23

Is there a need for new information security models?

S.A. Kokolakis
Department of Informatics
Athens University of Economics and Business
76 Patission Str., GR-104 34 Athens, Greece
voice: +301-8225268, fax: +301-8226204
email: sak@aueb.gr

Abstract

A considerable number of formal information security models have been developed during the last two decades. We present and discuss some of the most widespread ones that have been successfully applied to the traditional, centralised Information Systems of the past. We show the special security needs of modern information systems that are based on the concepts of Open Distributed Processing, the Object-oriented paradigm and multimedia technology. We argue that these Information Systems need new or enhanced information security models in order to address the information security issue effectively and present some efforts towards this goal.

Keywords

Information Security, Information Systems Security, Formal Information Security Models.

1 INTRODUCTION

Extensive use of information technology has brought out a number of security related issues. When referring to these issues several different terms are used including computer security, Information Technology (IT) security, information security and Information Systems (IS) security. We shall try to clarify these concepts in order to delineate the field of Information Security, which is going to be the subject of our discussion.

Computers security and **Information Technology security** are used interchangeably usually denoting every security issue that is related to computers or IT in general. This use of the terms amplifies the existing obscurity in the terminology used. Hopefully, a more concrete interpretation of IT security tends to be established focusing on the technical aspects of the information systems security issue. In EU's glossary of Information Systems Security (Commission of EC, 1993) an IT system is defined as "a specific IT Assembly installed in a specific (set of) location(s) with a specific operational environment responding

P. Horster (ed.), *Communications and Multimedia Security II*

to a particular set of purposes", where IT Assembly is "a collection of computer hardware, software, (and sometimes communication equipment or other IT components) capable of being used to handle information". The importance of communication equipment has lead to the use of the more complete term Information Technology and Communications Security.

Information Systems (IS) Security, on the other hand, has a broader meaning. Schoderbek (Schoderbek, 1990) defines the term system as follows: "a system is a set of objects together with relationships between the objects and between their attributes related to each other and to their environment so as to form a whole." The term Information System can be defined as a system that handles information and consists of 5 interrelated elements, namely hardware, software, data/information, people and procedures (Kiountouzis et al., 1996). So, we may say that *IS Security* is the scientific discipline that deals with the problem of protecting the five elements that consist an IS and the IS as a whole.

Information security concerns the preservation of the security related information attributes. Traditionally, these attributes are thought to be **integrity**, **availability** and **confidentiality**. The efficiency of these 3 attributes has been challenged, mainly by Donn Parker (Parker, 1995). Parker suggests that **utility** and **authenticity** should be included, but we can also come up with other attributes, such as **validity**, each of them showing a different aspect of the information (in)security problem. EU's glossary, for example, defines information security as "the combination of Confidentiality, Validity, Authenticity, Integrity and Information Availability". We can say that although all these attributes and many more need protection, when we refer to a particular application we should select the most appropriate ones and include them in the security requirements of the application.

Research in information security has been concerned with information security models since the early '70s. The most celebrated formal information security model was published by Bell and LaPadula in 1973. The need for this kind of models derives from the diversity of meanings the term "secure" can take in different applications. Formal models help the designers or evaluators of an IS to decide what "secure" means for their particular purpose. This could involve the selection of information attributes that need protection and the formal definition of each attribute. By selecting a formal information security model we set the security requirements which have to be met, so that later on we can examine whether the system has succeeded in meeting its security requirements or not. So, a formal security model is a tool for designing and evaluating information security in a particular IS. Finally, formal models serve as abstractions of reality, suitable for theoretical and scientific study of the information security issue.

2 INFORMATION SECURITY MODELS

2.1 The Bell-LaPadula Model

2.1.1 The formal model

The Bell and LaPadula model belongs to a class of models called access control models. The main characteristic of **access control models** is the use of the following features:

- a set of objects O, i.e. the set of protected entities, for instance files,
- a set of subjects $S \subseteq O$, for instance users or processes,
- a set of rights, for instance read write rights, and
- a set of rules, governing the manipulation of objects by subjects.

