Foundations of Secure Computation

EDITED BY
RICHARD A. DEMILLO
DAVID P. DOBKIN
ANITA K. JONES
RICHARD J. LIPTON

Foundations of Secure Computation

Edited by

Richard A. DeMillo
Georgia Institute of Technology
Atlanta, Georgia

David P. Dobkin
University of Arizona
Tucson, Arizona

Anita K. Jones
Carnegie-Mellon University
Pittsburgh, Pennsylvania

Richard J. Lipton
Yale University
New Haven, Connecticut

ACADEMIC PRESS NEW YORK SAN FRANCISCO LONDON 1978
A Subsidiary of Harcourt Brace Jovanovich, Publishers

ACADEMIC PRESS, INC.
111 Fifth Avenue, New York, New York 10003

United Kingdom Edition published by
ACADEMIC PRESS, INC. (LONDON) LTD.
24/28 Oval Road, London NW1 7DX

Library of Congress Cataloging in Publication Data
Main entry under title:

Foundations of secure computation.

Papers presented at a 3 day workshop held at Georgia Institute of Technology, Atlanta, Oct. 1977.
Includes bibliographical references.
1. Computers—Access control—Congresses. I. DeMillo, Richard A. II. Georgia. Institute of Technology, Atlanta.
QA76.9.A25F66 001.6'4 78-15034
ISBN 0-12-210350-5

PRINTED IN THE UNITED STATES OF AMERICA

Foundations of Secure Computation

List of Participants

Timothy A. Budd, Department of Computer Science, Yale University, New Haven, Connecticut 06520

James E. Burns, School of Information & Computer Science, Georgia Institute of Technology, Atlanta, Georgia 30332

Ellis Cohen, Computing Lab, University of Newcastle upon Tyne, Newcastle upon Tyne, England NEI 7RU

George I. Davida, Department of Electrical Engineering & Computer Science, University of Wisconsin-Milwaukee, Milwaukee, Wisconsin 53201

Richard A. DeMillo, School of Information & Computer Science, Georgia Institute of Technology, Atlanta, Georgia 30332

Dorothy Denning, Computer Sciences Department, Purdue University, Lafayette, Indiana 47906

David P. Dobkin, Department of Computer Science, University of Arizona, Tucson, Arizona 85721

Robert S. Fabry, Electrical Engineering & Computer Science Department, University of California-Berkeley, Berkeley, California 94705

Frederick C. Furtek, Mitre Corporation, Bedford, Massachusets 01730

R. Stockton Gaines, The Rand Corporation, 1700 Main Street, Santa Monica, California 90406

Robert Grafton, ONR-New York, 715 Broadway, New York, New York 10003

Patricia P. Griffiths, IBM Research, San Jose, California 95193

Leonard Haines, Office of Naval Research, Arlington, Virginia 22217

Michael A. Harrison, Computer Science Division, University of California-Berkeley, Berkeley, California 94720

Anita K. Jones, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania 15213

John B. Kam, Department of Computer Science, Columbia University, New York, New York 10027

Charles S. Kline, Department of Computer Science, University of California-Los Angeles, Los Angeles, California 90024

Richard J. Lipton, Department of Computer Science, Yale University, New Haven, Connecticut 06520

Nancy A. Lynch, School of Information & Computer Science, Georgia Institute of Technology, Atlanta, Georgia 30332

Larry McNeil, Management Science America, Inc., 3445 Peachtree Road, N.E., Atlanta, Georgia 30326

Jonathan K. Millen, Mitre Corporation, Bedford, Massachusetts 01730

Naftaly Minsky, Rutgers State University, New Brunswick, New Jersey 08903

Michael O. Rabin, Department of Mathematics, The Hebrew University, Jerusalem, Israel

Steven P. Reiss, Department of Applied Mathematics, Brown University, Providence, Rhode Island 02912

Ronald Rivest, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Walter L. Ruzzo, 1371 Shattuck Avenue, University of California-Berkeley, Berkeley, California 94709

Norman Z. Shapiro, The Rand Corporation, 1700 Main Street, Santa Monica, California 90406

Lawrence Snyder, Department of Computer Science, Yale University, New Haven, Connecticut 06520

Preface

The present book started as a four-way debate concerning the interaction of theory and design in computer security. To fix the discussion, let us note that many computational processes proceed on the assumption that a naïve or a malicious user may attempt to disrupt the process or to make undesirable inferences from observing aspects of the computation. Security is concerned with avoiding this sort of penetration. On one hand, we saw that, over several generations of systems, designers had addressed security issues with varying degrees of success and in the process a considerable body of folklore and genuine technology developed. On the other hand, we knew of theoretical work with models simple enough to permit rigorous analysis, and we wondered about the real-world implications of these theoretical results.

