Chapter #N

IMPROVING INFORMATION FLOW IN THE INFORMATION SECURITY MARKET

DoD Experience and Future Directions

Carl E. Landwehr
University of Maryland

Abstract:

Key words:

1. WHY DO PRODUCTS LACK SECURITY?

The market for information security has long been seen as dysfunctional [1]. Although there have been significant investments in research in techniques to improve information security (under various names, including computer security, information assurance, network defense, and so on), relatively little of that research is reflected in the products found in the marketplace or in the methods used to develop them. Why?

First, simply defining what "secure" means for some application or system is not an easy task. An intuitive notion of how a system is expected to behave, and not behave, rarely provides a sufficient basis to make any strong statement about the relative security or insecurity of a system.

Even if they have a clear understanding of the security they expect from a product, customers have a hard time knowing if they are getting it when they make purchase decisions. The features of a product are relatively easy to see and test; assessing the ability of the product to resist attack or abuse is much harder. Secure and insecure versions of the similar systems may behave indistinguishably except under attack.

Security properties are also unstable under system change. Even a small change in a system can make a big difference in its vulnerability, particularly if the system has not been constructed with security requirements in mind. Imagine a system in which a single bit may disable or enable a protection mechanism such as a virus scanner: flipping that bit clearly makes a major difference in vulnerability but little difference in observable behavior. The changes made in a typical software patch or system upgrade are of course much more substantial.

Security is also a system property. Two secure components connected inappropriately will make an insecure system. Assessing the security of individual components is hard enough; reasoning about entire digital systems, unless they are structured for this purpose, is extremely difficult.

Finally, system security depends on more than technology. Even systems in which significant resources have been devoted to security engineering may be abused by operators or users who are too trusting of others or too untrustworthy themselves.

Faced with these difficulties, and offered the choice between a system displaying a rich set of features, though cobbled together under the surface, and a system that provides only a few functions, albeit solidly implemented, is it surprising that for many years buyers have chosen sizzle security?

2. U.S. DEFENSE EFFORTS TO BRING SECURITY INFORMATION TO THE MARKETPLACE

2.1 Early Years

Security was a strong factor in the early history of electronic computers – the computations that motivated their development, such as decrypting intercepted messages, generating gunnery tables and developing weapons, had military applications. But the computers themselves were so big and so few that their computations were relatively easy to protect simply by limiting physical access to the machines. Further, users often shared these early machines sequentially, so there was relatively little opportunity for one user's computation to affect another's.

As multiprocessing and then timesharing of computers was developed in the 1960's, assuring the separation of different users' computations became more important, so that a single user could not bring down a system or steal another user's data. In this period, commercial and military concerns about computer security diverged. Commercial concerns naturally focused on the flow and protection of financial assets. The desire to prevent, detect, and prosecute commercial fraud motivated both security policy and technology development in this area and typically led to controls on application-level programs, so that only authorized individuals could invoke certain operations, and to the generation and preservation of audit trails so that potential fraud could be identified and the perpetrators identified, prosecuted, and convicted.

Military information security concerns at this time focused primarily on preserving the confidentiality of sensitive information. Computers were few, large, and expensive; it made economic sense to share them among users and applications. Yet leakage of sensitive information from a highly classified application to uncleared users might compromise an expensive intelligence collection system or a particular military operation and have a tremendous cost in dollars and lives; further, the compromise might not be detected.

These facts dictated a focus on preventive measures that would withstand a determined attack by a capable opponent. The notion of building systems that could protect sensitive information from a Trojan horse program – one which had full access to sensitive data and would try to export it without arousing suspicion – arose in this context. Although computer and software vendors sometimes asserted their systems could provide the kind of isolation desired by the military, when subjected to attack, their systems failed the test.

Military investment subsequently fueled much of the research and development in this area, though perhaps a declining fraction over the past decade or so, as computers have become critical to so many functions throughout society. The focus of this research, particularly in the early years was most often on securing the lower levels of the infrastructure – the operating systems, for example, rather than the applications, both because of the diversity of military applications and the belief that securing the applications without securing the infrastructure would be like building on a foundation of sand.

