Electronic services security management for the public administration

B. SZAFRAŃSKI, J. WILK

boleslaw.szafranski@wat.edu.pl, jaroslaw.wilk@wat.edu.pl

Military University of Technology, Faculty of Cybernetics
Kaliskiego Str. 2, 00-908 Warsaw, Poland

The public administration in Poland is constantly extending and improving electronic services provided to citizens, business and itself (e.g. one ministry providing services to another public administration institution). New expectations like integration using the multilateral interoperability model, the service oriented approach, different security categories and groups within different institutions are creating the requirement for the appropriate security model. According to the authors current models do not meet the new requirements, but can be used as a base for the new approach. The lattice and its variations in different models (Denning, Sandhu, Szafrański) was selected by authors to create the new electronic services security management model for integration platforms which are supporting realization of public tasks. This publication focuses on practical aspect of the new model and presents the case study on a public administration e-services example. Detailed definition of the model itself can be found in previous authors' articles.

Keywords: e-services, security, SOA.

1. Introduction

Over the last decade new approach to information processing in IT systems has become almost obligatory – change from the data oriented to the service oriented architecture. New architecture has brought new systems like cloud solutions and integration platforms together with new challenges.

One of them is a security management, which according to the authors is not keeping up with rapid changes. Data oriented models are still used for composite services security which is constantly leading to more complex to manage solutions. Current most popular models (fifteen models were analysed in previous author's research [1]) do not meet the new requirements.

In previous authors' publication [2] the Service Oriented Architecture, interoperability models and requirements for new security approach together with the new model proposition was presented.

This article discusses case studies of authors' information security model for service oriented systems used in public administration. It also consists of basic introduction to the proposed model and comparison to the approach used nowadays.

Authors' decision to focus on public administration services and electronic platforms was explained in previous publication [3]. The main reason is the observed problem with public e-services integration using the multilateral interoperability model. The lack of good security model is slowing down the rollout of new integrated public e-services with is impacting e-administration – State 2.0 [4] program in Poland.

The paper is organized as follows: first it defines the main components of author's model (section 2). Sections 3–6 presents case study of model usage. Section 3 is a case study introduction which focuses on the comparison of currently used solutions and the one presented by authors. Section 4 and 5 presents examples of data and execution management for atomic services and section 6 presents an example for data and execution management for composite services. In section 7, the conclusions of the paper are presented.

2. Main components of the proposed model

Although authors' model was explained in details in previous publications [1], [5], [6] it is necessary to define all basic parameters which are used in following case study.

The authors' information security management model (SM) for integration platforms is built from data access management model and e-service execution management model.

Security Management (SM) model for integration platforms consist of elements:

$$SM = \langle P, D, K, T, E, B, U, \rho, \tau, \delta, VF \rangle \qquad (1)$$

where:

- P – a set of entities,
- D – a set of data units,
- K – a set of protection classes,
- T – a set of operations,
- E – a set of atomic electronic services,
- U – a set of composite electronic services,
- B – a set of categories of an authorized execution,
- ρ – an access relation,
- τ – a scope of operations relation,
- δ – a relation of an authorized service execution,
- VF – a set of restrictions functions (restrictions generator for the security management model).

Additionally, in the set of atomic electronic services there are services, which are called initial and are started by an authorized entity (object): $R \subset E$.

Data units (elements of set D) are described by data protection classes, which are members of set K for example: public, confidential, secret, top secret. Electronic services are described by categories of an authorized execution, which are members of set B – for example: universal, special, restricted.

Each entity p from set P has the following parameters determining its access level:

- a protection class from the K set,
- a range of an authorized operation from T set – for example: reading,
- a category of an authorized execution from B set.

