3

SECURITY WITHIN FINANCIAL INFORMATION SYSTEMS as seen from an auditor's point of view*)

Drs W.Fred. de Koning,
registered accountant, registered EDP-auditor

Partner of Paardekooper & Hoffman, Management Consultants
(member-firm of Moores Rowland International),
Calandstraat 41
P.O. Box 23123
3001 KC ROTTERDAM
The Netherlands

tel. +31-10-4364944
fax. +31-10-4360466

ABSTRACT

This paper stresses the importance of integrity of financial data in commercial information systems for the financial auditor.

Integrity of data is defined as giving a true and fair view of reality. The usefulness of the TCSEC and ITSEC criteria are reviewed from this point of view. However, these criteria are mainly focused at the confidentiality of data.

In a business environment integrity of data tends to be more important than confidentiality of data. The model of Clark and Wilson [Clark87] is reviewed for practical use. The model is extended with some suggestions for checking the consistency of business data, based on the concepts of value streams, segregation of duties and comparison with standards. Additionally recommendations are given for the implementation of the adjusted Clark/Wilson model in a midrange computer environment.

Information Technology seems to be developing in the direction of distributed or decentralised data processing and more flexible information systems. A practical problem of the Clark/Wilson model is the heavy reliance on the integrity of application software. In a dynamic business environment the application programs are subject to continuing change. Besides which both development procedures and change management procedures are weakening. A solution for this dilemma might be the embedding of the separation of duties in the corporate data model.

---

*) This paper is a by-product a of research project into information technology and internal control by the Limperg Institute in Amsterdam, a collaboration between the auditing faculties of the Dutch universities. The complete results of this project, which had a wider scope than just integrity of data, will be published later this year.

J. H. P. Eloff et al. (eds.), *Information Security — the Next Decade*

SECURITY WITHIN FINANCIAL INFORMATION SYSTEMS as seen from an auditor's point of view

1. INTRODUCTION

In this paper the security of financial information systems will be reviewed from the point of view of the *financial auditor* (the public accountant who certifies the annual accounts of the entity). For the audit of the annual accounts the auditor has to investigate the reliability of the data that result from the accounting systems of the entity to be audited. This investigation can be limited in time when the transaction and accounting systems offer enough safeguards for the reliability of the resulting data. That means the information systems will have to be secure.

Opinions about security of information systems change over time. The first ideas about security in computer systems came from military circles and were especially aimed at guarding the secrecy or confidentiality of the data in information systems. The Bell/LaPadula-model [Bell74] and the "Orange Book" [DoD83], which is essentially based on this model, were for several years the guidelines for the opinions about security in information systems. Clark and Wilson [Clark87] were among the first who discussed the usefulness of the security concepts of the Orange Book for application in a business environment. They argued that in a business environment the integrity of information is usually more important than its confidentiality.

Integrity of data can be defined as giving a true and fair view of the reality, also called the external consistency of data. This definition complies with the needs of the financial auditor. The Clark/Wilson model is primarily based on the concept of segregation of duties. Segregation of duties (or separation of duties) is a well-known and important tool of internal control, of which financial auditors make use. In this paper the concept of segregation of duties will be defined and it will be shown how this measure of internal control can be implemented into well-known computer systems, such as the IBM AS/400 and Unix systems.

2. SYSTEMS-ORIENTED AUDITING

The primary function of the financial auditor is the evaluation and certification of annual accounts of companies and other organizations. With a non-qualified opinion the auditor declares that the annual accounts present a true and fair view of the assets and liabilities of the entity at the end of the year and of the results during the year. The annual accounts are based on the data derived from the accounting systems of the entity which is being audited (the year-end balances of the general ledger accounts). These balances can still be adjusted before they are summarized into the balance sheet and the income statement. For example inventories can be adjusted after taking stock, provisions for bad debts can be made, etc. The reliability of the data from the accounting systems however is an important condition for an auditor to be able to give an unqualified opinion on the annual accounts. For the evaluation

of the accounting data the auditor can choose between the data-oriented approach or the systems-oriented approach.

When following the *data-oriented approach* a sample from the data in the accounting systems is checked with data from sources outside the information system, e.g. input data, invoices from suppliers, confirmations from banks or debtors etc. Following the *systems-oriented approach* the auditor evaluates the way the system and the internal controls within the system are functioning; the actual checking of data will be limited as much as possible. When the system has enough safeguards to establish the reliability of the resulting data, the extent of the audit can be reduced substantially. Additionally the systems-oriented approach often makes it possible to give meaningful recommendations for improving the level of internal control related to information processing, a welcome by-product of the financial audit.