2.2 The DoD Strategy for Improving Computer Security through the Market

In the early years, the U.S. Department of Defense (DoD) often built its own computers and operating systems to suit its particular needs. By the late 1970's and early 1980's, U.S. government officials, and in particular, Stephen Walker of DoD, correctly recognized that, for cost reasons, the military would increasingly be driven to base its information systems on commercial, off-the-shelf, computers and software. These systems clearly lacked the properties sought by DoD to enforce its security policies. Yet the research already conducted under DoD sponsorship seemed to indicate that if vendors could be persuaded to pay more attention to security issues as they developed their systems, providing the structure and "hooks" needed for more secure modes of operation, commercial platforms might in fact provide an adequately secure base for DoD applications. Then DoD might simply acquire a high-security version of a commonly available product, taking advantage of the economies of scale. But how could DoD bring about this happy state of affairs in the commercial marketplace?

The strategy that Walker played the key role in developing was, in effect, one of trying to make better information about product security available to consumers. A set of criteria would be developed that could be used to evaluate the security of commercially offered computer systems. Products would be evaluated against these criteria and the results published. Once consumers (and in particular DoD acquisition program managers) could easily understand which systems had stronger security and which had weaker, they could make intelligent choices about which to buy for systems with security requirements. DoD could also make it a policy to purchase systems that achieved higher security ratings. The potential for increased sales in the defense market would give vendors the needed incentive to invest the presumably marginal added development cost needed to provide a base for higher security versions of their systems. In the best case, even non-defense sales would improve for systems with higher ratings and the market would produce better quality products for everyone.

This seemed a rational strategy, one based on improving the information available to the marketplace. It offered both the carrot of increased sales for systems with good evaluations, and the stick of an impartial, *Consumer Reports*-like mechanism to evaluate product security. The government had also followed a somewhat similar strategy in identifying equipment that met other kinds of security requirements in its TEMPEST Preferred Products List, an Endorsed Crypto Products List, and a Deguasser Qualified Products List.

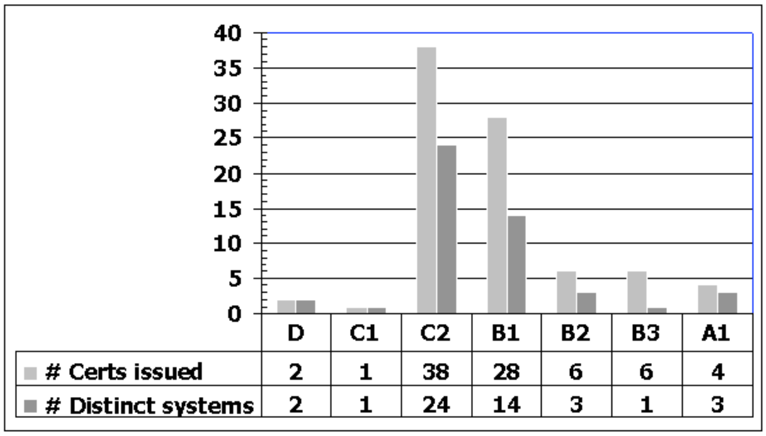
2.3 Implementing the Strategy

Major investments were needed to implement this strategy. The evaluation criteria had to be developed, a major effort in itself. The National Computer Security Evaluation Center (originally planned by Walker to be at the National Bureau of Standards, but ultimately created at the National Security Agency in 1981) was created to draft the criteria and subsequently evaluate products against it. The criteria became a book length document, the Trusted Computer System Evaluation Criteria (TCSEC), first published in 1983. Its cover color soon supplanted its lengthy title, and it was universally known as "The Orange Book." It defined seven ordered classes of overall product security ranging from level A1 (highest) through B3, B2, B1, C2, C1, to D (lowest). To meet the criteria for a given level, a product had to provide both an increasing set of security features such as audit functions, access control functions, information labels, and an increasing amount of evidence that the system correctly implemented the specified functions. This assurance came both from system documentation and increasing test requirements. Evaluators were required to review specified documentation and, depending on the evaluation class sought, assure that it corresponded accurately to the system as implemented. At the highest class specified (A1), no additional functions were required over the prior class (B3), but formal methods of specification and verification were to be applied to increase assurance that the system would behave as intended.

As added carrots to developers, the government agreed to bear the cost of evaluations (though the developer was responsible for developing the needed documentation), to evaluate only products submitted to it, and to conduct evaluations under nondisclosure agreements that would prevent detailed evaluation information from flowing to other parts of the government. The criteria took a good deal of expertise to apply, so the government also had to develop and train a workforce that was up to the job.