Detailed definition of a composite e-service u from the set U is provided in previous author's publication [1] – for the purpose of this article short introduction is provided.

$$u = (F, r, g, z) \qquad (2)$$

- $F \subset E$ – a set of atomic e-services being part of composite e-services,
- $r \in R$ and $r \in F$ – an initial e-service (placed on integration platform) which is also atomic e-service,
- g – a function transforming enforcements into started atomic e-services (defined in details in [2] – not relevant for this article purpose),
- z – a function transforming atomic e-services into new enforcements (defined in details in [2] – not relevant for this article purpose).

An access relation is built on pairs of protection classes: ρ c $K \times K$ and determines the hierarchy of allowed accesses to the data – for example: top secret > secret > confidential > public. The symbol > describes the pair relation – for example: secret > confidential means that pair (secret, confidential) belongs to relation ρ. A scope of authorized operations relation is built on pairs of operations: τ c $T \times T$ and defines a hierarchy of operations – for example: update > delete > write > read > search. Both relations meet requirements of the lattice theory, which allows building protection classes and operations lattices. It is allowed to build scope of operations on the combination of operation attributes changing the τ relation definition (like it is presented in the following example – Figure 2).

The entity is allowed to access the data unit using specific operations if the protection class of the entity and the protection class of the data unit satisfy the ρ relation and the scope of operations of the data unit and the scope of the operation being performed satisfy δ relation.

Defined relations are generating lattices which implies that all elements have to meet all lattice theory requirements [7]. An access relation is generating a hierarchical lattice from the protection classes set. Figure 1 presents an example of a protection classes lattice (arrows define the relation ρ). Similar as in the Sandhu model [8] it is possible to extend the set of partially ordered with a maximal element (eg. "Syshigh" from Sandhu model) or a minimum (eg. "k0: uncategorized") so they satisfy conditions of the lattice.

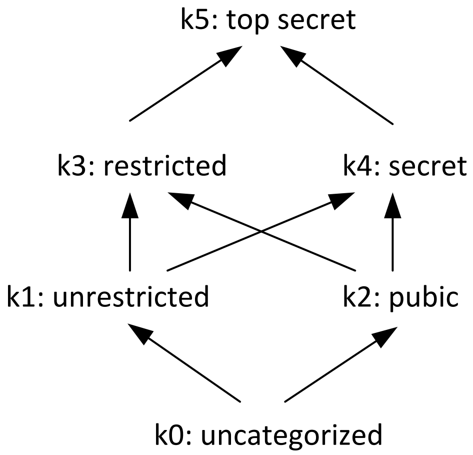

Fig. 1. An example of a protection classes lattice (with additional k0 element)

A scope of operations relation is generating a hierarchical lattice from the operations set Figure 2 presents an example of a scope of operations lattice based on Linux permissions (R: read, W: write, X: execute / access directory contents – (dash): additional symbol for no operation). In this case scope of operations is the combination of three operation attributes: $\tau \subset T^3 \times T^3$ where:

$$\mathrm{T} = \{R, W, X, -\} \tag{3}$$

In the proposed model execution is extracted from operations and applied only to services, so X should be interpreted only as an 'access directory contents' operation.

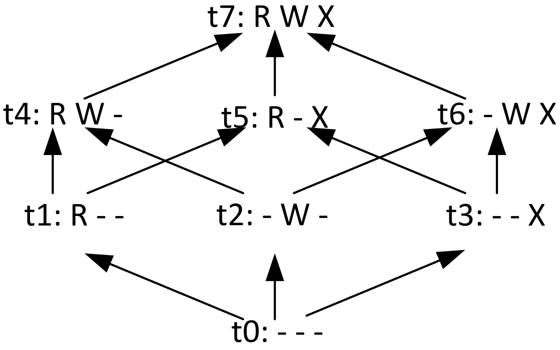

Fig. 2. An example of a scope of operations lattice

An authorized service execution relation is generating a hierarchical lattice from the categories of an authorized execution set. An example of an execution categories lattice is presented on Figure 3 (additional sys-low element B0 is added the same as to the protection classes lattice from Figure 1).

