# Path and Time based Access Control

## Final Report

Lakshya Tandon
Department of Computer Science
University of Calgary

## ABSTRACT

Access control is a process of restricting access to certain resources. It basically regulates the resources to which a user can have access to and the regulation is done via some access control policies. To implement these policies an access control system is designed which when implemented works according to the designed policies to grant access to the resources. A lot of research has been done on rule based, role based, temporal, location based and some other access control technique. This report presents a model which is based on Path and Time based access control. This takes in account the path travelled by the user and the time as factors to authenticate a user. This type of a model is nonexistent and very little research has been on Path based access control which is termed as an extension to Location based access control. Here I propose a model and then evaluate that model by designing a simulator to simulate a scenario which can be controlled using a simulator controller. The simulator works as it was supposed to which helped me in validating the model.

## 1. INTRODUCTION

Security is becoming an important matter of concern in today's world. With the introduction of new technologies and security management techniques we see a crack for them. We have an attacker or a group of attackers who are always working to breach the defined security levels. So if the security isn't well defined it becomes relatively easy for an attacker to breach it. When talking about access control if the policies are not well defined rather weak, the attacker can take advantage of it to get through. Another thing which affects security in access control is the type of access control model which is being used to preserve the security. As mentioned before there are a lot of access control models which are available and some of these are commonly used. So the more common a thing is the more knowledge an attacker has about it and it becomes easier for him to breach it. Some of the security breaches related to access control are mentioned below.

- On September 17th hackers obtained access to a large swathe of Adobe Customer ID's and encrypted passwords and removed sensitive information(i.e. names, encrypted credit and debit card numbers, expiry dates etc.). Approximately 36 million Adobe users were affected.
- Hacking group AntiSec said that they hacker an FBI laptop in March 2012 accessing more that 12 million Apple Unique Device Identifiers (UDIDs). AntiSec publisher a million of these UDIDs online.
- In September 2015 attacker used malware to get into Home Depot's systems and exposed 56 million debit and credit cards.
- In January 2015, hackers broke into the health insurance giant's records and pillaged names, Social Security numbers and other sensitive information for up to 80 million customers.

In each of the attack the access control system was compromised which eventually led to information leakage. To overcome the issue of breaching we have to come up with a secure and reliable access control model which is relatively new and unique to the computing environment with strong access control policies defined with it. So this work aims to do research on lesser known access control models and derive a more dependable and authentic access control model.

## 2. MOTIVATION

The main motivation for this work comes from studying about location based access control models. In the recent years researchers have proposed certain models which deal with providing access to a certain resource to a user based on his location. This is effective in a way that the user gets access only when he is within the scope of the requested resource. This type of model can be clubbed with some other access control models and can be applied to organizations such as banks, military organizations and some other organizations which deal with highly confidential data. My major motivation came from University of Calgary's door control policy. On weekends and after 6PM on weekdays the university doors get locked and are not accessible to everyone. The doors of a specific department are only accessible to the students of that department. This causes inconvenience to the students if they want to access a building which is locked. As the building are connected and due to cold weather in Canada students prefer to stay indoors while they are traversing their path. But during odd times when the doors get locked it gets really difficult for students to stay indoors while traversing their path, the only exception being when the path lies within the departments scope of which the student is a part of. While thinking about the solution to this problem, my aim was to develop a model which solves user's issue while keeping the university's security intact. This model would basically work with the path a user travels and at what time. Also motivation came from the fact that there is little work done on path based access control. I became really excited to work on a subject where less research has been done.

## 3. BACKGROUND AND RELATED WORK

The field of access control is not new to computer science. It has been there since the inception of computers in the real world. Access control is needed to regulate the authorization of resources at different levels. Be it an operating system, social networking website, credit card payment or an online bank transfer, nothing can work without access control. Now comes how should we implement access control so as to provide maximum security to prevent security breaches. So access control is classified into various sub categories each of which have their own predefined access control policies. Some of the basic access control models are mentioned here. [6]

a. *Discretionary Access Control(DAC)*
In this the policy is determined by the owner of the model. It is a very basic model where the owner decides who is allowed to access what resources and what privileges does the users have over a given resource.

b. *Mandatory Access Control(MAC)*
It refers to allowing a user to access a resource only if a rules exits according to which the access has to be provided along with the privileges mentioned. Rule based access control forms a major part in this. An example for this would be a software firewall which works on defined set of rules. Huiying *et al.* [1] effectively have studied the implementation of rule based access control and present their view on making it more effective and efficient.

c. *Role Based Access Control(RBAC)*
Here different roles are defined of different users and the access and privileges are granted on the basis of roles. For example in an organization the Director has the highest level of privileges which get reduced as you come down the hierarchy. This model forms a backbone of access control for most of the organizations.

