36

Aligning Information Security Profiles With Organizational Policies

Mrs D Pottas[a] & Prof SH von Solms[b]

[a] Sasol Group Information Division, PO Box 79512, Senderwood, 2145, South Africa
e-mail: potd0113@sasol.co.za

[b] Department of Computer Science, Rand Afrikaans University, PO Box 524, Auckland Park, 2006, South Africa

1. INTRODUCTION

This paper is a follow-up on ongoing research in regard of modelling the generation of information security profiles. In previous publications [Pot92, Pot93a, Pot93b] a model named MAPS (Model for Automated Profile Specification) was introduced. This model presents a synthesis for the generation and maintenance of information security profiles which are used in the enforcement of logical access control. The primary assertion made by the MAPS model, is that organizational policies such as the information security policy, job descriptions and business objectives should be used as basis for structuring security requirements. This will ensure conformance of the configured security controls to the high-level as well as more specific policies of an organization. It also provides a basis for the automation and tracking of access control specification.

In this paper the MAPS model is discussed in keeping with the results of research to date. Special attention is given to what should be included in the organizational policies and how this information will be used to generate profiles. Section 2 of this paper presents a bit of background information on the incentive for the creation of MAPS. In section 3 the context, scope and premises of this research are established as a precursor to the discussion of MAPS as a whole in section 4. A methodology (M-MAPS) providing a structured approach for the compilation of access control specifications, is introduced in section 4. This methodology consists of five phases (with phase 1 subdivided into two steps), which will each be discussed in sections 5-10. The paper is concluded in section 11.

2. BACKGROUND INFORMATION

The MAPS model originated due to the identification of inadequacies with regard to the generation of information security profiles in the mainframe environment. Following research done concerning the state of the art of the implementation and use of packages such as Top Secret

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

[Com91a], RACF [IBM92] and ACF2 [Com91b], the following became apparent:

- There does not exist a methodology for the generation of profiles that can to some extent, facilitate an automated implementation of the process.
- Once profiles are compiled, their contents are not traceable back to the originator and purpose thereof.
- There is uncertainty as to whether the profiles reflect and conform to the policies of the organization (eg division of duties and principle of least privilege).
- There is agreement in general, that security profiles never are quite up to date.
- There is uncertainty as to whether the grouping of user rights within the security database (containing the profiles), represents the granularity required in the distinction of these rights.

These scenarios resulted inter alia due to tedious manual procedures during the initial setup of security databases and as such, caused problems with maintenance procedures. In general, the responsibility of creating and maintaining security databases resort with security administrators. This leads to a situation where access rights to data and programs are granted without the advantage of discernment on an operational level.

Consequential to the identification of the above problems, was the development of the MAPS model, with the objective of automating the generation of information security profiles, while retaining the origin and reasons for the access rights/prohibitions they contain. This is achieved by providing a content and representation specification for organizational policies, which are in turn used to generate security profiles from.

3. MAPS - CONTEXT, SCOPE AND PREMISES

In this section some of the key issues along the lines of which the MAPS model is moulded, are briefly discussed.

3.1 The Organizational Security Repository

The organizational security repository (OSR) is the term used to denote a repository of organizational policies, which will be used as input for the structuring of security controls. The contents and representation of these policies are discussed in sections 5 and 6 respectively.

3.2 Enterprise Wide Security Administration

The problems in regard of security profiles that were discussed in section 2, were initially identified in the mainframe environment. However, these problems are not restricted to the mainframe environment, but are fundamental to every access control package presently on the market. The MAPS model therefore aims towards enterprise wide (all platforms included) security administration.

3.3 Role-Based Approach

In the literature various security models supporting role-based access control policies have been described [Abr91, Dob89, Fer, Fer91, Jon91, Ster91]. In fact, this approach is seen as central to the enforcement of security in commercial systems today. The MAPS approach is to abstract from

the actual users and model the functional operation of the organization solely in terms of roles. The users are then associated with the roles necessary to perform their duties.

3.4 Establishing the Role-to-Resource Relationship

The MAPS model follows a specific approach to resource analysis in order to accommodate the extraction of access rights and restrictions from organizational policies. This approach is illustrated in figure 1, within the context of the MAPS role-based approach.

