Journal of Social Sciences *Vol. III, no. 3 (2020), pp. 72 - 84*
Fascicle *Social Science* *ISSN 2587-3490*
Topic *Economics and Management* *eISSN 2587-3504*

DOI: 10.5281/zenodo.3971967
UDC 004.6:334

MODELING ACCESS CONTROL AND USER ACTIONS USING TRUST - BASED ACCESS CONTROL POLICIES

Marcel Danilescu*, ORCID ID: 0000-0002-6561-7955

"Danubius" University, 3 Galati Bd., Galați 800654, Romania
**marcel.danilescu@aswic.ro*

Received: 04. 22. 2020
Accepted: 06. 28. 2020

Abstract. This paper is a natural continuation of previous research on the implementation of confidentiality in small, medium, and virtual enterprises. This research began in 2009, with the work "*Xml Based Techniques for Data Privacy in E-Business*" which revealed that for small, medium, and virtual enterprises that often represent start-ups, trust is a basic element. Thus, in 2010, we published the paper "*Control access to information by applying policies based on trust hierarchies*". Present work shows a method of modeling the hierarchies of trust in computer systems. After a review of previous research and the presentation of the necessary conditions for accessing and applying actions on an object, published in *"Data security management applying trust policies for small organizations, ad hoc organizations and virtual organizations*" in 2012, this paper presents the four necessary steps, from the analysis of the manual system necessary to be automated to the determination of the access policies of the users.

- Stage 1 - analysis of the existing situation - refers to the identification of actors, and actions that apply to an object.
- Stage 2 - creating the computer system and optimizing the flow - presents the new, optimized system that will be implemented.
- Stage 3 - dealing with exceptions - separation of duty - presents various situations that may arise during the operation of the computer system and how they should be treated.
- Stage 4 - creation of the hierarchy of actions attached to the document, a model of the hierarchy of actions is presented, which will be attached to the document and can lead to the creation of access policies.

Keywords: *analysis, applications, authorization, organizations, relationships, workflow.*

Rezumat. Prezenta lucrare vine ca o continuare firească a cercetărilor anterioare privind implementarea confidențialității în întreprinderile mici, mijlocii și virtuale. Aceste cercetări au început în 2009, cu lucrarea " *Xml Based Techniques for Data Privacy in E-Business*" care ne-au relevat că pentru întreprinderi mici, mijlocii și virtuale care de multe ori reprezintă start-up-uri, încrederea este un element de bază. Astfel, în 2010, am publicat lucrarea "Control access to information by applying policies based on trust hierarchies". Lucrarea prezintă o metodă de modelare a ierarhiilor de încredere din cadrul sistemelor informatice.

După o trecere în revistă a cercetării anterioare și prezentarea condițiilor necesare pentru accesarea și aplicarea unor acțiuni asupra unui obiect, publicate în "*Data security management applying trust policies for small organizations, ad hoc organizations and virtual organizations*" în 2012, în cadrul lucrării sunt prezentate etapele necesare de parcurs, patru la număr, de la analiza sistemului manual necesar a fi automatizat până la determinarea politicilor de acces ale utilizatorilor.

Etapa 1-a – analiza situației existente – se referă la identificarea actorilor, și acțiunilor ce se aplică unui obiect.

Etapa a 2-a – crearea sistemului informatic și optimizarea fluxului – se prezintă sistemul nou , optimizat ce va fi implementat.

Etapa a 3-a – tratarea excepțiilor-separarea sarcinilor – prezintă diverse situații ce pot apărea în decursul exploatării sistemului informatic și cum ar trebui tratate.

Etapa a 4-a – crearea ierarhiei de acțiuni atașate documentului, este prezentat un model de ierarhie de acțiuni, ce va fi atașat documentului și va putea duce la crearea politicilor de acces.

Cuvinte cheie: *analiză, aplicații, autorizare, organizații, relații, flux de lucru.*

Introduction

The '90s of the last century were decisive in the development of computer systems, through the explosion of communication technologies, the opening to the Internet and the emergence of online applications.

Along with these, there were also risks of access to confidential data by unauthorized persons, which could lead to the compromise of computer systems and data to be processed.

In this context, there have been requirements to counteract unauthorized access actions and to carry out various actions on the details within the computer systems. Unauthorized action research teams have been set up and the aim has been to create the most effective solutions for preventing access and taking various actions in existing computer systems.

Prior to these times, two application design systems were imposed, MAC (Mandatory Access Control), which are characteristic of military multilevel applications and DAC (Discretionary Access Control) which are for governmental and civilian organizations.

