INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

Secure and Efficient File Access Using Location Proof Updating System

Drakshayini
M.Tech 4th Sem, Department of Computer Science and Engineering
SEA College of Engineering Technology, Bangalore, India
Shaik MD Ghouse
Professor, Assistant Professor, Department of Computer Science and Engineering
SEA College of Engineering Technology, Bangalore, India
Dr. B. R. Prasad Babu
Professor & HOD, Department of Computer Science and Engineering
SEA College of Engineering Technology, Bangalore, India

***Abstract**:*
Wireless networks have allowed organizations to become more mobile therefore organizations are now widely using wireless networks. Network security is a big concern for individuals and organizations because, all vital information is stored on the network and most critical process of the business are done through the network. If a network fails or security is compromised an organization could be completely crippled, this will be a big challenge in user centric location based services. In this proposed system, I am going to address this problem using location proof updating system. The system consists of Server, Prover, and client. Server contains sensitive files, clients always trying to download these files from server. The challenge here is client is allowed to download the files from server when they are in restricted location range, to achieve the downloading process location proof updating system produces a Prover application which will key monitor client location details and send this details in periodical intervals of time to the verifier service in server.

***Keywords**: security, location proof, location privacy, pseudonym, secure*

1. Introduction

The main challenge faced in security protocol design concerns the need to satisfy a number of conflicting security requirements. In the domain of location-based services, this conflict shows the tension between location assurance and location privacy. On the one hand, service providers must know their clients location with some level of assurance; while on the other hand, clients do not want to expose more location details other than needed for the requested service. A second factor complicating the design of a security protocol is its functional requirements and the assumptions. The system strongly depends on the intended usage scenario of the protocol. This factor clearly shows location-based service scenarios, each of them leading to essentially different solutions. In the following, we briefly exemplify three of such scenarios.

There are many kinds of location-sensitive applications. One category is location-based access control. For example, a hospital may allow patient information access only when doctors or nurses can prove that they are in a particular room of the hospital [2]. Another class of location-sensitive applications require users to provide past location proofs [3], such as auto insurance quote in which auto insurance companies offer discounts to drivers who can prove that they take safe routes during their daily commutes, police investigations in which detectives are interested in finding out if a person was at a murder scene at some time, and location-based social networking in which a user can ask for a location proof from the service requester and accepts the request only if the sender is able to present a valid location proof. The common theme across these location sensitive applications is that they offer a reward or benefit to users located in a certain geographical location at a certain time. Thus, users have the incentive to cheat on their locations.

Location-sensitive applications require users to prove that they really are (or were) at the claimed locations. Although most mobile users have devices capable of discovering their locations, some users may cheat on their locations and there is a lack of secure mechanism to provide their current or past locations to applications and services. One possible solution [4] is to build a trusted computing module on each mobile device to make sure trusted GPS data is generated and transmitted.

Geo-location data is gathered in a number of ways, including built-in Global Positioning System devices, IP address, or Wi-Fi network mapping. Location proof plays a vital role in location sensitive applications. Location sensitive applications such as Location based access

Control [3], Location aware routing [5] etc., are used in location proofs. They are also helpful in providing a history of location proofs and identifying a geographical location of users. Location proof is a piece of data that certifies a receiver to geographical location information can be eavesdropped by adversaries. It may cause vulnerability towards location privacy of the user. Public key Cryptographic operation is used for encryption and decryption of communicating [5]. In the location proof updating system, location messages and prevents from eavesdropping. The Process of hiding the identity of nodes is an approach to obtain identity privacy; the identity of the node is hidden by using pseudonym.

2. Proposed System

Aim of proposed system is to design architecture of system, such that it needs to provide secure and efficient access to system without compromising the security, privacy of user and preventing un- authorized to access the system to perform this functions, Location information is identified by using geographical representations through latitude and longitude points. We implement an Advanced System for Location Tracking and Updating in which co-located Wi-Fi enabled mobile devices mutually generate location proofs and update to a location proof server. By this it is easy to find the exact location of the client using a web portal, accessed by a Server by simply login into the system. The users must register with the CA (certificate authority). CA will generate credentials in the form of pseudonyms. These credentials send to the user mail ID by CA. Using these credentials user can able to login to the system and they can access the system, if user prove that they are at claimed location and they are trusted.