2.4 Experience

The first Orange Book evaluations were completed in 1984. Two business-oriented add-on access control packages, RACF-MVS and ACF2-MVS/SP achieved ratings of C1 and C2 respectively, and Honeywell's Secure Communications Processor (SCOMP) achieved an A1 rating. The SCOMP had been specifically developed in response to DoD requirements and its evaluation was actually underway as the Orange Book was being written. In 2000, the final TCSEC evaluation was complete: Sybase Adaptive Server Anywhere v. 7.0 achieved a C2 rating. In the 16 years from 1984-2000, a total of 85 evaluation certificates were issued; 29 more evaluations were initiated but not completed. The profile of systems evaluated over the period is shown in Figures N-1 and N-2. In many cases, after a system is evaluated, it is changed or updated, and the changed system must be re-evaluated; this is the basis for the distinction between number of certificates issues and number of distinct systems. Evaluations are now conducted under the international Common Criteria framework by a set of private, government-certified laboratories, addressed later in this paper.



	D	C1	C2	B1	B2	B3	A1
■ # Certs issued	2	1	38	28	6	6	4
■ # Distinct systems	2	1	24	14	3	1	3

Figure #N-1. TCSEC evaluations completed 1984-2000. Data from www.radium.ncsc.mil/tpep/epl

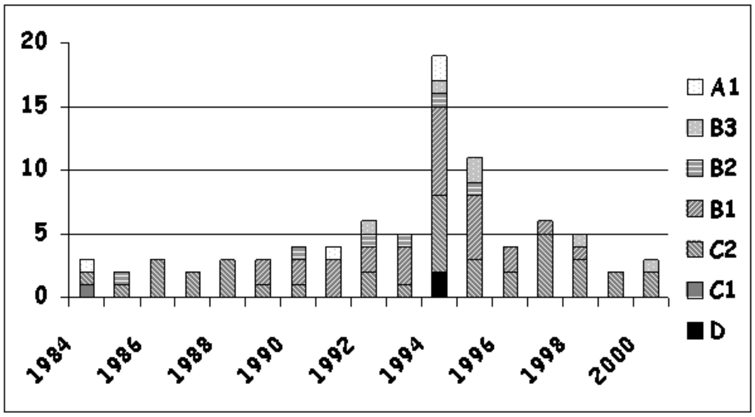


Figure #N-2. TCSEC evaluations completed by year and class 1984-2000. Data from www.radium.ncsc.mil/tpep/epl

This attempt to leverage market forces had some successes, but failed ultimately to have the effect its creators intended, for several reasons.

The evaluation process proved expensive and time-consuming. As noted, evaluations required expertise on the part of evaluators and developers. They were expensive for both the government and the developers, requiring substantial documentation to be generated by the developer and reviewed by the evaluators. Neither party felt in control of the resulting delays: because the process was voluntary, neither side could impose a schedule on the other.

The evaluation criteria were relatively abstract and interpretations had to be developed for different kinds of components. These generated a "case law" of interpretations that sometimes led to protracted discussions over "criteria creep" when vendors felt they were being subjected to more stringent interpretations than had been applied to earlier systems. Because the evaluation classes bundled assurance and features, they didn't apply well to high assurance devices with simple, specific functions.

There were significant startup problems in enforcing the intended procurement policy. In general, government procurements are required to be competitive. Initially, there were few evaluated products available. So, a procurement that required a product meeting, say, the B1 evaluation class might effectively designate a single supplier, thereby violating government procurement rules for open competition. Instead of having an advantage, the vendor to first achieve an evaluated product might be penalized, in effect, for having invested the resources needed to obtain it. Ultimately, officials controlling the procurements demanded the latest operating systems and features as long as there was some evidence of intention to have the product evaluated eventually; this gave vendors an incentive to start the evaluation process, but not necessarily to complete it.

Product changes and upgrades were difficult to accommodate. To keep up with advancing technology and competing products, and even to correct flaws, vendors need to update their systems regularly. Any change to a system would require its re-evaluation. In the end, a "ratings maintenance program" (RAMP) was developed to deal with this problem.

Systems, rather than products, are frequently the significant unit of procurement. Governments often procure systems, not single products. The evaluation criteria proved difficult or impossible to apply to systems – yet security is fundamentally a system property. During this time period, networking of systems became increasingly important, both to system function and to system security. Although a "network interpretation" of the criteria was developed, it was not effective.

