Technical Report CS83002-R
Teaching Protection in Computing:
A Research-Oriented Graduate Course

H. Rex Hartson

March, 1983

Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061
Teaching Protection in Computing: A Research-Oriented Graduate Course

H. Rex Hartson
Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

Abstract:

This paper describes a graduate course entitled "Protection in Computing" given at Virginia Tech. The course emphasizes selected Computer Science and research aspects of protection. Following a general course description, the various topics and reading references are detailed. A chronological course outline indicates the sequence of coverage and shows the correlation of reading references to the topical areas. The use of oral presentations is described.

Keywords: security, education, training, course syllabus.
1. INTRODUCTION

The discipline of Computer Science is one in which "future shock" [15] (dizzying disorientation due to the premature arrival of the future) is perhaps more directly felt than anywhere. Because it is the responsibility of educators to keep abreast of developments in their own fields, educators must work hard to remain current. The effects of rapidly developing technology are especially magnified for Computer Science educators. Though there are many new subject areas, such as distributed databases, in the ACM Curriculum Committee's 1981 Recommendations for Master's Level Programs in Computer Science [11], there is no course in computer security. Nor is one mentioned in the report of the ACM Curriculum Committee on Information Systems for educational programs in information systems [13]. The subject is missing as well from a suggested program of continuing education in Computer Science [5] for MS-level training of technical people with no formal background in Computer Science. Nonetheless, as a consequence of the rapid pace of research and development new subareas such as data security appear abruptly in the course offerings of the more forward-looking Computer Science departments. In an earlier article in this journal [12] William Neugent described a graduate-level course on computer security which he taught at The American University. This paper describes another such offering, a graduate-level course in the area of data security and protection currently given at Virginia Tech.
Another consequence of the fast rate of technical development is that an inversion in roles may occur. Sometimes industry takes the lead in research and occasionally even in teaching. Mr. Neugent is, in fact, employed in industry and taught his course as an adjunct faculty member at The American University.

As one whose primary livelihood derives from academe but who occasionally is involved with consulting for industry, I find my present professional role complementary to that of Mr. Neugent. Likewise, the course described here and the approach I have taken in presenting it are complementary to those described by Mr. Neugent. The overlap in subject material with [12] is minimal. Each covers the field broadly, but there are differences. Where one tends to give overviews of the subject matter, the other treats that material more specifically, and vice versa. For example, Neugent gives more emphasis to risk analysis, management of computer security, and physical and environmental security. Our Virginia Tech course deals more with models of protection in operating systems, database systems, and programming languages. In that regard this course is perhaps more academic and somewhat less an applied "how-to-do-it" course than the Neugent course. The present course gives more emphasis to Computer Science and research aspects, but shares the goal of the Neugent course to promote the understanding of general issues and concepts.

There is very little overlap in reading materials between the two courses, with the Virginia Tech course being based primarily on journal and conference papers. It is not the intent of this course to produce computer security specialists, but rather to give an expo-
sure to current topics and examples of their treatment in the literature.

The course described here is also taught as a short course to government and industrial personnel. What a university calls teaching is called training in industry; nevertheless, I use essentially the same material and viewpoint. The significant difference is in the nature of the class discussions. The computer professionals in government and industry have personally experienced many of the security problems which are discussed in the course. Class discussions often center around very specific situations in their working environment. These discussions never fail to increase my understanding of the "real world" needs and problems. Their increased level of experience and self-confidence also allows them to be much more critical in their treatment of the journal articles. They are quick to perceive the difference between useful contributions in the literature and those papers which make nice academic treatises, but which fall short of a correspondence with reality.
2. GENERAL DESCRIPTION AND REFERENCES

"Protection in Computing" is a one quarter (30 contact hours in 10 weeks) graduate level course given at Virginia Tech. The overall course purview includes access controls, flow controls, inference controls for statistical databases, and cryptographic controls. The emphasis of this course is on internal, logical protection in computing. External, physical security (especially various means for identifying individuals) is mentioned but not dwelt upon. The course begins with a "litany of war stories", the lore of computer abuse, fraud, crime, and catastrophe. In a course such as this it is important for students to understand the extent of the problem and, in the case of deliberate attempts, to realize how few perpetrators are caught. The importance of quality EDP auditing is obvious. Risk analysis and cryptography are two important topics that are not not heavily stressed, but which would be expanded in a semester length version of the course. These two topics are covered, at present, by student-delivered oral presentations, described later. Issues and concepts, rather than specific systems, are stressed. Several general models are presented and evaluated; policies and mechanisms are compared in the areas of operating systems, database systems, and programming languages. A discussion of security kernels indicates the relationship of software reliability as a prerequisite to security and illustrates the limitations of program verification as an approach to security. A system-oriented approach is used in the course and implementation, performance, and cost are touched upon, as
well as special topics such as the confinement problem and the safety question.