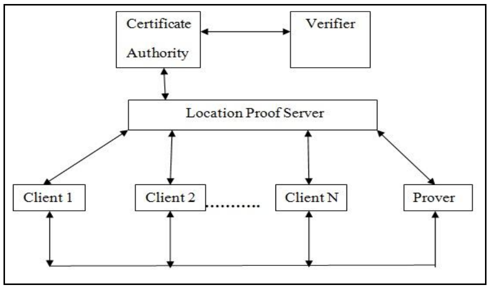


Figure 1: Proposed system Architecture

Architecture of system is the proposed work shown in Figure1. Using Wi-Fi services mobile user access the system and get register with CA. Where, CA will generate necessary credentials for each client of the system which contains uniquely generated pseudonym and sends to respective client. Using this credentials client can login to system to access the system. When client is login to system, Prover will track the location of client and sends to server. Apart from this, client also generates the location proof in an encrypted form by using server's public key and it signs using client private key in order to provide more security to prevent attacks. This location proof send by client will be verified by server using verifier application, it checks both location proofs send by client and Prover and verify whether client is at particular location or not. If client is at specified location then it will updated the server about client. If client is trusted enough then system will allow the client to download the sensitive files from server, else it will not allow the client to download files, by stating un-trusted user.

Proposed system architecture consists of three services, they are as follows

2.1. Prover service

Prover is the third party application who needs to collect location proof from the entire user when they are in communication range. Prover has to perform following functions.

- Time Event and Request for Proof: As soon as Prover will logged in timer will Start. Timer will send request to the all users who are all comes under his location. This timer event will happen for every interval of time. Use of request is to get proof from all users if it's belonging to that that location.
- Get proof and send it to verifier service at server: Prover will receive the proof of entire users. Then Prover will attach its proof also along with that and it will send to server.

2.2. User/client service:

User need to perform the following functions to prove it trustworthiness so that it can able to download files from server.

- Get the request from Prover and generate proof: User will get proof request from Prover along with Prover signature. If user ready to give proof means, then that user become witness of the Prover. Then user will generate proof to Prover, this proof consists of Prover signature, witness's current time stamp, witness's pseudonym, and their location.
- Encrypt the proof and send it to Prover: After generating proof, that will be encrypt by using Server Public key. Encrypted proof will send to Prover

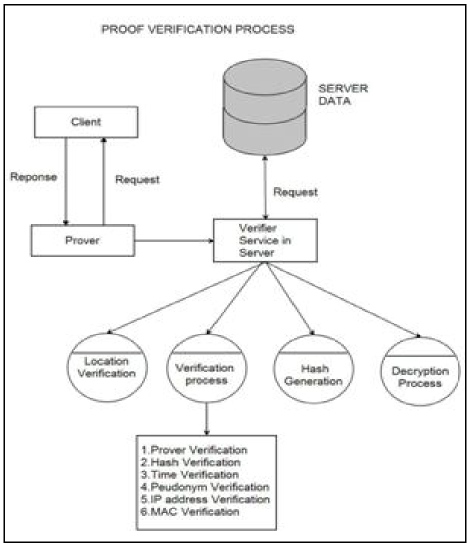


Figure 2: Proof Verification Process.

2.3. Server Service

Server has to perform following functions to verify client status.

- Get the Details from Prover: Server will receive all location proofs from Prover along with Prover signature. Then server will hand over location proofs to verifier for verification.
- Verify location proof and Update the Status in Server Data Base: Verifier starts to verify as shown in fig 2. First it will verify Prover signatures. After that it will verify witness/user proof. Verifier verifies users MAC address, IP address, Time stamp of user/witness, Pseudonym and witness location. If anyone verification is fails also those user/witnesses status consider as un-trusted and it will updated to database.

2.3.1. Algorithm1: Location Proof Generation

- User trying to access server through Prover.
- Prover will Generate two key (Pprov, Rprov) and give to the witness. Pprov=Previous Prover key, Rprov=Random key. If previous not present we generate dummy data and send to user.
- User accepts the request from Prover and generates **M= P_{prov}|| R_{prov} || T_{witt} || L** Then encrypt the following data by server public key. **Proof= E_{serv} (P_{witt} || S_{witt} (M) || H (M)).**
- Twitt=Time of witness, L=Location, Pwitt= Pseudonym of witness, Switt=Signature of witness, H (M) =Hash function of M. Finally send Proof to Prover.
- Prove send this proof to server along with two keys (P_{prov}, R_{prov}) and Prover location.