Based on the above features we can build a **reference monitor** (Millen, 1989). A reference monitor can be thought of as a state-transition machine whose current state is an *access matrix* showing, for each subject and object what set of access modes the subject currently has for that object. The Bell and LaPadula model defines a kind of machine that we shall call a "BLP machine" (Millen, 1989). A BLP machine has state set V, inputs R called requests, and outputs D={yes, no, ?} called decisions. Decision outputs are associated with transitions rather than states. A state has four components (b, M, *f*, H), which will be described below along with other elements of the model.

As a reference monitor, a BLP machine has a set of subjects S, which is a subset of a set of objects O, and it has a set of access attributes A={r, e, w, a}. Each state has an access set component, denoted with the symbol b, and representing current accesses as a set of triples (s, o, x) included in $S \times O \times A$. A BLP machine has a lattice L of security levels. Each level has two components: a classification from a totally ordered set C, and a subset of the set K of categories. Subsets of K are partially ordered by set inclusion, and the lattice ordering $\propto$ on L is induced as the direct product $C \times P(K)$. That is, $(c, x) \propto (c', x')$ if $c \leq c'$ and $x \subset x'$. For example, (Confidential, {NATO}) $\propto$ (Secret, {NATO, NUCLEAR}).

Security levels are assigned to subjects and objects by another component of the state, symbolised *f*. An *f*-component is actually a triple (f_s, f_o, f_c), where

$f_s : S \rightarrow L$ is the subject (maximum) security level function,
$f_o : O \rightarrow L$ is the object security level function, and
$f_c : S \rightarrow L$ is the subject current security level function.

The current security level is the one that plays a part in the *-property. The two levels are motivated by the idea that when a user logs in to a computer system, a process is created to communicate with the user's terminal and issue system commands. The process operates at a current security level requested by the user, and that level may be at or below the clearance of the user, which is recorded as the maximum level of the process. It is required that $f_c(s) \propto f_s(s)$. There are two axioms relating current access to level assignments: the simple security property and the *-property. The simple security property states that a subject can have read access only to objects at or below its maximum level.

Simple Security Property. For each state υ=(b, M, *f*, H),
if (s, o, r) $\in$ b or (s, o, w) $\in$ b, then $f_o(o) \propto f_c(s)$.

The *-property has an exception built into it for subjects in a distinguished set S_r of "trusted" subjects.

**-Property*. For each state υ=(b, M, *f*, H),
if (s, o, r) $\in$ b and $s \notin S_r$, then $f_o(o) \propto f_c(s)$;
if (s, o, w) $\in$ b and $s \notin S_r$, then $f_o(o) = f_c(s)$; and
if (s, o, a) $\in$ b and $s \notin S_r$, then $f_c(s) \propto f_o(o)$.

Two other components were added to the state to support discretionary access control. $M: S \times O \rightarrow P(A)$ is an access matrix whose elements represent access permissions rather than current access. The hierarchy component H is a function on O into $P(O)$, giving the set of subordinates of each object. The discretionary security property states that current accesses are restricted to accesses permitted in M.
Discretionary Security Property. For each state υ=(b, M, *f*, H), if (s, o, x) $\in$ b, then $x \in M(s,o)$.

The model is enhanced by eleven transition rules each rule being a function on R×V into D×V, giving the decision output and next state for each possible request and current state.

2.1.2 Discussion

The Bell-LaPadula model was constructed with military security in mind. Consequently, it focuses on information confidentiality. So, we may say that the enforcement of Bell-LaPadula model by itself does not make our system secure and an integration of Bell-LaPadula with other models should be needed in order to preserve a wider range of information attributes.

