A Trace-Based Model of the Chinese Wall Security Policy

Ann E. Kelley Sobel
Systems Analysis Department
Miami University
Oxford, OH 45056

Jim Alves-Foss
Center for Secure and Dependable Software
University of Idaho
Moscow, ID 83844-1010

Abstract

The Chinese Wall security policy is a well known information control policy used in the commercial world to specify control over information when conflicts of interest arise. A trace-based information flow control model for the Chinese Wall security policy is presented. This model is enhanced to permit conflict time frames for obtained information and revocation of access rights to a particular data object. The trace-based model is shown to be less restrictive, more robust, and more precise than existing Chinese Wall security policy models.

1 Introduction

The Chinese Wall security policy is a well known information control policy used in the commercial world that is used to specify control over information when conflicts of interest arise. Information in the computer system is grouped into single objects of the system, as defined by Brewer and Nash [BN89]. These objects are then grouped into datasets, where every object belongs to a single dataset, representing all the information about a single company. The datasets are then classified into conflict of interest classes. For example, a conflict-of-interest set may contain all of the datasets of banks that employ a certain advertising firm. Employees of the firm can work with many clients, but cannot work with more than one bank without exhibiting a conflict-of-interest.

A user is in violation of the Chinese Wall security policy if she holds information that conflicts with any other information that she already holds. Under this policy, a user who knows nothing is permitted access to any dataset. Once she has obtained information related to a particular dataset, A, a *Chinese Wall* is built around all datasets that conflict with A. She can still access other information in A and in any other dataset B which is not in conflict with A. After accessing dataset B, the Chinese Wall will be modified to include all datasets in conflict with B.

This policy differs greatly from traditional computer security policies that are based on the military classification model. In the military security model a user is given a clearance that defines a level of information up to which the user has complete access (more restrictive models are possible, but will not be discussed further in this paper). All information in the military system is then classified at a particular security level. It is this level that is compared with the user's clearance to determine if access is permitted. No past access history is evaluated and thus the information that the user can potentially access will not change.

Formalization of security policies for computer systems has typically been defined in terms of either access-control models or information flow models. In the access control approach, the system is defined in terms of *subjects* that represent the active entities of the system that can access and modify information, and *objects* that are information repositories. The access control policy defines which modes of access are available for objects (read, write, etc.) and under which conditions a subject can access an object. In the information flow approach, the system is defined in terms of subjects and *events* where each event is associated with activity at a particular security level. The information flow control policies define which events of the system are permitted to ensure that information does not flow in an unauthorized manner.

Information flow models provide a more general definition of security by permitting the specification of information flow that does not occur through normal access modes and more importantly, that occurs "accidentally" through authorized access modes. Brewer and Nash [BN89] have presented an access-control based approach to specifying the Chinese Wall security policy. Sandhu [San92, San93] has presented a Lattice-Based access control model for the policy. Although there are several general trace-based information flow policies for military style security models including non-interference [GM82], restrictiveness [McC87] and separability [McL94], there have been no published trace-based information flow models related to the Chinese Wall security policy. Brewer and Nash do address one aspect of the information flow problem through the introduction of a sanitization process, which will be discussed later. Sandhu addresses the same problem through adding new object labels that represent the possible information content of an object and using the standard lattice-based access control mechanisms of military-style computer security systems. Since most information flow control policies use a lattice-based approach for specifying security labels, it may seem natural to adopt Sandhu's lattice to an information-flow control model. However, as we will discuss later, there are limitations to Sandhu's approach.

The development of a more general trace-based information flow control model for the Chinese Wall security policy is the focus of this paper.

2 Information Flow and Traces

Information flow control policies are specified in relationship to a system or a system component. The events of the system are specified as the object access operations, and the security policy is defined in terms of the sequence of events over time, called a *trace.* In this case, the object access operations of the system correspond to the reading and creating of data objects. For the purposes of this paper, we assume that all data objects are immutable, hence the use of the object creation operation and not a write operation.