2.3.2. Algorithm 2: Verification of Location Proof

- Decrypt the content sent by the user /witness using server's private key, now Verifier has following data and it verifies , Twitt=Time of witness, L=Location, Pwitt= Pseudonym of witness, Switt= Signature of witness, H(M)=Hash function of M.
- Create a M using following formula and using HASH function create H(M)$^{|}$
  M=Pprov || Rprov || T_{witt} || L
  - H(M) and H(M)' are equal proceed else message is hacked in middle.
  - Verify Twitt from Witness message with Time validity when the Location Proof was conducted from sever table. If fails return Time Expired message.
  - Verify Pwitt from Witness message with DB. If fails return Pseudonym fails message.
  - Verify with Pprov get from encrypted message with Pprov given by Prover. If fails return Pprov fails.
  - Verify with Rprov get from encrypted message with Rprov given by Prover. If fails return Rprov fails.
  - Verify with Location (L) get from encrypted message with Location (L) given by Prover. If fails return Location fails.
- If all these ***six*** conditions are passed, make the status of Witness in Server table as **trusted** else make it as ***un-trusted*** and mention the condition which fails.

Algorithm 1 and 2 helps to generate and verify the location proof given by user respectively. If location proof verification is successful client can download the file from server else users deny accessing the files.

3. Implementation

In Figure 3, the process flowchart of the implementation is shown. The proposed system is implemented in 32 bit Windows operating system with 1 GHz Processor and 1GB RAM .The design environment is selected in java. User registers with system if it is success then he/she can view the files list and can access server for a file it need. Before accessing the server for file user needs to prove their location, means that they are in a restricted area for example within campus, within an office building so on. After the proof verification is successful then only client can download the file from server. Hence secure and efficient file access using location proof updating system is more effective and it gives more security to organization that uses the wireless network.

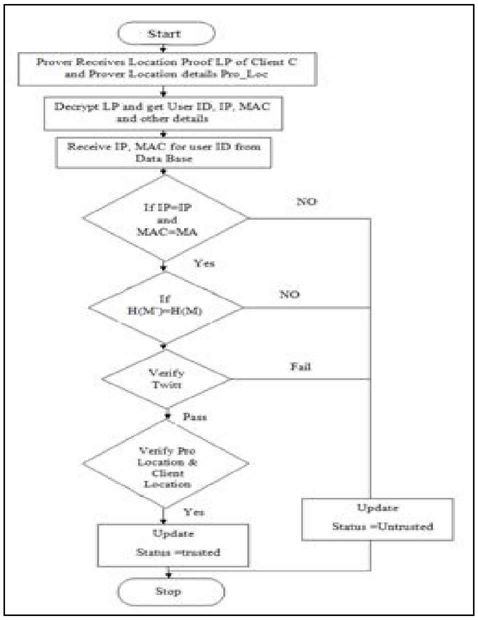

Start
Prover Receives Location Proof LP of Client C and Prover Location details Pro_Loc
Decrypt LP and get User ID, IP, MAC and other details
Receive IP, MAC for user ID from Data Base
If IP=IP and MAC=MA
NO
Yes
If H(M`)=H(M)
NO
Verify Twitt
Fail
Pass
Verify Pro Location & Client Location
Yes
Update Status =trusted
Update Status =Untrusted
Stop

Figure 3: Process flowchart implementation

4. Performance Analysis

In this section we check how system will able to identify the authorized user by checking location proof send by user as well as from Prover. It also tracks user's and Prover longitude and latitude, then it calculates entropy (To find client with in a restricted location or not). If calculated entropy is less or within a specified range then client is allowed to download the file. If it is larger than specified range then user is outside the restricted area hence system will not allow the user to download the file.

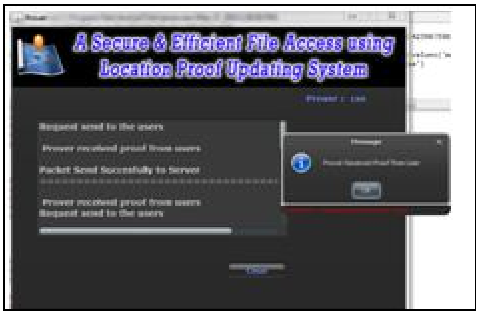
Performance of this system can be analysed by using following two criteria

- When an authorized user access the system
- When an unauthorised user access the system.

