New Directions in Information Protection

by Dr. Frederick B. Cohen ‡

Abstract

Recent developments in the area of 'secure' systems have been based on the assumption that we can attain perfection, and as a result, have concentrated on systems designed to be perfect. Unfortunately, we cannot be perfect in the design and implementation of protection systems without placing restrictions so severe as to make those systems useless for general purpose use. Hence, it is the thesis of this paper that we have been going the wrong way. In this paper, we propose a fundamental change in the way we pursue technical protection.

search terms: Information Protection, Access Control, Privacy, Integrity, Trusted Systems, Password Protection, Authentication, Auditing, Computer Security, Synergism

‡ This research was funded by ASP, PO Box 81270, Pittsburgh, PA 15217, USA

1 Background

Secure systems research and development has gone through a significant evolution in recent years because of a fundamental shift in the way we implement protection. Instead of adding protection onto an existing operating system by trying to 'plug the leaks', we have begun to implement protection deep in the design process. [1] This has resulted in a dramatic improvement in assurance, ease of implementation, reliability, security, and performance.

Unfortunately, along with this technological improvement, we have not made corresponding improvements in understanding the nature of the protection problem and how to solve it. For example, the 'Trusted System Evaluation Criteria' [2].

- It is widely used both for evaluating systems and for specifying them is based on protecting secrecy, but basically ignores integrity and availability.
- It assumes that we can achieve perfection and attempts to require proofs that we have done so, while simultaneously requiring an accounting for phenomena related to such imperfections as covert channels [3].
- It completely ignores many of the most pressing protection issues we encounter, such as computer viruses [4,5]
- It uses 20 year old policies [6,7] based on principles that have serious computational problems [8] and assumes military specific models, when there are simpler and more general models available [9,10] which cover the previous models and provide more insight into to problems we face.

The assumption that we can attain perfection in information protection flies directly in the face of strong scientific evidence to the contrary. For example:

- Any system that allows 'rights' to be altered over time has the potential problem of yielding uncontrollable configurations leaving the protection state undecidable, [8,11] and leads to time-transitivity of information flow which is generally ignored [10] in even the most 'secure' implementations.
- Covert channels, when combined with computer viruses, yield systems which cannot protect secrecy in practice dispite the 'mathematical proofs' of their soundness, because the assumptions in those proofs ignore fundamental issues. [3–5,12]

- In order to have a system which limits viral spread, we must limit sharing, transitivity of information flow, or functionality [4,5], none of which are done effectively in these 'secure' systems, and each of which makes current methods of computer use infeasable.
- Even the most rudimentary principle of being able to identify a user has basic limitations due to information and secrecy theory [12,13,14], and the common assumption of physical security is clearly untrue in almost every case in the current computing environment.

If we are to attain a reasonable degree of protection on any widespread basis at a reasonable cost and in the current computing environment, we must rethink our basic principles and allign them more accurately to the theoretical and practical realities of our computing environment. It is the basic thesis of this paper that we do so by 'taking computational advantage', and it is our further contention that this is the only practical method available to us today. To assess the value of protection, we must seek to understand the issues of complexity as they relate to protection technologies and characterize the costs associated with attack and defense with these measures.

In the remainder of this paper, we will explore the principles and current applications of taking computational advantage. We will begin by defining what we mean by 'taking computational advantage' a bit more elaborately and giving many examples of how this technique is used today as a basis for protection, even though it is not known under that name. Next we will explore recent advances in information protection which explicitly exploit this technique to dramatically increase the complexity of attack. Finally, we will summarize results, draw conclusions, and look into the possibilities of expanding its role in information protection to cover many other technical aspects of protection.

2 Taking Computational Advantage

When we speak of taking computational advantage, we generally mean exploiting the ability of a set of actors to solve a problem when in posession of some special knowledge or capability over the ability of a set of actors to solve the same problem when not in posession of that knowledge or capability. In other words, we are exploiting some sort of computational leverage given by the posession of something.

The most obvious case of this technique is the use of passwords as a means for authentication. In this case, the computational advantage comes from the legitimate actor's knowledge of the password, the illegitimate actor's lack of knowledge thereof, and the size and shape

of the space in which guessing is required in order for the illegitimate actor to gain the knowledge of the legitimate actor.

