SECURITY RATINGS FOR COMPUTER SYSTEMS

by

Lance J. Hoffman

Memorandum No. ERL-M444

20 May 1974

(cover)

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

SECURITY RATINGS FOR COMPUTER SYSTEMS*

Lance J. Hoffman

Computer Science Division
Department of Electrical Engineering and Computer Sciences
and the Electronics Research Laboratory
University of California, Berkeley, California 94720
(415)-642-4887 or (415)-642-1024

Abstract

A demarcation between security and protection is proposed. Then a framework is given which should allow the computation of security ratings for different computer systems and for the same computer systems at different installations. Most current computer systems will probably rate below 0.5, where a completely secure system is rated at 1 and a totally insecure system is rated at zero.

*Research sponsored by National Science Foundation Grant GJ-36475.

Introduction

For several years now, there has been increasing interest in computer security and protection techniques. With increasing work in these fields, there has been some tendency to loose use of the two words. The time has come to differentiate the terms explicitly before we all hopelessly muddy the waters.

Security

Security is a term which describes the methods for insuring confidentiality and integrity of data. There are many security methods, most of which are not technical. Procedural, administrative, legal, and personnel techniques[24] are all used in designing and implementing security systems; they antedate technical methods for computer security and are much more numerous. They are, however, beyond the scope of this paper.

The technical methods used for computer security can be broken up into three major categories: access management, privacy transformations, and threat monitoring techniques[18]. Notice that we do not make a distinction between hardware and software methods. Hardware memory access keys, for example, are lumped together with software passwords and a plethora of other techniques -- all under the umbrella of access management. Since the demarcation between hardware and software seems to change constantly, we don't consider it desirable to use that wavering line to delimit security technique categories.

Methods available within the three main categories of security techniques are well-known and numerous references to them exist[4,11,13,15].

To cite just a few examples,

access management techniques include identification cards readable by computer, hard-wired identification codes for computers, and handshaking procedures for user authentication, in addition to the memory protection keys and passwords mentioned above. Privacy transformations range from the ancient Caesar cipher through the more recent Vigènere systems to pseudo-random number generators and "finite" keys[12,8].

Threat monitoring may be done by software subroutines in the operating system or by dedicated software or hardware which monitors the action of the host computer[19,3].

Protection

The above classification of security measures is by now fairly standard. However, work on protection is more recent. It is important that the computing community agree on what protection is, and how it differs from security. In my view, protection is not concerned with malicious intruders or with unauthorized people obtaining information from a data bank; protection deals with reliability and integrity -- making sure the hardware does what it's supposed to do and the software of the operating system (not the security system) does what it is supposed to do. Protection is crucial to security. As pointed out in [22] and elsewhere, one cannot have security without a working protection mechanism. On the other hand, the presence of a proven correct protection system does not guarantee security at all.

Measurement of Security Effectiveness

Once the protection system is guaranteed to allow only access as specified by the security system, we can then consider the security system in toto and attempt to measure its effectiveness, independent of whether the protection system is matrix-based[14] or capability-based[9,10]. We shall then encounter another problem.

Most security measures cannot be proven or guaranteed correct. They can only be specified correct within certain bounds -- bounds which are often subjective. Many will only last so long before they can be broken. An authentication system which relies on physical possession of a magnetic striped card cannot be proven secure at all, since if an unauthorized person uses the card and the authorized user has not reported its loss, there is no way to detect that the system has been compromised. Even if the use of this card requires an accompanying password to be typed, an interloper can continually try different passwords with the same card until he or she is successful -- that is, unless threat monitoring is present in the security system.

We somehow must reconcile the ideas of the provability of the correctness of protection systems with the impossibility of this in security systems. Even if at some future date we are able to develop security systems which can be proven correct, these proofs will only deal with the technical aspects of security. In the long run, we will certainly have to concern ourselves with personnel security. And people cannot be "proven secure" in a non-Orwellian world.