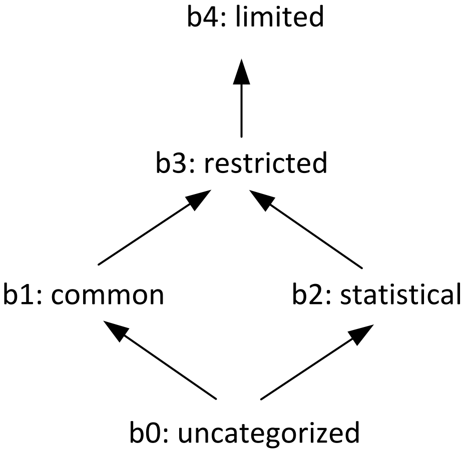

Fig. 3. An example of a scope of execution categories lattice (with additional b0 element)

A set of restrictions functions VF (restrictions generator for the security management model) was defined in details in previous authors' publication [1] as an access management and execution management restrictions functions. Additionally to already defined functions (V_1 to V_7 from data access management model and V_8 to V_{10} from e-service execution management model), there are three which are used for composite electronic services.

V_{11} – function checks correctness of data processing within the composite e-service u:

$$V_{11}: P \times U \rightarrow \{0,1\} \tag{4}$$

The verification process iterates over all atomic e-services F being part of composite e-service u.[1]. It checks if the access and the scope of operations relations are satisfied (within V_1 to V_7 functions defined in previous author's publications [1], [2]).

V_{12} – function checks correctness of execution of atomic services within the composite e-service u.

$$V_{12}: P \times U \rightarrow \{0,1\} \tag{5}$$

The verification process iterates over all atomic e-services F being part of composite e-service u. It checks if the relation of an authorized service execution is satisfied (within V_8 to V_{10} functions defined in previous author's publications [1], [2]). The sequence of all enforcements within an electronic service execution tree [1] is verified. The enforcement, which is starting a composite service has to be launched by the entity (user), all next enforcements are launched by electronic services.

Function V_{12} enforces very strict security policy – not only the entity must have sufficient authority to enforce the service, but services which are launching other services are checked. As a result, this prevents the placement of e-services with higher execution category at the end of the composite service tree as the successor of a service with a lower category. It is possible to mitigate the presented strict policy through the use of a softer version of the model (adapted for practical use), where function V_{12} is verifying all enforcements only form the entity execution category level (practically in this case the entity execution category is checked with the highest execution

[1] As described in formula (2)

category of atomic e-services F being part of composite e-service u).

V_{13} – function to verify correctness of data processing and services execution (full security management model) for composite e-service u.

$$V_{13}: P \times U \rightarrow \{0,1\} \qquad (6)$$

$$V_{13}(p,u) = V_{11}(p,u) \cdot V_{12}(p,u)$$
$$\text{where } p \in P, u \in U \qquad (7)$$

From the point of view of the full security management model for composite services it only allowed to execute a composite electronic service if and only if conditions of two security models – data access security and service security, are met.

3. Case study – introduction

This section presents the case study of authors' model for a simple composite electronic service with atomic services originating from different domain platforms. The presented e-service is integrated on an integration platform which in this example can be "e-PUAP" (Electronic Platform of Public Administration Services) or "Source" (original name in Polish is "Źródło" – a system that is integrating public registers). Domain platforms for this example purpose are systems owned by different ministries (Ministry of Justice – e-service which has access to the National Court Register and the of Ministry of the Interior and Administration – e-service which has access to PESEL Register). The composite e-service is allowing an official or citizen go get information if a person (based on PESEL) number was convicted or not. Each office / ministry has different IT systems, different security user groups and policies which makes it difficult to automatically define security management rules to access both e-services simultaneously (as a composite e-service). By default citizen can only request his own data but officials can check any citizen if needed (for internal administration processes). This means that for composite e-service execution security management system has to check access rights and access level (for example: read, write) to all atomic services and data entities originating from different systems.

Currently if there is no integrated services security model implemented usually a new access group is created for official who can execute the composite e-service. This means that:

- there is no automation in e-service integration (need of expert every time new integrated e-service is created or changed),
- new security groups are often created specifically for new integrated e-services,
- security compliance is not checked for every component in every integrated system.