In 1992 FERRAIOLO D. F. and KUHN D. R. [1] at the NIST (National Institute of Standards and Technology - National Computer Security Conference), presented RBAC (role-based access control), a new approach to civil and government applications, which become a de facto standard for access control and for which much research has been done over time [2 - 5].

In the context of large organizations, RBAC brings easy administration by managing roles and creating templates for them, allowing a role manager to assign them to different subjects.

However, the design of information systems did not allow the possibility that any changes in the organizational structure would be sufficiently dynamic, which impedes in the case of small, medium, or virtual organizations.

In general, these organizations, which are often start-ups, start-up staff consists of enthusiasts, who eventually did not know each other, but gained a reputation, based on experience, fairness, loyalty, etc. and who gained confidence, which led to their co-opting into the organization.

In this context, we have researched ensuring the confidentiality of data through trust-based access policies, which is the support for the cohesion of groups in these organizations [6 - 11].

In their research, Danilescu L. and Danilescu M. [12 - 14] presented and analyzed the relationships within an organization, and the fact that trust is important in creating formal hierarchies that lead to the creation of authorization policies for various actions within the enterprise.

In this research, the use of trust was considered as the main parameter, based on which to build policies, because at the base of these enterprises, at least in the start-up phase, it is very important, being necessary in the selection of staff and tasks assigned to them.

Thus, it sought to develop a first model for achieving policies based on trust.

Subsequent research has sought to develop concepts of trust, and ways to quantify them, especially in virtual organizations, as in many of them, partners may never physically interact. Research on how to assess trust in social networks was presented in the paper "Assurance model behavior in social networks based on trust" [15], where a first method of creating hierarchies of trust. In 2012, in the paper “Data security management applying trust policies for small organizations, ad hoc organizations and virtual organizations” [16] it was defined the relationship between users, objects (data, documents), and the hierarchy of user actions on objects by applying a policy based on trust.

In the following we briefly present this relationship.

Let be $Oi \in GO \wedge Pi \in P$ *where* $Pi=(p1,p2,...pk...pn)$ *, and* $pk=Hk(Ak)\ Hk(Ek)\ Fk$

for $\forall Ak, \exists(Uk \in Gm \Rightarrow \exists Ru\ ,\ Ru(Uk)=Ra(Ak) \wedge Ru(Uk)= \leq Rg) \oplus$

$\exists(Ux \in Gm \Rightarrow \exists Ru,\ Ru(Ux)=Ru(Uk) \wedge \exists dev(Uk)$ *for* $Ux) \oplus$

$\exists(Ux \in Gm \Rightarrow \exists Ru,\ Ru(Ux)=Ru(Uk) \wedge \exists dev(Uk)$ *for* $Ux \Rightarrow rev(Uk) \in RE$

$\wedge \neg \exists rev(Ux) \in RE\ )$

Where:

Ak = an action applied to one object;

dev = delegation received from a user *Uk*;

DE = the crowd of delegations;

Fk = flow sequences;

Gm = User group of which one user *Uk* is part;

GO = Group of objects;

Hk(Ak) = the corresponding action hierarchy to the pk subprocess;

Hk(Ek) = the corresponding hierarchy of events to the pk subprocess;

Oi = Object *i;*

Pi = The process applied to *Oi*;

p1..pn=numbers of subprocess of *Pi*;

Ru = confidence level of the U user, that is needed for the *Oi* object;

Rg = confidence level for the *GM* group;

Ra(Ak) = level of confidence necessary to the enforcement of the *Ak* action;

rev = restriction applied to the user *Uk*;

RE = the crowd of restrictions;

Uk= the user designated to execute the *Ak* action;

Ux = a user who belongs to the group *Gm*.

From the above we can determine the conditions for the implementation of the different types of access control policies, from the general type, to MAC or DAC.

Definition: We call a policy of type access control generally, a policy that does not include any restrictions and delegation of a user in the time of processing of the objects.

Basically, such a policy is applied in the first phase of creating an organization when there is no history of actions of its members, there were no events which had disturbed the organization, and its members were integrated into the organization on the required criteria applied subjectively, according to opinion made the recommendation received, the result of the interview, proposals, etc.

The paper established the importance of the workflow in the analysis of work processes for the design of the new IT system and trust-based policies.

This paper complements the previous works, by providing an example of modeling a business process and transforming it into computer systems, applying trust policies.

