Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 4915 – 4934

doi:10.3906/elk-1403-200

Multilayer authorization model and analysis of authorization methods

Alper UĞUR[1,*], **İbrahim SOĞUKPINAR**[2]
[1]Pamukkale University, Denizli, Turkey
[2]Gebze Technical University, Gebze, Kocaeli, Turkey

Received: 19.03.2014 • **Accepted/Published Online:** 03.10.2015 • **Final Version:** 06.12.2016

Abstract: There are various methods proposed in the literature to provide authorization control in workflows and information systems. Authorization implementations have deficiencies based on procedural scope. Basic login mechanisms grant system-wide access; the provided margins are broad. Access control lists provide limited definition on access restrictions; the authorization is bounded by these definitions. Role based authorizations do not cover regulations in institutions where the regulations describe specific operations and their operational procedures in institutional workflows. The proposed multilayer authorization model depicts the attributes of authorization mechanisms and analyzes the methods according to their authorization capabilities and contributions to the reliability of documents in the workflow. The layered structure provides comparative and integrated analysis of the authorization mechanisms. The incremental authorization structure would be a guide for implementations in that each layer presents the scope of authorization by providing analysis on deficiencies and the methods of solution. An institutional authorization mechanism on documents is also proposed. The proposed mechanism suggests and implements an authorization mechanism to enclose authorization restrictions in institutional regulations.

Key words: Authorization, information reliability, Petri net analysis

1. Introduction

Information security is the overall set of steps taken to protect valuable information assets from attacks and threats such as unauthorized access, modifications, destruction, and information reveal. Those steps may include security mechanisms, monitoring and control devices, software, regulations, standards, policies, and even security training. The security requirements of a system may vary, depending on the degree of importance of the information assets in the system. The attacks will also be advanced and complex in that case. The security depends on performed vulnerability analysis and also on the solutions provided for the detected security problems. Applied solutions determine the security level of the system.

In any information system, access to information starts with log in to the system. This first layer of authorization grants the requester system access as a "user". The identification of a user is done via authentication. Authentication is the operation of granting system access to an information asset by evaluating its attributes. Authorized users are granted access to the system and others are rejected at this phase.

An effective authentication system provides control for the whole system. However, there has to be an authorization mechanism inside the system for the authenticated users that will determine the permissions given to the user in the system. Recent studies have shown that a substantial fraction of total security vulnerabilities

*Correspondence: augur@bilmuh.gyte.edu.tr

are accomplished by internal users. According to a security survey [1], since 2004, the attack rate executed by internal users is about 28% (in 2013, the rate was 23%). The institutional damage caused by these attacks is around 46% of the total (in 2014) [2].

The internal users are clients who were logged in to the system by an authentication system. Due to the lack of security mechanisms, authenticated users have inadequate and broad authorization defined as system wide access. This may cause security vulnerabilities. In the UK, in 2007, a remarkable case of data disclosure and loss was exposed. An authorized officer had copied the entire 25 million records from a database of residents to compact discs and sent them with the postal service instead of fetching couple of thousand records and printing them out as a document. The discs disappeared in the post office [3]. Furthermore, users may cause damage to the reliability of the system. The intentional abuse of authorizations or the extending of authorization boundaries are examples of these types of acts. Sixty-three percent of internal threats are executed with unauthorized access to institutional information [1].

The authorization mechanism that implements authentication must be effective in preventing unauthorized operations. Authentication and access control based authorization is not sufficient for the security of sensitive information and records [4]. The malicious user is able to access and share personal, sensitive information, such as patient records, without any authorization mechanism other than authentication [4,5].

The authorization mechanism must have additional security layers in comparison with authentication based systems. The layers have a more composite structure than authentication, such as deciding who has authorization to execute a process in contradistinction to just deciding who has permission to enter the system. Basic login, Kerberos [6] authentication, RADIUS (remote authentication dial in user service) [7] authentication and access control, and role based access control (RBAC) are relatively complex methods that constitute the layers of authorization mechanism. These methods support reliability of documents by implementing user groups and roles [8–10].

The reliability of a document is bound to the authenticity, accuracy of the information contained and promised in the document, and confidence in the institutional and interinstitutional validness. A reliable document must be created through proper processes in an institutional workflow and must be produced according to institutional policy and regulations. The confidence in a document's validity is related to the convincing clues of the authenticity of the document. If a document was created in a secondary institution, the document must be reliable not only for the secondary institution where it was created but also the institution where it is going to be processed. The interinstitutional validness exists if the authenticity of the document can be confirmed in both institutions.

The authenticity of a document is supported by any method that proves the document has not been altered in an unauthorized way. The creator of the document and any authenticity information can be appended to the document with digital signature algorithms. Trust in the authenticity can be ensured with these cryptographically secure methods [11]. For the reliability and security of the document, it is important to examine the competence of authorization methods in different cases.

Management of a workflow's security consists of the execution of security rules. These rules are defined in security policies. The scope of a security policy includes basic institutional statements, government regulations, security standards, and even interinstitutional security politics. The security policies are defined generally as restrictions on roles, and operations in the workflow [12].

A workflow may be formed by processes of a unit or multiple units in an institution. The workflow may also involve different institutions, such as interinstitutional correspondence. In the application of security

policies with these variations, complications may arise while executing authorized operations in the workflow. Inconsistencies of restrictions may cause inaccuracies in the workflow [12].

In supplemental guidance on ongoing authorization [13] one of the three steps of authorization is reauthorization, where the authorization official or risk executive analyzes risk tolerance. After initial authorization and ongoing authorization steps, the information system must be reviewed during the operation/maintenance phase. This review triggers reauthorization according to the risk assessment and organizational risk tolerance. The scope of reauthorization may cover small changes, such as modification of parts, or complete and significant modifications, such as modification of regulations and security controls.

In this work, a multilayer authorization model is proposed. Each layer is evaluated through their contributions to document security and reliability. Possible security gaps are presented in sample cases and these are evaluated with reachability tests using Petri net models. Moreover, the solutions to identified problems are explained. The reliability of documents in workflows requiring authorization is examined. Then authorization problems and solutions are discussed within the model.

The multilayer model reveals objectives, process stages, and attributes of the authorization methods. The multilayer model facilitates the reauthorization process. The model enables layer based or cross layer analysis of the applications that require authorization, or authorization mechanisms that are implemented in institutional security policies and regulations. This allows detection of procedural authorization deficiencies and aids development of solutions if possible. The analyst can decide and plan what to do next using the proposed model. The reliability of the documents can be analyzed and proved through layers. The model makes it possible for new authorization methods and solutions to be specialized and implemented based on attributes of the layers.

The rest of the paper is organized as follows. In Section 2, related works on authorization are presented. In Section 3, a multilayer authorization model is described. In Section 4, Petri net reachability based authorization and reliability analysis of the model is given. The paper concludes with future works and solutions.

2. Background information and related works

In this section, brief background information on authorization mechanisms is given as the proposed multilayer model consists of authorization mechanisms. The section also summarizes their capabilities. The authorization mechanisms and frameworks that were excluded from the model are also presented.

Authorization is a security mechanism that determines user privileges in the system and forces them to operate in accordance with these permissions. The first authorization constraint applied to users is the system login. In most information systems, for structures such as secure web services, workstations, servers, and network devices databases, system access is given only to the users permitted to login. The user makes an access request to the system. The system analyzes the request, mostly by a challenge, and approves or denies the access request as a result. The requester must notify and prove its identity (ID) to the system. Many methods like basic login, Kerberos [6], and RADIUS [7] are used for the authentication. In the basic login method, an ID and password combination is requested from the user. In Kerberos, the user is authenticated with multiserver architecture. A session ticket is provided for the user to access the server for a certain period of time. The user could login to the server by using ID, password, and the ticket.

