Granular Deleting in Multi Level Security Models - an Electronic Engineering approach

Dirk Thorleuchter [1], Gerhard Weck [2], and Dirk Van den Poel [3]

[1] Fraunhofer INT, Appelsgarten 2, D-53879 Euskirchen, Germany
[2] Infodas GmbH, Rhonestraße 2, D-50765 Köln, Germany
[3] Ghent University, Faculty of Economics and Business Administration, B-9000 Gent, Tweekerkenstraat 2, Belgium
Dirk.Thorleuchter@int.fraunhofer.de,
dirk.vandenpoel@ugent.be, http://www.crm.UGent.be

Abstract. Data protection and information security can be assured by using a multi-level-security (MLS) access control model. However, a workflow between persons with different security levels is complicated by the fact that the exchange of information is only allowed in one direction: from persons that are assigned to a specific security level to person that are assigned to the same security level or to a higher security level (write up). Literature show solution approaches by using a MLS model with increased granularity. This enables distributing parts of documents to subjects of lower security levels without causing a security compromise. However, it does not consider an important aspect of workflows: the deleting of information. Thus, this work uses electronic engineering enlarges the introduced MLS model with increased granularity by integrating a deleting feature. This enables an improved workflow between persons with different security levels.

Keywords: Multi-level-security, Security, Modeling, Usability, Operating System, Electronic Engineering

Introduction

Today, many operation systems that are based on multi-level-security (MLS) are known [1]. As known from the use of these multi-level-security (MLS) models, a workflow among users (subjects) with different security levels is not permitted, because it is in contrast to the well-known ‘no write down’ and ‘no read up’ rules [2, 3, 4]. These rules say that a subject has to write information in a document (an object) that is assigned to an equal or a higher security level than the subject itself [5, 6]. Further, a subject can access to information from objects that are assigned to an equal or to a lower security level than the subject itself [7, 8]. This leads to two interesting aspects: a subject of lower security level could not read a text written by a subject of higher security level. Further, a subject of higher security level could not write information within a text written by a subject of lower security level. It could be clearly seen that a workflow between those two subjects is not supported by existing MLS operation systems [9, 10, 11].

To enable such a workflow, an increased granularity view on the data is nessesary. Such an MLS model is introduced in [12]. It is realized by storing information in objects of different security levels within one document. Thus, a document is defined as a set of objects. It is shown that this MLS model increase the usability by enabling a workflow among subjects. Further, it is shown that this model enables several knowledge extraction and text mining operations [13-22] on the data to extract relevant features e.g. an automatic assignment of textual patterns to security levels.

However, a disadvantage of this MLS model as well as of all further MLS models can be seen. They do not consider granular deleting operations. An object only can be deleted in total. Deleting parts of the object content causes a security compromise [23].

Granular deleting helps to increase the usability of workflows among subjects. In this paper, the existing granular MLS model is extended by implementing a granular deleting operation on the model. An example for the use of the electronic engineering based extended MLS model as well as conclusions and an outlook are given.

A new extended MLS Model

This new MLS model based on the Bell LaPadula model and it extends the high granular MLS model from [2, 19]. Here, we present a formal description of the new model.

Let an object $O\{i,j\}$ be defined as in Bell LaPadula model that consists of data, files, programs, subjects etc. The definition of a frame object $O^{sup}{}_{i}$ as a list of objects is taken over from [12] where $n \in N$ equals the number of frame objects in a multi-level-security system, $m_i \in N$ be the number of objects in $O^{sup}{}_{i}$, $i \in \{1,..,n\}$, and $j \in \{1,.., m_i\}$. Then, [12] formulize a frame object as

$$O^{sup}{}_{i} \underset{\cdot}{\equiv} [O\{i,1\}, .., O\{i,m_i\}] \qquad \mathbf{(1)}$$

As defined in [2, 12], C is the classification category (security level) and $C^{O\{i,j\}}$ is the corresponding classification category of an object. Further, K is the compartment information, P is the power set, and $PK^{O\{i,j\}}$ represents all needs-to-know categories of an object as calculated by the power set of all object specific compartment information. In contrast to [12], we define the deleting category of an object $Del^{O\{i,j\}} \in \{true, false\}$ as a boolean variable. Then, object categories can be formulized as

$$(C^{O\{i,j\}}, PK^{O\{i,j\}}, Del^{O\{i,j\}}) \qquad \mathbf{(2)}$$

We define a subject $S\{k\}$ as a process, programs in execution with subject categories $(C^{S\{k\}}, PK^{S\{k\}})$. This definition is not in contrast to the standard definition from Bell LaPadua because deleting of subjects is not a relevant feature in a workflow. We define $p \in N$ as the number of subjects in a multi-level-security system and we define $k \in \{1, ..., p\}$. Then, reading of object $O\{i,j\}$ by subject $S\{k\}$ is allowed if and only if