Implement reliable policies for access control and actions

Let a computer system *Si*:

Si = {Si1 ... Sin}

where *Siy* is a software module that can be:

- GUI (graphical user interface),
- a procedure,
- a function,
- or a manual or automatic procedure, acting on a set of objects (documents or data),

O = {O1,..., Om}.

For,

$\forall Si \Rightarrow \exists Gm = \{Gm1..GMx\}, \forall Gm(i) \neq \varnothing \wedge i=\{1..x\}$

where *Gm* is a plurality of user groups affecting various *Ak (y)* actions and,

Ak = {Ak1... Akz},

ordered within a *Hk (Ak)* hierarchy,

$\Rightarrow Hk(Ak) \Rightarrow \exists Hg(Gm)$,

$\forall Hk(Ak) \Leftrightarrow Hg(Gm)$,

where *Hk* and *Hg* are assigned a value, namely the value of the trust given to a group of users to perform the corresponding action.

This leads to the conclusion that for $\checkmark Oi$ during the actions applied and changes the status according to them, actions that are applied by a group of users who have a certain value of trust. Therefore, this value of trust, which applies to actions, users can also be applied to *Oi* who is in a position to bear the action of the appropriate user group.

Through this value assigned to *Oi* we can determine the state in which *Oi* is, what was the previous action, and what action is to be applied to it. Therefore, a tuple hierarchy *(Oi, Gmi, Aki)* can be created, which describes the status of the document or data at a given time, the action applied via the software module, or the manual or automatic procedure by a certain group of users.

Modeling access policies and actions

To model the access control policies and actions, the objects on which the users intervene must be analyzed, the necessary processes must be analyzed and the workflow established (Danilescu, 2012). As an example, we will take the completion of a request for leave of absence of an employee within a company, based on manual procedures that we want to automate.

Stage 1 - analysis of the existing situation

In this stage we establish the set of users *Ux* (*Uxi*∈*Gm* - where *Gm* is the working group involved), the set of actions *Ak* (Ak_j ordered in a hierarchy), and the set of objects *Oi*, on which we act. To analyze the existing situation, we will analyze the existing manual processes in an enterprise, which we want to automate. After conducting the interviews with the actors participating in the process flow, we will make a summary of their actions, which we will centralize in a table, containing the steps required to be completed by the application submitted, the location where actions are taken on the application and the actor involved.

The table that summarizes the ones listed above in Table 1.

Table 1

Operations required for the analysis of the application for rest leave

Step nr.	Actions	Application status	Locations	Actor
1	Completion of request for rest leave	Irresolute	Job	Applicant employee
2	Completing the employee manager's point of view	Irresolute	Job management	Applicant employee manager
3	Application registration	Irresolute	Registry office	Registrar employee (secretariat)
4	Verification of rest leave rights	Irresolute	Staff - salary (staff records)	Personnel employee (personnel record)
5	Completion of the number of days of rest leave for which the employee is entitled, the number of days remaining to be taken.	Irresolute	Staff - salary (staff records)	Personnel employee (personnel record)
6	Data validation	Irresolute	Personnel - payroll (management)	Personnel manager
7	Approval / Rejection of rest leave	Irresolute	Enterprise management	General Manager
8	Application solution registration	Approved / Rejected	Registry office	Registrar employee (secretariat)
9	Receive request solution	Approved / Rejected	Job	Applicant employee
10	Receive request solution	Approved / Rejected	Personnel - payroll (management)	Personnel manager
11	Recording of holiday leave	Application approved	Staff - salary (staff records)	Personnel employee (personnel record)

Continuation Table 1

12	Archiving rest leave application	Application approved	Staff - salary (staff records)	Personnel employee (personnel record)
13	Completing the gross income of the employee	Application approved	Staff - salary (salary)	Personal employee (salary calculation)
14	Calculation of holiday leave allowance	Application approved	Staff - salary (salary)	Personal employee (salary calculation)
15	Accounting Notice	Application approved	Staff - salary (salary)	Personal employee (salary calculation)
16	Approval of payment of indemnity	Application approved	Bookkeeping	Accounting manager
17	Bank payment order	Application approved	Bookkeeping	Accounting employee

Once the analysis of users' actions, for a better understanding of the application process, the hierarchy of actions, their locations, according to "Data security management applying trust policies for small organizations, ad hoc organizations, and virtual organizations" (Danilescu, 2012), we will create the flow work, which is important in establishing the actions that will be automated, and their hierarchies.

The workflow, based on Table 1, is shown in Figure 1.