Kerberos has a lack of authorization in distributed systems. There must be an authorization mechanism with the Kerberos authentication in order to ensure the required level of security [8]. Just like the session ticket in Kerberos, some information systems require additional information for authorization. Access control lists

(ACLs) determine the user's access privileges on a system [9]. The lists contain restriction entries for some operations in the system. These restrictions assist the authorization mechanism. In authorization control, the ACLs are checked and users are restricted by the entries.

Another common authentication and authorization method is RADIUS, which gives system access with authentication and uses ACLs for authorization [7]. This system and operation based multiple control provides more reliable authorization. However, ACL based authorization control is still not enough for the desired authorization. ACL entries have limited definitions for users and system operations and "permit" or "deny" decisions offered by the entries become inadequate when the procedures in a workflow get complicated [9]. For example, in an institutional structure, the privileges of an officer working in purchasing cannot be defined with "may" or "can't". The amount of purchase authorization is not clear in the list. The institutional structure and workflows require ACLs to be updated with this type of detailed entries. A huge and detailed list is difficult to control and manage [9].

Role based access control (RBAC) [10] methods were proposed as a solution to the shortcomings of ACLs. Users are grouped according to their specific institutional roles. A role is generally described as a collection or group of users who share the same position or perform the same operation [14]. Expansion, promotion, or demotion of roles can be achieved easily and efficiently. RBAC makes delegation of roles possible [15]. It ensures that users can only execute actions within the privileges defined to these groups.

Attribute based access control ABAC [16] is another access control mechanism that tries to solve the problems of RBAC in a dynamic environment. ABAC allocates dynamic attributes, including time and place, to objects to authorize execution of operations. However, a role can be defined in ABAC as just a role name; the definition does not contain its permissions. This property provides the dynamism of role definitions for the users but the authorization mechanism must query the permissions according to the role attribute. RBAC role definitions are also powerful in that the authorization mechanism can easily deduce defined permissions according to the role. As the model addresses authorization capabilities rather than dynamic management problems of authorization, RBAC is chosen to represent the access control mechanisms.

The proposed multilayer model analyzes authorization mechanisms in a workflow. The layered approach handles each mechanism in a separate layer according to their authorization control capabilities and contributions to the reliability of documents in the workflow. One of the layered authorization mechanisms in the literature is OAuth [17], which is an authorization framework providing an authorization layer that limits the access of a third party to an HTTP service. The OAuth framework addresses authorization problems where applications need access to restricted resources of the owner and the owner is required to provide its credentials to the application. This requirement exposes problems, as restrictions may revoke the given authorization or compromise credentials. The aim of the framework is to separate the role of the client and the owner of the resource with the authorization layer. As stated in the Request for Comments, the use of OAuth on any other protocols other than HTTP service is outside of the scope of the framework. As the proposed multilayer model is addressing authorizations on workflows, the OAuth framework is excluded.

3. The multilayer authorization model

Authorization is a security mechanism that determines user privileges in the system and forces the user to operate in accordance with these permissions. This work proposes a multilayer authorization model as depicted in Figure 1. The layer structures are established by their functionalities and the sensitivity of authorization control.

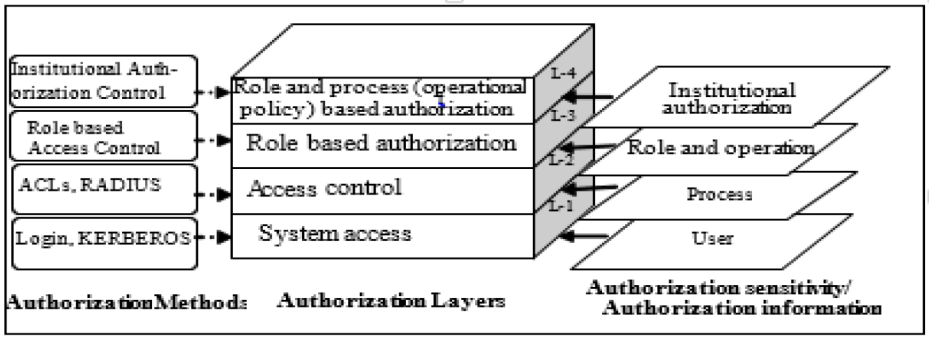


Figure 1. Multilayer authorization model.

Authorization mechanisms challenge the user with more precise and sensitive information and it encompasses more specific procedures from bottom to upper layers of the authorization model. The authorization information queried in each layer performs an authorization filter and elevates the user to the next layer. Authorization layers are fundamental structures that fulfill the required authorization in institutional workflows.

Authorization methods overlap the authorization layers in implementations as solutions. The authorization sensitivity filters are formed by authorization information required for each layer. The entity in an institutional workflow must provide this authorization information to access or execute processes in the corresponding layer.

3.1. Overview of the model

In this section, the multilayer authorization model is summarized by briefly presenting the scopes of authorization and the mechanisms employed in each layer.

System access layer: The first layer of the multilayer authorization model contains authorization for general system admission. The authorization for system access is provided by authentication mechanisms. It requires the ID and password. The authorization mechanism in this layer applies to all users. Because the authorization precision is low, only the user identity is used for governance. Also the privileges given to the user are high. The user gets full system access or else there is an absolute denial of access. There is no additional operational restriction to users who access the system. Login, Kerberos, and RADIUS authentication implementations are the practices that take place in this layer. Two-layer Kerberos authentication and session ticket generation have minimal authorization complexity as compared with the upper layers. Cryptographic algorithms are generally used in challenges to make the security level higher.

Access control layer: The second layer comprises the authorization for processes that can be executed by users logged in to the system by the first layer. As in RADIUS authorization mechanisms and access control lists, users' privileges for the operations are queried from access lists. The operations are approved or denied according to the authorization. This layer of authorization applies to more specific users. They are narrowed to a group of users who have been granted privileges to access the system by the first layer. The authorization is more precise as it includes user, process, and "approve/deny" expressions in the lists. The layer provides process based access control. Although it is limited by ACLs, its security level is high. Compared to system-wide access, mechanisms in this layer intensify the authorization scope of the processes.

Role based authorization layer: The third layer is a layer of role based authorization control. At this layer, users are grouped by their roles in the information system to provide a solution to the limitations of ACLs. The designated authorizations are customized not only based on processes but also by the rules that

execute those processes. As stated in related works, a role is a collection or group of users who share the same position or perform the same operation. The role is assigned to a user in order to perform an operation. Role assignment is safer than promoting a user to administrator, which gives gratuitously broad authority, as in the second layer. It enables the management of roles in an institutional structure. The user and their role can easily be promoted, revoked, and delegated. The complexity of authorization control is high but there are mechanisms that make authorization management easier. The precision of authorization is high as it utilizes the user-group-process-information asset. The scope of authorization is condensed to the process-information asset as the authorization is related to the specialized execution of the process according to the requested privilege.

Role and workflow process (operational policy) based authorization layer: The top layer, proposed as the fourth layer of authorization, is above role based authorizations. This layer tends to address mechanisms for institutional authorizations. In case of any insufficiency of role based mechanisms in an institutional structure, the authorization must be responsive enough to adapt with policies, regulations, and guidelines.

As a sample case, let an officer have the role of purchasing a part and approving its order document. Through this process, the purchase operation can be completed. However, the purchase operation is generally defined in institutional policy as "if the payment in purchase order is higher than a certain limit, it must be approved by the authorized administrator". The authorization control must take into account the institutional policy and the authorization information must be specified with respect to the policy of the information asset.