Security Ratings for Computer Systems

What is needed is a measure which combines the potential correctness proofs for protection systems[2,5] with a security rating for portions of the security system which fall outside the protection subsystem. We currently "have no easily expressible or accurate method to measure the different levels of protection afforded by different computer systems." At the same time, "although a certain amount of intellectual debate can occasionally be heard on the question of whether or not computer security is a bi-state condition in which the computer system is either secure or not secure, the reality is that computer security is a matter of degree"[7]. The following is offered as a first attempt at such a measure.

The illustrative values given in the example below are subjective, a gross approximation, and may be entirely wrong. But the idea of security rating given systems should be considered independent of the values assigned by any one person. Indeed, one parameter of the security rating we describe below will be the person (or organization) carrying out the evaluation. While the results obtained may not be as mathematically precise as some might like, there are no more pitfalls involved here than there are in using analogous indices such as the Dow-Jones Industrial Average or Standard and Poor's ratings. A blind reading of these or any other indices can lead to disaster; nevertheless, they do perform a useful function.

Consider a system to be composed of n features from a security point of view, as shown in Table I. Ideally, these features would be an orthogonal set of items all of which were necessary for security. In

Computer Science Based Features	F_1	encryption (privacy transformations)
	F_2	access management facilities (includes identification, authentication, authorization)
	F_3	threat monitoring ability
	F_4	tamper-proof software
	F_5	tamper-proof hardware
	...	
	F_m	
Non-Computer Science Based Features	F_{m+1}	nonattractiveness to potential interlopers
	F_{m+2}	administrative security procedures
	F_{m+3}	physical security procedures
	F_{m+4}	legal prohibitions which enhance security
	⋮	⋮
	F_n	

Table I. Features of a Security/Protection System

practice, the set of features will include an overlapping collection of techniques which, when used in conjunction with one another, augment the security of a given computer system. Hopefully as research continues into the areas of protection and security we shall be able to eliminate most or all of the overlap.

Let G_i be some "goodness measure" (e.g. confidence level[1], inverse work factor[18], etc.) of feature F_i. If feature F_i is not present in a given system, $G_i \equiv 0$.

Let W_i be the subjective weight of importance assigned to feature F_i by a given person or installation. Note that $0 \leq G_i \leq 1$ and $0 \leq W_i$ for $1 \leq i \leq n$.

We propose the use of a linear weight-and-score method to compute a "security rating" SR of a system s rated by a rater r. This method is used in many other fields, for example, in computer selection[21]. The security rating is given by the equation:

$$SR(s,r) = \frac{1}{n} \sum_{i=1}^{n} W_i G_i \quad . \tag{1}$$

Notice that equation (1) rejects to some extent the theory that "a chain is no stronger than its weakest link." If this theory is carried to its extreme, the lack of any feature would result in a zero rating for a given system. But this is totally unrealistic -- all systems will have some weak links. We are trying to get a measure of how strong a system is. For that reason, we reject the idea of using (say) the product rather than the sum in equation (1).

We shall denote any partial security rating (where not all F_i are known or considered) as $SR_p(s,r)$. Note that a perfectly secure system

has a rating of 1 and a completely insecure system has a rating of 0. To insure this mathematically, we require that $\sum W_i = n$. While the weights are supplied by each individual person or installation, there could certainly be industry-standard weights or weight ranges agreed on by industry or government, as well as a standard list F of features. Important (subjectively) features would have weights above 1, and relatively unimportant features would have weights below 1. Goodness measures would probably always be subjective.

Sample Security Ratings for Real Systems

Let us consider a few real systems and what their partial security ratings might be. In these examples only, we shall ignore the features which are not based on computer science since they are usually quite installation-dependent. We shall also set $m = 5$ since we do not have any reliable knowledge of the hardware subversion potential for most of these systems. Obviously, the figures used in Table II are the subjective values of rater r (the author); they are meant to be only illustrative of the potential use of equation (1). If an installation placed the following subjective weights on various features (note that $\sum_{i=1}^{n} W_i = n$):

privacy transformations	(F_1)	1.00
access management facilities	(F_2)	0.70
threat monitoring capabilities	(F_3)	1.10
difficulty of subverting software	(F_4)	1.20

then the partial security ratings of Table II could be computed; the results are shown in Table III.