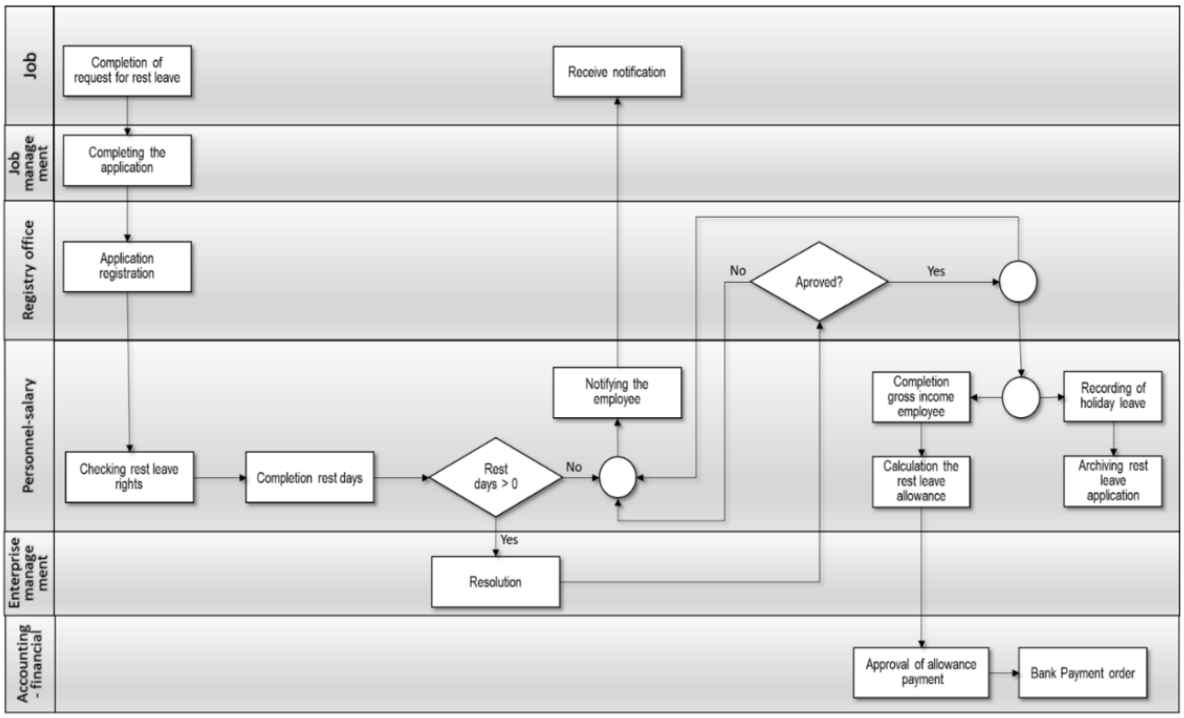

Figure 1. Workflow.

From table no.1 and figure no.1 we notice that there are situations in which the same actor during the data processing flow performs several tasks.

To optimize the automated process, a synthesis of user actions must be performed.

For this, we will try to centralize them in a work table (table 2), a table that will contain the actors, the number of stages in which they will intervene, the location and order of the actions performed.

Table 2

Synthesis of user actions

Actor	The step in which intervenes	Location	Actions
Applicant employee	1,9	Job	Create an application
Applicant Manager	2	Job manager	Completing the application
The employee at the registry office	3,8	Registry office	Application registration
Personnel employee records	4,5,11,12	Personnel	Verification, data completion, registration, archiving
Wages	14,15	Personnel	Compensation calculation, accounting information
Personnel manager	6,10	Personnel	Receipt of request, data validation
General manager	7	Enterprise management	Approval
Accounting's manager	16	Financial - Accounting	Approval of the payment of the allowance
Accounting's employee	17	Financial - Accounting	Completing the payment order

Following the steps, we will optimize the manual procedures, and design a new workflow, create a new summary of user actions, and create action hierarchies for users and documents.

Stage 2 - creating the computer system and optimizing the flow

To create the necessary computer system we will eliminate unnecessary manual procedures, and we will optimize the rest of the procedures, which we will automate.

From table 1, in the first phase we will eliminate the "Registry" location and the corresponding procedures, these being taken over by the newly created computer system. We will also abandon the procedure for verifying the documents regarding the record of the rest leave, creating an automatic procedure for this, using the documents registered in the previous records. We will keep the data validation by the personal compartment manager.

We also consider it unnecessary:

- registration of leave,
- (will be done automatically after approval),
- archiving it,
- supplementing the gross income,

- notification of the accounting service through the database,
- approval of the payment of the allowance,

these will be done automatically, through the computer system.
Thus we will obtain the following steps (table 3).