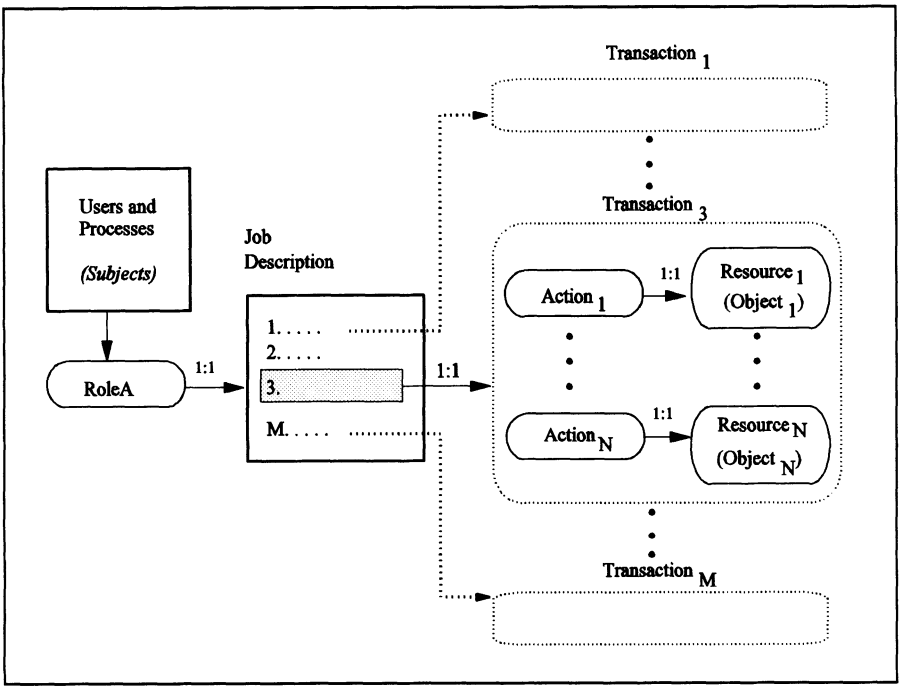

Figure 1 Establishing the Role-to-Resource Relationship

In general terms, the depiction in figure 1 may be described as follows:

1. Users and dynamically initiated processes are firstly associated with an applicable role (represented as RoleA), eg *Store Supervisor*.
2. Each role has a corresponding job description (JD) which states the functionality associated with the role, eg *Query Available stock of Item*.
3. The statements contained in a JD are each equivalent to computerized counterparts termed transactions. A transaction is representative of the entity to which access rights/prohibitions are granted in an information security profile.
4. Each transaction is in turn represented as several action-to-resource (A//R) couplings. For example, the transaction mapped from the JD statement *Query Available Stock of Item*, may be represented as follows:

Step	Action	Resource
1	dAccess	SERVJHB002
2	execute	f:\user\system.bat
3	execute	f:\appl\system.exe
4	execute	f:\appl\stockq.exe
5	read	f:\appl\data\invent.dat

Firstly the consultant will log on to the server *SERVJHB002* by supplying his user identification code and a correct password - note the specification of this type of access as a *dAccess*. Thereafter, the user has to execute the *system.bat* file in the directory *f:\user* which in turn executes the *system.exe* program in the directory *f:\appl*. The user will now make a choice from a menu indicating an intention to query the stock available of an item. Hereupon the *system.exe* program will execute the *stockq.exe* program (once again in the *f:\appl* directory) which will read the required information from the *invent.dat* data file in the directory *f:\appl\data*.

As may be seen from the above example, the MAPS model uses specific actions to define utilization or manipulation of a resource in the A//R analysis. These are defined to consist of *dAccess, Read, Write, Execute, Delete and Create*. These access types are more or less self-evident, except for the type *dAccess*. This term is used to indicate a direct access to a resource which can not be described by one of the other valid access types, eg the act of logging on to a server would be classified as a *dAccess*.

The A//R analysis has a dual purpose within the context of MAPS. Firstly, it is used to demarcate a transaction operationally (ie the transaction specifies WHAT is to be done, the A//R analysis specifies HOW it is to be done). At the same time, the A//R analysis specifies the path to be followed in the execution of a transaction. Simply stated, an *access path* may be seen as a sequence of entities which denote the path followed from a user to gain access to a resource. The inclusion of this concept in the MAPS model stems from security problems experienced as a result of the rapid deployment of new technologies such as networks, distributed processing, etc in the commercial environment. Due to lack of space this concept will not be expanded on at this stage.