Fortunately, we found support among our colleagues. The papers collected herein all lie near the "crack" between theory and practice; they all address issues at the foundations of security.

The contributing authors met in October 1977 in Atlanta, Georgia, for a three-day workshop. During this time, most of the technical details of the contributions were reviewed and discussed in informal presentations. We also met for an extensive round-table discussion concerning the history, current state, and prospects of research in secure computation. Many of these discussions were taped and edited. They appear sprinkled throughout the volume.

The atmosphere of our meeting in Atlanta was charged by an external (and unexpected) sequence of events. In the summer of 1977 the national news media began to release a series of stories concerning aspects of security research—these developments concerned results in which the interaction between theory and practice figured prominently. Even at this writing, there are news reports concerning security research. Clearly, the ideas discussed in these pages will have public impact. In a fashion, this is a resolution of our debate: theory and practice do interact visibly.

A few words about the level of the 19 papers contained in the sequel may help the reader. We anticipated that a considerable body of new technical results would issue from our meeting. We were pleasantly surprised to find ample survey material scattered among the research papers. Therefore, in addition to being a timely collection of research contributions, we offer the current collection as a

book suitable for collateral readings in a seminar or an advanced course in computer security.

This project was given generous support from a number of sources. The Office of Naval Research and the U.S. Army Office each provided grants to support travel to Atlanta and the assemblage of these papers.* Gordon Goldstein, Marvin Dennicoff, Robert Grafton, and Lenny Haynes of the Office of Naval Research, and Paul Boggs and Jimmy Suttle of the U.S. Army Research Office were particularly valuable in bringing about the meeting. Support was also provided by the Computer Science Departments of Carnegie-Mellon University and Yale University. In addition, the School of Information and Computer Science at the Georgia Institute of Technology cordially extended its considerable resources to us in holding the meeting and in providing the administrative support needed to assemble the papers into their final form.

Academic Press gave us valuable help in putting the volume together. Finally, Brandy Bryant deserves special thanks. She not only typed and retyped all of these papers, but she ran herd on the project. She made sure that we did not miss our deadlines by more than a month or two, and she insisted that we do things right.

*ONR grant no. N00014-76-G-0030, ARO grant no. DAAG29-77-M-0086.

Contents

THE FOUNDATIONS OF SECURE COMPUTATION*

Richard A. DeMillo

School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia

David Dobkin

Department of Computer Science
Yale University
New Haven, Connecticut

"How do you insure privacy?"

"By coding," I said. "A two-word signature is required to gain entry to a section of the memory bank. Each word is made up of fourteen bits, making a total of twenty-eight bits."

"Then the odds are about one hundred million to one against a chance guess" ... "What if I entered someone else's code by mistake?" ...

"Nothing would happen. A countersign is necessary which requires another fourteen bits ..."

Duckworth shook his head.

"I still don't like it," he said.

I was annoyed by his obstinancy and responded by behaving childishly.

"Here," I said. "I'll let you enter any two fourteen bit words..."

Duckworth seems startled at my suggestion, but he complied ... The Confirm register lit up.

"What does that mean?" asked Duckworth.

I bit my lip.**

---

* *The preparation of this paper was supported in part by the Office of Naval Research under Grant N00014-75-C-0450 and the National Science Foundation under Grant MCS 76-11460.*

** *L. Eisenberg, The Best Laid Schemes, MacMillan, 1971.*

ISBN 0-12-210350-5

I. INTRODUCTION

Professor Duckworth's canny guess retrieved information from a fictitious Federal Investigation Bureau. In this case, fiction is paler than life and therein lies a frightening fact. The computer is often used (simultaneously) as an excuse for an instrument of insensitive and destructive policy. Evidence is the maintenance of information in machine-readable form with only slight technical guarantees of security.

Computer security has been an important issue since the first computer was developed. However, with the advent of faster and more accessible machines used by many users and large quantities of shared data, this issue has achieved far greater importance. It is no longer sufficient to rely on a system of password control through which a user is protected by having a 7-letter code known only to himself, since while this may, in the best case, prevent other users from directly accessing the users area, it does little to prevent indirect access. The potential dangers from such indirect access increase manyfold. In this survey, we shall discuss protection in two forms. The first involves the problems of unauthorized users gaining access to restricted data. In this case, it is necessary to discuss access control mechanisms that can be brought to bear in order to protect each users security. A second and far more subtle method of compromising a system is through what is called "statistical inferencing". Here, the user obtains information that is available to him legally and uses this information to infer information to which he has no access privileges. As more secure access-control mechanisms are proposed to guard from illegal access to protected data, it is this problem which looms as the major important problem of data security. And, this problem can never be totally solved since we must grant to authorized users access to data of this type. As an illustration of this problem, consider a problem faced by the census bureau (or any other creator of administrative databases). In such a database, sensitive information is collected about a group of individuals while guaranteeing each individual that data collected about him will not be made available to users at large. However, in order to do research on large segments of the population, it is necessary for aggregated forms of this data to be made available to certain users. Suppose that a sociologist wishes to study correlations among a population with respect to various characteristics. Then, it might be necessary to give this sociologist access to the data. However, in order to guarantee each individual's privacy, we will wish to do this in a statistical manner. That is, we will refuse to answer questions about an individual or small set of individuals, but will make available information about larger segments of society in a manner that does not give information about any individual. And the problem arises as to how to insure that no malicious user can use