Table 3

Automating the submission and analysis application for leave

Step nr.	Actions	Application status	Locations	Actor
1	Completion of request for rest leave	Irresolute	Job	Applicant employee
2	Completing the employee manager's point of view	Irresolute	Management employee workplace	Applicant employee manager
3	Completion of the number of days of rest leave for which the employee is entitled, the number of days of leave taken remaining to be taken.	Irresolute	Salary staff (staff)	Personnel's employee (personal record)
4	Data validation	Irresolute	Personnel management	Manager
5	Approval / Rejection of rest leave	Irresolute	Enterprise management	General Manager
6	Application solution notification	Approved / Rejected	Job	Applicant employee
7	Rest leave approved	Application approved	Staff - salary	Personnel's employee (personal record)
8	Calculation of holiday leave allowance	Application approved	Staff - salary	Personnel's employee (personal record)
9	Completion of a bank payment order	Application approved	Financial - Accounting	Accounting's employee

Following the optimization of the procedures and their registration in table 3, we will move on to the next step, and we will design the workflow corresponding to the IT system.

This flow is shown in Figure 2, and its synthesis is no longer necessary as Table 3, shows the simplified procedures and their order.

From the analysis of Table 3 and Figure 2, we notice that the workflow has two components:

1. dynamic,
2. static.

The dynamic part of the flow is related to the employee's job.

The static part is the circuit of the document after leaving the workplace.

In the following we will deal with the static flow of the document.

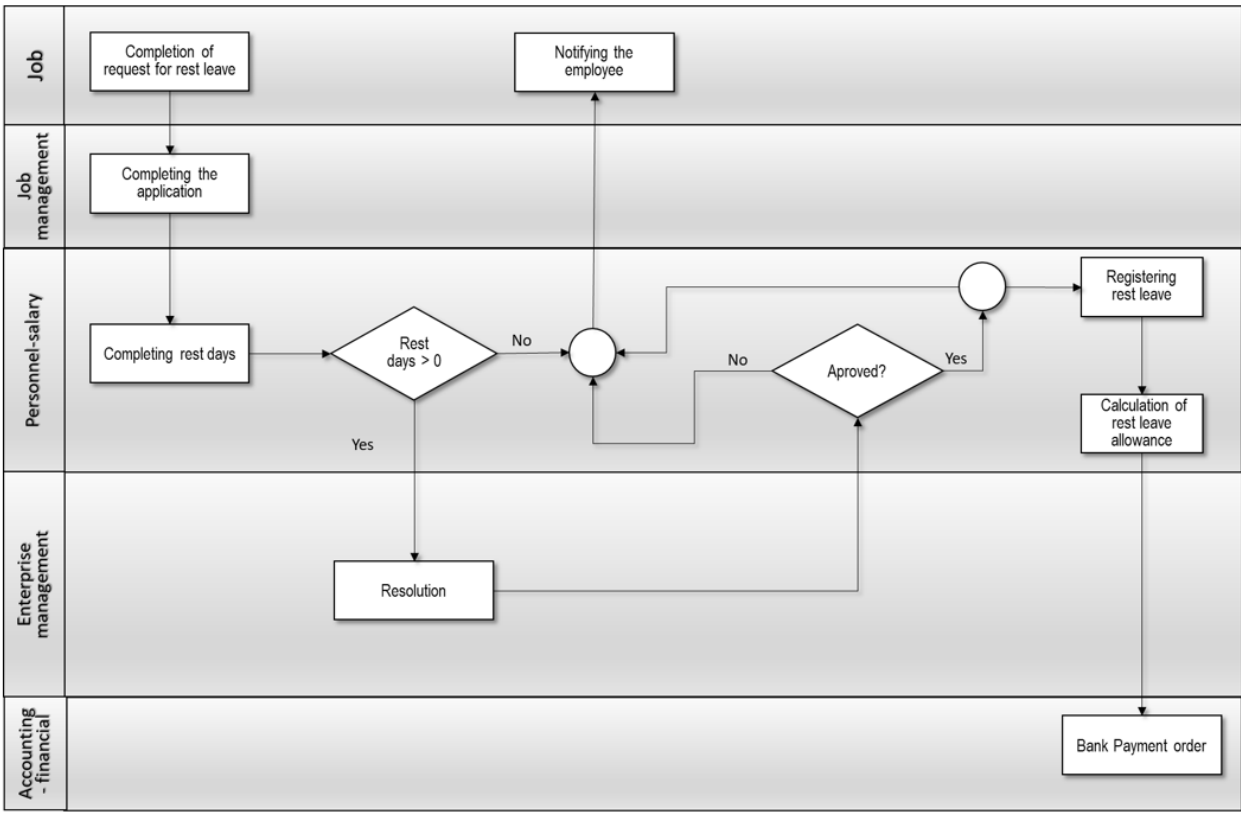

Figure 2. Workflow of the newly designed computer system.