3.5 The OSR Administrator

In the commercial environment, security administrators are normally employed to control and maintain the mechanisms (eg profiles) used to enforce logical access control. Contrary to this requirement, the MAPS model does not need an administrator in the context of administering the security system, because of the supposition that profiles are generated automatically from the OSR. What is needed, is an administrator in the role of specifying the contents of the OSR. This will facilitate the requirement as posed by MAPS that the identity of the originator of access authorizations should be retained. For this reason the concept of an OSR administrator is introduced as a person or persons responsible for the users/processes, role allocations, job descriptions and A//R analyses contained in the OSR. In order to facilitate the controlled devolution of authority, OSR administrators may be restricted to specific domains, as is done with security administrators in practice.

3.6 In Conclusion

In this section a general setting for the MAPS model was presented in order to ascertain the surrounding and connected circumstances. In section 4 these concepts will be used in the discussion of the MAPS model.

4. MAPS - MODEL FOR AUTOMATED PROFILE SPECIFICATION

The basic components of the MAPS model are as follows:

- Specific entities (such as the OSR and OSR administrator) are defined, which facilitate an accurate description of the phenomena of interest. A brief overview of these were provided in section 3.
- A methodology termed *Method-MAPS* (M-MAPS) is defined which facilitates an orderly and methodical approach to creating access specifications. A summary of this methodology is provided in the rest of this section.
- A specification notation based on the predicate calculus and termed *Representational-MAPS* (R-MAPS), is defined and used to represent and manipulate entities created with the application of M-MAPS. R-MAPS is discussed in more detail in section 6.

A high-level perception of the M-MAPS methodology is contained in figure 2. The basic structural components depicted in this diagram are levels, phases and arrows representing flow of operation.

Functional Levels

The first act in the construction of M-MAPS was the establishment of a structure which serves as a functional directive for the development of information security profiles. This structure basically provides an answer to WHAT is to be done by way of the following four functional levels:

Level 1: Planning & Preparation

The establishment of a content- specification from which profiles may be extrapolated.

Level 2: Development & Testing

The development of generic profiles from the established specification. The testing of these profiles to ensure that they reflect the requirements set out in the content-specification.

Level 3: Implementation & Testing

The adaption of the compiled generic profiles for utilization with a designated access control package and the checking of the final profiles in conjunction with this package.

Level 4: Maintenance

The continued updating of the profiles in consequence of any required changes.

An indication of the level of progress is provided with level 1 being the least advanced and level 4 the most advanced.