Based on the state-transition machine concept the Bell-LaPadula model defines a set of valid states and provides a set of transition rules that refuse any request that would leave the system in a state violating the security policy. In a centralised system is not very hard to implement controls that check the validity of the initial state of the system and then apply a mechanism that enforces the transition rules. In Open Distributed Processing (ODP) systems the possibility of implementing successfully a mechanism of this kind is rather limited. Especially when referring to a computer network, unless access of any subject to any object is controlled by one processing element (i.e. node of the network, computer, etc.) checks on the state of the whole network are not easy to implement.

The model has also been criticised because it permits the downgrading of objects that if not handled properly may result in total corruption of system's integrity. But as Bell points out "a model such as the Bell-LaPadula model that was constructed as an abstraction to allow analysis free of irrelevant detail never claimed to be a justification of "axioms" in a foundational sense, nor did it claim to capture all the facets of intuitive-security" (Bell, 1988). So, the model should be used in that sense as an abstraction to allow analysis free of irrelevant detail.

2.2 Biba's integrity model

2.2.1 Model presentation

Biba suggests that mandatory access controls could also be used for integrity even though they were originally intended merely to prevent compromise of information. Subjects and objects are labelled with integrity levels (e.g. crucial, very important, important).

If we think of a high-integrity level, e.g. Crucial, as dominating a low-integrity level, e.g. Important, the information flow policy for these levels is the opposite of that for sensitivity levels. Information flow from one entity to another should be allowed only when the destination carries an integrity level dominated by that of the source. Information can lose its integrity; it can never gain in integrity (Millen, 1989).

In Biba's model, a subject can observe or modify objects, and invoke other subjects. Invocation is meant to be interpreted as interprocess communication or procedure calls (into a different protection domain). Invocation causes information, in the form of a message or parameter values, to flow from the invoking subject to the invoked one.

Four different access control policies were proposed by Biba. The simplest and best remembered is the strict integrity policy, which permits a subject to

- observe access only to objects of a higher or equal integrity level,
- modify access only to objects of a lower or equal integrity level,
- invoke access only to subjects of a lower or equal integrity level.

The other three policies allow various relaxations of the axioms of the strict integrity policy. They are: a *low-water mark policy*, in which a subject can observe objects of lower integrity level, but its own integrity level is reduced accordingly; a *low-water mark for objects policy*, a low-water mark policy in which a subject can also modify objects of a

higher integrity level, but the integrity level of those objects is immediately reduced; and a *ring policy*, in which observation is unconstrained.

2.2.2 Discussion

Biba's model is a data integrity model that sees integrity as a measure of trust. The whole idea is that data items that belong to a low integrity level (untrusted) should not contaminate data items of high integrity (trusted).

The use of mandatory access controls implies that all objects and subjects should be assigned an integrity level label. This of course should be done by the security officer. So, any data entering the system should be given a label and this label should be of an integrity level at least equal to that of the subject that is supposed to use these data. So, according to this model all data entry should be done or at least observed by the security officer (Clark & Wilson, 1987). This requirement is obviously unrealistic.

2.3 The Clark and Wilson model

2.3.1 Model presentation

The Clark and Wilson model recommends the enforcement of two main principles, namely the principle of *well-formed transactions* and the principle of *separation of duty*. The concept of the well-formed transaction (Clark & Wilson, 1987) is that a user should not manipulate data arbitrarily, but only in constrained ways that preserve or ensure the integrity of the data. A very common mechanism in well-formed transactions is to record all data modifications in a log so that actions can be audited later. Separation of duties simply means that at least two people are required for a transaction to take place. When this principle is enforced a collusion among employees is needed for a fraud to take place. This can be made very difficult if we use additional techniques, such as random selection of the sets of people to perform an operation, so that any proposed collusion is only safe by chance.

The formal model presented by Clark and Wilson uses the term "Constrained Data Items- CDIs" to name the objects that need protection. Transformations are executed by Transformation Procedures (TPs). The model provides rules that ensure that if a CDI is in a valid state and a series of TPs are executed the CDI will reach a state that will be valid. Besides the enforcement of the two principles mentioned above, the proposed rules require user authentication, log keeping and certification procedures (i.e. procedures that certify the validity of a state and the integrity of TPs).