It will be obvious to you that I am a true supporter of the systems-oriented audit approach. However this approach is only feasible if the organization to be audited complies to certain minimum requirements of internal control. In this paper I will consider these internal control requirements, but first I would like to review some of the well-known security models.

3. THE BELL/LAPADULA MODEL (TCSEC)

The Bell/LaPadula model [Bell74] was the foundation on which the "Orange Book" [DoD83] was built, which contains the security standards of the American Department of Defense, also called the Trusted Computer Security Evaluation Criteria (TCSEC). These criteria are mainly focused on the secrecy of confidential information. Following the principle of "*Mandatory Access Control*" access to information will depend on the level of authority of the users. Documents will be given a classification label, such as "confidential", "secret" and "top-secret" for example. Users will be divided into classes and be granted access to the documents according to the class they are in. A user (a subject) is allowed to read a data file (an object) if his or her level of authorization is higher or equal to the classification of the object. A subject is allowed to write to an object if the authorization of the subject is lower or equal to that of the object.

Next to Mandatory Access Control, the TCSEC criteria give guidelines for the so called *Discretionary Access Control*; that means the control of the access of individual users to objects. The owner of an object is allowed to grant some of his authorizations to other users.

The TCSEC criteria are mainly used to qualify computer systems, especially operating systems and security packages, leading to a certain rating from C1 (the lowest level) to A1 (the highest level). For the C levels only Discretionary Access Control is required. For the B and A levels Mandatory Access Control must be provided as well. Most of the commercially used products are stuck at a C1 or C2 classification. Mandatory Access Control seems to be typically useful for a military environment, where the confidentiality of data is all important.

4. THE EUROPEAN ITSEC MODEL

The Commission of the European Communities has published security requirements for computer systems as well, the Information Technology Security Evaluation Criteria or ITSEC [Eur91]. These standards are partly based on the German criteria of the ZSI [ZSI89].

The ITSEC criteria are intended to have a wider scope than the TCSEC criteria. Not just the confidentiality of information, but also the integrity and availability of information are included. Security is defined within the ITSEC report as:
- *confidentiality* - prevention of the unauthorized disclosure of information;
- *integrity* - prevention of the unauthorized modification of information;
- *availability* - prevention of the unauthorized withholding of information or resources.

The definitions of integrity and availability are different from the common definitions and somewhat limited. Integrity is commonly defined as safeguarding the accuracy and completeness of information and computer software and availability as ensuring that information and vital services are available to users when required [DTI93].

The ITSEC criteria are grouped as follows:
- identification and authentication;
- access control, based on access rights and the verification of these rights;
- accountability, based on recording the exercising of rights by users;
- audit, based on recording of security-related events;
- object reuse, by control of the re-allocation of internal and external storage;
- accuracy, that means the detection and prevention of loss, addition or alteration of data;
- reliability of service, including functions intended to ensure that resources are accessible and usable on demand.
- security of the data exchange or data communication, broken down into:
  - authentication;
  - access control;
  - confidentiality;
  - integrity;
  - non-repudiation.

Although attention has been paid to data integrity, availability and data communication the ITSEC criteria correspond broadly with the TCSEC criteria. This year the TCSEC criteria, the ITSEC criteria and the Canadian criteria (CTCPEC) will be brought together to form the "Common Criteria".

5. THE CLARK/WILSON MODEL

During the 1987 IEEE Symposium on Security and Privacy Clark and Wilson [Clark87] wondered why software products, which are widely used within the business environment,

such as the security packages RACF, ACF-2 and Top-Secret, did not reach the B level of the TCSEC, but stayed at to the lower C level. They came to the conclusion that the security requirements of the industry are different from the security requirements of the "Orange Book". A major goal of *commercial data processing*, according to Clark and Wilson often the most important goal, is to ensure integrity of data to prevent fraud and errors. They claim there are two mechanisms at the heart of fraud and error control: separation of duties among employees and the "well-formed transaction".