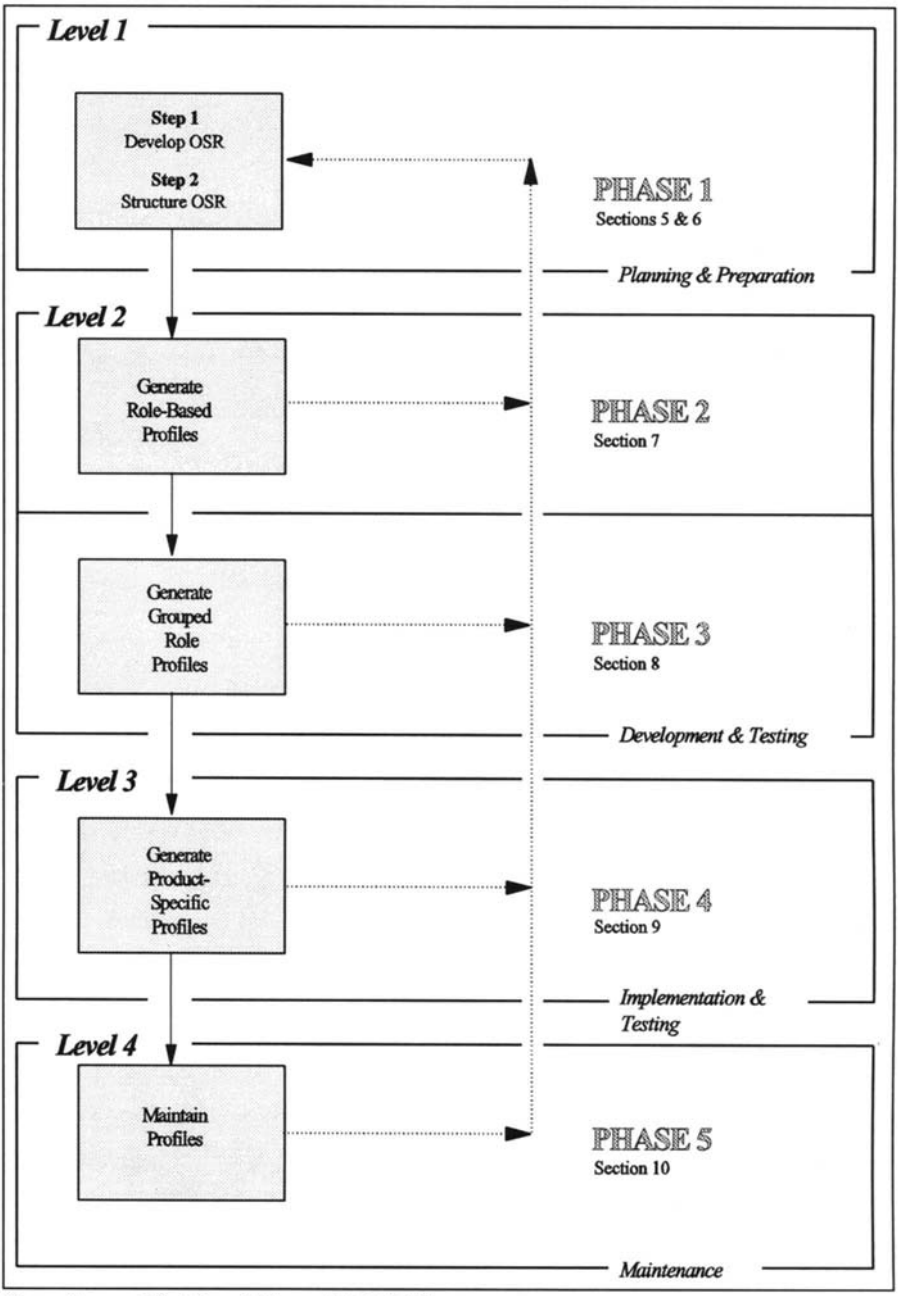

Figure 2 High-Level View of M-MAPS

Operational Phases
Within the functional levels of M-MAPS, five phases were identified on an operational level. These phases form part of a cyclic and sequential flow of operation and basically provide an answer to the question of HOW with regard to the development of profiles. These phases will be discussed in sections 5-10 (cf figure 2).

Flow of Operation
The phases comprising M-MAPS will always be executed sequentially (indicated with solid line directional arrows), starting from phase 1 and ending with phase 5. Provision is made for the execution of the phases preceding a current phase of execution, always commencing from phase 1 (indicated with dotted line directional arrows). This backtracking mechanism is defined as a regress and serves the purpose of facilitating the incorporation of any changes providing that they be initiated from phase 1.

This concludes the summary of the M-MAPS methodology as contained in the MAPS model. The phases comprising M-MAPS are now discussed individually in sections 5-10.

5. PHASE 1: STEP 1 - DEVELOP OSR

Phase 1: step 1 of M-MAPS is utilized exclusively for the gathering of the information necessary for the generation of comprehensively defined information security profiles. This information is incorporated in the OSR, in the form of specific organizational policies, documents or databases. Each of these will now be discussed and examples provided.

D1 - Information Security Policy
The information security policy is constituted by higher level policy statements which are indicative of top management's directives in regard of information security. These type of statements normally supply a broad outline rather than specifics which can be implemented directly [Ston89]. The approach therefore, is to require that D1 be used by the OSR administrators as mandate in the compilation of the other components of the OSR (D2-D5) which are to be discussed shortly. Since the profiles are generated from these components, it follows that D1 will be enforced by virtue of its use in the compilation of the OSR.

In some cases, a direct implementation of statements included in D1 is possible, for example the statement:

> An OSR administrator may not be associated with any role resorting in his/her administrative domain.

This statement can be converted to a rule and used to check generated profiles against, which in effect constitutes a direct implementation of the statement.