$$C^{S\{k\}} \geq C^{O\{i,j\}} \qquad \mathbf{(3)}$$

and

$$PK^{O\{i,j\}} \subsetneq PK^{S\{k\}}$$

and

$$Del^{O\{i,j\}} = \text{false}$$

In [12], an object $O\{i,j\} \equiv [data\{i,j,1\}, .., data\{i,j,q_{i,j}\}]$ is defined as a list of data units (e.g. line, sentence, text phrase, etc.). Further, $q_{i,j} \in N$ is defined as the number of data units in an object O{i,j} and $l \in \{1, .., q_{i,j}\}$ is defined as the position where a subject S{k} insert content. Then, a writing split $O_w\{i,j,l\}$ on position l of an object O{i,j} is defined as a list of three objects: $O_w\{i,j,l\} \equiv [O1\{i,j\}, O2\{i,j\}, O3\{i,j\}]$ with $O1\{i,j\} \equiv [data\{i,j,1\}, ..., data\_\{i,j,l\text{-}1\}]$ and $O3\{i,j\} \equiv [data\{i,j,l\}, \ldots, data\{i,j,q_{i,j}\}]$. Additionally, $O2\{i,j\} \in \varnothing$ is defined as a new and empty object. Writing of object O{i,j} by subject S{k} is allowed if and only if

$$C^{O1\{i,j\}} = C^{O3\{i,j\}} = C^{O\{i,j\}}. \tag{4}$$

and

$$PK^{O1\{i,j\}} = PK^{O3\{i,j\}} = PK^{O\{i,j\}}$$

and

$$Del^{O1\{i,j\}} = Del^{O3\{i,j\}} = Del^{O\{i,j\}}$$

and

$$C^{O2\{i,j\}} \equiv C^{S\{k\}}$$

and

$$PK^{O2\{i,j\}} \equiv PK^{S\{k\}}$$

and

$$Del^{O2\{i,j\}} \equiv \text{false}$$

and

$$O^{sup}_i \equiv [O\{i,1\}, \ldots, O_w\{i,j,l\} , \ldots, O\{i,m_i\}]$$

Let $[data\{i,j, h_1\}, ..., data\{i,j, h_2\}] \subsetneq O\{i,j\}$ be a list of data units that should be deleted from object O{i,j} by subject S{k}. Let $h_1 \in \{1, \ldots, q_{i,j}\}$ be the start position and let $h_2 \in \{1, \ldots, q_{i,j}\}$ be the end position, respectively. Let a deleting split $O_{del}\{i,j, h_1, h_2\}$ from position h_1 to position h_2 of an object O{i,j} be a list of three objects. $O_{del}\{i,j, h_1, h_2\} \equiv [O1\{i,j\}, O2\{i,j\}, O3\{i,j\}]$ with $O1\{i,j\} \equiv [data\{i,j,1\}, ..., data\_\{i,j,h_1\text{-}1\}]$ and $O2\{i,j\} \equiv [data\{i,j, h_1\}, ..., data\_\{i,j,h_2\}]$ and $O3\{i,j\} \equiv [data\{i,j, h_2+1\}, \ldots, data\{i,j,q_{i,j}\}]$. Let deleting from object O{i,j} by subject S{k} be allowed if and only if

$$C^{O1\{i,j\}} = C^{O2\{i,j\}} = C^{O3\{i,j\}} = C^{O\{i,j\}} \tag{5}$$

and

$$PK^{O1\{i,j\}} = PK^{O2\{i,j\}} = PK^{O3\{i,j\}} = PK^{O\{i,j\}}.$$

and

$$Del^{O1\{i,j\}}\ Del^{O3\{i,j\}} = Del^{O\{i,j\}}.$$

and

$$Del^{O2\{i,j\}} \dot{\equiv} \text{ true}$$

and

$$O^{sup}{}_{i} \equiv [O\{i,1\}, \ldots, O_{del}\{i,j, h_l, h_2\} , \ldots, O\{i,m_i\}]$$

An example for the use of this extended formal approach is given: Let the sentence: ‘The efficiency is 40 percent for a single cycle and 60 percent for combined cycle operations.’ be stored in an object. Deleting a text pattern ‘40 percent for a single cycle and’ is not possible in standard approach [2, 12, 24] because an object only can be deleted in total. Using the extended formal approach a new frame object is created that consists of two new objects labelled as not deleted: ‘The efficiency is’ and ‘60 percent for combined cycle operations’. It consists of one object labelled as deleted: ‘40 percent for a single cycle and’. A user can read in objects only if they are not deleted.

Conclusion and Outlook

This work shows a formal approach created with electronic engineering that theoretically enables subjects to delete objects without causing a security compromise. The default value for the object category ‘deleting’ is false. Deleting a textual pattern within a document means that a new object is created that contains the pattern. The object category ‘deleting’ is set to true for this new object while the deleting value of the other objects remain false. Thus, reading of deleted textual pattern is not possible. The new object itself remains in the MLS e.g. for possible restoring purposes. This new approach can be used to improve workflows among subjects without causing a security compromise. Future work should focus on realizing this new model in form of an operating system to evaluate the performance of this improvement. Further work should focus on a formulization for restoring information for this approach.

