



<b>Title</b>	A New Distributed Chinese Wall Security Policy Model
<b>Authors(s)</b>	Fehis, Saad, Nouali, Omar, Kechadi, Tahar
<b>Publication date</b>	2016
<b>Publication information</b>	Fehis, Saad, Omar Nouali, and Tahar Kechadi. "A New Distributed Chinese Wall Security Policy Model." Association of Digital Forensics, Security and Law, 2016. <a href="https://doi.org/10.15394/jdfsl.2016.1434">https://doi.org/10.15394/jdfsl.2016.1434</a> .
<b>Publisher</b>	Association of Digital Forensics, Security and Law
<b>Item record/more information</b>	<a href="http://hdl.handle.net/10197/9295">http://hdl.handle.net/10197/9295</a>
<b>Publisher's version (DOI)</b>	10.15394/jdfsl.2016.1434

Downloaded 2026-05-01 23:35:27

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd\_oa)



© Some rights reserved. For more information

# A New Security Policy based on Chinese Wall Model

..... · ..... · .....

Received: date / Accepted: date

**Abstract** In this paper, we propose a new security model based on Chinese Wall Security Policy (CWSP). In other words, the new model uses the concepts of CWSP on the access query type that the subjects attempt to make to the objects. We have two types of walls; the first type consists of walls that are built around the subjects, and the second type consists of walls that are built around the objects. The idea is to completely separate competing subjects/objects data by walls. We showed that this mechanism is a good alternative to deal with some previous models limitations. The model is easy to apply in a distributed system environment, such as Cloud-Computing. It is implemented using Object Oriented Programming and also it can be used in Cloud Computing at the SaS (Software as a Service) level. Or it can be used in the object capabilities as an access control in real distributed systems.

**Keywords** Security Policy · Chinese Wall Model · Information flow · Distributed System Security · Cloud Computing security · Protection mechanisms.

## 1 Introduction

The cloud computing technology comes with numerous advantages but also brings with it some disadvantages and challenges. One of the challenges of cloud computing is the security, protection, and trust caused by *Multi-tenancy*. For instance, it is possible to find two competitors using the same cloud infrastructure and the same provider. This may cause issues of how to control the information flow between those competing companies and also between subjects in general. The Chinese

Wall Security Policy (*CWSP*) is a very interesting candidate solution to the above problem. The CWSP has already been used in commercial applications. For instance, the UK's financial sector, which provides consulting services, uses the CWSP model. As consultants must respect the confidentiality agreements, the CWSP is used to prevent such confidentiality to be breached by avoiding the information flow that causes conflict of interest between involved parties.

The CWSP was defined and named by Brewer and Nash (*BN model*). They have developed their first model in 1989 [1]. The model became very attractive and therefore many other models based on the same idea have been proposed in subsequent years [2] [3] [4] [5] [6] [7]. The model and its variants have been successfully used in many applications [8] [9], in order to control the information flow between the competing subjects residing in the same system. More importantly, after an extensive state-of-the-art on the proposed security models based on Chinese wall, we find that the main goal of those models is to control the composite information flow (*CIF*) between competing companies (*a malicious Trojan horse's problem*), caused by the accesses in writing from the subjects on the objects. We can summarise these models by the following two important points:

- The main concept of the Chinese wall policy model is that users are only allowed access to information which is not conflicting with any other information that they already have [1]. So, each subject (a user or a process) has two key sets: *Granted set* and *Denied set*. *Granted set* contains subjects and/or objects that the subject has granted access to their data. *Denied set* contains subjects and/or objects data that the subject is not allowed to access. The pairwise (*Granted, Denied*) is called subject's

wall. This mechanism relies heavily on these two sets' content. Therefore, the definition of these sets is crucial.

- Information can flow between two objects only via a subject, and information can flow between two subjects only via an object [7], (*malicious Trojan horse's problem*). Using this rule, we can define object's wall in a similar way as for subject's wall. Again, one must ensure that no conflicting objects' data is contained in the same wall.

The two main definitions in relation to entities used in the model are *Objects* and *Subjects*. An object is a passive entity, while a subject is an "active entity". Each object has an "access right" Granted / Denied. An object is used to store data, so each object has a "Stored right", which are not the same as (*Access Right*). In this model, the object has two sets; *Allied* and *Conflict*, where the **Allied** set has the companies (*objects*) which have data stored inside the same object, and the **Conflict** set, contains the companies (*objects*) in conflict with the Allied set. The pairwise (Conflict, Allied), can be interpreted a wall around the object, called "*object's wall*".

Subjects are entities which can act on objects. In this model, each subject has a security label that consists of a pair; (*Granted, Denied*) and each object has a security label that consist of a pair (*Conflict, Allied*). The model's golden rule for executing a read or write query is that "we cannot find two competing data inside the same wall (*object's or subject's wall*)". Based on this rule, we can built walls between competing companies, which is the key concept of the Chinese wall policy [1]. This model ensures efficient control of the direct and composite information flow in the system.

To apply this security model in a distributed system, one needs the security labels either object or subject walls be managed by the server that hosts the objects or subjects. Each access query of a subject to an object will be associated with the security label of that subject, and this will be checked by the server that hosts the object. This model can be viewed as a distributed CWSP model (D-CWSPM). And, also it can be implemented using Object Oriented Programming paradigm. This model can be used in Cloud computing "Software as a service SaaS" or it can use the capabilities as an access control in real distributed system.

The remainder of the paper is organised as follows: In Section 2 we will present the main idea of the model and illustrate by an example. In Section 3, we will present the formal model and information flow problems of our model. In Section 4 we will present how the information flows within a system implementing our model. In Section 5, we will discuss D-CWSPM vs Ac-

cess Matrix; a model that uses a matrix to control the access of subjects to objects. In Section 6, we present some related research works in the area of this specific problem. In the section 7, we will show how the model can be implemented using an OOP language. Finally, we conclude and give some future directions in Section 8.

## 2 Proposed Model

In this work, we based our idea on the access query type of the subject to the objects and the philosophy of the Chinese wall security policy CWSP. Its rule is the building of the walls between the competing companies. In our model we have two types walls placement, the first is built around the subject, and the second around the object. We cannot find inside the same wall two data related to two competing objects. So, we start by these analysis:

The subject firstly, is freely to choose to access to any object, at this step it's important to known the nature of this access: reading or writing access, as in following:

### 2.1 Reading Queries

If the access is Read request, this induced to read from the object by the subject, and we can interpret this by the moving of the data from the object side into the subject side (*inside the subject's wall*). So, there is a related data (*information*) of the object inside this wall. Also, the access is denied of this subject to the competing objects with the objects inside subject's wall. So, we induced, that the subject has two securities labels:

- The Subject Wall Granted (*SWG*): The set of all objects, where their related data into the subject side.
- Subject Wall Denied (*SWD*): The set of all objects denied to moving them into the subject side.