2.1 Trace-based approach

Traditionally a single event of the system may appear as $< O_j, ?, I_m >$ which indicates that a read access, ?, to object O_j was granted to individual, I_m. To simplify the notation we will define for each data object O_j in the system, an associated trace sequence which records the creator of the data object and the individuals who have been given access to read the data object. In effect this is the sub-trace of the system trace consisting only of those events that contain O_j as the first element. Events in this sub-trace will consist only of the later two fields of the system trace.

The occurrence of the creation and read events are recorded by concatenating an element representing the particular event to the trace sequence. All possible trace sequences for the data object define the *behavioral* model of the data object [Sob98]. Enforcement of the Chinese Wall security model consists of allowing those system events which correspond to the allowable trace sequences for each O_j.

2.2 Trace Notation

The variable h_j is used to represent the event trace for the data object O_j. Trace operations are defined as follows.

Notation	*Definition*
$h[k]$	kth element of h
ε	empty sequence
$\#h$	the length of h
h^-	h with its last element removed
$h[k][0]$	first component of the kth element of h — the access operation
$h[k][1]$	second component of the kth element of h — the individual granted the access

Chinese Wall Model Notation

Notation	*Definition*
$\mathcal{I}$	The set of individuals (user's, principals, subjects, etc.)
$\mathcal{I}_j$	The j^{th} individual $\mathcal{I}_j \in \mathcal{I}$
O	The set of data objects
O_j	The j^{th} object $O_j \in O$
$\mathcal{C}$	The set of conflict of interest sets $\mathcal{C} \subset \mathbf{P}(O)$
$\mathcal{C}_j$	The j^{th} conflict of interest set $\mathcal{C}_j \subseteq O$
$Indvl(h_j)$	The set of individuals who have accessed the data object O_j, as specified by the trace h_j. (A formal definition is given in Section 2.3.)
$Inds(\mathcal{C}_j)$	The set of individuals who have accessed the data objects in the set $\mathcal{C}_j$. (A formal definition is given in Section 2.3.)

Using this notation, we can specify the no conflict-of-interest constraint with respect to objects accessed by individuals using Eq. 1. This equation states that an individual that has accessed O_j can not have accessed any of the objects that are in conflict with O_j. This equation expresses the same concept as Brewer and Nash Axiom 2, and the simple security policy of lattice-based access control mechanisms used by Sandhu.

$$\forall i \in \mathcal{I}.\ i \in Indvl(h_j) \Rightarrow\ i \notin Inds(\mathcal{C}_j) \tag{1}$$

It is also important to note that conflict-of-interest sets must be symmetric, in other words if O_j is in the conflict-of-interest set of O_k, then O_k must be in the conflict-of-interest set of O_j as expressed in Eq. 2. We can specify this property given the nature of our conflict of interest set notation, however, there is no equivalent notion in other Chinese Wall models.

$$\forall j,k.\ \mathcal{C}_j, \mathcal{C}_k \in \mathcal{C}.\ O_j \in \mathcal{C}_k \Leftrightarrow O_k \in \mathcal{C}_j \tag{2}$$

2.3 The Formal Model

Formally, the initial $\mathcal{C}$ set can be constructed by taking the cross product of each conflicting company's set of objects. Specifically, each conflict of interest set $\mathcal{C}_j$ is a subset of $\mathcal{C}$ which specifies a single object, O_j and all of the objects which conflict with O_j (all pairs (O_j, O_k) where companies

j and k conflict). To simplify notation, we will use C_j to directly denote the set of objects O_k that conflict with object O_j.