The attributes of processes as to who can execute them, how they can be accomplished, and which phase of the workflow they employ play a major role in determining authorizations. In the first layer, identity is used for general authorization. Superior layers oblige additional information such as access lists, role based relations, hierarchy, and delegations for the success of the authorization control. In an institutional structure, the role based authorization suggests a separation of duty (SoD) [18] rule for sale and purchase roles. The role based authorization is used successfully to separate and authorize the related procedures. However, institutional guidelines and regulations are not reflected in roles and these authorizations cannot be proved for the document.

At this layer, the precision of authorization is at its highest; the mechanism controls even the institutional regulations. The authorization control complexity increases at the same rate. The scope of authorization is isolated up to the process attributes.

3.2. Authorization layers on Petri net workflows

In this section each authorization layer of the model is presented with a Petri net on institutional workflows. The authorization mechanism and the scope of the authorization of each layer can be examined through these workflow models.

A Petri net is a graph that can be used to express the status, event, and the relation between these sets of a workflow. The Petri net N, is defined in Eq. (1)

$$N = \langle P, T, F, I, O, M \rangle \text{ where} \tag{1}$$

$$P = \{P0, P1, \ldots, PN\} \text{ is a finite place set (the status),} \tag{1.1}$$

$$T = \{T0, T1, \ldots, TN\} \text{ is a finite transition set (the event) where } P \cap T = \emptyset \tag{1.2}$$

$$F \text{ is } F \subseteq \quad (P \times T) \cup (T \times P) \text{ is a finite directed arc set,} \tag{1.3}$$

$$\text{where } ((\forall t \in T)\,(\exists p; q \in P)\,(p; t)\,;(t; q) \in F. \tag{1.4}$$

$$\text{Input function } I: (TXP) \rightarrow \{0,1\} \tag{1.5}$$

$$\text{Output function, } O: (PXT) \rightarrow \{0,1\} \tag{1.6}$$

The marking set of Petri net, M, is defined in Eq. (2):

$$M = \{M_0, M_1, \ldots, M_n\} \; where \tag{2}$$

$$M_0 \text{ is the initial marking and } \subseteq M \neq \emptyset \; and \; M \; P \tag{2.1}$$

If a transition t_1 is enabled at marking M_0 to M_1 it can be denoted as $M_0 \xrightarrow{t_1} M_1$ or $M_0[t_1 > M_1$. A finite sequence $\sigma = t_0 t_1 t_2 \ldots t_{n-1}$ of transitions is called a finite firing sequence, enabled at M_0, if there exists markings $M_1 M_2 M_n$ such that $M_0 \xrightarrow{t_0} M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_2} \xrightarrow{t_{n-1}} M_n$ and the notation can be condensed as $M_0 \xrightarrow{\sigma} M_n$ or $M_0[\sigma > M_n$. The $\xrightarrow{\sigma}$ notation will be used in the text.

A marking M_n is reachable from M_0 if there is a firing sequence leading from M_0 to M_n. The reachability can be denoted with $M_0 \xrightarrow{*} M_n$

Petri net models can be useful to present authorizations of an entity where authorization methods are applied. If an entity could reach a place in Petri net, he/she could execute the process in the workflow at that point. The reachability on Petri nets can be defined and used as:

Let $u_i u_{id} \in U$ where U is set of users; u_i is any user and u_{id} is an authenticated (identified) user in the system. If place p_n is reachable for user u_{ix} in the Petri net, user u_{ix} is authorized to execute process in place p_n.

It was stated before that the layer structures were established by their functionalities and the sensitivity of authorization control. In Petri net models of each layer, the functionalities are modeled with place and transitions in the workflow and the sensitivity of authorization controls are presented with information packets requested for authorization. The requested information for execution of an operation is defined with a 5 tuple information set as {*operation, execution type, user type, user,* authorization information} . The requested information is denoted with $\emptyset$ as it is not available or not required for the authorization. The requested authorization information is emphasized with brackets.

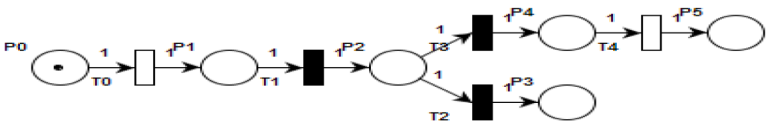
3.2.1. First layer of multilayer authorization model

Authorization is applied to the user for system access. The authentication mechanisms are executed in this layer. The user can perform any operation in the system with this authorization. For example method and implementation see user login systems and Kerberos authentication, below.

a. Login method: The login mechanism modeled with Petri net is given in Figure 2. The logged user who transits through places $\{p_2 p_4$, by the $M_2 \xrightarrow{t_3} M_4$ sequence, in other words the user who can trigger place p_5, could perform any operation in the system. With the initial marking [1000000] user u_i triggers $[t_0 t_1 t_3 t_4]$. According to the incidence matrices given in the Table below, in terms of $M = M_0 + \mu I$, reachability of user u_i is $[000001] = [1000000] + [11011] \bullet I$. The result sequence is $M_0 \xrightarrow{t_0} M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_3} M_3 \xrightarrow{t_4} M_4$ where the user could reach place p_5.

a. Kerberos authentication mechanism: The mechanism expands the basic login structure. The identification is implemented on an authentication server. If the request is affirmative, then a user can access

the server for a certain period. The basic Petri net model of this mechanism is presented in Figure 3. The user u_i could reach place p_{10} triggering $[t_0t_1t_3t_4t_5t_6t_8t_9]$ transitions. The reachability of user u_i is $[00000000001] = [1000000000] + [1101111011] \bullet I$. u_i could reach place p_{10} and execute operations by the sequence of $M_0 \xrightarrow{t_0} M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_3} M_3 \xrightarrow{t_4} M_4 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_8} M_8 \xrightarrow{t_9} M_9$



Places and transitions with the {*operation, execution type, user type, user, authorization information*} information set:

P0: operation execution request is sent by a

T0: System asks for identification

P1:ID&pwd $\{\emptyset, \emptyset, \emptyset, u_i, \emptyset\}$ information sent by user

T1: system checks for $\{u_i\}$

P2: $\{u_i\}$ is checked

T2: system denies $\{u_i\}$

P3:user is rejected

T3: system accepts $\{u_i\}$

P4:u_i is granted for execute operation

T4:u_i executes operation

P5: operation is successfully executed.

Figure 2. Petri net model of login mechanism.

Table. Incidence matrices for the first layer Petri net.

Forward IM I^+

	T0	T1	T2	T3	T4
P0	1	0	0	0	0
P1	0	1	0	0	0
P2	0	0	1	1	0
P3	0	0	0	0	0
P4	0	0	0	0	1
P5	0	0	0	0	0

Backward IMI^-

	T0	T1	T2	T3	T4
P0	1	0	0	0	0
P1	0	1	0	0	0
P2	0	0	1	1	0
P3	0	0	0	0	0
P4	0	0	0	0	1
P5	0	0	0	0	0

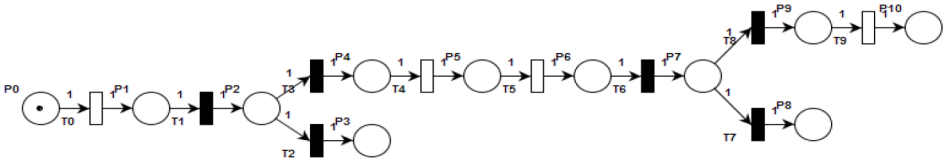
Backward IMI^-

	T0	T1	T2	T3	T4
P0	1	0	0	0	0
P1	0	1	0	0	0
P2	0	0	1	1	0
P3	0	0	0	0	0
P4	0	0	0	0	1
P5	0	0	0	0	0

3.2.2. Second layer of the multilayer authorization model

Basic access control and authorization mechanisms are performed in this layer to avoid operations that change or override the workflow by authenticated insiders. Control lists are designed and employed for user access restrictions to avoid the execution of all operations in the system.