2.3.2 Discussion

The Clark and Wilson model is a data integrity model and attempts to present in a formal, abstract way commercial data processing practices. We may say that the model is rather a formal representation of commercial security requirements than a formal information security model. The model is considerably flexible and addition of rules, if necessary, is quite straightforward.

The use of Transformation Procedures (TPs) instead of simple access rights allows a variety of security policies to be implemented based on the Clark-Wilson model. The model allows enforcement of different security policies in different applications within the same system. This model can be easily implemented in an object-oriented system as CDIs can be considered as protected objects having TPs as methods. Finally, the model includes two rules that are now considered essential for any multiuser system. The first rule states that the system must authenticate the identity of each user attempting to execute a TP and the second one requires that the system should keep a log of all operations concerning CDIs.

2.4 Chinese-Wall model

2.4.1 Model presentation

Commercial security policies may differ significantly depending on the particular field they are applied to. Financial institutions, for example, are strongly dependent on information validity and confidentiality. The Chinese-Wall security model (Brewer & Nash, 1989) was developed as a formal model of a security policy applicable to financial information systems. Consider a market analyst who advises different companies. Such an analyst must uphold the confidentiality of information provided to him by his firm's clients; this means he cannot advise corporations where he has insider knowledge of the plans, status or standing of a competitor. However, the analyst is free to advise corporations which are not in competition with each other, and also to draw on general market information.

All corporate information is stored in a hierarchically arranged filing system such as that shown in figure 1. Each individual object is associated with the dataset to which it belongs and each dataset belongs to a conflict of interest class. In our example, object *a* belongs to the company dataset *Wella* and *Wella* belongs to the conflict of interest class *Cosmetics companies*.

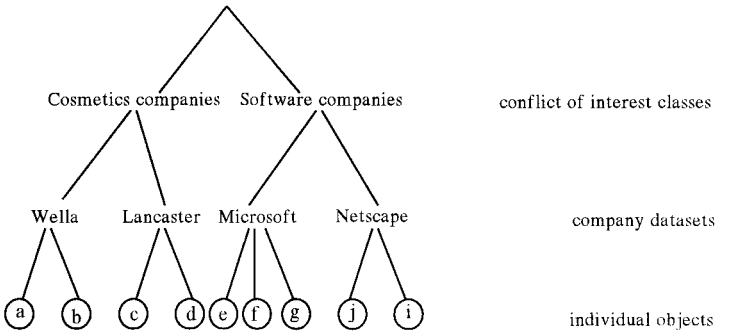

Figure 1 Object hierarchy in the Chinese Wall model.

The basis of the Chinese Wall policy is that people are only allowed access to information which is not held to conflict with any other information that they already possess. A new user may freely choose to access whatever datasets he likes. Suppose our user access object a belonging to dataset Wella. Sometime later he requests access to the object e belonging to dataset Microsoft. This is quite permissible since Wella and Microsoft belong to different conflict of interest classes. However, if he requests access to object c (Lancaster dataset) access must be denied since a conflict between Wella and Lancaster exists.

2.4.2 Discussion

This is a very specialised model applicable mainly to financial information systems. It covers only the information confidentiality problem. If other security attributes are to be protected integration of the Chinese Wall policy with other security models is required. It is not obvious whether this integration could be efficient and practically applicable or even plausible.

We may say that the Chinese Wall policy is very restrictive as enforcement of this policy may have a direct effect on organisational productivity. It is obvious that an organisation adopting this model should need to employ more experts. Another drawback of

the model is the need to sanitise information. That is information that is not thought confidential any more (e.g. obsolete information) should be put in a special class, where access is unrestricted or at least information in the class is not subject to the Chinese Wall policy. This requires significant cost in security management.

