Mandatory Access Control in Workflow Systems

Daniel Cvrček

ÚIVT FEI VUT Brno, Božetěchova 2, 612 66 Brno, Czech Republic

Abstract. This paper analysis problems of mandatory access control in strongly distributed information systems that solve computational tasks with long time durability (workflow management systems). We show that problem of mandatory access control may be solved on several levels. There is a strong difference between classification of data the system works with and classification of tasks' definitions. The latter situation is more difficult not only from the theoretical reasons but also from the administration and functional point of view. We are developing procedures for the task execution management that are based on the unified classification of data that are processed. This classification may be used not only for controlling the access to the resources but also for the determination of subjects that are allowed to execute parts of the workflow tasks (task-steps).

1 Introduction

Workflow management systems are primarily destined for commercial applications. This technology possesses special features that allow its implementation in business environment. One can say that it is opposite to classic mandatory access control models that allow very tight access control but the result is that there is not possible to use them in other environments because ”classification rules” make usage of such an information system very difficult. But workflow management systems are so valuable that there is a need to show possibility to use them in environments with high security needs - this is the reason why to define mandatory access control in workflow management systems.

The situation is not so easy, because authorization systems shall be viewed at like another layer above mandatory access control in particular autonomous information systems (s-nodes) that are parts of the given workflow system. Workflow management system defines sequences of atomic tasks (database or system resource accesses) so that the whole (workflow) performs some abstract operation (insurance claim, ordering material, creation of a new bank account, ...). The workflow authorization systems must ensure conditions settled for interoperation among s-nodes. Mandatory or discretionary access control (or other) models ensure correct resource accesses in particular s-nodes.

There are, from our point of view, two basic problems. The first one is to define mandatory requirements for inter-operational relations (redefine mandatory access control rules for workflows), the second one is to ensure the stated requirements. Solutions of the problems are demonstrated within frames of two different authorization systems. The first one is the system defined by Wei-Kuang Huang from Rutgers University ([4, 7]) the second one is ours (its basic ideas are based on works of Thomas [8, 9, 10]). Those two models are defined according to rather different concepts. The first model assumes that security requirements are part of the workflows' definitions. It means that processing of workflow steps, atomic

tasks, uses processing and security information at once. The second model defines security information separately from the functional information of a workflow. This approach allows implementation of a rather small unit that ensures fulfillment of security demands. There are some other works about authorization systems in workflow systems but they do not feature any integral view on the problem (e.g. [11]).

We show a classification of workflows for the mandatory access control on several levels. We show approaches that solve problems of mandatory access control on the previously defined levels. The results show necessity of two different authorization information that have to be managed during workflows execution.

The following section describes briefly discretionary access control to show its principles and disadvantages. The third section defines mandatory access control. Section 4 divides requirements for mandatory access control in workflow systems into consistent domains and outlines possible solutions. The last, fifth, section is dedicated to definition of tasks in the environment that contains autonomous information systems that have different security classification.

2 Discretionary Access Control

With discretionary access control is the access control for particular data object managed by it owner. A user or any of the user's programs or processes can choose to share objects with other users. The discretionary access control is therefore prone to certain kinds of leakage of information such as Trojan Horse attacks. On the other side, this approach is very flexible. The purpose of discretionary access control in workflow management systems is to provide controlled sharing of information by authorized subjects.

Workflow authorization system is destined for strongly distributed systems with large number of users and s-nodes. The administration of such a system is very difficult and security risks are gained its weight because of benefits that are obtainable by its defeating. The authorization system must support the following features:

1. Synchronize the delegation of authorizations with the execution of workflow.
2. Role-based access control: this is used to express a group if subjects (role) that possess the same authorization in performing the certain tasks.
3. Separation of duties - this imposes constraint to distinguished subjects that belong to the same role but perform different duties.
4. Temporal specification - adds the capability to specify the temporal constraint of an authorization such as the authorization is valid within a time interval.
5. Event-based authorization - provides a conditional delegation of authorization based on the existence of an event or a result of the previous task.