The authorization sensitivity differs from first layer as the second layer requires an ACL entry with a user group and a rule for the operation. The ACLs contain entries of users or groups, the operations and the access privileges as $<$*user/group, operation, permitordeny*$>$. The system checks the lists for the operation request



Places and transitions with the information set {*operation, execution type, user type, user, authorization information [ticket]*} are:
P0: operation execution on server S request is sent by a user
T0: System asks for identification
P1: ID&pwd$\{\emptyset, \emptyset, \emptyset, u_i, \emptyset\}$ information sent by user u_i
T1: system checks for $\{\emptyset, \emptyset, \emptyset, u_i, \emptyset\}$
P2: $\{\emptyset, \emptyset, \emptyset, u_i, \emptyset\}$ is checked
T2: system denies $\{u_i\}$
P3: user is rejected
T3: system accepts $\{u_i\}$
P4: ticket is granted to $u_i$for **server S**
T4: u_i gets **ticket**
P5: operation execution request to server *S* is sent by a user
T5: Server *S* asks for identification
P6: ID&pwd&ticket$\{\emptyset, \emptyset, \emptyset, u_i, [ticket]\}$ information sent by user u_i
T6: Server *S* checks for $\{u_i, [ticket]\}$
P7: $\{\emptyset, \emptyset, \emptyset, u_i, \emptyset\}$ is checked
T7: system denies $\{u_i\}$
P8: user is rejected
T8: system accepts $\{u_i, [ticket]\}$
P9: u_i is granted to execute operation with boundary of *ticket*
T9: u_i executes operation

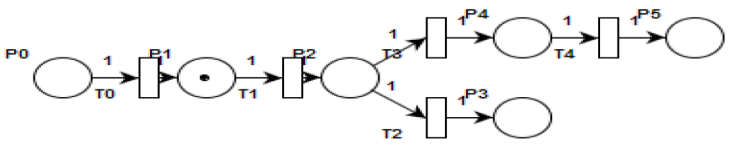
Figure 3. Petri net model of basic Kerberos authentication mechanism.

and permits or denies the execution according to the privilege on the list. The intention is to prevent users executing unauthorized operations. The Petri net model of the mechanism of an instance of access control lists is presented in Figure 4. The first layer of authorization is illustrated with M_0 initial marking. The second layer of authorization starts with marking M_1 .

Authenticated users u_i can trigger $[t_0\, t_1\, t_3\, t_4]$:[11011]. By initial marking [100000] reachability is [000001] = [100000] + [11011] $\bullet$ I. User u_i can reach place p_5 after the $M_0 \xrightarrow{t_0} M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_3} M_3 \xrightarrow{t_4} M_4$ sequence. In place p_5 *user* u_i could execute operation o_i in compliance with the rule $\{o_i, g, u_i\} \rightarrow$ permit in the access control list. User u_i can perform the $\{u_i, o_{approve}\}$ operation at place p_5, authorized with the $\langle heads, o_{approve}, permit\rangle \wedge u_i \in heads$ rule in place p_2.

3.2.3. Third layer of multilayer authorization model

This layer is built up with role based access control mechanisms to overcome the defects of the previous layer and advances the authorization capabilities of the system. The authorization is based on the roles and the operation privileges defined for these roles. The mechanism has more control of operations through detailed privilege definitions. The second layer mechanisms are applied to the low level operations such as folder, database,



Places and transitions with the {*operation, execution type, user type, user, authorization information*} information set are:

P0: operation execution on request ID&pwd, $\{\emptyset, \emptyset, \emptyset, u_i, \emptyset\}$ information is sent by user

T0: System authenticates u_i

P1: operation o_i execution request by u_i

T1: System checks $\{o_i, \emptyset, \emptyset, u_i, \emptyset\}$from **ACL**

P2: $\{o_i, \emptyset, g, u_i, \emptyset\}$ is checked

T2: system denies $\{o_i, u_i\}$

P3: user is rejected

T3: system accepts $\{o_i, g, u_i\}$

P4: u_i is granted for **execute operation**o_i **with** e_i

T4: u_i executes operation o_i with e_i

Figure 4. The Petri net model of access control lists based authorization.

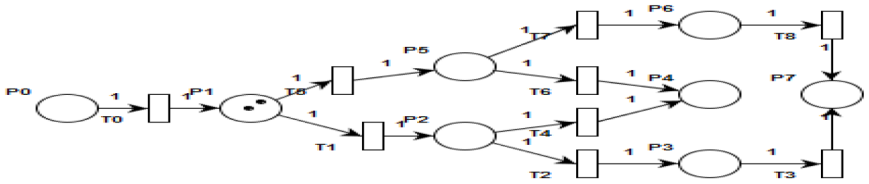
or hardware access. The authorization of institutional procedures in the workflow is handled with role based mechanisms in this layer. This multilayer approach reduces the administration load of authorization by filtering operations for their authorization requirements. The first layer filters authorized and unauthorized users for system access requests. The second layer of authorization filters system is based low level operations. The third layer of authorization deals with the institutional and procedural operations in the workflow to improve authorization control. The role based mechanisms of the third layer of authorization have more control over the detailed operations and the role structure facilitates the institutional procedures. The basic role based authorization mechanism forming the third layer is presented in Figure 5.

The first layer is illustrated with M_0 initial marking. The second layer starts with marking M_1 . If the operation is not defined in the ACL, the authorization decision will be given by role based authorization. The third layer of authorization sequence starts with $M_2 \xrightarrow{t_5}$. The system terminates at place p_4 and place p_7 on the graph.

Authenticated users u_{id} can trigger $[t_0\, t_5\, t_7\, t_8]:[100001011]$. By initial marking [10000000] the reachability is $[0000001] = [10000000] + [100001011] \bullet I$. User u_{id} can reach place p_7 after the $M_0 \xrightarrow{t_0} M_2 \xrightarrow{t_5} M_3 \xrightarrow{t_7} M_4 \xrightarrow{t_8} M_5$ sequence on $\{p_0 p_1 p_5 p_6 p_7$. At place p_7 user u_{id} could execute operation o_i where the role r of authenticated user u_{id} has privileges to perform operation o_i. User u_{id} can perform operation $\{u_{id}, o_{approve}\}$ at place p_7 with $\langle heads, o_{approve}, permit\rangle \wedge u_{id}$ has role "head of purchase unit".

3.3. Mechanisms of the model

In the previous section 3 layers of the model were presented with Petri nets. The authorization mechanisms in each layer were also examined with the reachability analysis of a user in the workflow. In this section the multilayer authorization model is analyzed by its features. The following notations are used in formulation of the features.



Places and transitions with the {*operation, execution type, user type, user, authorization information*} information set:
P0: operation execution on request ID&pwd $\{\emptyset, \emptyset, \emptyset, u_i, \emptyset\}$ information is sent by user
T0: System authenticates u_i
P1:operation $o_i$execution request by u_i
T1: System checks $\{o_i, e_i, \emptyset, u_i, \emptyset\}$from ACL.
P2: $\{o_i, \emptyset, g, u_i, \emptyset\}$ is checked.
T2: system accepts $\{o_i, g, u_i\}$
P3:u_i is granted for **execute operationo_i with e_i**
T3: u_i executes operation o_i with e_i
T4: system denies $\{o_i, u_i\}$
P4: user is rejected
T5:System checks $\{o_i, e_i, r, u_i\}$ from **user roles and operation rules defined in roles.**
P5: $\{o_i, e_i, r, u_i\}$ is checked.
T6: system denies $\{o_i, e_i, r, u_i\}$
P4: user is rejected
T7: system accepts $\{o_i, e_i, r, u_i\}$
P6: u_i is granted for **execute operation$o_i$with e_i by role r**
T8:u_i executes operation o_i with e_i by role r
P7: operation is successfully executed.