Stage 3 - dealing with exceptions-separation of duty

In this stage we deal with the exceptions within the static flow. These can occur when an actor in the static workflow is the one who initiates a rest leave request. For this, we analyze the actors who intervene during the workflow and seek to avoid conflicts of competence (we do not want an actor who intervenes in the flow, to participate in the actions of their request). At this stage, user restrictions and delegations of actions are analyzed and established. These restrictions are necessary in order not to allow an actor to be applicable and responsible for an action that may influence the application, throughout the entire workflow.

Stage 4 Creation of the hierarchy of actions attached to the document

Table 3 extracts the elements necessary to create the hierarchies of actions necessary for the IT system, which allow the assignment of actions to the actors involved, to which are added the delegations of competences and restrictions in Table 4.

Once created the above, then you can create policies for the newly created object.

This assumes that for the newly created Oi_1 application, a hierarchy of allowed actions *Hk (Ak)* is created, actions that are assigned to users within the enterprise, which are organized in a *Hg (Gm)* hierarchy.

In the following we will consider:

Oi_1= application,

Oi_2=order bank.

The hierarchy of allowed actions created by *Hk (Ak)* is:

Ak_1 = complete data,
Ak_2 = date validation,
Ak_3 = approval / rejection,
Ak_4 = notices,

Table 4

Delegations and restrictions

Step nr.	Actions	Initial actor	Exception condition	Re-strict action	Dele-gation of action	Delegate actor
3	Completion of the number of days of rest leave for which the employee is entitled, the number of days of leave taken, and the number of days remaining to be taken.	Personnel employee (personnel record)	Rest leave, Medical leave, Missing staff (unemployed, other situations)	Yes	Yes	The user designated by the unit management (personnel's employee - personal salary or accounting)
4	Data validation	Personnel office manager	Rest leave, sick leave, absent staff (unemployed, delegations, other situations)	Yes	Yes	The User designated by the unit management (personnel's office employed or responsible for the accounting department)
5	Approval / Rejection of rest leave	General Manager	Rest leave, medical leave, other situations	Yes	Yes	The user designated by the unit management
6	Application solution notification	Personnel office manager	Rest leave, sick leave, absent staff (unemployed, delegations, other situations	Yes	Yes	The user designated by the unit management (personnel's office employed or responsible for the accounting department)

Continuation Table 4

7	Notifying the employee	Personnel's employee (personnel record)	Rest leave, sick leave, absent staff (unemployed, delegations, other situations	Yes	Yes	The user designated by the unit management (personnel office employed, or accounting staff)
8	Calculation of holiday leave allowance	Personnel's employee (salary calculation)	Rest leave, sick leave, absent staff (unemployed, delegations, other situations)	Yes	Yes	The user designated by the unit management (accounting staff, or personnel's manager)
9	Bank payment order	Accounting employee	Rest leave, sick leave, absent staff (unemployed, delegations, other situations)	Yes	Yes	User appointed by the unit management (accounting staff to replace the missing person or department manager)

Ak_5 = holiday registration

Ak_6 = allowance calculation,

Ak_7 = bank order completion.

To these actions we add the conditions of restriction rev_1 ... $rev_7 \in RE\ (Oi)$ and the delegations $dev_1..dev_7 \in DE\ (Oi)$ set out in table 4.

We establish the Gm user group consisting of Ux_1, Ux_2, Ux_3, Ux_4, Ux_5, Ux_6, Ux_7 users, who have the following positions within the organization:

Ux_1 = personal employee (personal record),

Ux_2 = personal manager,

Ux_3 = general manager,

Ux_4 = personal employee (salary),

Ux_5 = accounting employee (bank order),

Ux_6 = accounting manager,

Ux_7 = replacement general manager

Ux_8 = accounting employee.