With use of the authors' model it is possible to create "super lattices" [6] access, execution and operations and then to process security management on integration platform according to calculations presented below.

Case study for integrated e-service is followed by examples for an atomic e-service for access and execution management models.

4. Data management security example atomic e-services

The protection classes lattice KL and the scope of the operations lattice TL are defined for this example purpose the same as on Figure 1 and 2.

Two users are analyzed p_1 i p_2 (who according to model are entities) with protection classes as indicated below:

$$V_1: P \cup D \rightarrow K \qquad (8)$$

$$V_1(p_1) = secret,\ V_1(p_2) = public \qquad (9)$$

The atomic service e_1, is forcing operations on two data units: d_1 (read operation: R--) and d_2 (update operation: RW-). Figure 4 presents the graphical representation of the analysed atomic service.

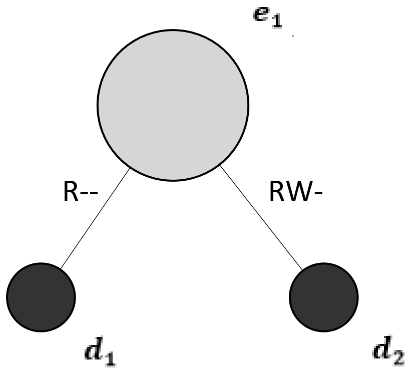

Fig. 4. The atomic service – example

Data units have the following protection classes:

$$V_1(d_1) = public,\ V_1(d_2) = restricted \qquad (10)$$

Additionally, entities (users) have the following set of allowed operation for data units d_1 and d_2 being part of analysed service.

Entity p_1 has „update" access right to both data units:

$$V_4: P \times D \to T^3 \tag{11}$$

$$V_4(p_1, d_1) = RW-, V_4(p_1, d_2) = RW- \tag{12}$$

Entity p_2 has „read" access right to the data unit d_1 and no access rights to data unit d_2:

$$V_4(p_2, d_1) = R--, V_4(p_2, d_2) = --- \tag{13}$$

Considering the case of an entity p_1 there will be two verifications when it tries to launch e_1 service.

1. Verification of the access to all data units within the service (iteration over all pairs of entity and data unit within e-service):

$$V_7: P \times E \to \{0,1\} \tag{14}$$

$$V_7(p_1, e_1) = V_6(p_1, d_1) \cdot V_6(p_1, d_2) \tag{15}$$

2. Verification of the access from the point of view of data flow and scope of operations for each data unit. General formula:

$$\bigwedge_{l_1, l_2 \in P \cup D} V_6(l_1, l_2) = V_2(l_1, l_2) \cdot V_5(l_1, l_2) \tag{16}$$

Case study example:

$$V_6(p_1, d_1) = V_2(p_1, d_1) \cdot V_5(p_1, d_1) \tag{17}$$

$$V_6(p_1, d_2) = V_2(p_1, d_2) \cdot V_5(p_1, d_2) \tag{18}$$

General formula:

$$V_2: (P \cup D) \times (P \cup D) \to \{0,1\} \tag{19}$$

Case study example:

$$V_2(p_1, d_1) = 1 \tag{20}$$

$$\text{because pair } \left(V_1(p_1), V_1(d_1)\right) \in \rho \tag{21}$$

$$\text{pair } (secret, public) \in \rho \tag{22}$$

$$V_2(p_1, d_2) = 1 \tag{23}$$

$$\text{because pair } (secret, resticted) \in \rho \tag{24}$$

General formula:

$$V_5: (P \cup D) \times (P \cup D) \to \{0,1\} \tag{25}$$

Scope of performed operation general formula:

$$V_4: P \times D \to T^3 \tag{26}$$

Case study example:

$$V_5(p_1, d_1) = 1 \tag{27}$$

$$\text{because pair } \left(V_4(p_1, d_1), V_3(p_1, d_1)\right) \in \tau \tag{28}$$