Let's mention two very important aspects of workflow authorization systems. Synchronization of authorizations and workflow processing is crucial for secure workflow execution. There exist several definitions of temporal predicates that restrict access rights but there are always defined for absolute time intervals (e.g. [1]). This is not sufficient for workflow environment because predefined specifications always allow access for more than the time required. The task itself does not need to start yet but it may already possess privileges to access the resources. It is highly appreciated to grant and revoke privileges not according to some predefined moments but to start and end of tasks.

The second aspect is separation of duties. The purpose is not to allow subjects to have sufficient authority to perpetrate a fraud on its own. Since the workflow decomposes a com-

plex activity into a number of smaller well-defined tasks, separation of duties naturally fits into workflow models. Current systems, however, do not support this feature.

3 Mandatory Access Control

Systems incorporating mandatory access control (MAC) are often called multilevel secure (MLS) systems. This arises from the fact that subjects (users and processes) and objects (files) have fixed security attributes that are used by the system to determine whether a subject can access a file or not. Those security attributes are assigned mandatory to all elements of the information system either administratively or automatically and can not be modified by the users or processes on request.

Information is categorized into classifications based on the data sensitivity and subjects obtain security clearance. Subjects are then permitted to access to a specified portion of the information. During determination of access granting the subject's clearance is compared with the classification of the data.

Security classifications are made up of two components. A hierarchical one, called the security level and a non-hierarchical one, called the category. The military security differentiates four security levels (unclassified, confidential, secret and top secret). The categories correspond to the content of information processed in the information system (e.g. Nato, Nuclear, Artillery,). The combination of security level and category forms security classes. Denning showed that security classes form a lattice [2] and presented a set of axioms that security policy must ensure.

The concept of mandatory access controls was first introduced by Bell and LaPadula [3]. The model uses two basic notions (1) subjects (active entities) and (2) objects (anything that holds data). The model, very briefly, does not allow read-up and write-down (security level). Subject acting on security level *secret* can not change content of objects with security level *confidential*. The subjects are in practice processes acting on behalf of users. A user with clearance *secret* must log in with security level *confidential* to be able to work with data classified by the latter security level.

The MAC models prevent Trojan Horse Attack but they are not free of security threats. Sensitive information can be still revealed by covert channels. Covert channels are paths not normally meant for information flow that could nevertheless be used to signal information. Two types of covert channels are identified (1) *covert storage channel* that is any communication path arising in the moment the object is written by a process and other process observers the effect (2) *covert timing channel* results from changes in the system performance that is observable and measurable by another process. The most important property of the covert channels is their bandwidth. Channels exceeding rate of 100 b/s are considered as having a high bandwidth.

Workflow in MLS workflows may belong to several security levels. Dependencies among workflows with the same security level are referred to as intra-level dependencies whereas those among workflows with different security levels are inter-level dependencies. From the security point of view only the latter are important.

Wei-Kuang Huang defines in [4] four levels of execution based on the degree of security and correctness it guarantees.

1. **SSSC-level** (strongly secure, strongly correct) An MLS workflow execution is said to of SSSC-level if it is secure and all the task dependencies are enforced. It assumes complete elimination of covert channels and still enforcing all dependencies.

2. **SSWC-level** (strongly secure, weakly correct) An MLS workflow execution is said to be of SSWC-level if it is secure but all the task dependencies need not be enforced.
3. **WSSC-level** (weakly secure, strongly correct) An MLS workflow execution is said to be of SSWC-level if it enforces all the task dependencies but may allow a low bandwidth covert channels.
4. **WSWC-level** (weakly secure, weakly correct) An MLS workflow execution that does not enforce all the task dependencies and still allows covert channels.

For commercial applications is this classification unusable. Majority of systems used, is supposed to perform workflows correctly and to satisfy security requirements. The author creates the classification for the purpose of redefinition of workflows in running time because of security dependencies that cannot be determined in advance.

4 MAC in Workflow Management Systems

Workflow management systems that implement mandatory access control are studied from various rather detailed aspects (atomicity of tasks [6], modeling MAC in role-based systems [12], ...). We want to analyze more abstract level.