The newly created access policies are expressed generically, below, by the expression

{Oi, Ukx, (Akx, revx, devx)}. (Danilescu Laura, Danilescu Marcel, 2011)

By filling in all the above values you will get the list of all access control policies created.

$$\{Oi_i,Ux_1,(Ak_1,rev_1,dev_1)\} \vee \{Oi_1,Ux_4,(Ak_1,dev_1)\} \vee \{Oi_1,Ux_5,(Ak_1,dev_1)\}$$

$$\{Oi_1,Ux_2,(Ak_2,rev_2,dev_2)\} \vee \{Oi_1,Ux_6,(Ak_2,dev_2)\} \vee \{Oi_1,Ux_4,(Ak_2, dev_2)\}$$
$$\{Oi_1,Ux_3, (Ak_3,rev_3,dev_3)\} \vee \{Oi_1,Ux_7,(Ak_3, dev_3)\}$$
$$\{Oi_1, ,Ux_2\ (Ak_4,rev_4,dev_4)\} \vee \{Oi_1,Ux_6,(Ak_4, dev_4)\} \vee \{Oi_1,Ux_4,(Ak_4, dev_1)\}$$
$$\{Oi_1,Ux_1,(Ak_5,rev_5,dev_5)\} \vee \{Oi_1,Ux_4,(Ak_5, dev_5)\} \vee \{Oi_1,Ux_5,(Ak_4, dev_1)\}$$
$$\{Oi_1,Ux_4,(Ak_6,rev_6,dev_6)\} \vee \{Oi_1,Ux_1,(Ak_6,dev_5)\} \vee \{Oi_1,Ux_2,(Ak_6, dev_1)\}$$
$$\{Oi_2,Ux_5,(Ak_7,rev_7,dev_7)\} \vee \{Oi_2,Ux_8,(Ak_7, dev_7)\} \vee \{Oi_2,Ux_6,(Ak_7, dev_7)\}$$

Conclusions and future work

In this paper, we analyzed and presented a methodology for modeling an access control system and actions based on trust for small, medium, and virtual enterprises.
From the above we can point out the following:

- understanding the existing situation, through the analysis performed, its tabular centralization, the creation of the workflow for the processed documents, is an important preliminary step in optimizing and achieving the new workflow,
- • also, the optimization of the new workflow by giving up unnecessary processes, and the transformation of processes manually into automatic processes is an important step in creating work hierarchies,
- the renunciation of the involvement of some actors repeatedly, in order to prevent some conflicts from the separation of tasks, is a necessity in the design of the new IT system
- the graphic realization of the two workflows, the existing one and the newly designed one allow the realization of the syntheses of the actors' actions.

Creating control policies for user actions is the last step before the actual implementation of the information system.

The paper aims to research and create a model of approach to modeling the control of actions based on trust, presented in previous papers.

This research covers an important stage in the development of IT systems to which trust-based access control policies apply.

The implementation of access control policies based on trust requires further research in order to improve the application model on the various objects, by refining the way in which access control is done.

In further research, we will address the dynamics of determining the degree of trust, the applicability of access control to various types of objects.

References

1. Ferraiolo D. F., Kuhn R. D. Role-Based Access Controls. 15th National Computer Security Conference In: *National Institute Of Standards AND Technology /National Computer Security Center*. (pp. 554-563). Baltimore MD: 1992.U.S.A. Retrieed from https://csrc.nist.gov/CSRC/media/ Publications/conference-paper/1992/10/13 /proceedings-15th-national-computer-security-conference-1992/documents/1992-15th-NCSC-proceedings-vol-2.pdf
2. Ferraiolo D., Kuhn R., Chandramouli R. *Role-Based Access Control* (2nd ed.). Norwood, MA 02062: Artec House, INC 2007.
3. Kuhn R., Coyne E., Weil T. Adding Attributes to Role-Based Access Control. *Computer (IEEE Computer)*, 43(6), 49-71. 2010 doi: DOI: 10.1109/MC.2010.155
4. Rajpoot Q. M., Damsgaard Jensen C., and Krishnan R. Integrating Attributes into Role-Based Access Control. *Proceedings of the 29th Annual IFIP WG 11.3 Working Conference on Data and Applications Security and Privacy* pp. 242-249 2015. Fairfax, VA, USA: Springer Verlag. doi:10.1007/978-3-319-20810-7_17
5. Sandhu R., Ferraiolo D., Kuhn R. The NIST Model for Role-Based Access Control: Towards a Unified Standard. In: *A. f. Machinery* (Ed.), RBAC '00: Proceedings of the fifth ACM workshop on Role-based access control (pp.

47–63). Berlin: Association for Computing Machinery, New York N.Y. United States. 2000. Retrieed from https://doi.org/10.1145/344287.344301.