Figure 5. Petri net model of basic role based authorization.

Let *UGORA* will be the set of users, user groups, operations, roles, and authorizations, respectively, where each user is an element of the user group as in $\forall u \in g, u \in Uve\, g \in G$. Roles are operations that user groups were assigned to; at least 1 role is defined for each operation, and the definition is given in Eq. (3):

$$r : g \to o, \exists r \forall o \in O,\ r \in Rg \in G \tag{3}$$

Authorizations are the roles of the users; an authorization a is defined for operations as given in Eq. (4):

$$a = \{\{u, r, o\} \mid u \in g \wedge r : o \to g\}\, r \in Rg \in Ga \in A. \tag{4}$$

authorization approval y is given as in Eq. (5):

$$y = \exists a \{a \in A \mid a = \{\{u, r, o\} \mid u \in g \wedge r : o \to g\}\} \to \{0, 1\}. \tag{5}$$

If a user has a role in the operation, authorization is approved. Otherwise it is rejected. If $u \in g$ then the authorization approval for group g of u can be written as in Eq. (6)

$$y = r : o \to u, \text{ where } u \in g. \tag{6}$$

The layers of authorization are proposed according to the scope of the authorization. The first layer of authorization is the system login layer. Authorization control is effective on all users set U. The operation definition is the most general definition as $o_0 \in O$ and consists of system access. The user set is defined as known and unknown users g_0: group of users known by the system $\wedge g_x$: group of unknown users of the system $g_0 g_x \in G$ As $o_0 \in O$ is defined as system access, the role of the user in authorization $r_0 : g \rightarrow o$ will be system wide access or system wide rejection. Authorization definition in this layer is stated in Eq. (7):

$$a_0 = \{\{u, r, o\} \mid u \in g_0 \wedge r : o_0 \rightarrow g_0\} \tag{7}$$

The user verifies system access with $y = 1$ approval. If the system has only the first layer authorization, the user u could perform any operation in the system $\forall o \in O$.

At the first layer, user identification and determination of a group is provided with authentication mechanisms. An identified and authenticated user logs in to the system as the authorization control allows. The definition set of $r : o \rightarrow g$ statement is $u \in g_0$ that is the definition of ID. If $u \in g_0$ then $r : o \rightarrow g \rightarrow \{1\}$ and $y = 1$. $u \in g_0$ is authorized system-wide access. If $u \in g_x$, then $r : o \rightarrow g \rightarrow \{0\}$ and $y = 0$, access will be denied.

In the second layer of authorization, access control is performed on users through their groups. In this layer, authorization is controlled with access control lists. The authorization control covers the filtered user form in the previous layer where $u \in g_0$ and $\subset g_0 U$. The operations are defined in ACLs where $\subseteq O_{ACL} O$. User groups and roles are also defined on ACLs. While $o_{ACL_1} \in O_{ACL}$, $g \in G$, $r : o \rightarrow g$ defined as $r : o_{ACL_1} X g$. The authorization information is updated with a $r : o_{ACL_1} X g$ restriction. The authorization for operation o will be given if $\in O_{ACL}$ and $r : oXg$ exists in the ACL. Excluding the systems with limited requirements, it is hard to include each {process, group} tuple to the ACL. The management complexity would also be high in that case.

The third layer provides a mechanism for the requirements of authorizations that are still a problem for layer 2. The third layer of authorization provides solutions for the problems of the prior layer. The authorizations for operations are defined in more detail with role based structures. The users and groups can be managed more efficiently with roles. Users are authorized over user groups. The set of users subject to control in this layer is not different from the prior one. In the second layer restrictions are defined in a list as $O_{ACL} X G$, in the third layer, roles are defined with more comprehensive mapping as $R : O \rightarrow G$. In the prior layer, expansion of authorization requires a group update and operational changes in the lists. Role based mechanisms have effective solutions such as delegation [15]. Separation of duties principle (SoD) [18] inhibits a user so they can have only 1 role for related operations. This principle improves the security of institutional operations. A person with a purchase role cannot have a purchase approval role at the same time. This static rule can be dynamically adapted as the person may have purchase and approval authorizations but cannot approve his own purchase operation. The authorization in this layer can be defined as in Eq. (8):

$$a = \{\{u, r, o\} \mid u \in g \wedge r : o \rightarrow g\} \tag{8}$$

Let o_t, o_h are 2 dependent operations in the workflow. r is bounded with the $, r_t : o_t \rightarrow u \wedge r_h : o_h \rightarrow u\, rule$. The authorization approval in this layer is defined in Eq. (9):

$$y = \exists a \{a \in A \mid a = \{\{u, r, o\} \mid u \in g \wedge r : o \rightarrow g\}\} \rightarrow \{0, 1\}. \tag{9}$$

The approval in an authorization delegation case is as shown in Eq. (10):

$$y^u = \exists a \{a \in A \mid a = \{\{u, r, o\} \mid u \in g \wedge r : o \rightarrow g\}\} \wedge y_{uu'} \rightarrow \{0, 1\} \tag{10}$$

where $y_{uu'}$ is simple delegation information stating the authorization is delegated from user u' *to user* u. Validation of this information will give authorization for the operation o. In the third layer, the purchase case given in the overview section is defined as follows. For o_p purchase operation and g_p purchasing group; let the purchase role be defined as $r_p : o_p \rightarrow g_p$. The authorization approval in Eq. (11) would be valid.

$$y = \exists a_p \{a_p \in A \,|\, a_p = \{\{u, r_p, o_p\} \,|\, u \in g_p \wedge r_p : o_p \rightarrow g_p\}\} \rightarrow \{0, 1\} \tag{11}$$

The role does not encapsulate institutional regulation statements like "if the value of the purchased good is over $50K, the head of unit will have the authorization". Authorizations are defined by automatic operations, but restrictions and exceptions are not included in roles. Role definition must contain institutional regulation or policy restrictions with operation and group descriptions. D will be defined as set of regulation conditions; $d \in D$ and d^r will be conditions for role r. The authorization will be expanded as in Eq. (12):

$$\subset a_p = \left\{\{u, r_p, o_p, d_p\} \,|\, u \in g_p \wedge r_p : o_p \rightarrow g_p \wedge o_p \, d_p^{r_p}\right\} \tag{12}$$

By this definition compliance with regulations of the operation o_p could be denoted in authorization, and authorization can be justified with y defined in Eq. (13):

$$\subset y = \exists a_p \left\{a_p = \left\{\{u, r_p, o_p, d_p\} \,|\, u \in g_p \wedge r_p : o_p \rightarrow g_p \wedge o_p \, d_p^{r_p}\right\}\right\} \rightarrow \{0, 1\} \tag{13}$$

There are many administrative benefits to defining institutional regulations as specialized operations in the operation set. Defining the same procedural workflow processes with multiple roles complicates the workflow. However, the realization of the operations can be evaluated rapidly in a workflow with institutional restrictions stated in the regulations. Regulations, policies, and institutional functions have a tendency to change and update with time. When this occurs, the operations will be updated and integrated to the system automatically by this structure. The fourth layer of authorization encloses institutional authorization definitions.

3.4. Fourth layer of multilayer authorization:

The role based mechanism provides detailed control over operations. However, none of the role based mechanisms implement institutional policies and regulation over institutional roles. SoD restrictions provide security mechanisms for sensitive operations on documents. However, this method will cause role assignment problems because of the different role definition and restrictions in interinstitutional transactions [19].

This layer ensures that the authorization controls support the regulations. The decision mechanisms take into account the restrictions defined in the regulations and authorize the user according to them. The authorization mechanism proposed in this layer can also be applied in cases where reliability of documents in a system is crucial. The institutional authorization mechanism in this layer provides convincing proof that the documents are established within the authorizations.