Talking about workflow management systems and mandatory access control we have to see several levels of abstraction due to the workflow system's distributivity. There are at least two criterion we should considered. The first one is sensitivity of data:

1. *Only data are classified* - we are securing only data we are working with. Definitions of tasks are public and there is no reason to suppose them to be sensitive. The classic example may be administration where all the information flows should be transparent.
2. *Workflows' definitions are classified* - the opposite possibility when workflows are viewed as data and access to their definition is granted according to their classification.

The case when workflows are classified opens very interesting problems concerning their distribution, definition of composite tasks and workflows and so.

The second criterion concerns the s-nodes of the distributed information system.

1. *All autonomous systems (s-nodes) are equal from the security viewpoint* - e.g. universities where all centers may be equal because there are users covering all possible security classes in the workflow management system.
2. *Each autonomous system has its own security class* - any form of organization that has got its structure; in the moment there is no uniformity, each node (autonomous system) has got its own security class because only certain data are processed there.

The second case is again very interesting. An autonomous system has to be used to perform certain part of a workflow but it can not see any result of the previous processing and in some cases it can not see the definition of a workflow that does not concert it.

We can talk about another one aspect and it is a scope of dependencies we are solving. The first one describes dependencies inside one node (intra-node dependencies), the second one describes dependencies among nodes (inter-node dependencies). We define four categories according to the first two criterion:

1. WNSN (workflows are not classified, systems are not classified) - this is the simplest situation when there is no need for special treatment of workflows among autonomous systems

2. WNSC (workflows are not classified, systems are classified) - from the security point of view there is no difference regarding to the previous situation, we may use the systems' classifications for optimization of workflows distribution.
3. WCSN (workflows are classified, systems are not classified) - distribution of workflows is without limitations but there is a need for securing confidentiality during the workflow transfer between autonomous systems.
4. WCSC (workflows are classified, systems are classified) - his the most interesting situation because the workflows and systems are classified it means that not all the systems are allowed to see certain workflows' definitions.

All four cases have to solve intra-node dependencies. The WCSC has to solve also inter-node dependecies. To demonstrate the situation let us assume the following situation. One of tasks in the workflow management system solves hiring of new employees - airfield personal. Generally, there are two types of employees, working for civil companies or for the army. The part of the hiring process is check of the candidate career and this differs for the two different customers (civil and army). The army does not even want to show the process used for this check to the world wide. All the environment knows is existence of the procedure, its inputs and results. This situation demonstrates existence of two s-nodes classified as low and high.

4.1 Secrecy vs. Integrity

The mandatory access control can be used not only for the confidentiality of data (no read-up, no write-down) but also for the integrity of data (no read-down, no write-up), see [5]. To make such a model effective it has to be set highest and lowest security level from which is the subject allowed to read (write) objects. Those boundaries allow subjects to access data not only on the same security level but also data that are "slightly" more or less secure. One can see it as a potential weakening of the original mandatory access models but it is a potential way to solve problems with practical usage of an information system when counting with data integrity.

Intra-node dependencies Intra-node dependencies on SSSC-level are extensively solved in [4]. Four categories of high-to-low dependencies are defined according to two basic definitions.

Definition 1: A dependency between two tasks $tw_i \rightarrow tw_j$ is said to be conflicting if there exist at least two conflicting operations $o_i[d]$ and $o_j[d]$, $(i \neq j)$.

Definition 2: A dependency $tw_i \rightarrow tw_j$ is said to be result-dependent if the result of executing the child is different when the dependency is or is not enforced.

The way the inter-level dependencies are solved lies in redefinition of tasks that are defined with Petri-nets and in versioning of data. That redefinition arises from assuring the atomicity of multilevel transactions in relational database management systems. The conflicting, result-independent dependencies (high task reads low data before their modification by another low task) is solved by versioning of data from [6] and by splitting the high task. The read operations are reordered such a way that all reads on low data occur before the first low operation on data items and the task is then divided into partitions based on the data items it is accessing. We assume that this approach is not generally appropriate for the running time of the task, from several reasons.