The pairwise (*SWG, SWD*), can be interpreted, by the building of the wall around the subject.

### 2.2 Writing Queries

If the access is writing query by the subject  $Sub_i$  to the object  $Obj_j$ , we can interpret this type of query by the writing of related information of the set of all objects inside the  $Sub_i$ 's wall into the object  $Obj_j$ . So, the second wall is built around the object, and this object's wall cannot contains the competing object's data. So, we induced, that the object's wall has two securities labels:

- The object’s wall in Allied (*OWA*):The set of all objects where their related data stored inside the object. So, consider they in allied with the object  $Ob_j$ .
- Object’s wall in Conflict (*OWC*): The set of all objects in conflict of interest with the objects stored inside the object  $Ob_j$ , so they denied to moving them into the the object side.

The pairwise (*OWA*, *OWC*), can be interpreted, by the building of the wall around the object.

### 2.3 Example

If we have two subjects  $Sub_1$  and  $Sub_2$ , and five objects  $Ob_1, Ob_2, Ob_3, Ob_4$  and  $Ob_5$ , where  $Ob_1$  in competition with  $Ob_2, Ob_3$  in competition with  $Ob_4$  but  $Ob_5$  neuter with the others objects (Table 1).

**Table 1** The Initial State of the Object’s Walls

Object	Object’s Wall
$Ob_1$	$OWA = \{Ob_1\}; OWC = \{Ob_2\}$
$Ob_2$	$OWA = \{Ob_2\}; OWC = \{Ob_1\}$
$Ob_3$	$OWA = \{Ob_3\}; OWC = \{Ob_4\}$
$Ob_4$	$OWA = \{Ob_4\}; OWC = \{Ob_3\}$
$Ob_5$	$OWA = \{Ob_5\}; OWC = \{\emptyset\}$

And we have this queries sequence in our system:

Q1: Subject  $Sub_1$  reading access from the object  $Ob_1$ , so the moving of data stored inside (*or related to*) of  $Ob_1$  to the subject side of  $Sub_1$ . So the updating the  $Sub_1$ ’s wall as following:

$$SWG_1 = \{Ob_1\} \text{ and } SWD_1 = \{Ob_2\},$$

Because the object  $Ob_1$  in conflict with the object  $Ob_2$ . And the subject’s wall composed by the two pairwise ( $SWG_1, SWD_1$ )

Q2: Subject  $Sub_1$  reading access from the object  $Ob_2$ , this access is denied, because there is inside of its wall a data related to the object  $Ob_1$ , where it is in competition with the object  $Ob_2$  (*Rules of Chinese wall security policy [1]*).

Q3: Subject  $Sub_2$  reading access from the object  $Ob_2$ , so the moving of the data stored inside (*or related to*) the object  $Ob_2$ . So the updating the  $Sub_2$ ’s wall as following:

$$SWG_2 = \{Ob_2\} \text{ and } SWD_2 = \{Ob_1\},$$

Because the object  $Ob_2$  in conflict with the object  $Ob_1$ . And the subject wall composed by the two pairwise ( $SWG_2, SWD_2$ )

Q4: Subject  $Sub_1$  reading access from the object  $Ob_3$ , we have inside the  $Sub_1$ ’s wall a data related to the object  $Ob_1$  and this object isn’t in conflict of interest with the object  $Ob_3$ , so the access is granted and also the moving of the data from  $Ob_3$  to the subject  $Sub_1$  side, inside of its wall. In the same time the access denied to the object  $ob_4$ , in conflict with  $Ob_3$ . So the updating the  $Sub_1$ ’s wall as following:

$$SWG_1 = \{Ob_1, Ob_3\} \text{ and } SWD_1 = \{Ob_2, Ob_4\}.$$

Q5: Subject  $Sub_1$  writing access to the object  $Ob_5$ , so the moving data from the  $Sub_1$  side into the object  $Ob_5$ . It’s clear, this data related to  $Ob_1$  or  $Ob_3$ . So probably, the object  $Ob_5$  contains a data related to the objects  $Ob_1$  or  $Ob_3$ , and cannot contains a two data related to two competing objects ( $Ob_2$  in conflict with the object  $Ob_1$  and  $Ob_3$  with the object  $Ob_4$ ). So we need to update the  $Ob_5$ ’s wall as follow:

$$OWA_5 = \{Ob_5, Ob_1, Ob_3\}, OWC_5 = \{Ob_2, Ob_4\}.$$

The  $Ob_5$ ’s wall composed by the pairwise ( $OWA_5, SWC_5$ ).

Q6: Subject  $Sub_2$  writing access to the object  $Ob_5$ , so moving data from its inside (*a data related to*  $Ob_2$ ) into the object  $Ob_5$ . However, the object  $Ob_5$  contains a data related to the object  $Ob_1$  who it is in competition with  $Ob_2$  ( $Ob_2 \in OWC_5$ ). And, this is in contradiction with the Chinese wall security policy, so this query is denied and not permitted.

Let now, if we have a third subject  $Sub_3$ , where it need to read a data from the object  $Ob_5$  and then write it to the  $Ob_2$ . So, the problem is that our malicious subject ( $Sub_3$ ) need to create a CIF between competing object  $Ob_1$  and  $Ob_2$ !

Firstly, the subject  $Sub_3$ , read data from the object  $Ob_5$ . The object  $Ob_5$ , contains a data related to two objects  $Ob_1$  and  $Ob_3$  (*from* Q5) and also the object has two sets information the  $OWA_5$  and the  $OWC_5$ . So, we have the two following steps:

1. The first step is the reading: After the reading access, we have inside  $Sub_3$ ’s wall a data related to three objects  $Ob_1, Ob_3$  and  $Ob_5$ . So, the updating of the subject’s wall as following:

$$SWG_3 = SWG_3 \cup OWA_5, \\ SWD_3 = SWD_3 \cup OWC_5$$

So,  $OWC_5$  set contains  $Ob_2$ .