D2 - Personnel Database
The personnel database should minimally contain the information necessary to establish

user-to-role relationships, viz the employee initials and surname, user identification numbers (UIDs) and user role allocations. Additional specifications such as workstation, time-of-day and day-of-week restrictions may also be included. An example of such a record from a personnel database is:

Initials &Surname	UID	Role Allocation	Workstation
H Suzmann	CS343093	JD-A&D003 Store Supervisor	S3031

D3 - Job Descriptions

The job descriptions database contains functional analyses of all roles identified in the working environment. These analyses assign actions, responsibilities and rights to each role. The job description for the role *JD-A&D003* is as follows:

JD-A&D003	Store Supervisor
003-01	Ship orders
003-02	Print delivery notes
003-03	Print partially shipped orders
003-04	Create purchase requisitions
003-05	Print purchase requisitions

D4 - Resource Database

The resource database contains specifications regarding the resources to be protected in the form of A//R couplings for each statement included in the job descriptions database. An example is not provided at this stage, as the example provided in section 3.4 refers. One issue that needs attention however, is the issue of the granularity of an A//R analysis. Theoretically speaking, an A//R analysis should contain all actions regarding the resources (hardware, software, data and information components) which are affected and/or effected in the execution of the transaction it represents. Practically speaking, some trade-offs will have to be made in terms of restricted A//R analyses and the risks posed by these restrictions. One criteria could for example be to measure the costs incurred in the enforcement of a certain chosen level of granularity for A//R analyses, against the possible loss in the case of a security breach. Another definite factor would be the capabilities of the security system(s) to be used.

D5 - Business Objectives

The business objectives database is used to contain any critical business functions, objectives, standards and procedures. These are issues that are related to business on a lower level, ie issues that are not universally applicable but more application-specific. The idea is to convert these specifications into a *Rules Database* (cf section 6) against which the contents of generated profiles may be checked. The content-specification of this database should be provided in a natural language. The following are examples of what might be included in the business objectives database.

Example 1	The rights associated with the tasks of ordering, receiving and shipping equipment may not be assigned to any one person, as this will allow fraud. It is acceptable, however, for an employee to receive rights regarding any two of these tasks.
Example 2	The act of creating supplemental purchase orders may under no circumstances be executed by an employee who creates purchase orders.

This concludes the discussion on step 1 of phase 1, the end-product of this step being the components D1-D5 as discussed above. These components now have to be converted to a format which facilitates the automatic generation of profiles. This action is executed during phase 1: step 2 and is addressed in the following section.

6. PHASE 1: STEP 2 - CONVERT OSR

During step 2 of phase 1, the OSR as constituted by D1-D5, is converted to a predicate-based form using R-MAPS, which accommodates the computerized representation, retrieval as well as manipulation of this information. This will facilitate the objective of automating the process of profile generation. After completion of this step, the OSR is constituted by four components, which are extracted from D1-D5 as follows:

D2	=>	F1
D3	=>	F2
D4	=>	F3
D1+D5	=>	F4

Each of the components F1-F4 are now discussed in terms of representation according to R-MAPS specifications. As mentioned earlier, R-MAPS is a predicate-based representational tool which is used to represent the components D1-D5 of the OSR. Note that the following conventions hold:

UPPERCASE_BOLD	: Predicate symbol
UPPERCASE	: Constant symbol
lowercase	: Variable symbol
lowercase_italics	: Function symbol

; = separation character for elements of a set
, = separation character for terms of a predicate / function

F1 - Personnel Database

The content of F1 is constituted by a direct mapping from D2 with each predicate relating to one row in the personnel database. The proposed R-MAPS representation is as follows:

ASSOCIATE-ROLES($\{r_1; ..; r_i\}$, u, $\{id_1; ..; id_j\}$, $[\{f_1; ..; f_n\}]$)

$\{r_1; ..; r_i\}$	= set of roles	(i>=1)
u	= user name	
$\{id_1; ..; id_j\}$	= set of user identification numbers	(i>=1)
$[\{f_1; ..; f_n\}]$	= optional set of functions	(n>=0)
f_1	= *workstation*(x_1, .., x_i)	(i>=1)

Note that the set of functions is indicated as optional in the proposed representation. It is included in the representation in order to facilitate customization through the use of functions which may be used to represent additional columns added to the personnel database (D2). For example the function workstation, assigns a workstation to each role in $\{r_1; ..; r_i\}$. Another function could be used to represent time-of-day restrictions regarding allocated roles.