The government's market leverage declined. As the commercial market for computers boomed, the government's share of that market declined substantially, reducing government leverage overall, and private purchasers did not in general perceive similar security needs.

In the end, the promised the carrot of lucrative government procurements of evaluated systems never really materialized for most of the vendors who participated in the program. The investment required of the vendors in order to meet the evaluation criteria, in dollars but more importantly in development time, proved more than they could justify economically.

3. GLOBALIZATION

A few years after the Orange Book was published, Canada, the UK, and several European countries began developing and adapting their own evaluation criteria and mechanisms. These related efforts eventually led to the Common Criteria that are in use today. These criteria provide a flexible means for specifying security functions and levels of assurance relatively independently. This flexibility imposes a corresponding specification burden, however. An independent laboratory, paid by the developer, performs the evaluation (and in many cases, a separate part of the same laboratory is paid by the same developer to produce the documentation to be evaluated). Government participation, though at a much lower level, is still required to certify the evaluation practices of the commercial laboratories. Although this process has shifted the financial burden of evaluations from government to industry and accelerated the speed of evaluations, it is not clear that it is contributing greatly to improved security in delivered products, particularly at the lower assurance levels.

4. CONCLUSIONS AND FUTURE DIRECTIONS

Rational market decisions depend on good information. All of the attempts to provide security evaluation criteria and to evaluate products can be seen as efforts to improve the flow of information in the market for secure computer systems.

The slow rate at which information about the security properties of products is generated impedes improvements in security of deployed systems. More precisely, the lack of specific information about the ability of specific components and system architectures to preserve information availability, integrity, and confidentiality in the face of failures and attacks, and the difficulty of developing this information quickly, is a strong factor in the current generally poor state of computer system security in many widely distributed computer systems

Experience shows that information about computer security properties is hard to obtain. The properties are not only difficult to specify and quantify, they are time-consuming to evaluate. Though asymmetric information may be a factor in this market, in that a seller may know more about the security properties of his product than the buyer can, in many cases even the vendor lacks full knowledge of his product. Because it takes significant time and energy to extract the information needed to support rational decisions, the buyer commonly faces a choice between new, unevaluated products or systems using the fastest hardware and providing the latest features, but with uncertain security properties, and older products with better known properties but poorer performance.

One way to improve the flow of this kind of information would be to seek measures and assessments of product security that can be obtained quickly and easily. Perhaps security could be viewed as a “hidden variable” and researchers might look for related exposed variables that could be assessed more quickly. For example, a developer might be able to certify that a piece of software is not subject to buffer overflow problems. The concept of proof-carrying code also help: some basic properties of the code can be certified immediately before it is executed.

One might also consider providing tools to help evaluate open source software. Security information about proprietary software can take longer to develop because only the proprietor has unrestricted access to the code and so the decision of whether to apply resources to security analysis of it is constrained. Opening source permits anyone who cares to apply resources to this task to do so [2]. DARPA's Composable High Assurance Trusted Software (CHATS) program funded some efforts to encourage security review of open source software [3].

Other kinds of information, beyond the internal properties of components or systems, are lacking from the security marketplace as well. These include reliable information on actual system behavior, actual security incidents, and actual losses. Other mechanisms are needed to foster bringing this kind of information, which is often considered sensitive by the parties who control it, to the marketplace, but may be releasable in aggregate form [4].

Market pressures can indeed influence vendor behavior. In January 2002, the dominant company in the software industry changed course significantly by announcing an initiative in "Trustworthy Computing" and, according to its own statements, has since invested hundreds of millions of dollars in trying to improve the security engineering in its software development processes and its products. This behavior was apparently triggered by a perception that the continuing stream of security incidents facilitated by features and flaws in their products would eventually deter buyers. Nevertheless, it remains difficult for their customers to make valid comparisons among different products.

Better information will make a better market for computer security. We need to explore how to bring that information to decision makers efficiently.

REFERENCES

1. National Research Council, System Security Study Committee, CSTB, *Computers at Risk*, National Academy Press, 1991. Chapter 6, "Why the Security Market Has Not Worked Well", pp.143-178. Also available at www.nap.edu
2. B. Witten., C. Landwehr, and M. Caloyannides. Does open source improve security? *IEEE Software 18*, 5, (Sept. 2001), 57-61.
3. Sardonix website: http://www.sardonix.org
4. D. Geer. Information security: why the future belongs to the quants*.* *IEEE Security & Privacy, 1*,4 (July/August 2003), 24-32.