Systems	G_1	G_2	G_3	G_4	$SR_p(s,r)$ in terms of W_1, W_2, W_3, W_4
SR_p (IBM OS/360 with Resource Security System,r)	0	0.5	0.7	0.90	$(0.5W_2 + 0.7W_3 + 0.9W_4)/4$
SR_p (ADEPT-50 Weissman 1969,r)	0	0.6	0.5	0.90	$(0.6W_2 + 0.5W_3 + 0.9W_4)/4$
SR_p (DEC PDP 11/45 with UCLA-VM Popek 1974,r)	0	0.6	0.0	0.99	$(0.6W_2 + 0.99W_4)/4$
SR_p (MULTICS Organick 1972,r)	0	0.8	0.5	0.70	$(0.8W_2 + 0.5W_3 + 0.7W_4)/4$
SR_p (CDC 6400,r)	0	0.4	0.2	0.80	$(0.4W_2 + 0.2W_3 + 0.8W_4)/4$
SR_p (IBM OS/360,r)	0	0.4	0.0	0.20	$(0.4W_2 + 0.2W_4)/4$
SR_p (INTEL MCS-8,r)	0	0.0	0.0	0.00	0

Table II

Subjective "goodness values" for security features
in selected systems rated by rater r.

We see from the foregoing that different hardware and software combinations certainly lead to different degrees of secureworthiness. Since too high a security rating may lead to an unwarranted feeling of confidence, we have set up equation (1) in such a way that it appears difficult (but not impossible) to take a standard third-generation system and achieve a partial security rating which exceeds 0.6. In fairness, allow us to note again that (a) several of the example systems were never designed with security in mind and (b) the weights W_i are quite subjective. Note also that the ratings might be changed if significant features were left out of the list $F_1, F_2, \ldots, F_m$. While the list in Table I is intended to be a first approximation to some "best" list of features, we are sure there are deficiencies. Obviously, the addition of one or two additional features might easily permute the ratings of many of the example systems. When the addition or deletion of one or two features makes little or no change in the security rating for most systems, we will have come close to this "best" list.

Problems with the Proposed Measure

There are some pitfalls with the system proposed above. One is its inherent subjectiveness. We attempt to handle this problem by making the security rating a function of the rater as well as of the system being rated. But what happens when an inconsistent rater is performing? Do we need some method to rate the raters? If one consistent person or entity measures everything, this would be far more desirable than combining ratings produced by several different "subjective" raters. For this reason, we will always incorporate our rater, r, as a parameter of the security rating.

Another problem is that these ratings are concerned only with

security and not at all with utility. A quite secure system may be almost useless and the security rating scheme described here considers that lack of utility immaterial.

Potential Extensions

The method given is just one possible method for evaluating security in computer systems; there are other contenders. One might, for example, extend the notion of "coverage" used in fault tolerance[6] to security measurements. If all possible threats could be enumerated, might one consider the fraction of them which would be able to breach security? Could coverage be estimated from the reaction of the system to a sample (the composition of which would be unknown to the designers of the security system) of these threats? Could F_i be "covered" only if $G_i \geq t$, where t is defined by the rater or by the installation? (Note that t might equal 1.) Most of the above extensions have been suggested by Professor Domenico Ferrari.

Another possible extension would be to use a fuzzy measurement approach[25] to arrive at some natural language description in non-precise terms of "how secure a system is."

Summary

It is hoped that the above remarks will put into some perspective the interactions between security systems on the one hand and protection mechanisms on the other hand. By adopting the above demarcation between security and protection we should be able to nicely categorize the research currently being done in all areas of security and protection. This in turn may help us get a feel for where the problems are pretty much solved

and where work remains to be done.

With the increasing national concern over privacy and security in computer systems[16], it behooves us to provide the public with something more than the conflicting claims where one computer scientist claims a given system is secure and another says it is not. The measure of security proposed herein will help give concreteness to the arguments involved and will perhaps provide a first step toward reasonable measurement of security in computer systems.

Acknowledgements

The author wishes to acknowledge the helpful constructive criticisms of his colleagues at the University of California at Berkeley, in particular Domenico Ferrari, Robert Fabry, T.D. Friedman, Charlie Bass, and Robert Pecherer. The responsibility for the content, of course, remains with the author.