Computers do not normally observe and record what happens in reality*). The recording of events by people may lead to intentional or unintentional errors. By *separation or segregation of duties*, or rather by segregated recording of events and by checking the consistency of the segregated recordings, the integrity of the data in the information system can be safeguarded. Consequently a user is not allowed to change data in an arbitrary way, but only by "*well-formed transactions*", that means certified or approved applications. These applications will be assigned to users in accordance with their duties and the related tasks and they must provide an audit trail.

This view is completely different from the Bell/LaPadula model, which requires that the authorizations of users to manipulate the data are defined within the operating system or a security package. The Bell/LaPadula model disregards the application programs completely. Yet the *application programs* are very important. Consider for instance the situation that a storekeeper is allowed to enter the receipt of goods into the computer system. That means he must be allowed to change the inventory file. This file however is also used to check the actions of the storekeeper. Thus it is clear that he can never be allowed to change the data in the file directly, but only by means of an application program, which will perform a number of additional controls, such as the reconciliation of the entered data with the purchase order file, the preparation of a control list or control file, the composition of a journal entry for the general ledger, etc.

6. SEGREGATION OF DUTIES FROM AN AUDITOR'S POINT OF VIEW

Gray and Manson [Gray89] describe segregation of duties as an important principle of internal control "which requires that no one person sees a transaction through from the beginning to end on their own" (page 131). According to them the following separation of functions can be considered as mandatory:

- authorization of transactions
- execution of transactions
- custody of assets
- recording of transactions and assets.

---

*) However, recording of events in the real world by computers will become feasible more and more with tools like scanning of data, RF/ID, ATM's etc.

It must be noted that the concept of separation of functions is somewhat changed in an automated and especially integrated environment. Nowadays it seems more important to separate the *successive phases* of a value stream or business cycle. Additionally the recording of the transactions has been taken over by the computer system. Provided it is not possible to execute transactions without regard to the computer system (in other words the use of the computer system to execute the computer system must be mandatory), this provides a good safeguard for the *completeness of the recording* of transaction data.

In an automated environment the concept of segregation of duties can be used to devise self-controlling systems. As an example we will review the purchase function of a company. The following duties can be distinguished:

- the preparation of the purchase order;
- the authorization of the purchase order;
- the recording of the receipt of goods;
- the recording of the incoming vendor invoice;
- the authorization of the payment to the vendor.

For all these separate duties, which are related to the successive phases of the purchase cycle, *separate application programs* will be available. When every step within the purchase cycle is carried out by an other employee and the consistency of the "segregated" data recorded in the information system has been established, then the reliability or integrity of the data in the purchase system has been safeguarded to a high degree (collusion disregarded).

7. CONSISTENCY OF DATA STREAM

In order to establish the consistency of the data stream the following interrelations in the value streams, which can be recognized within most *trading organizations*, can be used:

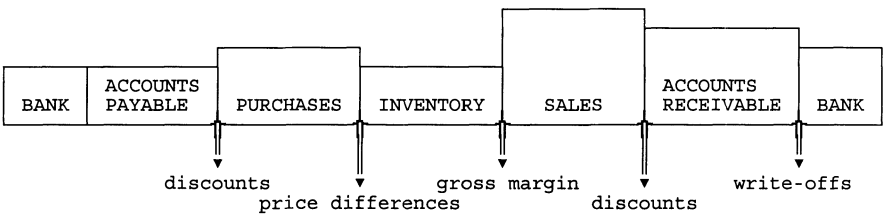

Figure 1. Value streams for trading companies.

Note the following *interrelations* in this value stream:

- payments to suppliers = debiting bank account;
- recorded purchases = increase in accounts payable;
- recorded sales = decrease in inventories;
- recorded sales = increase in accounts receivable;
- payments from customers = crediting bank account.

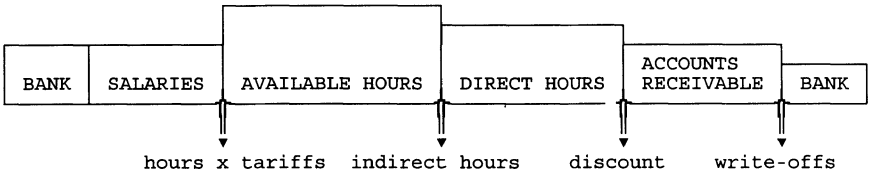

Figure 2. Value stream for service companies.