The transition from operating system security to database security introduces new requirements and calls for new models of protection. Case studies (for example, INGRES and System R) are used to present several experimental systems. Case studies are also used in a discussion of architectural approaches to secure data management and protection of distributed databases. Cary's work, Ohio State's Data Base Computer, and MULTISAFE are examples (see section 3 for references). Capability-based protection in programming languages includes a brief study of data abstraction and capability binding for parameters of procedure calls.

The prerequisites for the course are quite general—an undergraduate-level knowledge of operating systems, data structures, and programming—allowing the course to be taken by an occasional student who is not a Computer Science major (e.g., an accounting student). Those who have previously taken a database course find they get more out of the material on database security.

Although the course is based on journal articles, there are several books used for reference. Dorothy Denning's book [3] is the newest and most technically complete of any book available. Although a course like this can never do without articles from the current literature, Denning's book would be a good choice for a required text. It also serves as a rather complete reference book on the subject of cryptography as used with computers. The book by Leiss [10] is similar to Denning's and the material is, by and large, a subset
3. COURSE TOPICS AND READING

The next section contains a chronological outline of the course's ten week duration. Presentation topics are not considered secondary material; the course relies on them to round out the subject matter. If the presentations were not included, much of that material would have to be put into the main course outline.

3.1. CHRONOLOGICAL OUTLINE

The purpose of this comprehensive outline is to allow students to locate each lecture within the context of the entire course and to coordinate the reading assignments. The reading assignments are given as codes in parentheses. These codes are keyed to the reading list which follows the outline.

Week #1:

I. Introduction
   A. Motivation, background, terminology, and scope of course
   B. Privacy: a non-technical issue [SALTGS0]
   C. Policy vs. mechanism
   D. Personal identification problem

II. Operating System Protection [LAMPB71, DENND77, DENND79b]
   A. The access matrix
   B. Capability based models -- addressing with protection
   C. Access control lists (e.g., Multics)
D. Hybrid systems, caretaker programs

Week #2:

E. Revocation, review, and the accountability problem
F. The Safety Problem [HARRM76]
G. Security classes, the simple security condition, and the star-property
H. Reference monitors
I. The Confinement Problem [LAMPB73]

Week #3:

J. Security kernels: What they are and what they are not. The history of military interest in security kernels [HARTH81a, POPEG78]
--Kernelized Secure Operating System (KSOS) [MCCAEB79]
--UCLA Secure UNIX [POPEG79]
--Provably Secure Operating System (PSOS) [FEIRR79]
--The Hierarchical Design Methodology (HDM) [NEUMP78]
K. Information flow controls [DENND76]

Week #4:

III. Database System Protection
A. Overview [WOODC80, HARTH81a]
B. The transition from operating systems and file protection
C. The need for a new model
D. A procedure based model [HOFFL71]
E. A predicate based model, sensitive to system state [HARTH76]
Week #5:

F. Additional protection measures
   Access history keeping
   Auxiliary program invocation, triggering, alerters

G. Protection languages

Week #6:

H. Static and dynamic aspects of authorization and enforcement, cost and performance
   Access decision binding times, precision vs. performance [HARTH77]

I. Relational database protection
   INGRES and query modification [STONM74]
   System R protection system [GRIFP76]

J. Semantic integrity protection -- System R [ESWAK75]

Week #7:

K. Inference controls in statistical databases, trackers
   [DENND78, DENND79a]

L. Architectural approaches to database security [HARTH81a]
   --Distributed architecture security system [CARYJ79]
   --OSU'S Data Base Computer [BANEJ78]
   --MULTISAFE [TRUER80]
   --Petri-net model of enforcement [HARTH81b]
   --Implementation of MULTISAFE in a relational environment
   --Classification of types of access dependency
M. Distributed protection of distributed data

Week #8:

IV. Protection in programming languages
   A. The use of abstract data types
   B. Amplification and access control during procedure invocation [JONEA78]
   C. Capability variables and binding rules [CLAYB81]

Weeks #9 and #10: Presentations and discussions on current research topics.