These were the three main access control models which have been used since access control has been defined. There are some other recent access control models which have been derived with evolving technology and availability of devices to the masses. These models basically form the foundation of my research.

a. *Time based access control*
This model is basically based on time as factor to authenticate users. In an environment with high security requirements, lot of users and limited resources. All of the resources cannot be available to all the users at once, in this scenario time comes in effect where a user is given access to a resource only at a given time or a time period. A typical example of this would be CPSC Labs which are open to all from 9:00 AM to 5:00 PM but get locked after 5:00PM when no one is there at the technical help desk to monitor students. Time based access control is often clubbed with other access control models, majorly with role based to further enhance the security of the environment. Example remaining the same as previous, when doors are locked only CPSC students can gain access using their Unicard while others cannot.

b. *Location based access control*
This type of access control model has become more popular in the past 5 years with the increase in use of hand held Global Positioning System (GPS) devices. This basically works on the location of the user where a certain resource is only available to the user while he is at a specific location. His location is obtained using a GPS device which in majority of the cases is a smart phone and then it is matched against the location stored in the database. An Example would be an organization which wants certain documents on their server to be accessed by their employees when they are at a specific location within the organization like Wi-Fi projector's operational manual can only be accessed when the users are in the conference room where the projector is installed. The projector is only accessible to the employees when they are near to it so that no one is able to operate the projector from other room. Cleeff *et al.* [2] have studied how this model can be beneficial in real world scenario and how can it be used to solve real word authentication problems while making users life easy. Also Ray *et al*. combined location based access control with mandatory access control and presented some new techniques on this.

c. Location and time based access control
In recent years, Time based access control has been clubbed with location based access control to give rise to a more secure and reliable access control model. In this a user is granted access to a certain resource when he is present at a specific location at a given instance of time or over a period of time. Here the database is checked for the location first and then time, if both of them satisfy with the user's attributes only then the user is granted access. This model is often clubbed with Role based access control and cumulatively a Time and Location Role based Access Control model is derived. This is even more secure than the previously defined. Other models can also be embedded to it to make the whole model more secure and efficient. This when implemented properly is supposed to give highest level of security amongst all models discussed.

# 4. PROPOSED METHOD – THE CONCEPT

### *Path based access control*

A path is a collection of points and each point can be represented by x and y coordinates on a plane. So point is basically a location on the plane represented by x and y coordinates. When we join these we get a finite path. An access control system which follows a user's path to authenticate the user is termed as a path based access control system. This is important when a user wants to access multiple access points in an orderly manner. Here each access point will have its own location represented by x and y coordinates on the plane will be stored in the database in an orderly manner. The access point which is accessed first is stored first in the path followed by other access points. Now user will get authenticated in a way that he would only be granted access at an access point if it exists in his path and the previous access point he accessed was the one mentioned just before the one he wants to access now. For example in a path consisting of 5 access points, the user will be granted access at $4^{th}$ access point if he was granted access at $3^{rd}$ access point and the $3^{rd}$ access point was the last access point he visited.

### *Path and Time based access control*

Previously in this report we have seen that how to two different access control models can be clubbed together to give rise to a new more secure and effective model. In my case I combined path based access control model with time based access control model. This amalgamation of both the models can be explained as a user will only be granted access at an access point if it lies in his path and he tries to access that access point within a specific time frame. This model can be incorporated in environments which have multiple access points which are to be accessed only within a specific timeframe. The best environment for this would be University of Calgary where this model can be applied to door access. Lets take an example to explain this, Suppose a user has a path consisting of 5 access points and a time saved for each access point, then the user will be granted access at access point 4 if all of the following satisfy.

- The access point (AS) 4 is in user path.
- AS 3 was accessed just before accessing AS 4.
- The user accesses AS 4 within the time period defined for it.

If any of these conditions doesn't satisfy the user will be denied access at AS 4.

# 5. PROPOSED MODEL – THE WORKING

By now we know that path and time based access control model basically uses two attributes to provide access to user, namely the path of the user and the time at each point in path. These two are stored in the database and are then validated against users real time attributes. Now the real question is that in a real time environment how will the path and time be made and stored in the database so that the user can be authenticated against it. So basically we have to train the model first so as to use it later for authentication. A smartphone application will train the model and later the same application will be used to authenticate the user. This application should be installed on user's phone who will be a part of this implementation.