In *service companies* the interrelations can be:
- paid salaries = debiting bank account;
- paid salaries = available manhours;
- available manhours = chargeable hours + indirect hours;
- chargeable hours = invoiced hours = increase in accounts receivable;
- payments from customers = crediting bank account.

The interrelationships will not be as straight forward as shown above. There will be some *leakage* within the value streams and there are "*value leaps*". The information system should give enough detailed information for the analysis of these leakages and leaps. E.g. it must be possible to analyze the gross margin from different viewpoints, such as the margin of each product group, the margin of each customer group, region, sales representative, branch, etc. The "leakage" by indirect (that means not chargeable) hours can be analyzed by presenting the hours by category, by branch, by level of employee, etc.

Generally speaking one could say: at several stages within the value streams there have to be exchanges of information between the reality and the information system. These links between value stream and the representation of it by the information system have to be checked for consistency. A check on consistency can also be a comparison with standards, like budgets, production standards, sales targets, budgeted gross margin, and so on.

These checks can be represented in a diagram as follows:

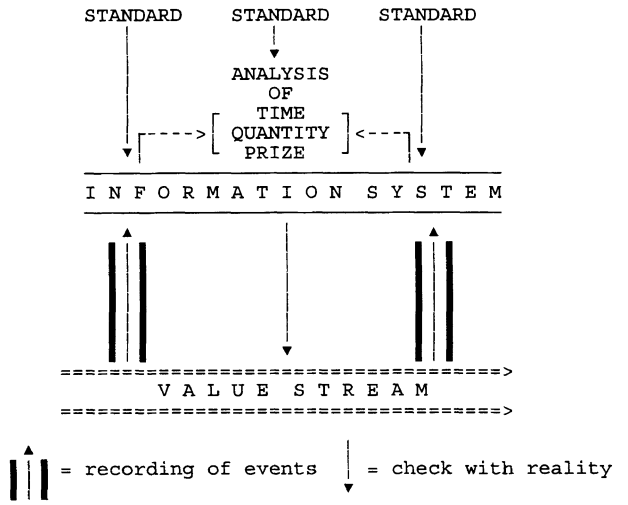

Figure 3. Adjusted Clark/Wilson model.

8. THE IMPLEMENTATION OF THE CLARK/WILSON MODEL ON MIDRANGE COMPUTERS

The Clark/Wilson model can be summarized as follows:

1. Each individual user will have to be identified and authenticated by the system, in order to control and audit the actions of the user.

2. Data items to which the integrity policy must be applied (so called Constrained Data Items) will only be manipulated by a restricted set of application programs (Transformation Procedures). These programs should be certified as being "well-formed transactions".

3. To each user a valid set of application programs has to be allocated. This allocation must follow the rules for separation of duties.

4. The system must keep an audit log, in which execution of programs by users will be recorded.

5. The system must contain mechanisms which enforce adherence to the procedures described above.

6. These mechanisms must be protected against unauthorized manipulation.

The first requirement is nowadays commonly accepted. Almost every multi-user operating system has facilities for identification of users and authentication by means of passwords.

The second requirement implies good procedures for *systems development* and *change management*, the latter being an increasingly important control. In order to control the production programs separation of the development environment from the production environment is essential. That can be realized by separate libraries, separate directories or even completely separate computer system. Procedures for the transfer of tested and accepted new or changed programs have to be established. The authorization for access to the production library is a key factor for the control.

The third requirement means that there must be communication between *responsible managers* in the user departments and the data centre, where the access rights and authorizations to use the system are defined by system managers, super-users or security officers. In an increasing number of cases it appears to be possible to transfer some of the competencies from the data centre to the user departments, which favours the involvement of responsible managers.

The fourth requirement can be solved by *history logging or audit files*, which are available on most midrange systems. For the analysis of these files analysis tools such as a query language or audit software can be useful.

The fifth requirement will depend on the procedures for the definition, maintenance and control of user profiles, object authorizations, authorization lists, trustee information and so on. *Regular reviews* of these procedures will be useful.

The sixth requirement is a difficult issue. Mistakes and omissions are possible and furthermore, most systems do not have facilities for a good separation of duties related to critical functions in the system. The consequence can be that too much responsibility rests on the shoulders of the *system manager or the security officer*. There are two ways to limit this risk: the facilities of the security officer can be limited and his or her actions can be reviewed (by analysis of log files) on a regular basis. Neither option is easily realizable.