2.5 Information flow models

There are ways to compromise information in a computer system that cannot be understood solely from access control considerations. One way to do this is by exploiting a *covert channel.* Covert channels are mechanisms by which a process operating at a high sensitivity level can send information to a lower-level process, in spite of an access control policy (Millen, 1989). Handling of security problems of this kind calls for formal definitions of information flow and precise restrictions on information channels. This need has motivated the development of specialised information security models, known as *Information Flow Models.*

2.5.1 Lattice model

The lattice model is an extension of the Bell-LaPadula model and was introduced to formalise information flow policies. A lattice is an algebraic structure (SC, $\leq$, $\oplus$, $\otimes$). SC is a finite set of security classes and $\leq$ defines a partial ordering on SC. Each subject and each object is assigned a security class. $\oplus$ defines the least upper bound operator and $\otimes$ defines the greatest lower bound operator. The greatest lower bound of SC can be interpreted as sanitised information which is accessible by anyone.

In a system modelled according to the lattice model information is permitted to flow within a class or upward. That is, information may flow from objects labelled with class $A \in SC$ to objects labelled with class $B \in SC$ iff $A \leq B$. The right of dissemination of information is independent of the object representing the information.

2.6 Database security

Security policies enforced at operating system level are insufficient for database protection. This results from the special security needs of databases. A database security policy should be independent from the file structures and relevant operating system protection mechanisms. Furthermore, protection at the level of tables, fields or records (in relational terminology) is not always sufficient and protection at the level of data elements is required. Other database security issues include **inference** and **aggregation**.

Methods to compromise information exist that take advantage of the ability of gathering statistical and other data about databases. Information not accessible by unauthorised users can be *inferred* based on statistical data provided by a database system. Suppose, for example, a database that contains wages of employees in a corporation. A user, that has access to all employees' data but his superior executive manager's can infer his superior's wage if he can get the average of all wages. The data aggregation problem simply refers to the fact that a large enough accumulation of records can become more sensitive than any of the individual records. Finally, the advent of *object oriented databases* calls for special security models that use object oriented concepts (Boulahia-Cuppens et al., 1993). An example of a contemporary object oriented database security model is DISCO (Olivier & vonSolms, 1992) (Discretionary Security Model for Object-oriented Databases).

This model uses *capabilities* to protect entities in the database. Capability is a protected identifier that identifies an object and specifies the access rights to be allowed to whoever possesses the capability. A subject that possesses a capability is authorised to access the corresponding entity. Additionally, under certain conditions a subject may pass the

capability on to another subject authorising this other subject to access the protected entity. In this way a discretionary security policy can be implemented.

The object-oriented model has a rich variety of entities with relationships between such entities. A subject that passes a capability on to another subject may (inadvertently) authorise the second subject to access more entities than intended. DISCO describes the restrictions that apply to the transfer of capabilities to safeguard against such an unintended disclosure of information.

Revoking of capabilities also has major implications: if a capability is revoked, it is possible that the user may still make inferences about the protected information. The model, also, considers the restrictions that should apply to revocation of capabilities.

The model, finally, indicates how the transfer of capabilities (and transfer of ownership) may be included in methods of a protected object. If such transfers are included in methods, the freedom of a subject to transfer capabilities to other subjects is limited to the extent determined by the method. This corresponds to the general object oriented philosophy that the manipulation of data encapsulated in the object is restricted to such manipulation permitted by the (encapsulated) methods.

3 DISCUSSION

None of the models described above can cover all aspects of information security. Bell-LaPadula model focuses on confidentiality and is more suitable for military applications. Clark and Wilson model is a data integrity model and can be thought of as a formal representation of commercial security requirements. Biba's model uses the concepts of mandatory access control for information integrity. Chinese Wall model is a very specialised model applicable mainly to financial information systems and deals with information confidentiality. Information flow models deal with information confidentiality and their prime objective is to detect covert channels.