6. Dasgupta P. Trust as a Commodity. In: D. Gambetta, ed., *Trust:Making and Breaking Cooperative Relations*. Oxford: Gambetta Diego. 2000. pp. 49-72 Retrieed from http://citeseerx.ist.psu.edu/viewdoc /download; jsessionid=6074C90B4EE45C43F7870DF2769592FB?doi=10.1.1.25.5180&rep=rep1&type=pdf
7. Einwiller S., Herrmann A. und Ingenhoff D. Vertrauen durch Reputation – Grundmodell und empirische Befunde im E-Commerce. *Marketing Zeitschrift für Forschung*.Ed.Springer Verlag Berlin ,pp. 24-40 2005.ISBN 3-824-7865-X Retrieed from https://books.google.ro/books?hl=ro&lr=&id=zsxrQ1H_AHUC &oi=fnd&pg=PR5&dq=Vertrauen+durch+Reputation+%E2%80%93+Grundmodell+und+empirische+Befunde +im+E-Commerce&ots=nCvdV1hJe-&sig=t_63sYhPZ1oMrxDNeDtwI4Z6C_E&redir_esc=y#v=onepage &q&f=false
8. Kasper-Fuehrer E. C. and Ashkanasy N. M. Communication trustworthiness. *Journal of Management,* ed. Academy of Management Vol 27 pp 235-254 2001 NewYork NY USA. https://doi.orh/ 10.1177/014920630102700302.
9. Lewicki R. J., Mcallister D. J., Bies R. J. Trust And Distrust: New Relationships And Realities. *Academy of Management Review*, (Academy of Management, Ed.) 23(3), 438-458. 1992 NewYork NY USA.
10. Mcknight H. D., Cummings Larry L. and Chervany Norman L. Initial Trust Formation in New Organizational Relationships In: *The Academy of Management Reiew*, 23(3), pp 473-490. (Academy. of. Management, Ed.) 1998 doi:DOI: 10.2307/259290
11. Möllering G., Bachmann R. and Lee S. H. Understanding organizational trust - Foundations, constellations, and issues of operationalisation.. *Journal of Managerial Psychology* pp.556-570 -2004 https://doi.org/10.1108/02683940410551480.
12. Danilescu L., Danilescu M. Control Access To Information By Applying Policies Based On Trust Hierarchies. In: *International Conference on Computer and Software Modeling*, ICCSM 2010 (pp. 285-290). Manila: Institute of Electrical and Electronics Engineers Inc. Printed in Chengdu. Ei Compendex, ISI Proceeding.
13. Danilescu L., Danilescu M. Organization's data access control policies based on trust. In: *EuroEconomica*. 2, 2010. pp. 113-122. Galați: Universitatea Danubius
14. Danilescu L., Danilescu M. Algorithms for Defining Trust-Hierarchies to Control Access to Information. In : E. A.S.E. (Ed.), *The Tenth International Conference On Informatica In Economy Bucharest* (p. CD). Bucuresti: A.S.E. - Bucuresti. 2011. https://www.ase.ro/upcpr/profesori/284/agendaIE2011.pdf
15. ADOMNICĂI C., DANILESCU M. Assurance model behavior in social networks based on trust. In: *3rd International Conference on Computer Technology and Deelopment* ICCTD Chengdu P.R.China. 2011.. ISBN:9780791859919 Retrieed https://asmedigitalcollection.asme.org/ebooks/book/187/chapter-abstract /35861/Assurance-Behaviour-Model-in-Social-Networks-Based?redirectedFrom=fulltext chengdu.
16. Danilescu M. Data security management applying trust policies for small organizations, ad hoc organizations and virtual organizations. In: *The Journal of Accounting and Management*, 2(3), pp 47-64 Galați: Universitatea Danubius 2012. Retrieed from http://journals.univ-danubius.ro/index.php/jam/article/view/1592
17. Backes M., Dürmuth M. and Rainer S. An Algebra for Composing Enterprise Privacy Policies. *Lecture Notes in Computer Sciences*, 3193.2004 Retrieed from https://doi.org/10.1007/978-3-540-30108-0_3 DOI:3193. 33-52. 10.1007/978-3-540-30108-0_3.
18. Danilescu L., Danilescu M. „Xml Based Techniques For Data Privacy In E-Business", *International Conference "Education and creativity for a knowledge society"* the third edition 2009, "Titu Maiorescu" University of Bucharest, ISBN 978-606-8002-36-1 pag. 15-18