References

1. Pfleeger, P., Pfleeger, S.L.: Security in computing. Prentice Hall, Old Tappan (2003)
2. Bell, D.E., LaPadula, L.J.: Secure Computer Systems: Mathematical Foundations. Mitre Corp., Bedford (1973)
3. Biba, K.J.: Integrity Considerations for Secure Computer Systems. Mitre Corp., Bedford (1977)
4. Gericke, W., Thorleuchter, D., Weck, G., Reilaender, F., Loss, D.: Vertrauliche Verarbeitung staatlich eingestufter Information - die Informationstechnologie im Geheimschutz. Informatik Spektrum 32 (2), 102--109 (2009)

5. McLean, J.: A comment on the Basic Security Theorem of Bell and LaPadula. Inf Process Lett 20 (2), 67--70 (1985)
6. Feiertag, R.J., Levitt, K.N., Robinson, L.: Providing multilevel security of a system design. In: Sixth Symposium on Operating System Principles, pp. 57--65. ACM, New York (1977)
7. Obiedkov, S., Kourie, D.G., Erloff, J.H.P.: Buildung access control models with attribute exploration. Comput Secur 28 (1-2), 2--7 (2009)
8. Saltzer, J.H., Schroeder, M.D.: The protection of information in computer systems. Proc IEEE 63 (9), 1278--1308 (1975)
9. Li, E.Y., Du, T.C., Wong, J.W.: Access control in collaborative commerce. Decis Support Syst 43 (2), 675--685 (2007)
10. Holeman, S., Manimaron, G., Davis, J., Chakrabarti, A.: Differentally secure multicasting and its implementation methods. Computer Security 21 (8), 736--749 (2002)
11. Lindgreen, R., Herschberg, I.S.: On the validity of the Bell-LaPadula model. Comput Secur 13 (4), 317--333 (1994)
12. Thorleuchter, D., Van den Poel, D.: High Granular Multi-Level-Security Model for Improved Usability. In: 2nd International Conference on System science, Engineering design and Manufacturing informatization, pp. 191--194. IEEE Press, New York (2011)
13. Thorleuchter, D., Van den Poel, D., Prinzie, A.: Analyzing existing customers' websites to improve the customer acquisition process as well as the profitability prediction in B-to-B marketing. Expert Syst. Appl. 39 (3), 2597--2605 (2012)
14. Thorleuchter, D., Herberz, S. and Van den Poel, D.: Mining Social Behavior Ideas of Przewalski Horses. In: 3rd International Symposium on Computer, Communication, Control and Automation. LNEE, vol. 121, pp. 649--656. Springer, Berlin (2012)
15. Thorleuchter, D., Van den Poel, D.: Companies Website Optimising concerning Consumer’s searching for new Products. In: International Conference on Uncertainty Reasoning and Knowledge Engineering, pp. 40--43. IEEE Press, New York (2011)
16. Thorleuchter, D., Van den Poel, D.: Semantic Technology Classification. In: International Conference on Uncertainty Reasoning and Knowledge Engineering, pp. 36--39. IEEE Press, New York (2011)
17. Thorleuchter, D., Van den Poel, D., Prinzie, A.: Extracting Consumers Needs for New Products. In: 3rd International Conference on Knowledge Discovery and Data Mining, pp. 440--443. IEEE Computer Society, Los Alamitos (2010)
18. Thorleuchter, D., Van den Poel, D., Prinzie, A.: Mining Innovative Ideas to Support new Product Research and Development. In: Locarek-Junge, H., Weihs, C. (eds.) Classification as a Tool for Research, pp. 587--594. Springer, Berlin (2010)
19. Thorleuchter, D., Van den Poel, D.: Extraction of Ideas from Microsystems Technology. In: 2nd International Conference on Computer Science and Information Engineering. Advances in Intelligent and Soft Computing, Springer, Berlin (2012) in press
20. Thorleuchter, D., Weck, G., Van den Poel, D.: Usability based Modeling for Advanced IT-Security - an Electronic Engineering approach. In: International Conference on Mechanical and Electronic Engineering. LNEE, Springer, Berlin (2012) in press
21. Thorleuchter, D., Van den Poel, D.: Rapid Scenario Generation with Generic Systems. In: Management Sciences and Information Technology. Lecture Notes in Information Technology. IERI, Delaware (2012), in press
22. Thorleuchter, D., Van den Poel, D.: Using Webcrawling of Publicly-Available Websites to Assess E-Commerce Relationships, Service Research and Innovation Institute (SRII 2012), IEEE Computer Society, Washington (2012), in press
23. Bell, D.E., LaPadula, L.J.: Secure Computer System: Unified Exposition and Multics Interpretation. Mitre Corp., Bedford (1976)
24. Landwehr C.E.: Formal models for computer security. Comput Surv 13 (3), 247--278 (1981)