The approval of purchase orders would be the sample case for document reliability. User u_m could generate a purchase order with his role as described in the previous section. User u_a may delegate an approval role to user u_m for a short period. Then user u_m could sign purchase orders to approve them. The authorization difference between the formerly signed order and the purchase approval signed after the delegation is ambiguous. Both documents are created and signed by acknowledged users in the system. However, neither of them indicate any authorization information as to whether the document is signed while user u_m is in personnel or user has a "delegated" head role. The digital signature operation has a timestamp mechanism and role delegation may

a have time interval defined in the system, but it may be hard to query these 2 discrete pieces of information in interinstitutional records and long-term documents. For the interinstitutional transactions, it is not secure to share institutional roles (and their attributes as time intervals) with the outside. The exterior institution has no chance to query authorization and evaluate the time of authorization.

Verification of authorization – control of a document as to whether it was created in an operation executed by an authorized user can be practically done with active authorizations in the workflow. The authorization subject and the scope of the authorization are available for verification on-time. But the dynamic structure of the workflow and continuous modifications on authorization make it difficult to fetch former subjects and scopes. Obtaining the proof gets harder on long-term stored documents such as patient or financial records, contracts, and governmental regulations, etc. [20,21]. It is complicated to investigate authorizations of multiple operations on a document such as contract signing or patient history over these intervals varying from months to decades.

The fourth layer of authorization, though the former layers focused on authorization control, presents audit of authorization. Related information, which is employed to verify authorization, is appended to the operations. The verification can be done through that authorization information. The layer proposes to encompass evidence of authorization for not only current authorization verifications but also is available to control on long-term documents. This mechanism maintains reliability of documents in workflows where authorization can be controlled. The approach has the same Petri net model (given in Figure 5) but differs in the procedures (places, transitions) as declared in Figure 6.

It differs at grant (p_3) and execution (t_3) nodes of the second layer and control and execution nodes of the third layer. Both executions are permitted/denied in the workflow according to the institutional policies and/or regulations. The system terminates at p_4 and p_7 on graph.

Authenticated users u_{id} can trigger $[t_0\, t_5\, t_7\, t_8\,]{:}[100001011]$. By initial marking [10000000] and the reachability $[0000001] = [10000000] + [100001011] \bullet I$ User u_i can reach p_7 after the $M_0 \xrightarrow{t_0} M_2 \xrightarrow{t_5} M_3 \xrightarrow{t_7} M_4 \xrightarrow{t_8} M_5$ sequence by $p_0 p_1 p_5 p_6 p_7$. At p_7 user u_i could execute operation o_i where role r of user u_i *has* privilege to perform o_i with regulation rule $d_p^{r_p}$. The authorization approval y is stated in the model is given in Eq. (14):

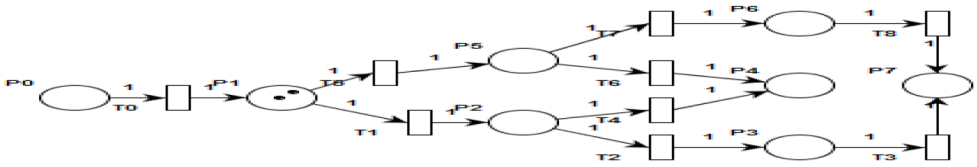
$$\subset y = \exists a_p \left\{ a_p \; = \left\{ \{u, r_p, o_p, d_p\} \mid u \in g_p \wedge r_p : o_p \rightarrow g_p \wedge o_p \, d_p^{r_p} \right\} \right\} \rightarrow \{1\} \tag{14}$$

4. Reachability analysis

In the introduction, the reliability of a document is defined as being bound to the authenticity of the document, the accuracy of the information contained and promised in the document, and the confidence in the institutional and interinstitutional validity of the document. A reliable document must be created through proper processes in the institutional workflow and produced according to the institutional policy and regulations. The confidence in the validity of the document can be achieved in this way.

In this section, the reliability of documents created in each layer of authorization is analyzed by examining the effectiveness of the authorization mechanisms on the reliability of documents and/or on authorization control. In cases given, the authorization mechanism is assumed as effective if it provides authorization for all users. If there exists any deficiency on authorization control the mechanism is assumed as ineffective for the case.

In the literature, Petri nets are used to analyze the security of protocols [22,23]. In this work, the workflow in each layer of authorization is modeled with Petri nets and reliability is observed on these models.



Information set for layer 4 is{operation,execution type,user type,user, autorization information}
P0: operation execution on request ID&pwd $\{\emptyset,\emptyset,\emptyset,u_i,\emptyset\}$ is sent by user
T0: System authenticates u_i
P1: operation o_i execution request by u_i
T1: System checks $\{o_i,e_i,\emptyset,u_i,\emptyset\}$from ACL.
P2: $\{o_i,\emptyset,g,u_i\}$ is checked.
T2: system accepts $\{o_i,g,u_i\}$
P3: u_i is granted for **execute operation** o_i **with** e_i
T3: $u_i$executes o_i with e_i and authorization information $[a_i]$ where $[a_i]$ is an institutional regulation rule on o_i
T4: system denies $\{o_i,u_i\}$
P4: user is rejected
T5: System checks $\{o_i,e_i,r,u_i,[a_i]\}$ from **user roles, operation rules and regulations.**
P5: $\{o_i,e_i,r,u_i,a_i\}$ is checked.
T6: system denies $\{o_i,e_i,r,u_i,[a_i]\}$
P4: user is rejected
T7: system accepts $\{o_i,e_i,r,u_i,[a_i]\}$
P7: u_i is granted for **execute operation** o_i **with** e_i **and** $[a_i]$ **by role** r
T8:u_i executes operation o_i**with** e_i **and** $[a_i]$ **by role** r
P7: operation is successfully executed.

Figure 6. Places and transitions of the 4th layer of authorization Petri net.

By reachability analysis, the authorization requirements are discussed with regards to creating and executing a document in a workflow. The adequacy of authorization control and reliability of the document are presented with the results of the analysis.

For comparative analysis of authorizations $u_a u_m$, $u_o \in U; u_a$ is any authorized user, u_m is a malicious user, u_o is an attacker (outsider) in the workflow. It is assumed that the attacker u_o has no information such as ID, or the password of any system users.

A user's reachability can be explained as follows: If attacker u_o or malicious user u_m could reach a place in the Petri net, it indicates that he could create an unauthorized document or simply execute an unauthorized operation in the workflow.

4.1. Reachability analysis for the first layer of multilayer authorization model

Case of login: Attacker u_o's reachability is as follows: As he could not pass authentication, attacker u_o could trigger $[t_0\, t_1\, t_2]$. The state is [11100] and the initial marking is [1000000]. From incidence matrix (M sequence) $M = M_0 + \mu I\,[000100] = [1000000] + [11100] \bullet I$ attacker u_o reaches place p_3 and is rejected. After the $M_0 \xrightarrow{t_0} M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_2} M_3$ sequence the workflow is terminated for u_o.

Case of Kerberos: Attacker u_o's reachability (as he could not pass authentication) is as follows: The transitions the attacker challenges are direct request to the server, $[t_6\, t_7]$, or normal flow transitions, which are $[t_0\, t_1\, t_2]\,.\,[00010000100] = [1000001000] + [1110001100]\bullet I$ The attacker u_o could reach places p_3 and p_8 and be rejected with the trigger $[1110001100]$ and the initial marking $[1000001000]$. The process will be terminated after $M_0 \xrightarrow{t_0} M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_2} M_3$ and $M_5 \xrightarrow{t_5} M_6 \xrightarrow{t_6} M_7 \xrightarrow{t_8} M_8$ sequences.