### *Training the model for a specific user*

As discussed, before a model grants access to the users according to the policies that are time and path, it beforehand needs to know the path of each user and the time when the user traverses it. For this purpose we need to train the model. This will be done by each user using the application which is developed. The application will have a training mode (Start Training button) which the user will use. The user will start the training mode and start walking on the path which he wants to traverse later. The application beforehand knows the location of access points. As the user reaches an access point the application stores that access point. Application determines whether the user has reached an access point by keeping a track on his location using GPS. This way the application will store a series of access points as the user traverses his path. Coming on to the time part of the model the application will compute the time taken by the user to travel between two access points. In the first case the time would be from the source of the user to the first access point. Once user has reached his destination there will be a button to stop training which when pressed will inform the application to stop tracing user. Now once this is done the application would ask the user the time at he wants to traverse that path at. Upon giving the time by the user the application will compute the times at each access point based on the time interval which it had stored earlier and then these time at each access point would be stored in the database. To explain this with an example if it takes 2 minutes for a user to reach access point 1 from his source, then if the user specifies the starting time as 9:00PM then the time stored for access point 1 would be 9:02PM and in a similar fashion for other access points in this path. When a user completes training, the model trained by him will go to University's security officer for approval. Once he approves it user can start traversing his path at odd times. A user can train more than one route for traversal. As this is a scenario for University of Calgary's students, it should be kept it mind that the training should be performed in daytime from 8:30AM to 5:30PM when all the doors are accessible. It should also be kept in mind that the user should avoid stopping in between for perfect results.

### *Working of Model*

Once the model has been configured and all the routes of all the users have been saved into the database, next comes the operational phase. This is the main functionality of the model where each user has to be authenticated at each access point according to the information present in the database. This model has:

- A list of users $U = \{u_1,u_2,u_3 \ldots\ldots u_n\}$
- A list of routes(paths) $R = \{r_1,r_2,r_3 \ldots\ldots r_n\}$ for each user in U
- A list of access points $A = \{a_1, a_2, a_3 \ldots a_n\}$ for each route in R.
- A list of time $T = \{t_1, t_2, t_3 \ldots\ldots t_n\}$ such that $t_i$ corresponds to $r_i$ is the time when the user starts to commute.
- A list of time between two access points TA $\{ta_1, ta_2, ta_3 \ldots ta_n\}$ for each route in R. Eg: $ta_1$ is the time travelled between source and a1 for route $r_1$ and tan is time between $a_{n-1}$ and $a_n$.

Now the access control policy for this model is defined as:

*At any access point $a_n$, the user $u_n$ while traversing route $r_m$ is granted access only and only if $a_n$ lies in $r_m$, the last access point accessed by the user was $a_{n-1}$ keeping in account all the previous access points and the user is still in the timeframe $t_f$ which is the current time such that $t_f \leq$ (time when then user accessed $a_{n-1}$) + $ta_n$.*

Lets discuss the above mentioned policy, When the user reaches an access point $a_n$ while he is traversing $r_m$ the first thing which is checked is that $a_n$ is a part of $r_m$. Then the next thing to be checked is that last access point which was accessed was $a_{n-1}$, also all the previous access points in the route are checked for to ensure he is in his correct route. The last condition is that the time has not expired while he has reached $a_n$. If all of these conditions evaluate to true the user gets access through the access point $a_n$ else not. At each access point the user's location is collected using GPS via application running on his. This would be service running in background which would be part of the application. Using this model the security would increase and will become more convenient for users to traverse their paths in off hours. That's how this model aims to combine path and time to and come up with a new Path and Time based access control technique.

### *Challenges to this model*

While the model seems simple and transparent but there are certain things which need to be taken care of while its implementation is done in real world situation. These things are:

- *Managing multiple users and their common access points*

  This model manages multiples users very well as it first makes a list of users and then for each user it defines the routes. Each route is a collection of access points. There may arise a situation where the two users with separate routes may have some common access points. Lets take an example to explain this:

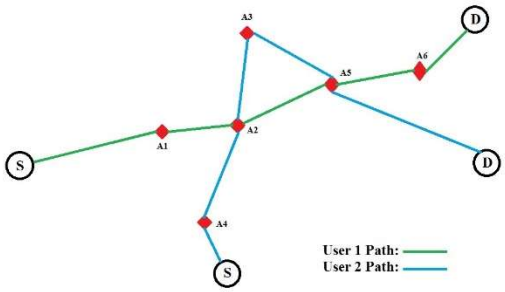


Fig 1: Managing Multiple Access Points

In the above figure we have two users User1 and User 2 which have S as their source and D as destination. Access points A2 and A5 are common in between their paths. Now there may be a situation where User 2 accesses A2 and then tries to access A5. He won't be granted access to A5 directly because he is supposed to access A3 after A2 and then A5. This scenario is helpful when there are some confidential resources in the path A2-A5 and User 2 should not be granted access to them.

