence, but the software industry has largely been immune to these pressures[4]. While reducing software vulnerabilities will not eliminate all security problems, eliminating the majority of today's enabling vulnerability classes would raise the barrier substantially. The expected result might well be fewer, but more professionally executed attacks, but for most users this would be a definite improvement.

This improvement is unlikely to occur unless the financial interests of the software developers can be aligned with improved security. This is likely to come about only through legal or regulatory pressures. While the later is unlikely in the current political climate[5], there is a growing movement to hold software developers responsible for the consequences of their software flaws. This trend may be accelerated by a movement to hold the owners of compromised platforms used in attacks responsible for the attack (even if they had no knowledge of the compromise) by claiming that they were negligent in failing to secure their machine. One can see the operator of such a platform attempting to shift the liability to the developer of the compromised software in turn.

In summary, the current situation distributes the costs associated with achieving a more secure Internet in such a way that those who stand to benefit from improved security are largely disjoint from those who would have to spend to achieve it. Unless costs and benefits can be aligned, the situation is unlikely to change.

[3]Microsoft claims to have "gotten religion" in this area recently. Only time will tell if this actually changes their behavior.

[4]Attempts at legislation such as UCITA indicate an interest by the software community in institutionalizing the status quo.

[5]This might change if releasing software with security flaws were to be considered an act of terrorism.