The Petri net reachability analysis reveals the mechanisms in the first layer of authorization and prevents document access for the attacker u_o. However, the system is vulnerable to attack from a malicious insider u_o who could execute any operation. Systems performing only the first layer of authorization are prone to attacks of malicious insiders. The malicious insider u_m could create or change documents without adequate authorization control in the workflow.

4.2. Reachability analysis for the second layer of multilayer authorization model

While $[t_0]$ was not triggered in the second layer, the attacker u_o was not authenticated and the initial marking was never $[1000]$. The following case analyses authorization control on a malicious user u_m by reachability analysis on Petri net.

Case of ACL: Through this mechanism, u_m could execute operations if he is authorized as in list entries. The authorization mechanism seems to work properly but the sample case below presents the deficiency of authorization in workflow.

Let o_{sign} be the signature operation on a document and $o_{approve}$ be the approval of a document with digital signature. In the institutional structure u_a and u_m are users who have authorization to sign a document by performing the operation o_{sign} where u_m is a person in the purchasing office and u_a is the head of the office. Furthermore, authorized user u_a has the authority to approve purchase order document d, in that he is authorized to perform operation $o_{approve}$ ACL includes $\langle users, O_{sign}, permit\rangle$, $\langle heads, O_{approve}, permit\rangle$, $\langle personnel, O_{approve}, deny\rangle$ rules. $u_a\, u_m \in users$, $u_a \in$ heads ve $u_m \in personnel$.

If the malicious user u_m tries to perform the $\{u_m, o_{approve}\}$ operation according to the rules in the control list entry $\langle personnel,\ O_{approve}, deny\rangle \wedge u_m \in personnel$ then the triggers obtained from incidence matrix will be $[001000] = [100000] + [11000]\, I$ Consequently by following the $M_0 \xrightarrow{t_0} M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_3} M_3$ sequence, the malicious user u_m could not perform the operation and the workflow terminates in the place p_3 .

Document approval is the signing operation of a purchase document by authorized person u_a. The $o_{approve}\,(d)$ operation is actually the $o_{sign}\,(d)$ operation. In the workflow the malicious user u_m cannot perform the $o_{approve}$ operation, but u_m could bypass the authorization control using the control list rule $\langle personnel, O_{sign}, permit\rangle \wedge u_m \in personnel$ over p_2 and could perform the $o_{sign}(d)$ operation in place p_5 The authorization deficiency in the mechanism makes it possible to sign a document as an approved purchase order.

It has been mentioned that access control lists have authorization in institutional operations [9]. The mechanism cannot provide solutions for promotion, demotion, revocation, and delegation requirements of an institutional authorization structure. As stated before, the adaption of ACLs to this requirement causes management difficulties.

4.3. Reachability analysis for the third layer of multilayer authorization model

While $[t_0]$ was not triggered in the third layer, the attacker u_o does not get authenticated and the initial marking will never be $[1000000]$. The following case analyzes authorization control on a malicious user u_m by reachability analysis on Petri net.

Case of RBAC: Through this mechanism, the malicious user u_m could execute operations that his role permits. The ACL mechanism has an override deficiency, as stated in the second layer. The role based system is designed to be a solution to cover institutional procedures and operations by allocating institutional roles and privileges to the users.

If the malicious user u_m tries to perform the $\{u_m, o_{approve}\}$ operation according to the rules $\langle personnel, o_{approve}, deny\rangle \wedge u_m has\, role$ "$personnel$" by the $M_0 \xrightarrow{t_0} M_2 \xrightarrow{t_5} M_3 \xrightarrow{t_6} M_4$ sequence, on places $\{p_0 p_1, p_5, p_4$, according to his reachability, u_m could not perform the operation and the workflow terminates at place p_4.

Authorized user u_a can perform the $\{u_a, o_{approve}\}$ operation at place p_7 with the $\rangle heads, O_{approve}, permit\rangle$ $\wedge u_a$ has role "head of purchase unit" authorization.

This operation can be described in detail as the malicious user u_m cannot perform the $o_{approve}$ operation. According to the SoD rule of RBAC, any user u_i in the system cannot perform an approval operation if the order is prepared by him. The SoD forces $r : o_{purchase} \rightarrow u \wedge r : o_{approve} \rightarrow urule$ onto role r.

The institutional workflows have promotion, demotion, and revocations of roles in the role hierarchy. Role based authorization also supports delegation of roles, which is a common implementation in institutions. Delegation is the assignment of the role of a user to another user within set intervals. The user will possess privileges that he did not previously have.

4.4. Reachability analysis for the top layer of multilayer authorization model

Through this mechanism, a malicious user u_m is forced to execute operations that his role permits according to institutional regulations.

Case of regulations: According to the reachability analysis in Figure 6:

If malicious user u_m tries the $\{u_m, o_{approve}\}$ operation according to the rules $\langle o_{approve}, e_{purchaseapproval}, personnel, u_m, d_i\rangle \wedge u_m has\, role\, personnel \wedge d_i states\, no$thing for $personnel$ role over $o_{approve} \rightarrow \{0\}$ By the $M_0 \xrightarrow{t_0} M_2 \xrightarrow{t_5} M_3 \xrightarrow{t_6} M_4$ sequence, u_m will not be able to perform the operation and the workflow terminates at the place p_4.

If malicious user u_m tries to present a signed document as an approved document to the workflow, the system detects the unauthorized operation by the authorization information supplemented with the procedure.

The malicious user u_m can perform the $\{u_m, o_{sign}, e_{personalsign}\}$ operation at the place p_7, authorized with the $\langle o_{sign}, e_{personalsign}, personnel, u_m, d_i\rangle \wedge u_m\, has\, role\, personnel \wedge d_i \;\; states\, no$thing for $personnel$ role over $o_{sign} \rightarrow \{1\}$ Operational type prevents the signed document being treated as order approval. The operation must be declared as $\{u_m, o_{sign}, e_{personalsign}\}$.

The malicious user u_m can perform the $\{u_m, o_{approval}, e_{delegatedapproval}\}$ operation at the place p_7, authorized with the $\langle o_{approval}, e_{delegatedapproval}, personnel, u_m, d_i\rangle \wedge u_m\, has\, delegated\, role\;\, approval\, authority \wedge d_i$ rule, which states that delegated authorities could only sign up to \$50K orders over $o_{approval} \rightarrow \{1\}$ The operation is restricted by d_i over delegated role r_d. While the operation is described as $\{u_m, o_{approval}, e_{delegatedapproval}\}$, the operational type confirms that the signed document will be treated as an approval of the order. Thus the malicious user u_m could not perform an unauthorized operation or present an actual unauthorized procedure as an authorized operation.

The fourth layer of authorization implements the restrictions of institutional regulations and policies. The authorization mechanism provides authorization information for the critical operations. These functions increase the reliability of the documents generated in the workflow.

Each layer has positive and incremental effects on reliability but these contributions are not adequate for total reliability. The reason for this is deficiencies in authorization mechanisms, which are presented in the analysis section. In each layer the effect of the authorization control is an enhancement. Incremental authorization information appended to the validation supports the reliability of the document in the workflow.

4.5. The overview analysis of the model

The authorization information employed in authorization control is the measurement for precision of authorization. In the first layer, the information is formed by the identity of user. In upper layers the authorization information is updated incrementally with operations defined in ACLs, roles and processes, role delegations, and restrictions in institutional policy and regulations, respectively. Definitions of authorization a and approval y stated in each layer provide incremental precision for authorization.

Complexity of authorization control is related to the scope of the authorization. At the first layer, authorization control grants system access by user identification. At the top layer of authorization the regulatory restrictions must be controlled for authorization. The management and verification of y in each layer becomes more complex than the prior one.