EXAMPLE

ASSOCIATE-ROLES({JD-A&D003},H SUZMANN,{CS343093}, {*workstation*(S3031)})

F2 - Transaction Database

The content of the transaction database (F2), is extracted from the job descriptions database (D3) and may be represented as follows:

EXEC-TRANSACTIONS(r, $\{t_1; ..; t_p\}$)

r	= role of consequence	
$\{t_1; ..; t_p\}$	= set of transactions associated with r	(p>=1)

EXAMPLE

EXEC-TRANSACTIONS(JD-A&D003, {003-01; 003-02; 003-03; 003-04; 003-05; 003-06; 003-07})

F3 - A//R Database

The A//R database (F3) is converted from the resource database (D4) and may be represented as follows:

TRANSACTION-SCOPE(t, $\{a//r(a, r)_1; ..; a//r(a, r)_j\}$)

t	= transaction of consequence	
$\{a//r(a, r)_1; ..; a//r(a, r)_j\}$	= set of A//R couplings associated with t	(j>=1)

EXAMPLE

TRANSACTION-SCOPE(003-01,
{*a//r*(DACCESS, S3031); (Workstation)
a//r(DACCESS, CTNSERV001); (Server)
a//r(EXECUTE, F:\USER\SYSTEM.BAT); (*.bat* file starts application)
a//r(EXECUTE, F:\APPL\SYSTEM.EXE); (System shell displays main menu)
a//r(EXECUTE, F:\APPL\SHIP.EXE)} (Program executes shipping)

F4 - Rules Database

The contents of the rules database are comprised by the information security policy (D1) and business objectives database (D5), which were compiled during phase 1: step 1. These documents are composed as natural language statements. In order to represent these statements in a predicate-based format, the constructs of the predicate calculus [Pit88, Wii87] are used. This comprises the composition of well-formed formulas which consist of atomic formulas and connectives in a combination as required to represent the statement. The following general representation is proposed, as there are no hard and fast rules:

PREDICATE($term_1$, .., $term_t$) [{AND; OR; =>; '} **PREDICATE**($term_1$, .., $term_t$)]

($term_1$, .., $term_t$) = terms of the predicate (constants,variables and/or functions) (t>=1)
[] = optional construction of compound statements
{AND; OR =>; '} = set of connectives

The above representation indicates that the statements will be compiled as predicates, optionally combined by way of the indicated connectives to form compound statements. Note that the negation of a predicate is indicated with an apostrophe (').

EXAMPLE 1 (Representation of D5 - Example 1 in section 5)

ALLOW-TRANSACTIONS(u, {003-01; 003-02; 003-03; 003-04; 003-05})
OR
ALLOW-TRANSACTIONS(u, {003-04; 003-05; 003-06; 003-07})
OR
ALLOW-TRANSACTIONS(u, {003-01; 003-02; 003-03; 003-06; 003-07})

EXAMPLE 2 (Representation of D5 - Example 2 in section 5)

ALLOW-TRANSACTIONS(u, {004-03}) OR **ALLOW-TRANSACTIONS**(u, {004-04})

Note that there are usually several choices about how to represent natural language statements in the predicate calculus. The designer of the representation selects the alphabet of predicates and terms and defines what they will mean [Nil80].

This concludes the discussion on step 2 of phase 1. The components F1-F4 which now constitute the OSR, will subsequently be used to structure the security profiles. This process is discussed in section 7.

7. PHASE 2 - GENERATE ROLE-BASED PROFILES

After completion of phase 1: step 2, the OSR is in a format as may be used to generate role-based profiles from. This basically comprises selecting from F1-F3 and structuring the profiles according to this information. Note that F4 is not used at this point, but after the profiles have been generated, to check for inconsistencies. The example for the role of *Store Supervisor* that has been used so far, will yield the following role-based information security profile:

ROLE(JD-A&D003)
USER(H SUZMANN, *uid*(CS343093), *workstation*(S3031))
TRANSACTIONS(003-**)
A//R(003-01, *action*(EXECUTE), *resource*(SHIP.EXE))
A//R(003-02, *action*(EXECUTE), *resource*(DNOTE.EXE))
A//R(003-03, *action*(EXECUTE), *resource*(P-ORD.EXE))
A//R(003-04, *action*(EXECUTE), *resource*(P-REQ.EXE))
A//R(003-05, *action*(EXECUTE), *resource*(PP-REQ.EXE))
A//R(003-06, *action*(EXECUTE), *resource*(RECEIV.EXE))
A//R(003-07, *action*(EXECUTE), *resource*(PREC.EXE))

Note that the A//R analyses have been scaled down to include only the program which has to be executed to perform the duty. Also note that the profile is still in a predicate-based form, as further manipulation is required. This format will be used up to the execution of phase 4, when product-specific profiles are generated.