$$\text{pair } (RW-, R--) \in \tau \tag{29}$$

$$H_5(p_1, d_2) = 1 \tag{30}$$

$$\text{because pair } (RW-, RW-) \in \tau \tag{31}$$

Entity p_1 will be authorized to launch service e_1 from the data security management perspective, because:

$$V_7(p_1, e_1) = 1 \tag{32}$$

Similar verification is performed for the p_2 entity, who obtains a negative result (denial of service execution e_1), because it does not have sufficient protection class to initiate the flow of data between itself and a data unit d_2 and at the same time does not have permission to perform any operations on the data unit d_2 (as minimum it must have update rights):

$$V_7(p_2, e_1) = 0 \tag{33}$$

$$V_2(p_2, d_2) = 0 \tag{34}$$

$$\text{because pair } (public, resticted) \notin \rho \tag{35}$$

$$V_5(p_2, d_2) = 0 \tag{36}$$

$$\text{because pair } (---, RW-) \notin \tau \tag{37}$$

5. Execution management security example – atomic e-services

BL is the scope of execution categories lattice as in the Figure 3.

Two user p_1 and p_2 (called entities according to the model) have the following execution categories permissions:

$$V_8: P \cup E \to B \tag{38}$$

$$V_8(p_1) = limited, \ V_8(p_2) = statistical \tag{39}$$

The atomic electronic service e_1 has the following execution category:

$$V_9 : E \rightarrow B \tag{40}$$

$$V_9(e_1) = restricted \tag{41}$$

The first case is when an entity p_1 tries to execute the e_1 electronic service (security management system will perform following check). General formula:

$$V_{10} : (P \cup E) \times E \rightarrow \{0,1\} \tag{42}$$

Case study example:

$$V_{10}(p_1, e_1) = 1 \tag{43}$$

$$\text{because pair } (V_8(p_1), V_9(e_1)) \,\epsilon\, \delta \tag{44}$$

$$\text{pair } (limited, restricted) \,\epsilon\, \delta \tag{45}$$

Entity p_1 will be permitted to execute the service e_1 from the e-services security management perspective.

The second case is when entity p_2 tries to execute the e_1 electronic service (security management system will perform following check).

$$V_{10}(p_2, e_1) = 0 \tag{46}$$

$$\text{because } (statistical, restricted) \notin \delta \tag{47}$$

The p_1 entity obtains a negative result (denial of the e_1 service execution), because it does not have sufficient execution permissions to launch the e_1 service.

6. Data and execution management security example – composite e-services

The protection classes lattice KL, the scope of operations lattice TL and the scope of execution categories lattice BL are defined for this example purpose the same as on Figure 1, 2 and 3.

KL, TL and BL are super-lattices created from domain systems lattices, which were defining security boundaries within one administration institution (for example – ministry).

Two users p_1 i p_2 have the following protection classes and execution security categories:

$$V_1(p_1) = secret,\ \ V_1(p_2) = public \tag{48}$$

$$V_8(p_1) = limited,\ V_8(p_2) = statistical \tag{49}$$

The composite electronic service u_1**,** analysed in this example, is graphically presented on Figure 5.

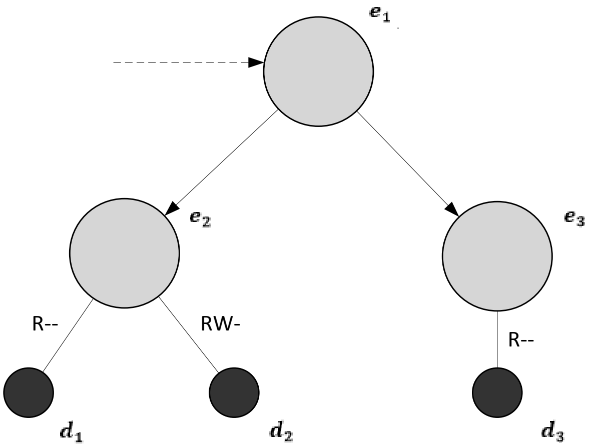

Fig. 5. Graphical representation of the composite electronic service u_1 – example