We will associate with each data object O_j an event trace h_j. Each event trace is initialized with an element that represents the individual who created the data object, $\langle\ !,\ \mathcal{I}_k\ \rangle$, given that the owner of O_j is $\mathcal{I}_k$. Elements are added to the trace each time that an individual is granted access to this data object. These elements are of the form, $\langle\ ?,\ \mathcal{I}_m\ \rangle$, where individual $\mathcal{I}_m$ has been granted access to the data object O_j. Read access to data object O_j is granted to individuals who have not accessed any object in the conflict-of-interest set for O_j, $\mathcal{C}_j$. This check is defined in Eq. 3 for individual ($\mathcal{I}_m$) that wishes to access object (O_j):

$$\mathcal{I}_m \notin Inds(\mathcal{C}_j) \tag{3}$$

An invariant approach to specifying that this check always holds is given in Eq. 4

$$\forall i \in \mathcal{I}.\ \forall k.\ 0 \leq k < \#h_j.\ i = h_j[k][1] \Rightarrow i \notin Inds(\mathcal{C}_j) \tag{4}$$

Where:

$$Indvl(h_j) \quad = \quad \begin{cases} \Phi & \texttt{if } h_j \ = \ \varepsilon \\ Indvl(h_j^-) \ \cup \ \{h_j[\#h_j \Leftrightarrow 1][1]\} & \texttt{otherwise} \end{cases} \tag{5}$$

$$Inds(\mathcal{C}_j) \quad = \quad \forall k.\ O_k \ \in \ \mathcal{C}_j.\ \bigcup_k Indvl(h_k) \tag{6}$$

The creation of data objects requires a modification of the set of conflict-of-interest sets, $\mathcal{C}$. This is accomplished by adding a new conflict-of-interest set $\mathcal{C}_n$ to COI. Any data object that is in conflict with a data object previously accessed by the creator ($\mathcal{I}_m$) must also be considered in conflict with the newly created data object (O_n), and thus be in $\mathcal{C}_n$. $\mathcal{C}_n$ is defined as in Eq. 7.

$$\forall k.\ \mathcal{C}_k \in \mathcal{C}.\ \mathcal{I}_m \in Inds(\mathcal{C}_k) \Rightarrow [\forall j.\ O_j \in \mathcal{C}_k \wedge \mathcal{I}_m \in Indvl(h_j) \Rightarrow C_j \subseteq C_n] \tag{7}$$

Or, more simply put, if $\mathcal{I}_m$ has accessed something in the conflict-of-interest set for an object O_k, then O_k should now be in it's conflict-of-interest set as specified in Eq. 8.

$$\forall k.\ \mathcal{I}_m \in Inds(\mathcal{C}_k) \Rightarrow O_k \in C_n \tag{8}$$

To create the reciprocal conflict-of-interest, as required by Eq. 2, we must ensure that Eq. 9 holds.

$$\forall k.\ O_k \in \mathcal{C}_n.\ O_n \in \mathcal{C}_k. \tag{9}$$

3 Enhanced Chinese Wall

The Chinese Wall security policy, as stated in the introduction and in previous work by Sandhu [San92, San93] and Brewer and Nash [BN89], is a very restrictive security policy. For example, if a user has information about Bank-A, then at no time in the future will the user be permitted to access information about Bank-B. In reality, conflict-of-interest information may actually only conflict over a certain time period, after which point there is no conflict. We need to augment the Chinese Wall security policy defined in the previous section to permit the specification of a conflict time-frame for information obtained.

Time Frames. To permit the specification of a conflict time-frame for information obtained, we only need to modify $\mathcal{C}$. If two data objects, say O_j and O_n no longer conflict, then we must remove the object reference from the other object's conflict-of-interest set:

$$\mathcal{C}_j \Leftrightarrow \{O_n\} \wedge \mathcal{C}_n \Leftrightarrow \{O_j\} \tag{10}$$

Data Relinquishing. In addition, as a user's job responsibilities change and time progresses, it may be desirable for the user to relinquish all information related to one data object and thus tear down part of the Chinese Wall. Such an action must be approved by an external agent who authorizes the reassignment/relinquishment. A modification of the Chinese Wall security policy that allows revocation of access rights to a particular data object may prove useful in this case.