Testing
After role-based profiles have been generated for all roles included in the personnel database, it is imperative that these profiles be checked for inconsistencies. It is for example possible that irregularities are contained in the original policy specifications, which are contradictory to the rules contained in the rules database (F4). In the example cited earlier, the profile created for the role of *Store Supervisor* and associated with *H Suzmann*, granted rights for the ordering, receiving and shipping of equipment. This is not acceptable, as indicated in example 1 provided in the rules database. This error will not be corrected at this stage, but rather in section 10 as an example of handling the maintenance of profiles.

8. PHASE 3 - GENERATE GROUPED ROLE PROFILES

The inclusion of phase 3 execution in M-MAPS stems from practical considerations. As it will be impracticable to implement a separate profile for each identified role, it becomes essential to move towards a grouped structure (eg departmental profiles). Grouping may be implemented on an organizational (divisions, departments, etc), role, resource and/or ad hoc basis. This is achieved by searching for communal entities within the generated role-based profiles, depending on the selected grouping criteria. Examples of grouping criteria are:

- All roles resorting in an organizational unit
- All roles with similar access rights / prohibitions
- All resources with similar access rights / prohibitions
- Matching roles with regard to any specified condition

Testing
After grouping, a new arrangement of role-based and grouped role profiles serves as output. These profiles once again have to be checked against F4, the rules database. A good example of why this is necessary, is to check that accumulating rights being assigned to users in the form of multiple role allocations, do not exceed restrictions specified in F4.

9. PHASE 4 - GENERATE PRODUCT-SPECIFIC PROFILES

The next logical step after completion of phase 3, is to compile product-specific profiles in a format as required by the access control packages which are to be used.

With regard to this phase, suffice to say that a limited implementation of the MAPS model has been developed by a prominent company in the banking sector. This application was developed in the mainframe environment for use with the Top Secret access control package. All requests for changes to security profiles are firstly channelled to this product, which handles them according to procedures contained in the program. It therefore basically serves as a front-end to the Top Secret package. Resource owners (equivalent to the MAPS OSR administrators) are assigned the responsibility of specifying requirements regarding the resources in their administrative domains. Nothing can be done without the approval of the resource owners (ROs). An interface has been written which converts the instructions received from the ROs, to profiles in a format as required by Top Secret. The program furthermore updates the Top Secret profiles on a daily basis, after performing a search to check for the relevance of profile contents. For example, if in a profile access is granted to a non-existing resource (eg deleted dataset), it will be removed from the profile. Developmental work on this implementation is continuing and certainly warrants discussion in more detail, in a separate paper.

10. PHASE 5 - MAINTAIN PROFILES

Inherent to the process of administering logical access control is coping with change, whether it originates on an operational (eg launching of a new project), managerial (eg change of policy) or

technological (eg introduction of smartcards as additional authentication measure) level. Phase 5 execution therefore comprises the continued updating of profiles to reflect any changes.

As an example of the maintenance of profiles, the inconsistency with F4 regarding the example profile for role *JD-A&D003*, will now be discussed. As mentioned earlier, the problem with this profile is the fact that the user can order, receive and ship equipment, which is inconsistent with the rule in example 1 supplied for the rules database. Let us assume that correction of this situation has to be done as part of maintaining the generated profiles.

PHASE 1
Firstly the OSR administrator will be alerted to the problem. The only logical way to solve the problem, is to create a separate role, for which one or two of the tasks of ordering, receiving and shipping is included in the job description. The personnel database (D2) and the job descriptions database (D3) therefore have to be updated in this regard. Note that the new role has to be assigned a different user than *H Suzmann*, in order to ensure the required division of duties. Next, the updated OSR will be converted according to R-MAPS specifications.