IBM OS/360 with Resource Security System	0.550
ADEPT-50	0.513
DEC PDP-11/45 with UCLA-VM	0.402
MULTICS	0.486
CDC 6400	0.365
IBM OS/360	0.130
INTEL MCS-8	0.000

Table III

Partial Security Ratings for Selected Systems Rated by rater r.

References

1 "Report of the ACM/NBS Conference on Controlled Accessibility," Rancho Santa Fe, Ca., December 1972.

2 Andrews, G., "A Protection Mechanism For Computer Systems," Ph.D. Thesis, University of Washington, 1974.

3 Basic Computing Arts, Inc., Palo Alto, Ca., advertising literature.

4 Beardsley, C.W., "Is Your Computer Insecure?," IEEE Spectrum, January 1972. Reprinted in (11).

5 Bell, D. and LaPadula, L.J., "Secure Computer Systems: A Mathematical Model," Vols. I-II, MTR-2547, the MITRE Corp., Bedford, Mass., November 1973.

6 Bouricious, W.G., Carter, W.C. and Schneider, P.R., "Reliability Modeling Techniques for Self-Repairing Computer Systems," Proc. 24th ACM National Conference 1969, pp. 295-309.

7 Bushkin, Arthur A., "A Framework for Computer Security," System Development Corporation, Document TM-WD-5733/000/00, March 31, 1974, System Development Corporation, Falls Church, VA 22041.

8 Carroll, J.M. and McLelland, P.M., "Fast Infinite-Key Privacy Transformation for Resource-Sharing System," Proc. FJCC 1970, pp. 223-230. Reprinted in (11).

9 Dennis, J.B. and Van Horn, E.G., "Programming Semantics for Multiprogrammed Computations," Comm. ACM 9, 3 (March 1966), pp. 143-155.

10 Fabry, R.S., "Capability-Based Addressing," Comm. ACM, June 1974.

11 Hoffman, L.J., Security and Privacy in Computer Systems, Melville Publishing Co., Los Angeles, CA., 1973.

12 Kahn, D., The Codebreakers, Macmillan Co., 1967.

13 Katzan, H. Jr., Computer Data Security, Van Nostrand Reinhold Co., New York, 1973.

14 Lampson, B.W., "Protection," Fifth Annual Princeton Symposium on Information Sciences and Systems, March 25-26, 1971, pp. 437-443. Reprinted in Operating Systems Review 8, 1 (January 1974), pp. 18-24.

15 Martin, James, Security, Accuracy, and Privacy in Computer Systems, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1973.

16 Oelsner, L., "President Picks Ford to Head Panel to Guard Citizens' Privacy," The New York Times CXXIII, 42400 (Feb. 24, 1974), p. 1.

17 Organick, E.I., The Multics System: An Examination of Its Structure, MIT Press, Cambridge, Mass., 1972.

18 Petersen, H.E. and Turn, R., "System Implications of Information Privacy," Proc. AFIPS 1967 SJCC, pp. 291-300. Reprinted in (11).

19 Pfister, John, Position paper, ACM/NBS Workshop on Controlled Accessibility, Rancho Santa Fe, CA, December 10-13, 1972.

20 Popek, G.J. and Kline, C., "Verifiable Secure Operating System Software," Proc. NCC 1974.

21 Sharpe, The Economics of Computers, Columbia University Press, New York (1969), p. 284ff.

22 Tsichritzis, Denis, "System Security," IBM T.J. Watson Research Center Report RC 3989, Yorktown Heights, N.Y., August 17, 1972.

23 Weissman, C., "Security Controls in the ADEPT-50 Time-Sharing System," Proc. 1969 FJCC, 119ff. Reprinted in (11).

24 Wooldridge, Susan, Corder, Colin and Johnson, Claude, Security Standards for Data Processing, Halsted Press, New York, 1973.

25 Zadeh, L.A., "A Fuzzy-Set-Theoretic Interpretation of Linguistic Hedges," ERL Memo M335, Electronics Research Laboratory, University of California, Berkeley.