3.2. READING LIST

As the course is based on journal and conference articles, the reading is not something extra; it is the course. Timely completion of the reading assignments, assured by a required written one-page synopsis of each article, is essential to successful classroom interaction. Lectures are designed to explain, support, and extend the material in the papers. Much classroom time is devoted to critical analysis and interpretation of the reading.

Following is an annotated list of course readings which is keyed to the chronological outline above.
SALTG80  Salton, Gerard, "A Progress Report on Information Privacy and Data Security," J. of ASIS (March, 1980), 75-83. Some interesting cases and points of view on the privacy problem are presented in this paper.

DENND77  Denning, Dorothy E., and Peter J. Denning, "The Limits of Data Security," AFIPS Abacus 0, 0 (June 1977), 22-30.

DENND79b  Denning, Dorothy E., and Peter J. Denning, "Data Security," ACM Computing Surveys 11, 3 (September 1979), 227-249. The above two are nice, easy-to-read surveys of the general problem of protection in computing from a computer science viewpoint.

HARRM76  Harrison, Michael A., Walter L. Ruzzo, and Jeffrey D. Ullman, "On Protection in Operating Systems," Comm. of the ACM 19, 8 (August 1976), 461-471. This paper presents a theoretical question about the decidability of the "safety question" in computer protection systems. Class discussion centers on the relevance, applicability, and usefulness of the results in real computer systems.

LAMPB73 Lampson, Butler W., "A Note on the Confinement Problem," Comm. of the ACM 16, 10 (October 1973), 613-615. Short, but interesting, description of a problem typically ignored by lots of other researchers.


POPEG79 Popek, Gerald, J., et al., "UCLA Secure UNIX," Proc. of the AFIPS NCC, vol 48, (1979), 355-364. This and POPEG78 provide a well-written summary of one particular approach to the design of operating system security kernels.


DENND76 Denning, Dorothy E., "A Lattice Model of Secure Information Flow," Comm. of the ACM 19, 5 (May 1976), pp. 236-243. The definitive work on information flow controls (of which policies such as the "star-property" are a subset).


HARTH76 Hartson, H. Rex, and David K. Hsiao, "A Semantic Model for Data Base Protection Languages," *Proc. of the International Conf. on Very Large Data Bases* Brussels (September 1976).

HARTH77 Hartson, H. Rex, "Dynamics of Database Protection Enforcement--A Preliminary Study," *Proc. of the IEEE Computer and Software Applications Conf.* Chicago (November 1977), 349-356. These two papers introduce a predicate-based model of database access control and discuss the dynamics and cost of various kinds of enforcement.

STONM74 Stonebraker, Michael, and Eugene Wong, "Access Control in a Relational Data Base Management System by Query Modification," *Proc. of the ACM Annual Conf.* San Diego (November 1974), 180-186. An interesting approach to database security based on front-end query processing to modify each query so that it cannot request anything it shouldn't. Class discussion deals with an analysis of the advantages and disadvantages of this approach.

GRIFF76 Griffiths, Patricia P., and Bradford W. Wade, "An Authorization Mechanism for a Relational Database System," *ACM Trans. on Database Systems* 1, 3 (September 1976), 242-255. The authorization and enforcement processes proposed for System R. Major issue is consistent handling of chains of authorizations under revocation operations. Class discussion analyzes the applicability of the policies and mechanisms.
ESWA75  Eswaran, Kapali P., and Donald D. Chamberlin, "Functional Specifications of a Subsystem for Data Base Integrity," *Proc. of the International Conf. on Very Large Data Bases*, Framingham, Mass. (September 1975), 48-68. Representative of a genre of work about that time on semantic data integrity.


DENND79a  Denning, Dorothy E., Peter J. Denning, and Mayer D. Schwartz, "The Tracker: A Threat to Statistical Database Security," *ACM Trans. on Database Systems* 4, 1 (March 1979), 76-96. The two above, supplemented with results from other related papers, represent the area of inference controls in statistical databases.

BANEJ78  Banerjee, J., R.I. Baum, and D.K. Hsiao, "Concepts and Capabilities of a Database Computer," ACM Trans. on Database Systems, 3 (4), (December 1978), 347-384. This paper is one of many that describes the work of David Hsiao while he was at Ohio State University. The Data Base Computer is a high-performance database machine with functionally specialized architecture and built-in security functions.