2. The second step is the writing: the writing access to the object  $Ob_2$ , this, is not permitted, because the object is in the Denied set ( $SWD_3$ ) of the subject. So, the access is denied.

So, as a result, our malicious subject cannot create a CIF between competing objects. And in end we can view in table 2 and 3, the final state of the Subjects’ and Objects’ walls.

**Table 2** The End State of the Subject's Walls

Subject	Subject's Walls
$Sub_1$	$SWG_1 = \{Ob_1, Ob_3\}; SWD_1 = \{Ob_2, Ob_4\}$
$Sub_2$	$SWG_2 = \{Ob_2\}; SWD_2 = \{Ob_1\}$
$Sub_3$	$SWG_3 = \{Ob_1, Ob_3, Ob_5\}; SWD_3 = \{Ob_2, Ob_4\}$

**Table 3** The end state of the object's walls

Object	Object's Walls
$Ob_1$	$OWA = \{Ob_1\}; OWC = \{Ob_2\}$
$Ob_2$	$OWA = \{Ob_2\}; OWC = \{Ob_1\}$
$Ob_3$	$OWA = \{Ob_3\}; OWC = \{Ob_4\}$
$Ob_4$	$OWA = \{Ob_4\}; OWC = \{Ob_3\}$
$Ob_5$	$OWA = \{Ob_1, Ob_3, Ob_5\}; OWC = \{Ob_2, Ob_4\}$

### 2.4 Queries' running conditions

Let the subject  $Sub_i$  has two sets' object: the granted set  $SWG_i$  and the denied set  $SWD_i$ , and the object  $Ob_j$  has two object's set:  $OWA_j$  and  $OWC_j$

After, the prior interpretation of the query access type (*Reading / Writing*), we can induce the necessary condition to running the query of the access of the subject  $Sub_i$  to the object  $Ob_j$  is:

"we cannot find inside the same wall, data related to two competing objects"

Formally as in the following:

$$SWG_i \cap OWC_j = \emptyset \text{ and } SWD_i \cap OWA_j = \emptyset$$

## 3 Distributed CWSP Model

After, the prior illustration, we can now present the formal model.

Let:

- $OB = \{obj_1, \dots, obj_n\}$ , denote the set of all objects,
- $SU = \{s_1, \dots, s_m\}$ , denote the set of all subjects,
- $Comp(obj_i)$  or simply  $Comp_i$  be the company dataset of object  $obj_i$ .

### 3.1 Dataset Organisation

In our model we keep the dataset organisation proposed by Lin [2]-[6], where:

- **Lowest Level**: we consider individual items of information, each concerning a single corporation. We will refer to the files in which such information is stored as objects [1].

- **Intermediate level**: we group all objects which concern the same corporation together into what we call a company dataset [1].
- **Highest level**: we associate with each company dataset, say X, a Frechet neighborhood, denoted by  $CIN(X)$  Conflict of Interest Neighborhood of X, where  $CIN(X)$  is the set of all company datasets that are in conflict of interest with X.

### 3.2 Conflict of interest relation CIR

Let  $CIR \subseteq OB \times OB$  as a binary relation, satisfies the following properties.

- CIR-1: CIR is symmetric.
- CIR-2: CIR is anti-reflexive.

It should be clear CIR-2 is necessary, a company cannot conflict to itself. If company A is in conflicts with B, B is certainly in conflicts with A, so CIR-1 is valid.

### 3.3 Model

Our model is 3 tuples  $(SU, OB, Query)$  where:

#### 3.3.1 OB

Denote the set of all objects, where each object  $obj_i$  has or associated with two subsets of the set  $OB$ :

- $OWA(obj_i) \subseteq OB$ , Or simply  $OWA_i$ , the set of all objects, where they have a related data stored inside the object  $obj_i$ . If there is an object  $obj_j \in OWA_i$ , so the object  $obj_i$  contains (*or stored inside itself*) a data related to the object  $obj_j$ .
- $OWC(obj_i) \subseteq OB$ , Or simply  $OWC_i$ , the set of all objects denied to be stored inside the object  $obj_i$ . So, if there is an object  $obj_j \in OWC_i$ , that, the object  $obj_i$  cannot will contain any related information of the object  $obj_j$ . Otherwise, the object  $obj_i$  has a data related to another object in the conflict of interest with the object  $obj_j$ .

And they are initially as following:

- $OWA(obj_i) = obj_i$ , initialized by its self,
- $OWC(obj_i) = \{obj_j \in OB | (obj_i, obj_j) \in CIR\}$

The pairwise  $(OWA_i, OWC_i)$  can it interpret by the building of the wall around the object  $obj_i$ . So, we cannot find inside the same wall, two data related to two distinct competing objects.

### 3.3.2 SU

Denote the set of all subjects, where each subject  $S_i$  has or associated with two subsets of the object  $OB$ :

- $SWG(S_i) \subseteq OB$ , Or simply  $SWG_i$ , the set of the objects have a related data inside the subject wall of  $S_i$  (*read by the subject*). So, if there is an object  $obj_j \in SWG_i$ , so the subject  $S_i$  contains a related data of the object  $obj_j$ .
- $SWD(S_i) \subseteq OB$ , Or simply  $SWD_i$ , the set of the objects denied to will be read by the subject  $S_i$ . So, if there is an object  $obj_j \in SWD_i$ , that the subject  $S_i$  cannot will contain (*or read*) any related data of the object  $obj_j$ .

And they are initially as following:

- $SWG(S_i) = \emptyset$ ; initialized by an empty set, because the subject isn't yet read any object;
- $SWD(S_i) = \emptyset$  initialized by an empty set, because the subject is free to choose any object

The pairwise ( $SWG_i, SWD_i$ ) can it interpret by the building of the wall around the subject  $S_i$ , so, we cannot find inside the same wall two data related to two distinct competing objects.

### 3.3.3 Query( $S_i, obj_j, mode$ )

Any query made by a subject  $S_i$  to access to the object  $obj_j$  with the *mode* equal to:

- *read*: to read from the object
- *write*: to write into the object

The access is authorized, if and only if, this condition is verified:

$$SWG_i \cap OWC_j = \emptyset \text{ And } SWD_i \cap OWA_j = \emptyset$$

And in the same time:

If the mode is Reading Query (*Writing into the subject side, inside the wall that round the subject*):

- $SWG_i = SWG_i \cup OWA_j$
- $SWD_i = SWD_i \cup OWC_j$

If the mode is Writing Query (*In Object side, inside the wall that round the object*):

- $OWA_j = SWG_i \cup OWA_j$
- $OWC_j = SWD_i \cup OWC_j$

Otherwise, the access is denied.