this information in order to determine the characteristics of a single individual. A common method that has been proposed is to refuse access to information about any set of individuals which consists of too few people and in this manner restrict access to individual data. When data is given about a set of individuals, it will then be given in an aggregated form consisting of mean or median characteristics or counts of the number of people having a certain characteristic. However, as shown below, such a limitation is often not sufficient to guarantee individual privacy. Furthermore, refusing to answer a question often gives as much information as an aggregated answer since one might be able to infer information from the reason for a non-answer. Another area where this problem is of great significance is in the problem of medical record-keeping. Here, we may wish to track a set of people having a certain ailment in their early life (or people who have been exposed to certain phenomena) in order to determine long range effects of medications or exposures. In so doing, we want to make the data as helpful as possible to medical researchers while guaranteeing individual privacy. Because of the nature of such data, it is of great value to malicious users.

In this survey, we shall study the recent developments in these two areas, improved access-control mechanisms and guaranteeing statistical confidentiality. We begin with a study of the former problem in the next section. The problem of statistical security, which seems to be a major problem in the area, will be studied in detail in the third section. The goal of the survey shall be to highlight issues and recent developments in those areas. Because of our limited space, we cannot go into any issues in any amount of detail. The interested reader is referred to Hoffman's book [13] for an elementary survey of these issues and remaining papers [a] [3] for details of the state of the art on such problems. It will be clear in what follows that the interplay of theoretical and practical research has led us to question the *limitations* which we place in the notion of security as well as to create "secure" systems.

II. ACCESS CONTROL MECHANISMS

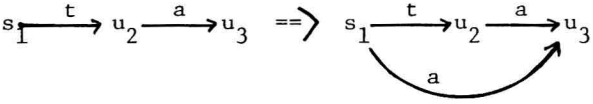
In operating systems, the most common forms of protection access-control are the access control mechanisms first introduced by Graham and Denning [11]. Access control mechanisms are capable of enforcing rules about who can perform what operation or who can access an object containing certain information. For example, users may be able to access objects via READ, WRITE, SORT, DELETEFILE, or APPEND commands with different users allowed restricted access to individual files. Access control may be represented by a subject-object matrix through which a subject

i's privileges for object j are represented as element ij in the matrix. Given such a system, one will wish to determine if it defines a secure system: can a subject obtain access to restricted objects by combining a set of privileges? In general, the problem of determining security is undecidable by a result of Harrison, Ruzzo, and Ullman [12]. While this result is of theoretical interest, it does not address the problem in a practical manner, since for particular access control mechanisms, it may be possible for specialized algorithms to solve the security problem. Thus, it may still be possible for the designer of a given system to determine the security of his system by an efficient algorithm, even though no general procedure exists for testing the security of arbitrary access control matrices.

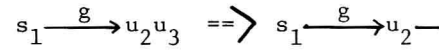
A basic question is whether it is possible to design a protection mechanism of sufficient richness so as to be capable of admitting a complex variety of sharing relationships, while being of a sufficiently simple form to have an efficient algorithm for checking its integrity. One important step toward answering this question has been made by Jones, Lipton and Synder [15, 16, 26]. Under a restricted model called the Take-Grant System, there is a linear time algorithm for testing subject security [15,16] and hence the system can be regarded as having a high degree of integrity. Furthermore, the rich instances of this system demonstrated by Snyder [26] suggest that this system will also be satisfactory in an environment where complex sharing is desired.

A Take-Grant model can be represented by a finite, directed, edge-labelled state graph and a set of rewriting rules to allow for state transitions. Vertices are labelled as either subjects (represented as s_i), objects (represented as o_i) or unknown (represented as u_i). A vertex u_i may be either a subject or an object. Edges are labelled with rights consisting of either t (for take), g (for grant) or t,g. We have four rewriting rules. Rules allow for transitions by allowing subgraph a to be replaced by subgraph b if a==>b is one of our rewriting rules. The rules are then given as a take rule, a grant rule, a create rule and a remove rule which serve to handle sharing and file handling in the user environment.