Atomic services e_1, e_2, e_3 have the following execution security categories:

$$V_9(e_1) = restricted, \quad V_9(e_2) = \\ = restricted,\ V_9(e_3) = limited \tag{50}$$

Data entities, used in the example, have the following protection classes:

$$V_1(d_1) = public,\ V_1(d_2) = restricted, \\ V_1(d_3) = restricted \tag{51}$$

The atomic service e_1 is the initial service published on the integration platform (for example – “e-PUAP”, e_2 and e_3 are e-services originating from different domain systems (for example: one is published by the Ministry of Justice and another by the Ministry of the Interior and Administration).

Following operations are defined to run on selected data units within the u_1 composite electronic service: read operation t_1: R--, update operation t_2: RW- and write operation: t_3: R--.

Entity (user) p_1 has access rights to all three data units with following operations level:

$$V_4(p_1, d_1) = RW-, V_4(p_1, d_2) = RW-, \\ V_4(p_1, d_3) = RW - \tag{52}$$

Entity (user) p_2 has access rights only to d_1 and d_3 data units with following operations level and no access rights to d_2 data unit:

$$V_4(p_2,d_1) = R--, V_4(p_2,d_2) = ---, V_4(p_2,d_3) = R-- \quad (53)$$

When the user p_1 is trying to launch u_1 composite electronic service the security management system will perform verification as follows.

From the data and e-services security perspective:

$$V_{13}(p_1,u_1) = V_{11}(p_1,u_1) \cdot V_{12}(p_1,u_1) \quad (54)$$

The data security management part will be checked for every atomic electronic service and then for every operation and a data unit within the service. The result will be positive – all pairs are within the access or scope of operations relations.

$$V_{11}(p_1,u_1) = V_7(p_1,e_1) \cdot V_7(p_1,e_2) \cdot V_7(p_1,e_3) = 1 \cdot V_6(p_1,d_1) \cdot V_6(p_1,d_2) \cdot V_6(p_1,d_2) = V_2(p_1,d_1) \cdot V_5(p_1,d_1) \cdot V_2(p_1,d_2) \cdot V_5(p_1,d_2) \cdot V_2(p_1,d_3) \cdot V_5(p_1,d_3) = 1 \quad (55)$$

The service security management can be divided into two cases:

- Strict version of the model:

$$V_{12}(p_1,u_1) = V_{10}(p_1,e_1) \cdot V_{10}(e_1,e_2) \cdot V_{10}(e_1,e_3) = 0 \quad (56)$$

In this case although entity can execute initial service e_1 the whole process will be blocked as e_1 service is not permitted to launch e_3 service.

- Soft version of the model:

$$V_{12}(p_1,u_1) = V_{10}(p_1,e_1) \cdot V_{10}(p_1,e_2) \cdot V_{10}(p_1,e_3) = 1 \quad (57)$$

In this case, the result will be positive, because the entity is entitled to launch all atomic services occurring within the composite service.

Similar procedure will be performed when the p_2 entity is trying to launch the composite service. It will end up with the negative result of the data security check due to inappropriate (too low) access and operations levels.

As proved on presented example, with use of authors' model:

- there is automation in e-service integration possible (maximum there is need of expert only one time during systems integration to supervise super-lattice creation [6], e-services can be created and changed with automatic security level recalculation with use of presented verification schema),
- no need of new security groups (available security groups are integrated in super--lattices),
- security compliance can be checked for every component in every integrated system as they share common security schema defined with super-lattices.

Presented example is simple but integrated e-services can become very complex consisting of many electronic services from many different systems. Lack of well-defined security model for public integration platforms is a delaying factor in the development of this area of public administration.