## 4 Information flow's and D-CWSPM

We give the following definition:

**DIF** : A direct information flow between two companies  $A$  and  $B$ : denoted by  $A \rightarrow B$ , is a sequence of read query data from a company  $A$  by any subject, then writing query of this data into the other company  $B$  by the same subject.

**CIF** : A composite information flow from  $A$  to  $B$ , is a sequence of DIFs (*direct information flow*) maded by many sybjects, which starts from  $A$  and end at  $B$ :

$$A = A_0 \rightarrow A_1 \rightarrow \dots \rightarrow A_n = B$$

**D-CWSPM's Theorem**: The Distributed CWSPM assures that no DIF and no CIF between competing companies.

**Proof**: Let we have the two following propositions:

1.  $(A, B) \in CIR \Rightarrow A \in OWC_B (B \in OWC_A)$  and  $A \notin OWA_B (B \notin OWA_A)$  by symmetry.
2. There exists a CIF from  $A$  to  $B$ , that is, a composite direct information flow of size  $n$ :

$$A = A_0 \rightarrow A_1 \rightarrow \dots \rightarrow A_n = B$$

We will use proof by Recurrence; on the number of DIFs between companies.

Let  $n$  the number of DIFs between these two companies.

With  $n = 1$ :

First, the initial assertion: Since  $A = A_0 \rightarrow A_1 = B$  is a DIF, or read data from object  $A_0$  by any Subject  $S_i$ , then write it by the same subject to  $A_1 = B$ :

- Reading by the subject  $S_i$  from the object  $A_0$  is granted, if and only if:

$$SWG_i \cap OWC_{A_0} = \emptyset \text{ And } SWD_i \cap OWA_{A_0} = \emptyset$$

And the result is:

$$SWG_i = SWG_i \cup OWA_{A_0}$$

and

$$SWD_i = SWD_i \cup OWC_{A_0}$$

- Writing by the subject to object  $A_1$  is granted, if and only if:

$$SWG_i \cap OWC_{A_1} = \emptyset \text{ And } SWD_i \cap OWA_{A_1} = \emptyset.$$

This condition, assures that  $A_0$  isn't in the conflict of interest with  $A_1$ . However, in our case we have,  $A = A_0$  in conflict with  $A_1 = B$  ( $A$  in conflict with  $B$ ), so no DIF between  $A$  and  $B$  if they are in competing, so the query is denied QED.

So, if  $n$  equal to 1, there isn't a DIF between two competing companies.

Otherwise, in the case of  $A_0$  isn't in conflict of interest with  $A_1$ :

$$\begin{aligned} OWA_1 &= OWA_1 \cup SWG_i \\ OWC_1 &= OWC_1 \cup SWD_i. \end{aligned}$$

So the result of the DIF from  $A_0$  to  $A_1$  is:

$$\begin{aligned} OWA_0 &\subseteq OWA_1 \\ OWC_0 &\subseteq OWC_1. \\ \text{So, } A_0 &\in OWA_1. \end{aligned}$$

Let now our theorem is true with  $n - 1$  DIFs and we need to verify, if is it true for  $n$  DIFs. So we have:

$$A = A_0 \rightarrow A_1 \rightarrow \dots \rightarrow A_{n-1}$$

And we need to extend it to  $A_n = B$ , where  $A \in OWC_B$ . From the sequence of size  $n - 1$  of the DIFs, we have:

$$A = A_0 \in OWA_0 \subseteq OWA_1 \subseteq OWA_2 \subseteq \dots \subseteq OWA_{n-1}.$$

And it is the same with the set of  $OWC$ .

Let, there is a subject  $S_j$  need to create a DIF between  $A_{n-1}$  and  $A_n$ . So, the subject  $S_j$  needs to read from  $A_{n-1}$  then writing to  $A_n$ , so we have two following steps:

- The first step is reading from  $A_{n-1}$  by the subject and the consequence is:

$$\begin{aligned} SWG_j &= SWG_j \cup OWA_{n-1} \\ \text{And} \\ SWD_j &= SWD_j \cup OWC_{n-1}. \end{aligned}$$

So  $A \in SWG_j$  and  $B \in OWC_A \subseteq SWD_j$ .

- The second step is writing to  $A_n$ , with the condition, that  $SWG_j \cap OWC_n = \emptyset$ . However, in our case, we have  $A \in SWG_j$  and  $A \in OWC_n$ , so the intersection is different from the empty set ( $\emptyset$ ).

So, the creating query is denied because  $A_0 \in OWC_n$ . And the result is that no CIF between those competing companies  $A$  and  $B$ . QED

We conclude, that our distributed D-CWSPM assures that no CIF between any two companies if they are in conflict of interest.

## 5 D-CWSPM vs Access Matrix

In this section, we will present how to implement our distributed model by using the matrix as mechanism, and to compare with the previous proposed models.

### 5.1 Access matrix Model

Firstly, in our model, the object's wall of the object  $obj_i$  is represented by the pairwise  $(OWA_i, OWC_i)$ . And The set of the walls can be represent they, as a binary relation between objects and can be represented by a matrix, where we call it, the object's wall matrix ( $OWM$ ).

The  $OWM$ , be a matrix with element  $OWM(i, j)$  corresponding to the members of  $OB \times OB$ , where the value of  $OWM(i, j)$  is:

- 1: The object  $obj_i$  contains (or stored inside itself) a related information about the object  $obj_j$ ;
- 0: The object  $obj_i$  cannot contain any related information of the object  $obj_j$ . Or the object  $obj_i$  has data in the conflict of interest with the object  $obj_j$ .
- 1: There isn't any data related to the object  $obj_j$  or to its competing objects, stored inside the object  $obj_j$ ;

Initially,

- $OWM(i, j) = 1$  if  $i = j$ ,
- $OWM(i, j) = 0$  if  $(obj_i, obj_j) \in CIR$ ,
- otherwise  $-1$ .

And we can define also two subsets of  $OB$ :

- $OWA(obj_i) = \{obj_j \in OB | OWM(i, j) = 1\}$  the set of all objects, have their data stored inside of the object  $obj_i$ ,
- $OWC(obj_i) = \{obj_j \in OB | OWM(i, j) = 0\}$  the set of all objects denied to be stored inside of the object  $O_i$ ,

The second wall in our model is the subject's wall. So for any subject  $S_i$ , its wall can be represented by, the pairwise  $(SWG_i, SWD_i)$ . So the set of the walls can be represented by a binary relation between subject and object, and can be represented by a matrix, where we call it, the Subject's wall matrix ( $SWM$ ).