CLAYB81  Claybrook, Billy G., and H. Rex Hartson, "Language Extensions for Specifying Access Control Policies in Programming Languages," accepted for publication in Journal of Systems and Software. These two address the matter of access controls built into the binding mechanisms of programming languages.
4. ORAL PRESENTATIONS

Computer scientists (and lots of other people) must be able to make oral technical presentations. One's work might be very good, but if the ideas cannot be communicated, the work might well be ineffective. Each student in this course must give a technical presentation on a related system or topic. The goals of each presentation include:

a. Explain--give an expository presentation of system or concept.
b. Interpret--provide intuition and insight into the concept.
c. Analyze--give a critical evaluation of the approach being presented.
d. Relate--tie the presentation to the course material by translating terminology and drawing analogies.

Some rules the students are asked to observe in making their presentations:

a. Do not read the presentation.
b. Avoid specialized jargon without explaining it.
c. Prepare carefully and practice it a few times.
d. Make sure it can be finished (including a few minutes for questions) in the allotted time; allow others their full time.
e. Use a top-down approach, starting with a broad overview and
description of structure, then fill in details as time per-
mits.
f. If overhead transparencies are used, make them simple and
readable. (No source code listings, text copied from a
paper, etc.).

No written report is handed in as part of the presentation.
However, a Xerox copy of the transparencies and notes are handed in
at the time the talk begins. Also required is a one-page bibliogra-
phy listing the sources of information that were used in preparing
the presentation. If the course were a semester in length, the pre-
sentation assignment could profitably be extended to include a
research paper as well. Following is a partial list of acceptable
presentation topics. Approval of these or other topics is negotiated
on an individual basis.

2. take-grant and other formal models of authorization
3. the safety problem
4. auditing
5. ADP trackers and statistical inference
6. capability-based mechanisms and their implementation
7. cost models for security and privacy
8. electronic fund transfer protection
9. a unified model for OS and DBMS protection
10. protection as a general systems theory problem
11. synergistic authorization and transport of privileges
12. legal and technical aspects of privacy
13. physical security
14. encryption
15. public key cryptography
16. formal program verification for security
17. transborder data flow
18. Army ADP security regulations, AR-380-380
19. DES and the surrounding controversy
20. protecting software for micro-computers
21. computer abuse by students
On the following page is a copy of the form which is used to grade each presentation as it takes place. The students are given this form to guide them in the preparation of their talk. These forms are returned to each student when all presentations are completed.
Presentation Grading Form

Name ___________________________ Topic ___________________________

Date ___________________________ Total points/100 __________

Preparation (35%)

____ Knowledge of topic (5%)
____ Quality of transparencies (size & neatness of lettering, limited
detail and busyness, legibility, contrast) (10%)
____ Completeness of coverage (5%)
____ Good use of examples (5%)
____ Good use of diagrams and figures (5%)
____ Material well organized for presentation, using top down approach,
going from general overview to details (5%)

Delivery (30%)

____ Clarity of presentation, easy to understand (5%)
____ Communication (how well are ideas conveyed?) (5%)
____ Avoids use of unfamiliar jargon (5%)
____ Good pedagogical progression (takes listener from known concepts
to new ones smoothly) (5%)
____ Length (finished in time so as not to use other people's time?)
(10%)

Content (25%)

____ Related to concepts and terminology of course (15%)
(e.g., Does topic correspond to a concept, issue, or model
discussed in class? How does it differ? What are the
constraints and limitations? Draw analogies.)
____ Evaluates critically (5%)
____ Level (avoids getting bogged down in details, boils down to major
points) (5%)

Overall Excellence (10%)

____ This category allows me to give a positive reward to that very
small number of really outstanding presentations. Normally no
points are given here. (If you didn't get any points here, it's
still not necessarily true that you didn't do a good job.)

Comments:

20
5. CONCLUSIONS

Each year the material in this course is reviewed for relevance and significance and some parts are replaced with newer material. The short duration of an academic quarter limits the choices for material, which could easily be expanded to fill a semester course. Each year the course is taught I wonder, as did Mr. Neugent, if any of my students are taking this course as part of their training as future computer criminals. Although there is a risk in exposing students to the vulnerabilities of computing systems, it is done in the hope that exposure and understanding contribute more to the solution than to the problem. The reality is that students already know much about the weaknesses of computing systems, anyway. A course such as this one is an ideal opportunity to address the issue of computer ethics, an obligation of every Computer Science curriculum.

REFERENCES