7. Conclusions

This article presented only a small part of authors' research on the electronic services security for public administration, which can be found in already published and planned (in print or in review) papers.

The analysed theoretical example has to be tested on real environment of public administration in Poland in three steps:

1. Use and test cases have to be defined – using UML language.
2. The proposed theoretical model has to be tested on real life composite e-services (using modelling language).
3. The authors' model has to be implemented into tools used for e-services creation and modelling and tested by real integration platform administrators.

If all three steps are passed the proposed model can then be easily used by public administration to create the central management security system for integration platforms as an open standard. In the future, it can be extended with operational elements like:

- authentication mechanisms,
- intrusion prevention and elements (IPS and IDS),
- antivirus protection,
- encryption mechanisms,
- event logging mechanisms.

This could allow to use the model with extensions to create a central security management for public administration instead of many domain systems (usually more than one security policy and system for each public institution – ministry, office, etc.).

8. Bibliography

[1] Wilk J., “Zarządzanie bezpieczeństwem w środowisku rozproszonym”, *Rozwój działalności przedsiębiorstw: wielowymiarowość uwarunkowań i konsekwencji*, WAT, Warszawa, 2015.

[2] Wilk J., “Security of Composite Electronic Services”, *International Journal of New Computer Architectures and their Applications* (*IJNCAA*), Vol. 5, No. 3, 127–140 (2015).

[3] Wilk J., “Wykorzystanie teorii krat w modelowaniu procesów zarządzania bezpieczeństwem w platformach usług elektronicznych administracji publicznej”, *Roczniki KAE*, No. 33, 581–597 (2014).

[4] Boboli A., Jeruzalski T., Olszewska M., Paleń R., Ręgowski A., Siejda A., *Państwo 2.0. Nowy start dla e-administracji*, raport pod redakcją Michała Boniego, Ministerstwo Administracji i Cyfryzacji, Warszawa 2012.

[5] Wilk J., “Information security management model for integration platforms”, in: *e-Technologies and Networks for Development* (*ICeND*) *– IEEE Xplore*, pp. 22–27, Fourth International Conference on e-Technologies and Networks for Development (ICeND 2015), September 21–23, 2015, Lodz, Poland.

[6] Szafrański B., *Modelowanie procesów ochrony baz danych ze szczególnym uwzględnieniem ich integracji*, WAT, Warszawa, 1987.

[7] Birkhoff G., “Lattice theory”, *American Mathematical Society Colloquium Publications*, XXV, New York 1940, II edition 1948.

[8] Sandhu R.S., “Lattice-based access control models”, George Mason University, (11.1993), p. 18.

Zarządzanie bezpieczeństwem usług elektronicznych dla administracji publicznej

B. SZAFRAŃSKI, J. WILK

Administracja państwowa w Polsce ciągle rozszerza i rozwija usługi elektroniczne dostarczane obywatelom, do biznesu i do samych siebie (np. jedno ministerstwo świadczące e-usługi na rzecz innego ministerstwa). Nowe oczekiwania, takie jak integracja z wykorzystaniem wielostronnego modelu interoperacyjności, podejście zorientowane na usługi, różne kategorie i grupy uprawnień w ramach różnych instytucji tworzą wymóg zastosowania odpowiedniego modelu bezpieczeństwa. Według autorów dostępne obecnie modele nie spełniają nowych wymagań, ale mogą stać się podstawą do stworzenia nowego podejścia. Krata i jej modyfikacje w różnych modelach (Denning, Sandhu, Szafrańskiego) zostały wybrane przez autorów w celu stworzenia nowego modelu zarządzania bezpieczeństwem usług elektronicznych dla platform integracyjnych wspomagających realizację zadań publicznych. Niniejsza publikacja skupia się na praktycznym aspekcie nowego modelu i przedstawia studium przypadku na przykładzie e-usług administracji publicznej. Szczegółową definicję samego modelu można znaleźć w poprzednich publikacjach autorów.

Słowa kluczowe: usługi elektroniczne, bezpieczeństwo, SOA.