The  $SWM$ : is an access matrix with elements  $SWM(i, j)$  corresponding to the members of the  $SU \times OB$ , where the value of  $SWM(i, j)$  is:

- 1: The subject  $S_i$  contains related information of the object  $obj_j$ .
- 0: The subject  $S_i$  cannot contain any related information of the object  $obj_j$ .
- 1: There isn't any information related to the object  $obj_j$  or their competing objects, inside of the subject  $S_i$ ;

Initially,  $SWM(i, j) = -1$  for all  $(i, j)$ .

And From this matrix, we can also define two subsets of:

- $SWG(S_i) = \{O_j \in OB | SWM(i, j) = 1\}$  the set of the objects, have a data inside of the subject wall of  $S_i$ ,

- $SWD(S_i) = \{O_j \in OB | SWM(i, j) = 0\}$  the set of the objects denied to the subject  $S_i$ ,

In the end, and after the representation of the objects' wall and the subjects' wall by using the matrix ( $OWM$ ,  $SWM$ ) as mechanism, we will show, how updating them, for every access query.

Let the  $Query(S_i, obj_j, mode)$ , any query made by a subject  $S_i$  to access to the object  $obj_j$  with the  $mode$  equal to *read* or *write*. The access is authorized, if and only if; this condition is verified:

$$SWG_i \cap OWC_j = \emptyset \text{ And } SWD_i \cap OWA_j = \emptyset$$

And in the same time:

If the mode is Reading Query (*In Subject side*): So the updating of the subject's wall as following:

- $SWM(i, h) = 1$  where  $obj_h \in OWA_j$ , (as  $SWG_i = SWG_i \cup OWA_j$ )
- $SWM(i, h) = 0$  where  $obj_h \in OWC_j$ , (as  $SWD_i = SWD_i \cup OWC_j$ )

If the mode is Writing Query (*In Object side*): So the updating of the object's wall as following:

- $OWM(j, h) = 1$  where  $obj_h \in SWG_i$ , (as  $OWA_j = SWG_i \cup OWA_j$ )
- $OWM(j, h) = 0$  where  $obj_h \in SWD_i$ , (as  $OWC_j = SWD_i \cup OWC_j$ )

Otherwise, the access is denied, and in the same times:

- $SWM[i, j] = 0$

## 5.2 Information flows, Objects, Companies and their Allies

In our model, we focused our work, on the relations between objects and the information flows between them. However, what about of the flow between the objects in the same company?

The answer is the information flows is freely between them. So, we build the object wall around the allied objects (*allied companies*).

However, in our proposed model the update of the matrix  $OWM$  is focused on the object, and not on the companies. So, to fix this problem, the solution is the mapping of  $OWM$  from of  $OB \times OB$  to  $Comp \times Comp$ . By this mapping, we will assure that, we can't find the data of two competing companies stored inside the same company (*in different objects of the same company*).

## 6 Related Work

In this section we will present a set of related works, where in the first section we present the previous proposed models based an access matrix, and theirs problems. Then, in the next section we will present the related application works of the CWSP in the environment as distributed system (*Cloud-Computing, workflow*).

### 6.1 Related works based on access matrix

The CWSPM was identified and so named by Brewer and Nash (*BN's model*), where they developed a mathematical model for this policy [1]. Their idea is the grouped of the dataset of companies in conflict of interest classes (*COI*), so a set of partitions and applying to subjects a mandatory ruling, where all subjects are allowed access to at most one dataset belonging to each such conflict of interest class (*security rule*). Where, the access is only granted if the object requested:

- Is in the same Company Dataset as an object already accessed by that subject, or,
- Belongs to an entirely different Conflict of Interest Class.

Access, means read or write. So, this to answer that no direct information flow (*DIF*) between competing companies. And also they prevents the CIF by the application of the **start-property** rule to write access, where is only permitted if, the:

- Access is permitted by the **simple security** rule, and,
- No object can be read which is in a different Company Dataset to the one for which write access is requested and contains un-sanitized information.

The proposal was a great idea. Unfortunately, BN's model was based on incorrect assumption that corporate data can be partitioned (*decomposed*) into mutually disjoint conflict of interest classes (*COI-classes*), such a disjoint collection is called a partition in mathematics. COI-classes seldom disjoint, they do overlap, and hence BN theory collapses. Also, the authors did not distinguish between human users and subjects that are processes running on behalf of users. The BN's write rule (*\*-property*) is successful in preventing such information leakage by Trojan Horse. However, it does so at an unacceptable cost.

It is easy to see that the BN write rule has the following implications [10]: A subject which has read objects from two or more company datasets cannot write

at all. And, a subject which has read objects from exactly one company dataset can write to that dataset. These implications are clearly unacceptable (*if the computer system is to be used for something more than a read-only repository of confidential information*)[10]. Under this regime a consultant can work effectively so long as he or she is assigned to exactly one company (*however, even then the consultants is forbidden to write public information*). When the consultant is assigned to a second company, he or she will be unable to write any information into the system. Consequently, the model proposed is very restrictive as it allows a consultant to work for one company.

Sandhu [10] improves upon this model by making a clear distinction between users, principals, and subjects, defines a lattice-based security structure, and shows how the CWP complies with the Bell-Lapadula model [11].

In the same year, Lin announces a modified model, called an aggressive Chinese Wall Security Policy Model (ACWSPM)[2] to fix the errors of BN. The error is that the conflict of interest (COI) is a binary relation (CIR), and not, an equivalence class (partitions). The CIR is non-reflexive, symmetric and Anti-transitive. The Lin's idea is the construction of a partition, so an equivalence relation. Lin, extend the CIR relation to an equivalence relation (partition). However, then the known properties of a binary relation do not support the elegance and crispness of an equivalence relation, the enthusiasm was lost.

Based on the work of Pawlak [12] [13], Lin in [4], show that the lack of crispness of the ACWSP, since CIR cannot produce a partition. However, the partition, can be re-captured by the induced equivalence relation of a binary relation. The idea is "each binary relation induced equivalence relation" [4]-[6]. The induced equivalence relation named by IAR "In allied with" relation used by Lin is the complement of CIR (*Theorem: CIR is a symmetric and anti-reflexive and anti-transitive binary relation. Its complement IAR is an equivalence relation*).