So this is how this model tackles this situation.

- *Managing speed of the user*

  This model behaves strictly on the time policy that is an access point is available to the user for a certain period of time. After the time expires user cannot access that access point even if he's going the correct path. Now there might be a situation that a user meets his friend and has to stop in between for a small chat. A user may stop in between A2 and A5 if he encounters a friend. In that case the application running on his phone will keep a track of his walking. There is a hardware in modern smart phones known as accelerometer. Whenever the phone moves the values of accelerometer change rapidly, so when the person is stationary the accelerometer values won't change rapidly and that is when the app would detect that the person has stopped. It would start counting the time when the accelerometer doesn't show big changes and stop when big changes appear that is when the person starts to move again. I would then append the time to the access time of the next access point and access would be granted when the user reaches the next access point.

- *Time based access in access point*

  The main objective of an access control model is to ensure security of the resources. Time based access has been applied on the access points to further increase the safety of the model. Although it has made the model a bit complex and introduced some challenges but those challenges have been addressed.

## 6. MODEL SIMULATION

Although the real life implementation of this model is not achievable in the given timeframe of the project and due to non-availability of university resources but a simulator for simulating the access control policies can always be developed. In this section we'll take about the simulator which was developed to simulate the movement of users for such an environment. Simulator is a windows application which was developed using C# as the programming language and .Net as the supporting framework. A local server was also made to maintain the database so that the users can be authenticated. The server used was MS SQL Server 2014 and it ran locally on the development system.

Talking about the simulator, it appears as windows form on the computer screen which when initialized has predefined access points at different predefined locations.

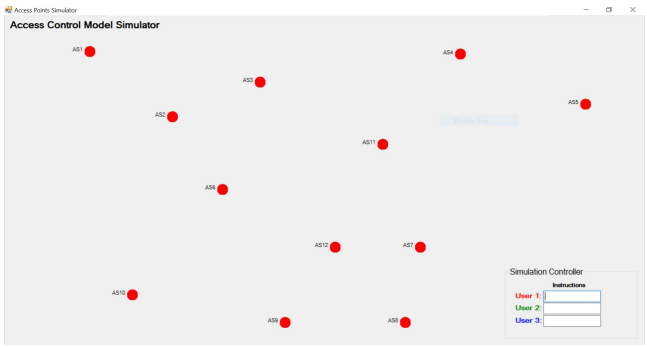


Fig2: Simulator Initialization

Figure 2 shows the initialization of simulator. You can see that the access points are represented by red dots and are distributed all over the screen. On the bottom left corner of the screen you can see a simulator controller which controls the user movements. This simulator is designed to work for three users here. You can pass the name of the next access point, 'SLEEP' command or 'WAKE' command in each of the text box for corresponding user.

Passing the name of access point will take user to that access point if he is authenticated. 'SLEEP' command is used to stop a user in between two access points which would eventually demonstrate the scenario when a user stops in between for a chat with a friend. Using 'WAKE' command you can again make the user to move. The time is calculated between 'SLEEP' and 'WAKE' command and is then appended to the time at next access point. When the user is sleeping there is an 'isStatioanry' message displayed besides textbox. Once the user gets access to an access point the access point turns green as seen in figure 5.

The simulator works with the database which is developed on SQL Server. The simulator check the database for user paths and user times. The database isn't a very large one and has only two tables, one for storing path and other for storing time.

Results | Messages

| | uid | uname | upath |
|---|---|---|---|
| 1 | 1001 | User1 | AS1:AS2:AS3:AS11:AS5 |
| 2 | 1002 | User2 | AS6:AS12:AS7:AS11:AS5:AS4 |
| 3 | 1003 | User3 | AS10:AS9:AS8:AS12:AS7:AS11:AS5 |

Fig 3: User path table

Figure 3 shows the table which stores user's path. It has three columns where uid stores the user ID, uname stores the user name and upath stores the user path which are basically the access points separated by ':'.