Most midrange computer systems provide means for access control of data files. In practice it appears that these facilities are not intensively used. The access control to files is often regulated by *menu control* in application programs. When the users are only allowed to execute application programs, this might be a satisfactory control. However one must be sure that users do not have direct access to files by using editors, SQL-tools, query languages, upload-facilities and so on. Therefore it seems desirable to protect the data files as well. Especially in an environment where users are allowed to make queries or other retrievals from files, you will want to limit users to read-only access to files.

However, users need to be able to change data when they are executing the application programs for which they are authorized. The IBM AS/400 has a very useful facility to cope with this dilemma, called *adopted authority*. When the user executes an application program, the program can adopt the authorization of the user who owns the program. That can be a "virtual" user, whose profile is only used for the implementation of new software into the production libraries. Bear in mind that the user profile of the owner must be of a low authorization class, e.g. no authorization as security officer.

It has to be mentioned that there is an inherent risk that the adopted authorization will become valid outside the application program (however, such an application cannot be called a "well-formed transaction").

Unix offers similar possibilities with the *SETUID or SETGRID commands*. With this commands the user-id can be changed during the execution of an application program into a user-id which is authorized to change data.

With adoption of authority or the setting of the user-id it becomes possible to protect the data files for direct change by end-users. In that way the most important requirement of Clark and Wilson can be fulfilled, namely: "No user of the system, even if authorized, may be permitted to modify data items in such a way that assets or accounting records of the company are lost or corrupted." (page 186).

9. RECENT DEVELOPMENTS

Recently there have been developments in the field of information processing in the direction of distributed or decentralised data processing. The Clark/Wilson model relies heavily on the integrity of application software. This requires, as argued before, good procedures for system development and change management, including separation of duties between systems development and data processing. The tendency towards *decentralisation and distributed processing* implies fewer possibilities for the realization of such procedures, which means that it will become more and more difficult to consider application programs as "well-formed transactions".

Moreover the dynamics of the business environment have increased requiring more *flexibility* from the information systems. System development procedures and change management procedures can become an obstacle in this regard. Flexibility of retrieval and presentation of data can relatively easily be realized by report generators, query languages and so on. End-users only need a read-permission for the files they want to access. By the use of adaption of authorization or the SETUID command, as outlined before, the write-permission can be granted to a program instead of an end-user.

Flexibility within the application programs that actually change data is more difficult to realize. Karger already pointed out that the certification of transaction procedures is extremely difficult [Karger88]. When programs are frequently changed, even by end-

users using "end-user tools" (fourth generation languages), certification of Transformation Procedures, and consequently the implementation of the Clark/Wilson model, becomes virtually impossible. Besides the audit of the effective separation of duties becomes a very time consuming affair.

10. SEGREGATION OF OBJECTS

For dynamic or distributed environments I would like to suggest a different solution. In this solution segregation of duties is not realized by separation of application programs, but by separation of objects in the data model of the organization.
Ramackers [Ramackers94] recently wrote a thesis in which he tries to relate business modelling to information modelling.

Within the *strategic business model* Ramackers defines an object as the primary resources of an organization that are required for its functions, such as an order or a cargo.

Within the *operational business model* an object is defined as a resource unit that is perceived as central to the business domain of the organization. It may be an information object (conceptual) or a material (physical) object.

Within the *information system model* an object is defined as a basic component of an information system that consists of a unique identity, a number of properties (attributes and relationships) and a number of actions. Properties form the basis for expressing constraints that may be associated with an object. Actions form the basis for formulating events for the object with associated pre en post conditions. An information object at the business level is reflected by one or more object types at the information system level.

The objects in the data model should reflect the organization and procedures within the organization, including the separation of duties. The conceptual information objects, such as orders, receipts or payments, are especially interesting from the point of view of separation of duties. One could for instance define a "receipt-of-goods" object. A storekeeper can be allowed to add receipts to the goods-received table. He will **not** be allowed to change directly the inventory file (if it still exists).

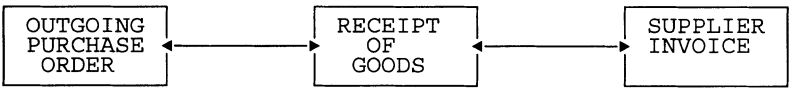

Figure 4. Objects related to purchase.