In the end Lin applied a rules to subjects based on CIR relation and an allied partition, all this to answer that no information flow can will occur between the competing companies.

From these previous models, we are observing, that both models BN and Lin, based on the partitioning of the companies in set of class (partitions). In BN's model the competing companies in the same partition, however, in Lin's model the allied companies in the same partition. The partitions in BN's model can overlap (*example A in conflict with B, B with C and C with D*). In Lin's model the partition based on the complement

of the CIR relation (*induced equivalence relation*), can overlap if the CIR relation is not "ant-transitive", case named by Lin as a "Bad CIR Relation" (*page 10 in [5]*), so a real case excluded by Lin's model!

Also, we are observing, that, Lin in [2]-[6] fix the problems of the DIF / CIF by the using of the same BN's idea [1], un-sanitized/sanitized information and also the read/write access type, where: In BN the un-sanitized information is confined to its self-company but in Lin to allied dataset. So, the inheriting of the problem [10] "consultant to work for one company only" to "consultant to work for one allied companies only". So, no difference between the problem of the DIF and the CIF inside the same company / allied companies (*as a set of objects in the same of a single company*).

There is a recent and interested work proposed by Sharifi and al. [7], where they proposed a Least-Restrictive Enforcement of the CWSPM based on graph representation. Their enforcement mechanism mediates read attempts only to prevent subject-violations, and write attempts only to prevent object-violations. However, there is a strong mathematical confusion between the notion of the class, partition, equivalence binary relation and the transitivity property (*page 3*). Also, their graph representation is very complex for the implementation.

Finally, our new model in this article is easy for the implementation based access matrix and fixing the problems of the previous proposed models.

In the first, our main objective is the application of the CWSP in the Cloud Computing, and the social network and not the proposition of a new model for the CWSP. However, and after analyses of the previous proposed models [1]-[10], and theirs applications [8] [14] [15] [16] [17] [18] [19] [9], we are surprising by many problems. For example the problem of the Conflict of interest (COI) is a set of disjoint class or a binary relation (CIR), the error was fixed in 1989 by Lin [2]-[6], but to our days there are many applications based on COI classes and not a CIR binary relation.

## 6.2 Security Models and Distributed Systems

In the past, there are many attempt to applying the CWSPM in environment as distributed system.

Firstly in the Cloud-Computing (*as a distributed system*), The CWSP used in [16], is to fix inter-VM attack from competitors, which targets at the VMs running on the same physical machine, so each two competing VMs cannot hosted in the same physical machines so that physical isolation. The authors use the conflict of interest and the graph colouring algorithm for the

VM deployment. However, the authors were based on centralized control mechanism.

Also, in [15] use the CWSPM for the "Information Flow Control in Cloud Computing", at the IaaS level. Based on the concept of the conflict of interest is partition (BN's model), however, it is a binary relation [2], and so they based on wrong model, and the same problem with two other work [17]-[18] in Cloud Computing.

In the end, there is an interesting work [9], named by "A Decentralized Treatment of a Highly Distributed Chinese-Wall", however, they based on wrong model, the Brewer and Nash model [1].

## 7 D-CWSPM Implementation

To valid our approach, we are developed a prototype, based on Object Oriented Programming (OOP), and using the C++ as programming language. In this model implementation, we defined, four following classes:

- System Class: This class is the main class of the implementation, where each instance from this class, contains in its private section, the companies' identification, the definition of the CIR Relation and the subjects' identification in the system. The instance, is responsible for the creation/destruction of the companies and the updating of the CIR Relation.
- Company Class: This class is the set of all companies, where each company's instance contains the following proprieties: The list of all its objects identification and the company's wall, (*the pairwise (OWD, OWC)*). The company instance responsible for the creation / destruction of their objects.
- Object Class: This class is the set of all objects in our system, where each instance is related to a single company, and has its "object wall", the pairwise (*OWA, OWC*). The class has also a set of interface for the communication with the other object in the system.
- Subject Class: This class is the set of all subject in our system, where each instance has an single identification, and can access to any object in our system, by using the object interfaces, and CWSP rule.

## 8 Conclusion

The Chinese wall security policy model (*CWSPM*) is very interesting solution, to control the information flow between competing companies in cloud computing (*multi-tenancy*) and the social network in general. The CWSPM's

idea is the building of the wall between the dataset of the competing companies by the applying of mandatory rules to the subjects (*people are only allowed access to information which is not held to conflict with any other information that they already possess [1]*).

So, the CWSPM can be used as mechanism to control the information flow, and in the same time as an access control imposed to the subjects. However, the previous proposed model have a many problems [1]-[6], [7], with application of these model in different way, with the same problems, and based on a matrix as mechanism without distribution.

In this work, we proposed a new model for this policy, where we named it a Distributed Chinese Wall Security policy Model (*D-CWSPM*). Our model is real interpretation of the CWSP, (*we can't find inside the same wall a data related to competing companies*). The D-CWSPM's idea is the building of the wall around the subject [1], as the same to the objects. The model fix the problem of the "a malicious Trojan horses", based on the concept of "the information can flow between two objects only via a subject and information can flow between two subjects only via an object [7]".

Our model is based on a mathematical model, where the Conflict of interest is a binary relation and not a set of partitions (*Class*) [2]-[6]. And, by the interpretation gave of two kinds of the queries (*Reading / Writing*). Our model assures that we can't find any information flow between two competing companies, Direct or Composite information flow. So the fixing of the malicious Trojan horses problem.

Our model is easy to implementing it, in any way, basing on an access matrix between subjects/objects (*can be compared with the previous proposed models*) or in real distributed system (*as Cloud Computing*).

We valid our model, by an implementation prototype, based on the technique of, Object Oriented Programming. Where, the entire security labels are distributed among the system's elements (*object and subject*). Which, any element had its security label (*its Wall*). This, prototype, can will be applying it, in the Cloud-Computing (*Software as a service SaaS*).

In the future works,

1. Initially, our main objective is the application of the CWSPM in the Cloud-Computing and note the proposition of a new model. So, our next step is the application of this model in the Cloud-Computing at the service level "Infrastructure As a Service (*IaaS*)", to control the information between Virtual machines (*multi-tenancy*). Then the application of the model at the service level "Software As a service (*SaaS*)", by the developing of prototypes, based on the technique of Object Oriented Programming.