Results | Messages

| | uid | utime |
|---|---|---|
| 1 | 1001 | 4:13AM;4:15AM;4:17AM;4:18AM;4:20AM |
| 2 | 1002 | 4:11AM;4:12AM;4:13AM;4:16AM;4:18AM;4:19AM |
| 3 | 1003 | 4:15AM;4:16AM;4:17AM;4:17AM;4:18AM;4:19AM;4:21AM |

Fig 4: User time table

Figure 4 shows the user time table which the simulator uses to check for time. Here the columns are uid which has the user ID and utime which has user time which is time at each access point separated by ';'.

Also in the simulator we don't have an access point available only at a specific time. It is available for a period of time that lasts for 2 mins. So if the time at access point 2 for user 2 is 4:12AM then the access point 2 is available from 4:11AM to 4:13AM that one minute before and one minute after the specified time.

In the proposed model the user trains the model with an application which stores the values in database but in this simulator I have manually filled in the values. There is no such training application as training application requires realtime user movements and real time access points which weren't available here.

The following link shows a quick the working of the simulator:

https://www.youtube.com/watch?v=rVDKgNNtn4Y&feature=youtu.be

## 7. SIMULATION RESULTS

The simulator has access points marked with red dots and for each user a path and corresponding time is given in the database. The paths used for testing are:

- User1- AS1:AS2:AS3:AS11:AS5
- User2- AS6:AS12:AS7:AS11:AS5:AS4
- User3- AS10:AS9:AS8:AS12:AS7:AS11:AS5

The corresponding time stored for each access point in the user path is:

- User1- 4:06AM;4:07AM;4:08AM;4:10;4:13AM
- User2- 4:05AM;4:07AM;4:09AM;4:10AM;4:11AM;4:12AM
- User3- 4:06AM;4:07AM;4:07AM;4:08AM;4:09AM;4:11AM; 4:13AM

As instructions are passed in the simulator you can see lines being drawn between two access points. In the simulation controller window each user has textbox for its instructions. When you pass the name of the next access point in the textbox the simulator checks the database for validation according to the defined policy of the model and if access is granted the access point turns green and a line is drawn.

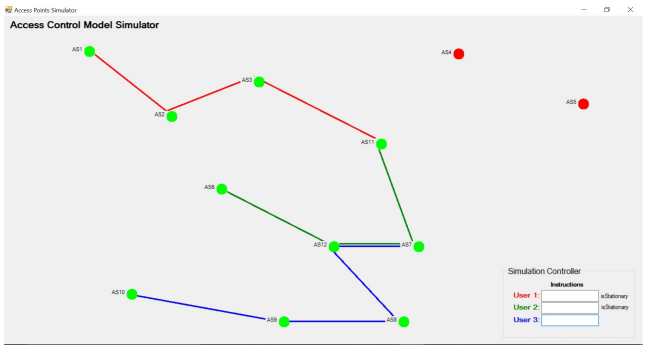


Fig 5: Intermediate results of simulator

Figure 5 shows intermediate results of simulator. You can see user1 has traversed AS1, AS2, A3 and AS11. All of these have turned green as access has been granted. You can also see that user 1 is stationary after gaining access through AS11. 'isStationary' message is displayed besides user1 textbox. For user 2, it has traversed AS6, AS12, AS7, AS 11 and has become stationary after that and is waiting for 'WAKE' command to be issued. User 3 has covered AS10, AS9, AS8, AS12, AS7 and is still moving towards its next access point.

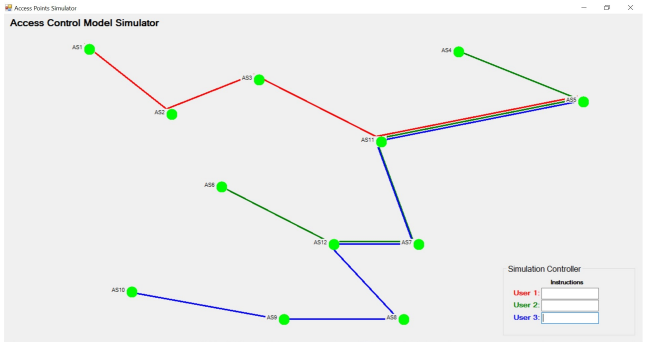


Fig 6: Full Simulation

Figure 6 shows full simulation of all the three users, you can see that all of them have traversed their paths and all the access points have turned green.

### *Policy Check*

Lets see how the simulator addresses challenges we discussed before.