To permit an individual to relinquish a particular data object, say O_j, then we must represent this action by adding another element to the sequence h_j. Let the element, $\langle !, \mathcal{I}_m \rangle$, represent that individual $\mathcal{I}_m$ relinquishes data object O_j. The addition of this new element changes the property of reading a data object and the definition of $Indvl(O_j, h_j)$. We must strengthen Eq. 3 so that an individual may access data object O_j if she does not currently hold any information in conflict with O_j (this permits her to have previously read information in conflict, but then to have released that information). The modification of this equation is is completely limited to changing the definition of $Indvl(h_j)$ which specifies the individuals that have current access to object O_j. The definition of this function is presented in Eq. 11.

$$Indvl(h_j) = \begin{cases} \Phi & \texttt{if } h_j \;=\; \varepsilon \\ Indvl(h_j^-) \;\cup\; \{h_j[\#h_j \Leftrightarrow 1][1]\} & \texttt{if } h_j[\#h_j \Leftrightarrow 1][0] = \,?\, \vee\; \#h_j = 1 \\ Indvl(h_j^-) \;\Leftrightarrow\; \{h_j[\#h_j \Leftrightarrow 1][1]\} & \texttt{if } h_j[\#h_j \Leftrightarrow 1][0] = \,!\, \wedge\; \#h_j \neq 1 \end{cases} \tag{11}$$

Sanitization. As mentioned in the Brewer and Nash paper, data sanitization is often useful, where a new object is created containing information that is not in a sensitive format and thus does not present any conflict-of-interest. As long as an external agent authorizes data santization, it must be possible to create a new sanitized object, O_n, with the property specified in Eq. 12.

$$\forall k.\ O_k \notin C_n \wedge O_n \notin C_k \tag{12}$$

4 Comparison to Other Chinese Wall Models

It is important to compare the information flow control model of the Chinese Wall security policy with the access-control based models presented by Brewer and Nash [BN89]and by Sandhu [San92, San93]. Regardless of which model we examine, there are some advantages to our approach:

- We don't limit a dataset/company to a single conflict of interest set. This is important for real-world modeling where a corporation may have interests in many areas, for example banking and telecommunications.

- We permit the specification of conflict-of-interest at the object level. There is a wide-range of information about a company, and not all of it will conflict with all of the data in another company. Allowing a finer level of granularity permits a wider range of behavior for the user. However, if desired, the specification of the conflict of interest sets could force all objects of a company to have the same conflict-of-interest sets, duplicating the original more restrictive policy.

- We provide a more precise notion of security by specifying behavior in terms of the actions on objects in the system. All security information in our system is tied directly to the objects themselves and not to system-dependent security labels or access matrix. The set-based approach allows more flexibility in the model that can be solidified to a more implementation-based approach at a later time.
- We provide some useful extensions to the Chinese Wall Model to more accurately reflect the real-world situation. These include time frames for conflict-of-interest, data relinquishing and data sanitization.

4.1 Sandhu's Lattice Based Approach

If we follow the approach of Sandhu [San92, San93], we define the conflict-of-interest sets of the system as each consisting of a set of datasets from conflicting companies. These sets are statically defined and each company is in precisely one such set. We define the security label of an event of the system in terms of it's information content relative to these sets. Specifically, the label will be an n-tuple for a system with n distinct conflict-of-interest sets. Each field of the n-tuple will represent either the name of the dataset from which this information could be derived (e.g., Bank-A), or $\perp$ to represent no information from any member of the conflict-of-interest set. As we have discussed, our model does not limit conflict-of-interest sets in this way, but rather specifies all conflicting objects of each object.