As shown above all models discussed concern information confidentiality and integrity. Availability, though widely accepted as a prime security attribute, is not covered by those models. This can be attributed to the fact that the concept of information availability is intrinsically fuzzy (Gritzalis, 1994). So far, availability is dealt with integrity mechanisms that protect operating system's components and communication software and hardware. We may argue that an information availability model should integrate fuzzy logic concepts. Utility, authenticity and validity lack of formal definitions that would serve as a basis for relevant formal models. Authenticity, however, can be preserved by mechanisms that monitor all modifications of protected objects and, possibly, keep a log of objects' history.

Since organisational needs cannot be covered by a single model, usually a combination of models is required. These models, however, are not compatible with each other and integration of various models is hard to implement and involves a considerable cost. So, usually, when designing a new application we have to build our own security model, using concepts and ideas proposed by formal models like those presented above. Obviously, there is a need for new models that cover a wider range of security attributes, are flexible enough to be applied in a variety of IS and to be adapted accordingly and can be implemented at an acceptable cost.

4 CURRENT TRENDS IN INFORMATION SECURITY

Most models mentioned above have been quite successful within the traditional centralised application environment. Modern IS have gone beyond this centralised concept introducing

Open Distributed Processing (ODP) systems and *Multimedia* technology. Consider, for example, a teleworking application. Such an application involves people with no special training in IT communicating through a computer network and using a multimedia interface. The security requirements of this application should include non-repudiation of messages, entity authentication and perhaps monitoring of information exchange, where information could be in the form of video, audio, etc. This simple example show us how distributed systems and multimedia technology have changed significantly the nature of IS. Within this framework security policies have become more complex and concepts such as *multiple security policies*, *metapolicies*, and *application dependent security policies* have emerged (Kuhnhauser, 1995), (Rieb, 1990).

Cooperation of IS with different security policies generally requires a definition of the relationships between the involved security policies, a concept referred to as *metapolicies*. *Multipe security policies* occur when instead of applying a general security policy for every subject and object in an IS we have a number of distinct security domains with different policies within the same IS. Some examples of modern security models that take into consideration the special security needs of ODP systems follow (Kuhnhauser, 1995).

4.1 Access Control Programs

Access Control Programs (ACPs) were introduced in Theimer et al. (1992). In a distributed environment with a client/server-based cooperation scheme, ACPs permit a fine-grained delegation of access rights to intermediates that act on behalf of a client. The concept is based on the observation that existing delegation protocols have only very limited possibilities to restrict delegated rights to precisely the minimum the intermediate needs, which in general results in too much rights given to an intermediate.

ACPs are programs that a client passes together with a request to a server. An ACP precisely describes the rights a client is willing to delegate to an intermediate for each single request. A digital signature prevents intermediates from tampering with it. As a part of the permission check within the server, the server executes the ACP and grants the intermediate's access if the ACP approves.

The basic model focuses on delegation, ACP transmission and protection, ACP execution and on the ACP language. Nevertheless, the concept of ACPs has another very interesting facet. It withdraws from the traditional scheme of passive, databased descriptions of access rights to *algorithm-based access decisions*, and thus is well in tune with the increasing need for *ambitious security models*.

4.2 Custodians

The custodian paradigm is a concept to support *multiple*, *application-dependent* and *user-defined* security policies for distributed applications (Kuhnhauser, 1995). Similar to an ACP, a custodian is a shell for a security policy that constitutes an annex to a policy neutral reference monitor. While maintaining the reference monitor properties tamperproofness, complete access mediation and verifiability, custodians prepare the reference monitor to assimilate user-defined security policies. Custodians are glued to arbitrary system entities, hereby submitting the entities to the custodian's security policy. No semantical restriction is imposed on the contained security policy. A custodian may emulate a simple traditional access control list or contain a sophisticated rule based security policy.