Since the legitimate actor requires only a memory recall in order to provide the password, the amount of computation required for the legitimate actor is quite small, while an attacker must either guess the password by some sort of trial and error, observe the password as it is entered or stored, determine an acceptable password by reversing the techniques used by the system to verify its propriety, or bypass the mechanism used to authenticate. In each case, the complexity to the attacker may be far greater than that of a legitimate actor, and so we say that we are taking computational advantage.

A fundamental issue that remains to be settled it the degree to which we actually attain a computational advantage through the use of some particular technique. Here we run into a serious problem, in that we need to find some model for the system in order to analyze it. In the case of password analysis, the best model we seem to have is based on Shannon's information theory [12]. Under a 'syntactic' analysis of passwords, we take the degree of surprise associated with each password in the password space, and associate this with the expected number of guesses required to guess the password. By factoring the time per guess into the equation, we can get an expected time to guess the password. We can then try to model the costs associated with using different password spaces and get a model of attack time versus costs for this aspect of password usage.

Given this result, we can then perform similar modeling and analysis on the other aspects of password protection. For example, we can find a way to model the complexity of an attacker observing a password as it is entered or stored, model the determination of an acceptable password by reversing the techniques used by the system to verify its propriety, and model bypassing the mechanism used to authenticate. The result will be a fairly complex, but hopefully accurate model of the costs versus attack difficulty associated with password use.

Given this result, we then extend the model further to consider the impact of password guessing on a system as a function of the manner in which the system operates. For example, particular large grained mandatory access control schemes limit the extent of both corruption and leakage attained by entry into a particular domain. If guessing a password grants access to some number of domains, we can then assess the potential damage due to this entry from legitimate use. To remain accurate, we must also consider the complexity associated with bypassing the access control mechanism and factor the likelihood of that into our calculation of the impact of password guessing on systemic damage.

3 Synergistic Effects

As we continue down this path, it becomes apparent that all of the protection mechanisms in a system interact in a very complex manner, and that these complex interactions appear as synergistic effects when we only examine protection techniques in isolation. When we assume that mechanisms are perfect and independent, we greatly simplify the analysis process, but we lose a significant piece of the picture.

The degree of significance of what is abstracted out becomes apparent when we look at an example, like the exploitation of covert channels by a computer virus. In this case, even the best available systems designed to prevent the leakage of secrets can be made to leak secrets at a high rate. [3,4,5] Before the publication of the results on computer viruses, it was widely assumed that these systems were very nearly invulnerable without a classified user acting as an attacker, even to the extent that mathematical proofs of their safety were given. After this development, it became clear that privacy without integrity is infeasible if we allow sharing, transitivity, and general purpose function.

Clearly, a theoretical basis that collapses with the ease of the one in place in 1984 leaves a lot to be desired, particularly if we intend to rely on it for protection decisions in systems where protection failures are substantive. Furthermore, any new basis that is to be practical in light of synergistic effects will likely have to either directly model the entire space or model synergistic effects adequately to cover the space. Two major impediments to this effort are the fact that the protection space is both poorly defined and based on a concept called 'protection by extension'.

Solving the lack of adequate definition requires that we find a sound and appropriate basis for specifying protection objectives. At this time, we only have the rudimentary and poorly defined concepts of 'privacy', 'integrity', 'availability', and 'accountability'. Each of these is constantly redefined in the linguistic sense, and only a small number of mathematical efforts have been put forth to define them or combine their definitions and understand the ramifications of their use and combination.

The use of protection by extension is closely related to our mathematical understanding of protection policies. We take basic policy concerns and attempt to derive techniques which we hope will address them. In the same way, we try to implement these techniques by extending physical protection mechanisms. This is called 'protection by extension' because it attempts to expand the boundaries of protection provided by physical techniques to cover logical boundaries. We run into difficulties here because the theoretical basis of the extension may be unsound, the implementation may be imperfect, and the physical mechanism may be inadequate for the purpose. For example, over 90% of the world's computers are currently

IBM based personal computers with virtually no physical protection capability. All of the effective non-hardware based protection systems for these machines operate by extending an unsound mechanism, and can thus be easily bypassed.

Even though these system can be 'easily' circumvented, that does not make them ineffective or inadequate for some tasks. For example, the vast majority of PC users know little or nothing about how computers operate, and it may be adequate to our goals to have minimal protection provided the users don't have the motive or understanding required to bypass it. Again, the effectiveness of the technique is related to the complexity of attack given a particular amount of knowledge and effort.