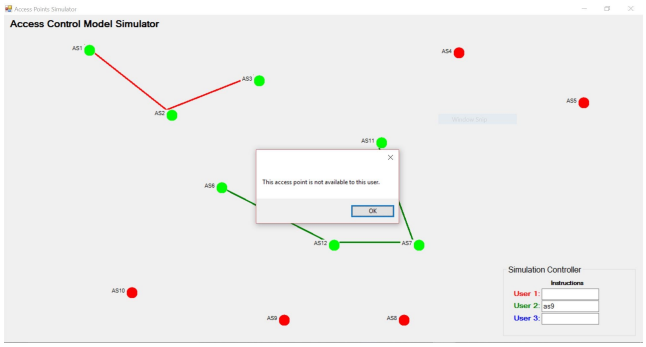


Fig 7: Access point not available

Figure 7 shows a situation when user 2 tries to access AS9 which isn't in its path, a message is displayed which says that access point is not available and AS9 remains red. Thus a user can only access an access point which is present in its path. This shows that the simulator is following the path based policy.

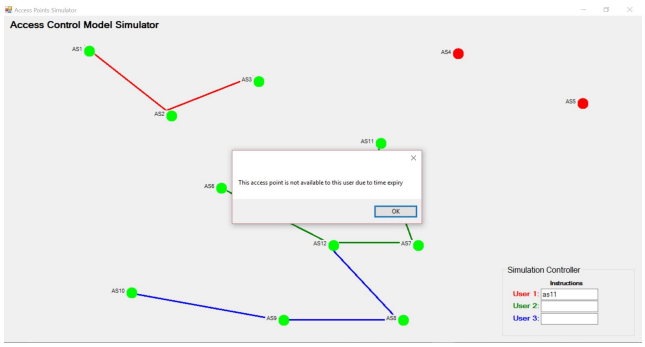


Fig 8: Time Expiry

In figure 8, you can see a message being displayed which says that the access point is not available due to time expiry. We get this message when user1 tries to access AS11 after 4:09AM. This shows that the simulator is also following the time based policy.

Thus the results show that the simulator works as per the proposed model which is Path and Time based access control.

## 8. LIMITATIONS OF SIMULATOR

First of all the simulator doesn't simulate the complete model because it is not possible to simulate the complete model with available resources. The complete model basically needs real time working smart phone application and real time walking to demonstrate things. Also only three users are taken in the simulator, due to the lack of time the simulator was made in a manner that I didn't had provision to increase the number of users without making changes to the code. If I had more time to work on simulator I would have made the user addition to the simulator as dynamic. Also one major limitation is that the database which the simulator uses is maintained manually while in the model a training application is used to do the same. So whenever the simulator has to run the times in databases are to be updated every time as per the system's time as simulator uses system's clock. This process becomes really tedious as you don't want to miss the updated time in database while simulating so you need to perform manual updating really fast. In real application it would use the server's clock but in our case the development system is the server. Despite of these limitations the simulator has been successful in demonstrating what it was designed for.

## 9. CONCLUSION AND FUTURE WORK

From the proposed model and the simulator it can be said that Path and Time based access control is an effective access control technique which has high efficiency and provide great security in scenarios where you have many access points and they are to be used by various users. It is a very secure model and can be extended to make more secure as it is scalable for new augmentations. The future work on this could be to work on a special case where a user who is not allowed to access an access point crosses that access point along with the user who has access to it. This can be avoided by monitoring the users who are accessing an access point and ones who are breaching the rules can be identified and acted against.

## 10. REFERENCES


[1] Huiying Li, Xiang Zhang, Honghan Wu, Yuzhong Qu, *Design and Application of Rule Based Access Control* Policies, ISWC2005.

[2] Andre van Cleeff, Wolter Pieters, Roel Wieringa, *Benefits of Location-Based Access Control: A Literature Study*, 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom).

[3] Clara Bertolissi, Maribel Fernandez, *Time and Location Based Services with Access Control*, New Technologies, Mobility and Security, 2008. NTMS '08.

[4] I Ray, M Kumar, *Towards a Location-Based Mandatory Access Control Model*, Computers and Security, 2006 – Elsevier

[5] Ardagna, Claudio A., et al. "*Access control in location-based services.*"Privacy in Location-Based Applications. Springer Berlin Heidelberg, 2009.

[6] Pierangela Samarati and Sabrina de Capitani di Vimercati, *"Access Control: Policies, Models, and Mechanisms"*, Dipartimento di Tecnologie dell'Informazione, Universit`a di Milano Via Bramante, Crema (CR), Italy