Review of Abstraction and Refinement of Layered Security Policy

Nathan Balon
SN# 3210 1717
CIS 544

Title

Abstraction and Refinement of Layered Security Policy by Marshall Abrams and David Bailey

Introduction

The paper "Abstraction and Refinement of Layered Security Policy" deals with the creation of a security policy for an enterprise. A security policy is used to capture the security requirements for an organization. The authors use the technique of defining a policy by basing it on layers and abstraction. The layered approach is used to show how different parts of an organization view the computing assets. Abstraction is used to hide the details of the security policy from the layers below.

Abrams and Bailey divide the security of an organization into three levels. First, top management has its own view of the security needs of an organization. Second, computer users have a separate view of the company's security and computing needs. Third, the process level view sees security at an even lower level.

The paper also looks at various ways to state the policy of an enterprise. The authors give three ways that policy can be stated. A security policy can be stated in natural language, mathematically, or in formal statements. Each approach to stating security policy has it own advantages and disadvantages. Furthermore, there must be a mapping of security policies at the different levels. A meta-policy, a policy that describes all of the policies, can also be used. The authors of the paper feel that by defining a security policy with different terms for different users they will have a complete security policy.

Problem

The dilemma that Abrams and Bailey are concerned with is creating a security policy for an enterprise. The problem in creating a security policy is that security can be viewed differently by different people within the same organization. Different people in an organization use different terms to describe the same asset. Top management's view of security is different from that of the computer users of an organization. The computer users of an organization also have a different view of security than the individual processes that are run on the organization's computers. Because different users have different needs and views of security, there must be a way to design the security for an enterprise and take the view of each level into account. The paper explores ways to create an enterprise level security policy, by using abstraction and refinement to overcome the problems related with creating multi-level security.

Review

The authors suggest that the solution to this problem is to use abstraction and take a layered approach to security. Different layers are used to describe the security policy for an organization. By doing so, individuals can help to define the security for an

organization in a way in which they are familiar. The authors state, "Abstraction serves the very useful purpose of suppressing details not of interest to the user." So the users at any one level can describe their concerns. The details from the lower levels are suppressed from the users. Users at high levels in the organization shouldn't have to concern themselves with how security is defined at lower levels. There is no need for top management to know in great detail how the security is actually implemented. By using a layered approach, security can be defined to different users of the information system in a language with which they are comfortable. The authors state that different users of the system use different metaphors to describe the same thing. In Abrams and Bailey's view, it is perfectly acceptable to allow these different people in the organization to use different metaphors because each person in the organization will see the enterprise from a different perspective.

Layers of Security Policy

In the paper there are three levels at which a security policy can be defined. Each of these levels defines security from a different perspective. The three levels used to describe a security policy are:

1. Top level management
2. Computer users
3. Process level

To start, top level management defines what they need from a security policy. Security needs of management are described at a high level of abstraction. Management views security as a whole for the organization and usually describes security in broad terms. An example of this could be management stating that there will be no unauthorized usage of the organization's computer system. As Abrams and Bailey state, management usually doesn't view security only on the basis of protecting computers but for the protection of the organization's information in general.

The next layer by which security is described is an individual computer user. They describe security based upon individual needs within the organization and upon the functions of the information system they use to complete their job. Examples at this layer are users entering new customers into the information system, updating inventory levels, or re-coding a new sale. These individual functions must have security policies associated with them.

The lowest level of the security policy is process based. At this level, security is concerned with individual bits of data flowing throughout the system. The process level security is concerned with the actual bits that are held in the computers buffers and flow throughout the computer system. An example at this level is how bits are transferred across the organization's network. Security policies must be in place to protect the enterprise system at the process level as well.

By using a layered approach to security, policies can be defined at each of these layers. The advantage of using a layered approach is that it allows those at each level to define security based on their domain of knowledge. Top management should not have to be concerned with how bits are represented in the enterprise system. Management then can define the policy based on the areas of expertise and not have to worry about the bits representation.

Techniques to Describe Policy

At each level, various techniques can be used to describe the security policy of that layer. The authors propose three ways for describing the security policy:

1. Natural Language
2. Mathematical Statements
3. Non-mathematical statements based on a formal model

First, a policy can be described using plain English. This approach is the easiest to use, but it isn't the best for all situations. The problem English presents is the ambiguity that is introduced by the language. For example, more than one word can have the same meaning in English or definitions can vary slightly. The advantage that English offers is it can be understood easily by all. Using natural language is the least effective way of describing a policy because of the ambiguity of language.