2. Introduction of this security model in the conception and development process of the solutions' kind as SaaS. So, the PaaS level (*Platform as a Service*).
3. We believe to applying our model in any previous application of the Chinese wall in the past based on the wrong models.
4. The model can be used in Inter-process communication (*IPC*), so not always and only between subject and object. So to extending it between processes (*subjects*) as active entity in the same system.
5. In the end, the implementation of our model by using the *capabilities* as an access control in real distributed system, and to control the information flow between competition groups in the social network.

### 9 Appendices, Previous Models

In this section we will present the two main proposed model of CWSP, the BN's and Lin's model

#### 9.1 BN's Model:

##### 9.1.1 Database organisation

in the BN's model, all corporate information is stored in a hierarchically arranged filing system such as that shown in Figure 1.

- At the lowest level, we consider individual items of information, each concerning 3 single corporation. In keeping with BLP, we will refer to the files in which such information is stored as objects; There are three levels of significance:
- At the intermediate level, we group all objects which concern the same corporation together into what we will call a company dataset;
- At the highest level, we group together all company datasets whose corporations are in competition. We will refer to each such group as a conflict of interest class.

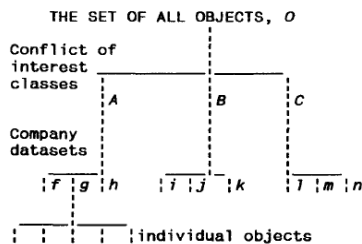


Fig. 1 The composition of the objects [1]

#### 9.1.2 Basic Model

Let  $S$  be a set of subjects,  $O$  be a set of objects and  $L$  a set of security labels  $(x, y)$ . One such label is associated with each object. We introduce functions  $X(o)$  and  $Y(o)$  which determine respectively the  $x$  and  $y$  components of this security label for a given object  $O$ . We will refer to the  $x$  as conflict of interest classes, the  $y$  as company datasets and introduce the notation  $x_j, y_j$  to mean  $X(O_j)$  and  $Y(O_j)$  respectively. Thus for some object  $O_j, X_j$  is its conflict of interest class and  $y_j$  is its company dataset.

- **Axiom 1:**

$$y_1 = y_2 \rightarrow x_1 = x_2$$

In other words, if any two objects  $0_1$  and  $0_2$  belong to the same company dataset then they also belong to the same conflict of interest class.

- **Corollary 1:**

$$x_1 <> x_2 \rightarrow y_1 <> y_2$$

In other words, if any two objects  $0_1$  and  $0_2$  belong to different conflict of interest classes then they must belong to different company datasets.

- **Definition 1:**

$N$ , a boolean matrix with elements  $N(v, c)$  corresponding to the members of  $S \times C$  which take the value true if subject  $s_v$  has, or has had, access to object  $O_c$  or the value false if  $S_v$  has not had access to object  $O_0$  Once some request  $R(u, r)$  by subject  $S_u$  to access some new object  $O_r$  has been granted then  $N(u, r)$  must be set true to reflect the fact that access has now been granted. Thus, without loss of generality, any request  $R(u, r)$  causes a state transition whereby  $N$  is replaced by some new  $N, N'$ .

- **Axiom 2:**

Access to any object  $O_r$  by any subject  $s_u$  is granted if and only if for all  $N(u, c) = true$  (i.e. by D1,  $s_u$  has had access to  $O_o$ )

$$((X_o <> X_r) or (y_c = y_r)).$$

- **Axiom 3:**

$N(v, c) = false$ , for all  $(v, c)$  represents an initially secure state.

- **Axiom 4:**

If  $N(u, c)$  is everywhere false for some  $s_u$  then any request  $R(u, r)$  is granted.

- **Theorem 1:**

Once a subject has accessed an object the only other objects accessible by that subject lie within the same company dataset or within a different conflict of interest class.

- **Theorem 2:**  
A subject can at most have access to one company dataset in each conflict of interest class.
- **Theorem 3:**  
If for some conflict of interest class  $X$  there are  $X_v$  company datasets then the minimum number of subjects which will allow every object to be accessed by at least one subject is  $X_v$ .

### 9.1.3 Sanitized Information

- **Definition 2:**  
For any object  $O_a$ ,  
 $Y_a = Y_o$  implies that  $O_a$  contains sanitized information  
 $Y_a \ll Y_o$  implies that  $O_a$  contains unsanitized information
- **Axiom 5:**

$$Y_o \iff X_o$$

In other words, if an object bears the security label  $Y_o$  then it must also bear the label  $X_o$  and vice versa. theorem 2 tells us that all subjects can access this company dataset.

- **Axiom 6:**  
Write access to any object  $O_b$  by any subject  $S_u$  is permitted if and only if  $N'(u, b) = \text{true}$  and there does not exist any object  $O_a$  ( $N'(u, a) = \text{true}$ ) which can be read by  $S_u$  for which:

$$Y_a \ll Y_b \text{ and } Y_a \ll Y_o.$$

- **Theorem 4:**  
The flow of unsanitized information is confined to its own company dataset; sanitized information may however flow freely throughout the system.

## 9.2 Lin's Model:

In this section and before to present the Lin's model we start by some by presenting some properties related to the binary relations, then the Lin's model.

### 9.2.1 Binary Relation Property

Let  $V$  a set of objects (or elements), and we recall some definitions:

- A binary relation is a subset,  $B \subseteq V \times V$  for each object  $p \in V$ , we associate a set  $B_p$  defined by:  
 $B_p = \{v \in V | pBv\}$  or  $B_p = \{v \in V | (p, v) \in B\}$ .  
That consists of all elements  $v$  that are related to  $p$  by  $B$ .  $B_p$  is called a binary neighbourhood.  
If the binary relation is an equivalence relation, then  $B_p$  is the equivalence class containing  $p$ .

- A symmetric binary relation  $B$  is a binary relation such that for every  $(u, v) \in B$  implies  $(v, u) \in B$ .
- A binary relation  $B$  is anti-reflexive: if  $B$  is non-empty and no pair  $(v, v)$  is in  $B$ . That is,  $B \cap \Delta = \emptyset$ , where  $\Delta = \{(v, v) | v \in V\}$  is called diagonal set.
- A binary relation  $B$  is anti-transitive: if  $B$  is non-empty and if  $(u, v)$  belongs to  $B$  implies that for all  $w$  either  $(u, w)$  or  $(w, v)$  belongs to  $B$ .