- There is always an administrator that is the only one competent for the workflows' definitions.
- It is not desirable that system changes definitions of workflows during their execution according to the processed data.
- Definition of workflow should not create sequence when there are no reasons for it. Such relations should be solved by parallel processes.
- This approach solves problem with secrecy of information but says nothing about their integrity

Moreover, there is a problem that we are not generally able to prove equivalence of the original and converted task and any changes in the task's definition are very problematic. This is one of the basic idea in the authorization model that separates functional and security description of the task.

Creation of new tasks is always done by linking elementary, atomic actions. Such actions have defined their classification because we know type of data they are working with. Those information may be used during the task generation to avoid covert channels in this very beginning phase. There is one condition for the previous statement. There has to be defined one and only one general classification of data that is used throughout the workflow management system. We are only checking abidance of the specified rules and in case of their breach is the task execution terminated.

Inter-node dependencies Very interesting area are dependencies among autonomous systems (nodes). We can not find any solution of the problem in the open literature but we thing that this is what the MLS workflow management systems are about. We can look at the problem as generalization of intra-node dependencies with one exception. We do not have to look at underlying access control systems.

We have shown above, that this situation appears in the moment when as tasks as nodes are classified. You can see that this point of view is not important for a common user but for system administrator. The classification of nodes strongly determines the way the tasks are defined and executed.

Definition 3: Classification of the workflow management system $\mathcal{G}$ is a system of sets (equivalency classes) if each node from the set of available nodes is positioned in just one set $\mathcal{G}_i$. Assume there is defined partial-order relation ($\leq$) on $\mathcal{G}$ and there are two sets $\mathcal{G}_{\mathrm{o}}$ and $\mathcal{G}_{\infty}$ that satisfy the following conditions: $\forall \mathcal{G}_i : \mathcal{G}_i \leq \mathcal{G}_{\infty} \wedge \mathcal{G}_i \geq \mathcal{G}_{\mathrm{o}}$. The classification of WfMS $\mathcal{G}$ forms then a lattice.

Assume that we have a finite set of tasks that define actions available in the workflow management system. We can define classification of tasks $\mathcal{H}$ the same way as the previous definition.

In the moment we have two classifications $\mathcal{G}$ and $\mathcal{H}$ (algebraic lattices). The function we have used for their definition works always the same way. The function places a node or a task into an equivalence class according to the classification of data it is allowed to process or to its own classification.

We can define an abstract set of data classifications (security levels). We can define two projections e.g. f and g that assign one security level to each equivalency class ($\mathcal{G}_i$ or $\mathcal{H}_j$). In the moment we are able to say that a node from a class $\mathcal{G}_i$ is able to process a task from $\mathcal{H}_j$ iff $f(\mathcal{G}_i) \Leftrightarrow g(\mathcal{H}_j)$. We are even able to define other relations. It may be comparison ('$<$', '$>$', '$\leq$', '$\geq$' or '$\neq$' - incomparability). We can define other mappings that specify sublattices those set may be used for the comparison purposes.

5 Workflows Definitions

The most interesting situation, from the inter-node communication is WSCS. We have already declared that the minimum requirement is the existence of global data classification. This is the necessary condition for any inter-node dependency treatment. The basic assumption is that there are tasks that are processed in several nodes of the workflow system. It means that there must be a way to define those tasks. For the purpose we have to know about existence of tasks that is possible to process on particular nodes. We do not need definitions of the tasks but we need at least know that they exist, way to identify them, their inputs and their results.

Of course that there may be tasks that are internal for the particular node but those have to be assumed as private and cannot be used outside.

We have the following definition of workflow task security information - protection data.

Definition 4: Protection data is a 6-tuple:

$$PD = (\{\alpha | \alpha \in \mathcal{A}\}, id_{task}, id, id_{AU}, \mathcal{U}, \mathcal{G})$$

where $\mathcal{A}$ is a set of authorization units (security information for the particular task steps), id_{task} is an identification of the associated workflow task instance. id is identification this instance of the protection data and id_{AU} is an identification of the last processed authorization unit in this PD. The last two elements of the 6-tuple are an authorization usage log ($\mathcal{U}$) and classification of the PD ($\mathcal{G}$).