Mathematics can be used to describe a security policy. The advantage that mathematics has is that the security policy can be described very formally and precisely. There are some drawbacks to using mathematics to describe security. First, not all users have the background to understand the policy when it is expressed mathematically. A CEO of an organization would have a hard time understanding the security policy of the organization if it is written formally in mathematics. The programmers of the system may also have a hard time implementing a system just from mathematics. Second, not all users at the different levels can efficiently describe the policy by using mathematics. Top management would have a much easier time describing their needs of security by using natural language. Mathematics is the best approach to use to describe the policy of an organization, but it is not without its drawbacks.

The last way that a policy can be defined is by using a nonmathematical formal statement based upon a model. There are various models that can be used to describe the security policy of an organization. The Access Control Matrix is a model that is an example of how this type of policy can be defined. The Access Control Matrix can be easily used to show which individuals are allowed to access which assets in an organization. For example, the model can be used to show which employees can access payroll records. The problem with this type of model is that it can't be easily used at the process level that is concerned with bit level security. A formal model does have the advantage of being easily learned by others.

The best approach is to use a variety of these three methods to describe the security policy of an organization. At different layers, certain approaches are more effective at describing the security policy. For example the CEO could describe what they want as far as security is concerned by using formal language. At the users' level of the organization, the security policy could be constructed by using nonmathematical based model such as the Access Control Matrix. The Access Control Matrix can be used to show which users have access to certain objects in an organization and which objects a user does not have access to. The process layer of the organization could use a mathematically based model.

Mappings between Layers

Once the different layers of security policy are created, correct mapping between the layers must be ensured. Security vulnerabilities can exist if the mappings aren't correct. The authors believe the best approach to mapping between the layers is to take a conservative approach. If the conservative approach is not taken, it is possible to introduce security vulnerabilities. Abrams and Bailey give the following rules for mapping policies between layers.

The test of the connection between different levels of policy is that:

1. Every access allowed by the higher level policy should be supported by the lower level policy, and
2. No action allowed by the lower level policy should be forbidden by the higher level policy.

The conservative position is:

1. that there may be an action allowed by the higher level policy that is forbidden by the lower level policy, but
2. No action or combination of actions allowed by the lower level policy can violate the higher level policy.

The authors recommend taking the conservative approach. The policy should only be modified if system is unusable by the users.

Policy can also be described in different areas of security. The organization can have policies that deal with integrity, confidentiality, and availability. Creating separate policies for each of these areas can simplify the policy making process. Making one policy that combines various subjects can be difficult. Using these separate policies can improve the overall policy of the organization. The organization can have one policy that deals with integrity and another policy that deals with availability.

After the organization has defined all of its policies for the security at all the separate levels, the enterprise has one remaining step. The organization should then create a meta-policy, which is a policy that describes the other polices of the organization. The meta-policy is similar to meta-data in a database that describes the data in the database. By

creating the meta-policy the organization has one unified document that describes all of the policies of the enterprise.

Discussion

The authors make some good points in their paper. The use of layers to create a security policy would create a more complete policy for an enterprise. Because many different people would be involved in the creation of the policy, it is less likely that things would be overlooked when the policy is created. This would, in effect, not leave the system vulnerable to attacks. It would take more time and resources to develop the policy, but in the long run the extra effort would be worth the sacrifice. An approach that involves many users of the system along with the security specialist of the enterprise could be a great benefit.

The one point that is very interesting in the paper is the meta-policy that the authors describe. The authors don't go into great detail about how to create the meta-policy. In the paper, the meta-policy seems similar to meta-data that is stored the data dictionary in a database. The meta-data in this situation describes the data that is used in the database. The topic of using a meta-policy to control of the policies of an enterprise is a great research issue. From this paper more research could be put into ways of developing a meta-policy for a multi-layered security policy.

The one weakness of the paper is that the authors don't give much detail in creating a policy for the process level. They discuss the other two layers in depth but don't explain much about creating the actual process level policy. The authors basically state that less policy models have been established for the process level, but there are process level models available. Security policies at the process level are another area where further research can be conducted to improve on the model that Abrahams and Bailey suggest in the paper. Overall, creating a layered policy for an organization would be much better than one security specialist creating the entire policy on his/her own.

Summary

Using a layered approach in creating a security model has many benefits. At each layer of the organization, the security model can be expressed in different terms. By using layers, the users of the system can express the model in a way in which they are comfortable. By using abstraction, the users can express their ideas of security and not concern themselves with the details at lower levels. There can be more then one way to express the same idea at different levels. The layered model supports these different views of the organization's assets. As the policy is modeled at each successful lower level, more detail is given to the policy. Using layers can help prevent an organization from overlooking parts of their security concerns.

References

Abrams, M. and Bailey, D. Abstraction and Refinement of Layered Security Policy. Information Security -- and Integrated Collection of Essays (Abrams, Jajodia and Podell, eds.), IEEE Computer Society Press, Los Alamitos, CA, 1995. http://www.acsac.org/secshelf/book001/book001.html.