Let the complement,  $B' = V \times V \sim B$ , is called the complement binary relation (CBR) of  $B$ .

**Proposition:** if  $B$  is symmetric, anti-reflexive and anti-transitive, then  $B'$  is an equivalence relation [6].

**Corollary 4:** If  $B$  is symmetric, anti-reflexive and anti-transitive, then  $B'$  is the induced equivalence relation  $E_B$ .

### 9.2.2 Model

In spite of their error, Brewer and Nash's intuitive idea is a fascinating one. To keep their spirit, in [2] Lin reformulated the model based on a general binary relation; however, the expected sharpness and crispness of the model, which are reflections some characteristics of equivalence relations, are lost. With the notion of the induced equivalence relation, in this section, we will present the mains points of Lin's model based on induced equivalence.

Let  $O$  be a set of objects; an object is a dataset of a company. In Lin's model the conflict of interest is a binary relation, noted by  $CIR$ . Where  $CIR \subseteq O \times O$ , satisfies the following properties:

**CIR-1:** CIR is symmetric.

**CIR-2:** CIR is anti-reflexive.

**CIR-3:** CIR is anti-transitive.

It should be clear CIR-2 is necessary; a company cannot conflict to itself. If company  $A$  is in conflicts with  $B$ ,  $B$  is certainly in conflicts with  $A$ , so CIR-1 is valid.

To see CIR-3, let  $O = \{USA, UK, USSR\}$  be a set of three countries. Let  $CIR$  be "in cold war with". If the relation "in cold war with" were transitive, then the following two statements:

(1) USA is in cold war with USSR.

(2) USSR is in cold war with UK.

would imply that

(3) USA is in cold war with UK.

Obviously, this is absurd. In fact this argument is applicable to any country; In other words, (2) and (3) cannot be both true for any country (*that replaces UK*). So we have anti-transitivity for CIR.

Let  $E_{CIR}$  be the induced equivalence relation of  $CIR$ . In this model a new "axiom" will be explicitly

added, though it is implied by the others (*See Proposition 2*)

**CIR-4:** The granulation of CIR and partition of  $E_{CIR}$  are compatible, in the sense that each CIR-neighbourhood is a union of  $E_{CIR}$ -equivalence classes.

In [2], So, the placed the Chinese walls on the boundary of a CIR-neighbourhood, this "new axiom" implies that that such a boundary is actually on some boundary of some unions of ECIR-equivalence classes.

**CIR-5:** If we interpret CIR as "in cold war with" - relation, then the complement is "in ally with"-relation (IAR). IAR is an equivalence relation, by Corollary 4.

Here are the same views of theorems in [1] and [2].

– **Theorem 1:**

Once a agent  $S_i$  has accessed an object  $O_j$ , the only other objects  $O_k$  accessible by  $S_i$  is either inside the allied dataset of  $O_j$  or outside of  $CIR_{O_j}$ .

– **Theorem 2:**

The minimum number of agents which allow every object to be accessed by at least one agent is  $n$ , where  $n$  is the number of  $E_{CIR}$ -equivalence classes.

– **Theorem 3:**

The flow of un-sanitized information is confined to its allied dataset; sanitized information may, however, flow freely through the system.

## References

1. D.F. Brewer, M.J. Nash, in *Security and Privacy, 1989. Proceedings., 1989 IEEE Symposium on* (IEEE, 1989), pp. 206–214
2. T.Y. Lin, in *Computer Security Applications Conference, 1989., Fifth Annual* (IEEE, 1989), pp. 282–289
3. T.Y. Lin, in *24th International Computer Software and Applications Conference (COMPSAC 2000), 25-28 October 2000, Taipei, Taiwan* (2000), pp. 122–127
4. T.Y. Lin, in *26th International Computer Software and Applications Conference (COMPSAC 2002), Prolonging Software Life: Development and Redevelopment, 26-29 August 2002, Oxford, England, Proceedings* (2002), pp. 966–974
5. T.Y. Lin, in *Data and Applications Security XVII: Status and Prospects, IFIP TC-11 WG 11.3 Seventeenth Annual Working Conference on Data and Application Security, August 4-6, 2003, Estes Park, Colorado, USA* (2003), pp. 275–287
6. T.Y. Lin, in *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on* (IEEE, 2007), pp. 3027–3028
7. A. Sharifi, M.V. Tripunitara, in *Proceedings of the 18th ACM symposium on Access control models and technologies* (ACM, 2013), pp. 61–72
8. V. Atluri, S.A. Chun, P. Mazzoleni, *Journal of Computer Security* **12**(6), 799 (2004)
9. N.H. Minsky, in *Policies for Distributed Systems and Networks, 2004. POLICY 2004. Proceedings. Fifth IEEE International Workshop on* (IEEE, 2004), pp. 181–184
10. R.S. Sandhu, *Computers & Security* **11**(8), 753 (1992)
11. D.E. Bell, L.J. La Padula, *Secure computer system: Unified exposition and multics interpretation*. Tech. rep., DTIC Document (1976)
12. Z. Pawlak, *International Journal of Man-Machine Studies* **21**(2), 127 (1984)
13. Z. Pawlak, in *Joint Conference of Information Science, Research Triangle Park, North Carolina* (1997), pp. 350–352
14. Y.C. Hsiao, G.H. Hwang, in *Parallel and Distributed Processing with Applications (ISPA), 2010 International Symposium on* (IEEE, 2010), pp. 574–581
15. R. Wu, G.J. Ahn, H. Hu, M. Singhal, in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2010 6th International Conference on* (IEEE, 2010), pp. 1–7
16. T.H. Tsai, Y.C. Chen, H.C. Huang, P.M. Huang, K.S. Chou, in *Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific* (IEEE, 2011), pp. 1–4
17. A. Kesarwani, C. Gupta, M.M. Tripathi, V. Gupta, R. Gupta, V.K. Chaurasiya, in *Emerging Trends in Networks and Computer Communications (ETNCC), 2011 International Conference on* (IEEE, 2011), pp. 411–413
18. X. Xie, I. Ray, R. Adaikkalavan, R. Gamble, in *Proceedings of the 18th ACM symposium on Access control models and technologies* (ACM, 2013), pp. 89–100
19. S.M. Alqahtani, R. Gamble, I. Ray, in *Services (SERVICES), 2013 IEEE Ninth World Congress on* (IEEE, 2013), pp. 298–305