Graphically, these rules are:

(1) Take: 

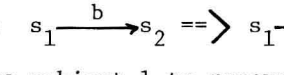
allowing subject 1 to take the privilege of u_2 to u_3 since s_1 has take rights.

(2) Grant: 

allowing subject 1 to grant his privileges to u_3 to u_2 since s_1 has grant rights.

(3) Create: 

allowing s_1 to grant u_2 a subset a of his rights.

(4) Remove: 

allowing subject 1 to remove rights of a from u_2.

We then phrase the security question as a test of whether or not x can "a" y. This situation corresponds to being given an initial configuration and asking whether we apply a set of rewriting rules to obtain a graph containing an edge from x to y containing the label a. In contrast to the results of [12], a test is available under which security in this model can be determined in linear time [15,16]. Furthermore, Snyder [26] demonstrates implementations of this method in which sufficiently rich user sharing is available.

III. SECURITY OF STATISTICAL DATA BASES

While the methods mentioned above are important for securing operating systems, they are of limited value in considering the data base security problem. Here, we are dealing with an environment where most users have only READ access to the information in the data base. The problem is to determine whether users can manipulate this access to compromise the data base. It is no longer the case that we may determine whether a user may obtain rights which should not be available to him, since every user has the same rights and no rights can be taken or granted beyond these basic rights. The issues run deeper. Users are granted access to information regarding the population served by a database and we wish to guarantee that no user may use this information to *infer* data about protected individuals (or groups) served by the data base. We are, thus, dealing with nebulous inference mechanisms

rather than simple security violations. We must discern whether a user can infer information about guarded individuals from the information we have made available to him. With the additional considerations of inferences, the problem becomes more complex. We are still faced with the tradeoff between richness and integrity: we wish to produce a system rich enough to supply useful information to those using the database while assuring the system's integrity in protecting those represented in the database.

A simple example of the subtlty of such a problem was first given by Hoffman and Miller [14] who showed that with sufficient queries a dossier could be complied on an individual represented in a database. Typically, one wishes to be able to ask questions of a database of the form:

"How many people in the database satisfy properties $P_1, P_2, \ldots, P_k$?"

"What is the mean (or median) value (of a parameter) of people satisfying properties $P_1, P_2, \ldots, P_k$?"

Such a parameter might be "salary" or "number of times hospitalized with a certain disease." Typical properties might be "male", "over 50", or "having an income greater than $10,000." Such questions or *queries* are necessary in a variety of applications. For example, suppose that one wishes to dtermine the incidence of cancer among workers in plants using certain types of chemicals [25], to track a population having a certain ailment in childhood to determine their adjustment to society [18], or to draw correlations between salary and standard of living.

As an example of the ease with which such a database can be compromised*, we consider the following example from [7] consisting of the characteristics of a number of persons who have contributed to a political campaign.

---

* *We will say that a data base has been compromised (or cracked) if a user may infer from the response to valid queries a characteristic of a person served by the data base.*

Person	Business Area	Party	Favoritism Shown by Administration	Geographic Area
P1	Steel	D	High	Northeast
P2	Steel	R	Medium	West
P3	Steel	I	Low	South
P4	Sugar	D	Medium	Northeast
P5	Sugar	R	Low	Northeast
P6	Sugar	I	High	West
P7	Oil	D	Low	South
P8	Oil	R	High	South
P9	Oil	I	Medium	West

Suppose that in order to protect individual integrity, we are only willing to make available to a user the average contribution of people sharing a common attribute, e.g., contributions from the steel industry consisting of the average of the contributions of the first three people. In this manner, we might hope to secure the database. Observe, however, that we may generate a system of twelve equations in the variables $C_1,\ldots,C_9$ with C_i corresponding to the contribution of P_i (e.g., $C_1+C_2+C_3$ corresponds to the contribution from people in the steel industry) and may then solve these equations to determine the individual values of C_1, C_2, ... C_9. While this example provides only a simple view of the problem in securing a database, it forebodes the difficulties that actually occur in large administrative databases. This issue has been previously investigated by [2,9,10] from a statistical point of view and [21,22] has considered the implications of such schemes from a medical point of view.

We are, therefore, led to consider the techniques that might be applied to *enhance* the integrity of the database. The enhancements are basically of two types both dealing with restricting data flow. We might either restrict the number and types of queries which a user might be allowed or we might restrict the form of the answer given to a query. In both of these instances, we must take care to insure that the restrictions we place on the model do not sacrifice its richness. Previous studies of this problem have appeared in [1,6,4,7,8,19,20,23,24]. In [5], this problem is shown to be basic to the study of combinatorial inference and is related to a number of well-known combinatorial problems including group testing and balance problems.

We turn now to exposition of the methods which have been proposed to handle this problem. For each, we also describe the known results concerning its efficiency.