PHASE 2
The updated OSR will result in the generation of new role-based profiles, reflecting the changes. Testing at this stage will succeed as the rule contained in F4 is now adhered to.

PHASE 4
Phase 4 execution will render updated product-specific profiles. (Phase 3 is not mentioned here as it is irrelevant.)

Note how these changes are incorporated by backtracking to phase 1. This is in accordance with the discussion of MAPS (figure 2) in section 4.

11. IN CONCLUSION

In this paper the MAPS model was discussed in its capacity of automating the generation of information security profiles. The application of the methodology M-MAPS ensures that the generated profiles and the OSR are kept in synchronization at all times. This ensures that there is always an up to date point of reference to work from. It will always be possible to see who originated the contents of a profile (*OSR administrator*) and why it was done (*Job Descriptions*). Furthermore, the rules database ensures that the high-level as well as more specific policies of an organization are applied at all times. It will not be possible for an administrator to unwittingly include access rights in a profile, which compromise security.

The novelty of MAPS lies in the use of organizational policies as vehicle for the automation of profile generation. Access rules are compiled as a direct corollary to the policies concerned. The success achieved so far with the development of the application based on MAPS (cf section 9), is considered as proof-of-concept. Although this application was developed in the mainframe environment, many possibilities and definite incentive exist for expansion to distributed and networked environments. The complexity of today's computing environments demands that

solutions be found for the dilemma that security administration has become. We believe that MAPS provides such a solution, which has also been shown to be viable in practice.

REFERENCES

[Abr91] M Abrams et al, *Generalized Framework for Access Control: Towards Prototyping the Orgcon Policy*, in Proceedings of the 14th National Computer Security Conference, pp 257-266, October 1-4, 1991.

[Com91a] Computer Associates International Inc, *CA-TOP SECRET: The Innovative Security Package for MVS - Product Concepts and Facilities Manual*, 1991.

[Com91b] Computer Associates International Inc, *CA-ACF2: The Access Control Facility for MVS - Product Concepts and Facilities Manual*, 1991.

[Dob89] JE Dobson & JA McDermid, *Security Models and Enterprise Models*, in Database Security II, Status and Prospects, ed CE Landwehr, Elsevier Science Publishers, pp 1-39, 1989.

[Fer] D Ferraiolo & R Kuhn, *Role-Based Access Controls*, Details of Publication Unknown.

[Fer91] D Ferraiolo & K Ferraiolo, *Another Factor in Determining Security Requirements for Trusted Computer Applications*, in Proceedings of the 14th National Computer Security Conference, pp 37-44, October 1-4, 1991.

[IBM92] IBM World Trade Corporation, *Resource Access Control Facility (RACF): General Information*, Document Nr GC28-0722-15, 16th Edition, April 1992.

[Jon91] D Jonscher & W Gerhardt, *A Role-Based Modelling of Access Control with the Help of Frames*, in Proceedings of the 7th International Conference and Exhibition in Information Security, pp 131-142, 1991.

[Nil80] NJ Nilsson, *Principles of Artificial Intelligence*, Tioga Publishing Company, 1980.

[Pit88] J Pitrat, *An Artificial Intelligence Approach to Understanding Natural Language*, North Oxford Academic Publishers Ltd, 1988.

[Pot92] D Pottas & SH von Solms, *MAPS - Model for Automated Profile Specification*, in Proceedings of the IFIP TC11 8th International Conference on Information Security, pp 131-144, 27-29 May 1992.

[Pot93a] D Pottas & SH von Solms, *Superseding Manual Generation of Access Control Specification - From Policies to Profiles*, in Proceedings of the IFIP TC11 9th International Conference on Information Security, pp 327-342, 12-14 May 1993.

[Pot93b] D Pottas & SH von Solms, *The Automatic Generation of Information Security Profiles*, in Proceedings of the CompSec 93 10th World Conference on Computer Security, Audit and Control, 20-22 October 1993.

[Ster91] DF Sterne et al, *An Analysis of Application Specific Security Policies*, in Proceedings of the 14th National Computer Security Conference, pp 25-36, October 1-4, 1991.

[Ston89] J Stoner et al, *Management*, 4th edition, Prentice-Hall, 1989.

[Wii87] SA Wiitala, *Discrete Mathematics: A Unified Approach*, Mc Graw-Hill Book Company, 1987.