4 An Analogy to Cryptography

Our discussion is closely related to recent progress in the field of cryptography, and we would be remiss if we did not point out the similarities. The only 'perfect' cryptographic system we know of is the 'one-time-pad' [14], which is perfect in the information theoretical sense in that no information is available to an attacker after observing any number of messages that was not already available to the attacker before observing those messages. In certain applications this technique is practical, but because it requires a bit of shared secret key for each bit of message, it is impractical in many other applications.

For 'imperfect' systems, Shannon also points out the concept of a workload. [14] The workload concept is that even though we may be able to solve a problem (i.e. break a cryptosystem), the amount of effort required to do so may be prohibitive, and this difficulty can be exploited as the basis for providing protection. When we design a system so that the effort required to perform the normal cryptographic function with posession of the 'secret' knowledge is far less than the effort required without that knowledge, this is an example of a computational advantage.

Until the late 1970s, cryptosystems were primarily designed based on Shannon's 'diffusion' and 'confusion' principles [14] which were designed to attempt to drive up the worload, but have little theoretical basis other than their obvious flattenning of statistics and scattering of message elements over a large number of cyphertext elements. The 'Data Encryption Standard' (DES) is a good example of this. [17] From its inception, there were lingering and very public doubts about the security of the DES [18]. The major problems were the lack of any reason to believe it was secure, the small key size which made 'brute force' and 'birthday' attacks possible, and the secrecy surrounding the basis for designing the key components.

In the early 1980s [15,16], 'public key' cryptosystems began to appear. These systems had

two important features; they were based on well known mathematical problems that have been worked on for hundreds of years by many very highly regarded mathematicians, and none of them were able to reduce the complexity of the problem solution below an acceptable level; and they were 'two-key' systems in which the holder of one key could authenticate or encrypt messages, but not forge or decrypt messages of the holder of the other key.

The two major systems were based on the very well known relative computational complexity of solving the well known factoring and knapsack problems with and without knowledge of how the composite number or knapsack were formed. The 'knapsack' system has become widely discredited in the cryptographic research community because the published techniques are relatively easily broken, but it is still used in a number of practical systems. The 'RSA' system is increasingly used because, dispite its poorer performance, it is far more secure.

The current trend in cryptography is almost entirely in this direction, and for most researchers in the cryptographic community, any system for which we cannot predict the computational advantage based on current mathematical knowledge is simply not trustworthy. There is also widespread recognition that mathematical breakthroughs could dramatically change the situation at any time.

5 A Call To Research

6 Summary, Conclusions, and Further Work

7 References

1. **1)** C. E. Landwehr, "The Best Available Technologies for Computer Security", IEEE Computer, V16#7, July, 1983.
2. **2)** TCSEC

3. **3)** B. W. Lampson, "A note on the Confinement Problem", Communications of the ACM V16(10) pp613-615, Oct, 1973.
4. **4)** F. Cohen, "Computer Viruses", Dissertation at the University of Southern California, 1986.
5. **5)** F. Cohen, "Computer Viruses - Theory and Experiments", DOD/NBS 7th Conference on Computer Security, originally appearing in IFIP-sec 84 (1984), also appearing as invited paper in IFIP-TC11, "Computers and Security", V6#1 (Jan. 1987), pp 22-35 and other publications in several languages.
6. **6)** D. E. Bell and L. J. LaPadula, "Secure Computer Systems: Mathematical Foundations and Model", The Mitre Corporation, 1973.
7. **7)** K. Biba ...
8. **8)** M. Harrison, W. Ruzzo, and J. Ullman, "Protection in Operating Systems", CACM V19#8, Aug 1976, pp461-471.
9. **9)** D. Denning, Lattice
10. **10)** F. Cohen, "Protection and Administration of Information Networks with Partial Orderings", IFIP-TC11, "Computers and Security", V6#2 (April 1987) pp 118-128.
11. **11)** A. Turing, "On Computable Numbers, with an Application to the Entscheidungsproblem", London Math Soc Ser 2, 1936.
12. **12)** C. Shannon, "A Mathematical Theory of Communications", Bell Systems Technical Journal, 1948.
13. **13)** F. Cohen, "Algorithmic Authentication of Identification", Information Age, V7#1 (Jan. 1985), pp 35-41.
14. **14)** C. Shannon, "A Communications Theory of Secrecy Systems", Bell Systems Technical Journal, 1949.
15. **15)** R. Rivest, A. Shamir, and L. Adleman, " "
16. **16)** W. Diffie and M. Hellman, "New Directions in Cryptography"
17. **17)** DES
18. **18)** DES doubts