The custodian paradigm is based on a tamperproof reference monitor providing total communication control and separation. Additionally, the reference monitor provides the glue between application and security policy. The glue is a binding mechanism that associates a custodian to an arbitrary collection of application objects. Any such association

results in a detour of any object communication to its associated custodian. Depending on the security policy, the communication can be suppressed, modified, or forwarded to the original recipient. The binding mechanism thus expands the total communication control property of the reference monitor to include custodians.

5 CONCLUSIONS

Information security concerns the preservation of a number of security related information attributes. Integrity, confidentiality, availability, utility, authenticity and validity are a few (perhaps the most important ones) of the many possible information attributes to be protected. Varying views on which attributes should be included in the definition of information security and what meaning should be given in each of them exist. This justifies the need for formal definitions of information security. Formal information security models serve as indisputable definitions of what "secure" means in our particular application. Moreover, formal models set the security goals of our application and their formalism facilitates the process of verifying that these security goals are finally met. Finally, we should not underestimate the importance of formal models as abstractions of reality suitable for theoretical and scientific study of the information security issue.

We have presented and discussed some of the most widespread formal information models. We have reached the conclusion that none of these models can cover all aspects of information security. Models differ on the area of application, the information attributes they refer to and the level of abstraction. Their main features are presented in table 1. We should notice that none of the models presented covers attributes other than confidentiality and integrity.

We have shown some of the new concepts in information security that have emerged with the advent of ODP systems. The most important ones seems to be application dependent security policies, metapolicies and multiple security policies. Multiple security policies allow distinct security policy domains, administered by different organisational entities, each with complete policy autonomy in its domain, to be modelled in an information system. The coexistence of multiple policies requires metapolicies that would control the interaction of individual policies. Finally, this framework allows us to establish application dependent policies. This can also occur when we have a number of distinct applications within the same system or application that are established (sometimes ad hoc) between IS communicating through a network or maybe client-server applications.

So, ODP systems require a new approach towards information security. Quoting from Kuhnhauser (Kuhnhauser, 1995) "in the next decade the integration of application dependent security policies in a distributed computer system will become a major challenge in computer security". ACPs and custodians presented above consist an attempt to tackle this issue. Though ACPs and custodians provide a technical solution to the problem, they cannot be considered formal information security models in the sense the term was used in this paper. Moreover, defining metapolicies remains an issue and new methodologies are needed that will guide our effort to define particular metapolicies.

The object-oriented paradigm has influenced information security in two ways. First, we have to develop information security models for object-oriented systems. Traditional models are not capable of providing security to object-oriented systems and especially to object-oriented databases and new models are being developed. The need for new models originates from the fact that the basic object-subject concept of traditional models does not apply (at least not at this simple form) to object-oriented systems (Boulahia-Cuppens et al., 1993). On the other hand, object-oriented concepts such as encapsulation and polymorphism can be used by information security models, thus creating object-oriented models of information security (Kang et al., 1993).

Table 1 Main features of information security models.

Models	**Main features**
Bell-LaPadula model	Confidentiality. Military security. Centralised IS. Low implementation cost.
Biba's model	Integrity. Mandatory access controls. Hard to apply. Centralised IS.
Clark-Wilson model	Integrity. Commercial security. Can be used in object-oriented systems. Can serve as a basis for application dependent policies. Flexible.
Chinese Wall model	Confidentiality. Financial institutions. Possible negative effect on productivity. High cost in security management.
Information flow models	Confidentiality. Covert channels detection.
DISCO	Database security. Object-oriented. Use of capabilities. Discretionary security policy.
ACPs	Distributed environment. Application dependent security policies. Algorithm-based access decisions.
Custodians	Distributed environment. Application dependent security policies. Metapolicies.

Multimedia technology has changed the form of information to be protected. Images, audio and video are more complex packages than mere text. Data take a different meaning depending on their context and attaching security labels to information in a multimedia form is a far more complex task. Multimedia security research besides developing more efficient encryption techniques should provide information security models, methodologies and techniques that take into consideration the special features of multimedia technology.

