in essence, different models implement protection binding in different levels of the data base and therefore can be viewed as special cases of our four level model.

2. It shows that protection specifications and mechanisms can be spread through different levels of a data base, and that more than one protection mechanism can be used. This in effect increases the security of the system. In particular, storing protection specifications in the storage and access levels may be advantageous in the future with dedicated *data base machines* (Baum and Hsiao, 1976).

This 'spreading' will enable a designer to achieve decentralisation of protection mechanisms if desired. It is possible to achieve decentralisation of control by having different people responsible for protection at different levels. Another important aspect of this model is that it can be used for investigating the *security engineering* problems. For example, we can ask questions such as: Which protection mechanism to use at each level? How should the protection specifications be spread between the levels? Both cost estimates and security measures are needed for answering these questions. More research can be done in this area of security engineering.

4. Summary

In this paper we have investigated the area of data base security. Beginning with the general subject area we showed why the security problems in data base systems should be discussed separately from those in operating systems. We then defined some basic concepts such as: protection specification, protection mechanism and protection binding. Using these concepts we analysed the current models of data base systems and pointed out some serious disadvantages to security, in particular the problem of centralisation. We then developed a multilevel model which can incorporate many of the security models known today. We also showed that with this model we can achieve decentralisation if desired. The model allows for the spreading of protection specifications and mechanisms at all levels of the data base, thus providing the designer with high flexibility and better performance/security tradeoffs.

References

ASTRAHAN, M. M., ALTMAN, E. B., FEHDER, P. L. and SENKO, M. E. (1972). Concepts of a Data Independent Accessing Model, *SIGFIDET Workshop*.

BAUM, R. I. and HSIAO, D. K. (1976). Database computers – a step towards data utilities, *IEEE Transactions on Computers*, Vol. 25 No. 12.

CONWAY, R. W., MAXWELL, N. L. and MORGAN, H. L. (1972). On the implementation of security measures in information systems, *CACM*, Vol. 15 No. 4.

FERNANDEZ, E. B. and WOOD, C. (1977). The Relationship Between Operating System and Data Base System Security: A Survey, *Proceedings of the First International Computer Software and Applications Conference*, November 1977.

FRIEDMAN, T. D. (1970). The Authorization Problem in Shared Files, *IBM System Journal*, Vol. 7 No. 4.

GRAHAM, S. G. and DENNING, P. J. (1972). Protection Principles and Practice, *AFIPS Conference Proceedings*, SJCC.

GUDES, E., KOCH, H. S. and STAHL, F. A. (1976a). Applications of Cryptographic Transformations to Data Base Security, *NCC Proceedings*.

GUDES, E. (1976b). The Application of Cryptography to Data Base Security, Ph.D. Dissertation, The Ohio State University.

LAMPSON, B. W. (1969). Dynamic Protection Structures, *AFIPS Conference Proceedings*, FJCC.

MCCAULEY, E. J. III (1975). Highly secure attribute based file organization, *Proceedings of the Second USA-JAPAN Computer Conference*, Tokyo, Japan, August 1975.

SIBLEY, E. H. and TAYLOR, R. W. (1973). A Data Definition and Mapping Language, *CACM*, Vol. 16 No. 12.

STONEBRAKER, M. and WONG, E. (1974). Access Control in a Relational Data Base Management Syste m by Query Modification, *Proceeding of 1974 ACM National Conference*, San Diego, CA, November 1974.

Book reviews

Compatible Fortran by A. Colin Day, 1979; 99 pages. (*Cambridge University Press*, £5·95)

In many ways Compatible FORTRAN (CF) could be regarded as Colin Day's own personal FORTRAN standard. Why is CF necessary when the 1966 and 1977 ANSI FORTRAN standards have been produced for much the same purpose? Firstly, the two ANSI standards are not completely compatible. Secondly, the standards are said to be 'permissive' and this leaves some parts open to interpretation. Thirdly, and perhaps the most important reason, is that many current FORTRAN processors (i.e. compilers, etc.) do not conform to the standard.

Compatible FORTRAN highlights the way to write portable programs despite these difficulties. This is one of the few books available that directly meet the great need for the teaching of programming standards, which should accompany the teaching of programming methods and design. The published ANSI standards are reference documents, not teaching documents. Each of the restrictions in CF which go beyond ANSI FORTRAN is well supported by the reasons for it. The restrictions relate to such things as machine dependent features, site dependent features, compiler implementation methods, compiler errors, language extensions and obsolete features and are based on Colin Day's experience in transporting FORTRAN programs. The major problem areas with transporting FORTRAN programs are 'character handling' and 'numerical precision'. The recommended way of handling characters in CF is to convert each character to a code number which can be manipulated by the arithmetic facilities of FORTRAN. Some useful functions are listed to assist the conversions. The recommendations for writing portable numerical algorithms are scantily covered in 12 lines (and three references are given). More advice about portability techniques and functions to support them should have been described in the text to a similar level of detail as the character handling (i.e. ten pages), without entering into the specialised subject of numerical analysis.

Overall *Compatible FORTRAN* is a carefully designed, practical language, capable of expressing solutions to most real life computing problems. It is the result of much scholarly attention to detail and worthy of study by all who are faced with the immediate problem of writing programs to run on several computers. It is also a good introduction to the reasoning behind FORTRAN standards and will help the reader who wishes to study ANSI standards, which is where the long-term solutions to portability lie.

P. A. CLARKE (Harpenden)

Fundamentals of Digital Computers by D. D. Spencer, 1978; 311 pages. (*Prentice-Hall*, £7·25)

I was unable to find anything in this book that would warrant its inclusion in a reading list. There are several errors in the text. The section on the history of computing notes American achievements only. New terms are introduced and then defined (if at all) several pages later.

In a book that purports to give a basic understanding as a foundation for further study, the lack of any form of bibliography is astonishing.

D. B. JOHNSON (Bromborough)