Sandhu defines active entities in the system in terms of user's, principals and subjects. A user is the human being accessing the system and has associated rights and privileges. As the user logs into the system they activate a principal, a user may be associated with several principals but a principal is associated with only one user. The principal is a login session with the system and is bound to a particular security level (this is consistent with most military-style security models.) The principal activates one or more subjects (computer processes) at the same security level as the principal. The subjects are restricted to reading from objects at or below their current security level (i.e., objects from companies for which they are already authorized) and to writing up to new objects. These objects may consist of consolidated information which can be represented by adding a dataset identifier to a previously undefined $\perp$ field for the security label. For the user to read from such a new data set, they must activate a new, higher security principal(s) and subject(s).

In our approach, we do not distinguish between active entities in this manner, but rather define a generic individual that can represent any of these entities. We place no restrictions on the execution behavior of the individual other than those specified by the Chinese Wall security policy. We leave such restrictions to implementation-dependent refinements of the model.

4.2 Brewer and Nash

The Brewer and Nash approach [BN89] to the Chinese Wall Policy defines conflict-of-interest sets in a manner similar to Sandhu. Axiom 1 of their formal model specifically restricts a dataset from being a member of more than one conflict-of-interest set. Our model is not that restrictive, allowing a better mapping to the real-world.

Sandhu, in [San92], points out that the Brewer and Nash model has the following implication:

> *A subject which has read objects from two or more company datasets cannot write at all.*

This is clearly either a mistake in the Brewer and Nash approach or a very restrictive limitation. Sandhu allows the ability to write to a new composite dataset that is labeled for each company the user has accessed. Our approach creates a new conflict-of-interest set for the new object and includes in this set all objects that are in conflict with objects the user has already accessed. We also add the new object to the conflict-of-interest sets of all conflicting objects in it's new conflict-of-interest set. This is the most general approach to writing we can find and satisfies the intent of the Chinese Wall policy.

5 Conclusion

In this paper we have presented a simple, yet powerful, trace-based model of the Chinese Wall security policy, which is more general that those presented in the literature. This model specifies, at the object-level, all conflict-of-interest restrictions of the Chinese Wall policy. After an individual has read information from an object (or created a new object) they are not permitted to access data from an object that is classified as having a conflict-of-interest. The specification of the conflict-of-interest sets is left to the system policy with the restriction that there is no inconsistency in the specification (Eq. 2).

We have extended the standard Chinese Wall policy to include real-world issues such as a time-frame for conflicts and the redefinition of a user's job duties. Formal specifications of these restrictions, along with informal constraints on their use (i.e., external authorization) are provided.

We plan to continue this work by showing formally the relationship between our model and those of Sandhu and Brewer and Nash. It is our contention that our model is a generalization of these models yet still satisfies the important, general, restrictions of the Chinese Wall Policy. An investigation into efficiency issues related to implementations of this model are also warranted. Although we specify traces with respect to individual objects of the system and conflict-of-interest sets, an implementation can use a different approach that enhances efficiency.

References

[BN89] Dr. D.F.C. Brewer and Dr. M.J. Nash. The chinese wall security policy. In *Proc. IEEE Symposium on Security and Privacy*, pages 206–214, 1989.

[GM82] J.A. Goguen and J. Meseguer. Security policies and security models. In *Proc. IEEE Symposium on Security and Privacy*, pages 11–20, 1982.

[McC87] D. McCullough. Specifications for multi-level security and a hook-up property. In *Proc. IEEE Symposium on Security and Privacy*, pages 161–166, 1987.

[McL94] J. McLean. A general theory of composition for trace sets closed under selective interleaving functions. In *Proc. IEEE Symposium on Research in Security and Privacy*, pages 79–93, 1994.

[San92] R. Sandhu. A lattice interpretation of the chinese wall security policy. In *Proc. 15th NIST-NCSC National Computer Security Conference*, pages 329–339. US Govt. Printing Office, 1992.

[San93] R. Sandhu. Lattice-based access control models. *IEEE Computer*, 26(11):9–19, November 1993.

[Sob98] A.E.K. Sobel. Formal requirements specification analysis of the attitude control subsystem of the international space station alpha. In *Proc. HICSS'31*, pages 348–355, January 1998.