Concluding, we may say that although researchers have provided a plethora of formal information security models in the last two decades, the evolution of information technology has changed the nature of IS and new formal models are needed in order to tackle the security issue within this new framework.

6 ACKNOWLEDGEMENTS

The author would like to thank Prof. Evangelos Kiountouzis and dr. Dimitris Gritzalis for their support and encouragement.

7 REFERENCES

Bell, D.E. (1988) Concerning "Modelling" of Computer Security. In *proceedings of 1988 IEEE Symposium on Security and Privacy.*

Boulahia-Cuppens, N., Cuppens, F., Gabillon, A. and Yazdanian, K. (1993) Multilevel Security in Object-Oriented Databases. In *Security for Object-Oriented Systems*, (eds. B. Thuraisingham, K. Sandhu and T.C. Ting), proceedings of the OOPSLA-93 Conference Workshop on Security for Object-Oriented Systems, Springel-Verlag.

Brewer, D.F.C. and Nash, M.J. (1989) The Chinese Wall Security Policy. In *proceedings of the 1989 IEEE Symposium on Security and Privacy.*

Clark, D.D. and Wilson, D.R. (1987) A comparison of Commercial and Military Computer Security Policies. In *proceedings of the 1987 IEEE Symposium on Security and Privacy.*

Commission of the European Communities (1993) Glossary of Information Systems Security, contract S2001, Definitions Within Information Systems Security.

Gritzalis, D (1994) Information Security in Dependable Systems. PhD Thesis, University of Aegean, Greece, April 1994.

Kang, M.H., Costich, O. and Froscher, J.N. (1993) Using Object Modeling Techniques to Design MLS Data Models. In *Security for Object-Oriented Systems*, (eds. B. Thuraisingham, K. Sandhu and T.C. Ting), proceedings of the OOPSLA-93 Conference Workshop on Security for Object-Oriented Systems, Springel-Verlag.

Kiountouzis, E.A. and Kokolakis,S.A. (1996) An Analyst's View of IS Security. In *proceedings of the 12th International Conference on Information Security*, IFIP'96, Samos, Greece.

Kuhnhauser, W.E. (1995) On Paradigms for Security Policies in Multipolicy Environments. In *Information Security - the Next Decade* (eds. Ellof,J. and S. von Solms). Proceedings of the 11th International Conference on Information Security, IFIP'95, Chapman & Hall, London.

Landwehr, C.E. (1981) Formal Models of Computer Security. *ACM Computing Surveys* vol. 13(3), 1981.

Millen, J.K. (1989) Models of Multilevel Computer Security. *Advances in Computers*, vol. 22. Academic Press Inc.

Olivier, M.S. and vonSolms, S.H. (1992) DISCO: A Discretionary Security Model for Object-oriented Databases. In *IT Security: the Need for International Cooperation* (eds. G.G. Gable and W.J. Caelli). Proceedings of the 8th International Conference on Information Security, IFIP'92, North-Holland.

Parker, D.B. (1995) A New Framework for Information Security to Avoid Information Anarchy. In *Information Security - the Next Decade* (eds. Ellof,J. and S. von Solms). Proceedings of the 11th International Conference on Information Security, IFIP'95, Chapman & Hall, London.

Rieb, H.P. (1990) Modeling Security in Distributed Systems. In *Computer Security and Information Integrity* (eds. K. Dittrich, S. Rautakivi and J. Saari), Proceedings of the 7th International Conference on Information Security, IFIP SEC '90, Elsevier Science Publ., 1991.

Schoderbek, P., Schoderbek, G. and Kefalas, A. (1990) *Management Systems.* Conceptual Considerations, 4th ed., Irwin, Boston, 1990.

Theimer, M.M., Nichols, D.A. and Terry, D.B. (1992) Delegation Through Access Control Programs. In *proceedings of the 12th International Conference on Distributed Systems.* IEEE Computer Society Press.