And let's define some rules for change of the PD classification - $\mathcal{G}$.

1. $\mathcal{G}_{\mathrm{o}}$ (the initial classification) is defined in the task template (description of the task) and this is derived from the minimum requirements stated by data that are to be processed with the task instances.
2. $\mathcal{G}_i$ (classification after the i-th task-step execution) is maximum from $\mathcal{G}_{i-1}$ and classifications of data actually processed. (Potentially we may assume some other factors).

Now assume the previously described example. There may be a need to define as few tasks as possible. We have to define two tasks for one action only because of different performing of career check. In the situation we have tasks classified we are able to define one task with several contents and selection is done in runtime according to the actual classification of the performed task. The trigger is $\mathcal{G}_i$. That may change during the task processing according to data processed.

This can be said to be a way of dynamic definition of workflow tasks that results from the security considerations. This approach may be used in the inter-node as well in the intra-node context.

6 Conclusion

We have shown a possibility to define workflows the way that is applicable not only in the homogeneous distributed systems but also in the environment that does not offer unified conditions for the workflows execution. The basis for the introduced method is the global data classification. This classification has to be known to all autonomous information systems (s-nodes), that the distributed workflow management system comprises from, and ideally does not change in time.

We have controverted some ideas that assume dynamic reconfiguration of workflows during execution. We say that the approach is applicable only during the workflow's definition

because of general security reasons. On the other side, we say that it is possible to use one definition of workflow for different *tasks*, working with distinct classification.

We have defined a way that can be used for determination of possible execution points for the workflows that is based on security classification of workflows and s-nodes. The presented procedure may be basis for the dynamic selection of the proper workflow definition.

References

[1] Bertino E., Bettini C., Ferrari E., and Samarati P. , *A temporal access control mechanism for database systems*, IEEE Transactions on Knowledge and Data Engineering, **8**(1), 1996, p. 67-80.

[2] E.D. Denning, *A lattice model of secure information flow*, Communications of the ACM, May 1976, p. 236-243.

[3] D.E. Bell, L.J. LaPadula, *Secure computer systems: Unified exposition and multics interpretation*, Technical Report MTR-2997, The Mitre Corporation, Bedford, MA, March 1976.

[4] W.K. Huang *Incorporating security into workflow management systems*, PhD thesis, Rutgers University, 1998.

[5] L.C. Dion, *A complete protection model*, Proc. IEEE Symposium on Security and Privacy (Oakland, CA), April 1981

[6] O. Costich and S. Jajodia, *Maintaining multilevel transaction atomicity in mls database systems with Kernelized architecture*, Database Security V: Status and Prospects (North Holland), 1992, p. 173-189

[7] V. Atluri and W.K. Huang, *An Authorization Model for Workflows*, Proceedings of the Fourth European Symposium on Research in Computer Security (Rome, Italy), September 1996.

[8] , R.K. Thomas and R.S. Sandhu, *Towards a Task-based Paradigm for Flexible and Adaptable Access Control in Distributed Applications*, Proceedings of the Second New Security Paradigms Workshop (Little Compton, Rhode Island), IEEE Press, 1993.

[9] R.K. Thomas and R.S. Sandhu, *Conceptual Foundations for A Model of Task-based Authorizations*, Proceedings of the IEEE Computer Security Foundations Workshop (New Hampshire), IEEE Press, 1994.

[10] R.K. Thomas and R.S. Sandhu, *Task-based Authorization Controls (TBAC): A Family of Models for Active and Enterprise-oriented Authorization Management*, Proceedings of the IFIP WG11.3 Workshop on Database Security (Lake Tahoe, California), Chapman & Hall, August 1997.

[11] E. Bertino, E. Ferrari, and V. Atluri, *A Flexible Model Supporting the Specification and Enforcement of Role-based Authorizations in Workflow Management Systems*, Proceedings of the Second ACM Workshop on Role-Based Access Control (Fairfax, VA), November 1997.

[12] N. Nyanchama and S.Osborn, *Modeling Mandatory Access Control in Role-based Security Systems*. In Database Security IX: Status and Prospects, 1996, p. 129-144.