Constraints on the update could for instance be: the received goods must be ordered, that means be recorded in the purchase-order table and the supplier must be available in the supplier table. The authorizations and the constraints are included in the object definition. This means that the most important controls are recorded next to the data definition which safeguards the proper functioning of these controls and which simplifies the audit of the controls.

The concept of objects, *including authorizations and constraints*, forms the ideal situation which will be reached within a few years, I anticipate. Present database management systems, however, do have some opportunities for access control and for validation of data entry. Further developments in the area of data dictionary/directory systems and repositories will have to be awaited.

11. VERIFICATION PROCEDURES

Clark and Wilson not only speak of Transformation Procedures, but also of *Integrity Verification Procedures* (IVPs). The IVPs are application programs which check that all Constraint Data Items (CDIs) conform to the integrity specification. These IVPs still remain useful in a situation where well-formed TPs no longer exist. The IVPs can be used for checking the consistency of recorded data, based on the relationships within the value streams. Moreover they could derive journal entries from the recorded data. An IVP could for instance check that all recorded receipts of goods match the purchase order file and consequently derive a journal entry for goods received based on the unit costs from the article table. When the separation of duties is based on the access to the objects, there is no clear need for well-formed transactions for the update of data (from the point of view of separation of duties). However it is very important that the IVPs are still well-formed transactions: the consistency control depends on them to a large extent.

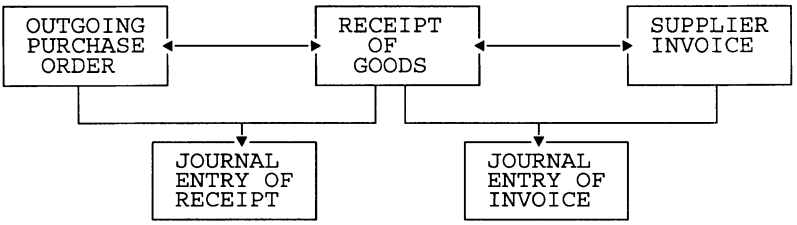

Figure 5. Journal entries related to purchase.

12. CONCLUSIONS

The Bell/LaPadula model and the "Orange Book" are focused mainly at the *confidentiality* of data. In business environments *integrity* of data is often more important than confidentiality of data. The ITSEC criteria mention integrity of data, but the definition (prevention of the unauthorized modification of information) is unusual and only relates to access control.

The Clark/Wilson model is useful in a business environment with quite stable systems and good procedures for systems development and change management. The concept of value streams is useful for a automatic check on the *consistency of data*.

In a more *decentralised environment* it is hardly possible to implement the Clark/Wilson model. The systems development procedures and the change management procedures will be below the required standards. In a dynamic business environment the certification of transformation procedures can be a hindrance to the *flexibility* of the information systems.

In such cases the separation of duties can possibly be embedded in the *data model*, which requires object-oriented systems. The objects must be a representation of the business model including the related responsibilities (separation of duties).
In the systems currently in use a practical solution can be found by using the facilities or data dictionary/directory systems or repositories.

December 30th, 1994

REFERENCES:

[Bell73] D.E. Bell and L.J. LaPadula, Secure Computer Systems, A Mathematical Model, 1973, Mitre Corporation, Bedford Mass.

[Clark87] D.D. Clark and D.R. Wilson, A Comparison of Commercial and Military Security Policies, Proceedings 1987 IEEE Symposium on Security and Privacy, page 184-194, IEEE Computer Society Press, Oakland CA.

[DoD83] Department of Defense, Trusted Computer Security Evaluation Criteria, 1983, Computer Security Centre, Fort Meade, MD.

[DTI93] Department of Trade and Industry (UK), A Code of Practice for Information Security Management, 1993.

[Eur91] European Commission, Information Technology Security Evaluation Criteria (ITSEC), version 1.2, 1991, ECSC-EEC-EAEC, Brussels-Luxembourg.

[Gray89] I. Gray and S. Manson, The Audit Process, 1989, Van Nostrand Reinhold, London.

[Karger88] P.A. Karger, Implementing Commercial Data Integrity with Secure Capabilities, Proceedings 1988 IEEE Symposium on Security and Privacy, page 130-139.

[Ramack94] G.J. Ramackers, Integrated Object Modelling, 1994, Thesis Publishers, Amsterdam.

[ZSI89] Zentralstelle (or Bundesamt) für Sicherheit in der Informationstechnik, IT-Sicherheitskriterien, 1989, Bonn.