The approved operation set after the authorization process is another attribute of the layers of authorization in the model. The scope of authorization control on operations is determined by this operation set. In the first layer, authorization grants system access and it covers the largest set of operations. At the higher layers operations are specialized and it narrows the scope. The operation $o \in O$ in authorization a in each layer establishes the scope. The user scope has identical properties with scope of operations. In the first layer, authorization control encloses all users $u \in U$. Afterwards, the authorization is specialized on related users by roles, groups, and operations.

4.6. Fields of use

The multilayer authorization model is primarily proposed as a framework to analyze authorization methods by presenting their relationship with each other and also their contributions to the authorization process. A fourth and top layer is also proposed in this work to solve authorization problems caused by unhandled institutional regulations.

The model would be a basis guide for those implementing authorization in institutional workflows. They can build up the system by requirements according to the facilities of the layers. The decision makers can settle on adequate authorization in accordance with the scope and the operational boundaries of the layers of the model. The model provides a system workflow template that practitioners can use to analyze their system. If the applications in the system cannot accomplish the requirements, they may choose to upgrade authorization mechanisms as in the upper layers in the model.

The proposed model would not be useful for single user systems where users have full authorizations. These types of systems have a single big layer of authorization that permits the user to execute all operations or denies any access. From mobile clients to distributed systems the authorization model may be the initial analysis step to make decisions on the implemented authorization mechanisms and authorizations.

The model is built based on institutional workflows where authorizations are crucial. The institutional authorizations that are defined by regulations are generally missing or have not been addressed in most systems. The top layer of the model is proposed to reveal and overcome this authorization vulnerability. The model seeks to depict that the authorization formation is not complete yet. There may be another top layer addressing authorization requirements of a special application. As stated before, the proposed top layer is proposed to

solve the authorization problem in institutional regulations. The multilayer model can be a guide to examine the actual authorizations in multirole/authorization systems. This can trigger an upgrade to the authorization scope by replacing the current layer of authorization with a superior one. The model would be an incentive to analyze and expose any unnoted but critical deficiencies.

5. Conclusion and future works

In this work a multilayer authorization model is proposed. The model is constructed on functionality, precision and scope of authorization, operational range, and authorization effectiveness of the authorization mechanisms. The reliability of documents in a workflow is analyzed by reachability analysis on Petri net models of the layers. The institutional authorization deficiency of the layers is presented and a solution based on authorization with institutional regulations is proposed. A reliable document must be created through proper processes in institutional workflows and must be produced according to institutional policy and regulations. A document in the workflow could be analyzed with the reachability analysis by the proposed model. If the document was created or altered by an unauthorized user, the analysis identifies it.

Also the policy based authorization mechanism proposed for the fourth layer improves reliability of the document in a workflow. The mechanism provides authorization control according to institutional policy and regulations where known authorization mechanisms fail. The Petri net models and analysis were designed to present functionality of the mechanisms in the workflow, but were also kept simple to explain the authorization deficits. Reachability analysis on advanced workflows may reveal new problems of authorizations in institutional workflow.

Petri net analysis is generally used for analysis of workflow flaws. To the best of our knowledge, the paper is novel for using reachability analysis for authorization purposes in a workflow.

The proposed model and reachability analysis on authorization can be used as an effective tool for ongoing reauthorization analysis in workflows. The regulation based authorization solution is simple and effective to detect unauthorized operations in a workflow and provides authorization proofs for verification of reliability. The administrative cost of the proposed authorization solution is high as the method comprises institutional policy and regulations as authorization information.

For simplicity, only the fundamental authorization mechanisms are presented in the model. The layers of the model can be extended by supplementing other authorization mechanisms according to their authorization capabilities. The layers may not be a bulk layer in that case, where multiple mechanisms may split a layer.

References

[1] PwC, CSO Magazine, the U.S. Computer Emergency Readiness Team (CERT) Division of the Software Engineering Institute at Carnegie Mellon University, and the U.S. Secret Service .2014 US State of Cybercrime Survey. CSO Magazine April 2014.

[2] PwC, CSO Magazine, the U.S. Computer Emergency Readiness Team (CERT) Division of the Software Engineering Institute at Carnegie Mellon University, and the U.S. Secret Service 2013 Cyber Security Watch Survey. CSO Magazine, 2013.

[3] Schneier B, Ranum M., Schneier-Ranum Face-Off: Is perfect access control possible? Information Security Magazine, 2009.

[4] Poovendran, R., Narayanan, S. Protecting patient privacy against unauthorized release of medical images in a group communication situation. Computerized Medical Imaging and Graphics, 2005; 29: 367-383.

[5] Fakhari P, Vahedi E, Lucas C. Protecting patient privacy from unauthorized release of medical images using a bio-inspired wavelet-based watermarking approach. Digital Signal Processing 2011; 21: 433-446.

[6] Neuman BC, Ts'o T. Kerberos: An authentication service for computer networks. IEEE Communications 1994; 32: 33-38.

[7] Rigney C, Rubens A, Simpson W, Willens S. Remote authentication dial in user service (RADIUS). RFC 2138, April 1997.

[8] Jie W, Arshad J, Sinnott R, Townend P, Lei Z. A review of grid authentication and authorization technologies and support for federated access control. ACM Computing Surveys 2011; 43: 12.

[9] Barkley J. Comparing simple role based access control models and access control lists. In Proceedings of RBAC '97, ACM. NY, USA, 1997, pp. 127-132.

[10] Ferraiolo DF, Kuhn R, Sandhu R. RBAC standard rationale: comments on a critique of the ANSI standard on role based access control. IEEE Security & Privacy 2007; 5: 51-53.

[11] FIPS PUB 186-3 Digital Signature Standard (DSS), 2009.

[12] Tan K, Crampton J, Gunter C. The consistency of task-based authorization constraints in workflow. In Proceedings of the 17th IEEE Computer Security Foundations Workshop. IEEE, 2004, pp. 155-169.

[13] Dempsey K, Ross RS., McGuire KS. National Institute of Standards and Technology (NIST) Supplemental Guidance on Ongoing Authorization (OA). June 2014.

[14] Ferraiolo DF, Kuhn R. Role Based Access Control, In: 15th National Computer Security Conference, Oct 13–16, 1992. pp. 554-563.

[15] Lui RWC, Hui LCK, Yiu SM. Delegation with supervision. Information Sciences, 2007; 177: 4014-4030.

[16] Coyne E, Weil TR. ABAC and RBAC: Scalable, flexible, and auditable access management. IT Professional, 2013; 15: 14-16.

[17] The OAuth 2.0 authorization framework. IETF, RFC6749, 2012.

[18] ANSI, American National Standard for Information Technology—Role Based Access Control, ANSI Int'l Committee for Inf. Tech. Stds, 2004, pp. 359.

[19] Yuqing S, Qihua W, Ninghui L, Bertino E, Atallah M. On the complexity of authorization in RBAC under qualification and security constraints. IEEE T Dependable Secure Computing, 2011; 883-897.

[20] Fakhari P, Vahedi E, Lucas C. Protecting patient privacy from unauthorized release of medical images using a bio-inspired wavelet-based watermarking approach. Digital Signal Processing 2011; 21: 433-446.

[21] Freudenthal E, Das B. VPAF: a flexible framework for establishing and monitoring prolonged authorization relationships, In: CollaborateCom, IEEE, 2009.

[22] Jensen, K. Coloured Petrinets. Basic concepts, analysis methods and practical use. Monographs in Theoretical Computer Science, Vol. 1. 1992.

[23] Al-Azzoni I, Down DG, Khedri R. Modelling and verification of cryptographic protocols using coloured Petrinets and Design/CPN. Nordic Journal of Computing 2005; 12: 200-228.