Let us see how system will respond to above criteria through user interface.

4.1. When an Authorized User Access The System

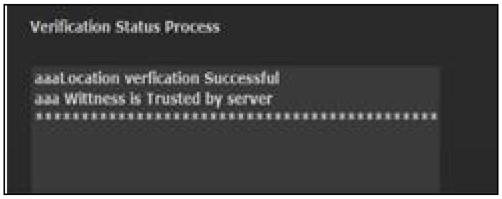
Prover tracks all users who are in communication range and sends request for proof, after receiving the proof then it will send proof to server.


A Secure & Efficient File Access using Location Proof Updating System

```
Request send to the users
Prover received proof from users
Packet Send Successfully to Server
Prover received proof from users
Request send to the users
```

Figure 4

After proof verification server will decide, user is trusted or not.



```
Verification Status Process

aaaLocation verfication Successful
aaa Wittness is Trusted by server
********************************************
```

Figure 5

Then it will allow the user to download the file from server.

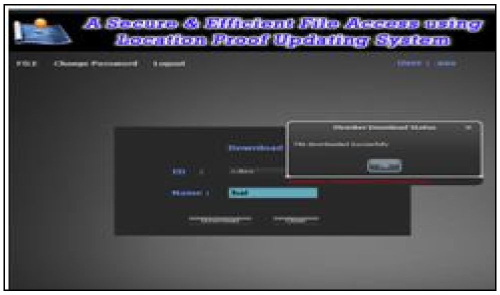

A Secure & Efficient File Access using
Location Proof Updating System

Figure 6

4.2. When an Unauthorized User Access the System

When an unauthorized user is trying to access the system from different machine and different user ID which is not assigned to that user, then system will not allow him to download files and gives the error report as you are un-trusted user or trying to change IP address or you are not a valid user of the system.

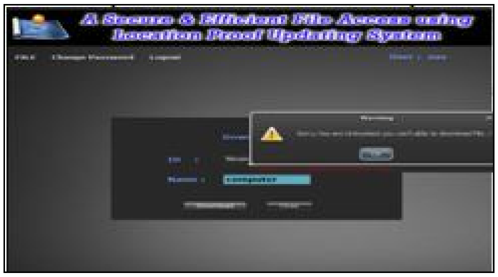
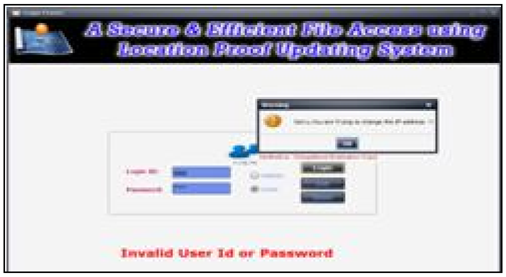

A Secure & Efficient File Access using
Location Proof Updating System

Figure 7


A Secure & Efficient File Access using
Location Proof Updating System

Invalid User Id or Password

Figure 8

5. Conclusion

This system is developed in java technology and tested with network of computers. The execution results shows this system satisfy the all the functional requirements which are specified in design phase. For more security reasons not only IP address, MAC address is also included in this system for verification. This will be help full in secure location based file accessing system. I can say this system is more suitable in any sensitive file management system that uses wireless network such as hospital record management, any corporate companies file management, and very confidential military file management system.

6. References

1. Z. Zhu and G. Cao. Applaus: A privacy-preserving and collusion resistance in location proof updating system IEEE INFOCOM 2011.
2. W. Luo and U. Hengartner, "Proving Your Location Without Giving Up Your Privacy," Proc. ACM 11th Workshop Mobile Computing Systems and Applications (HotMobile'10), 2010.
3. S.Saroiu and A. Wolman, "Enabling New Mobile Applications with Location Proofs," Proc. ACM 10th Workshop Mobile Computing Systems and Applications (HotMobile '09), 2009.
4. V. Lenders, E. Koukoumidis, P. Zhang, and M. Martonosi, "Location-Based Trust for Mobile User-Generated Content:applications Challenges and Implementations," Proc. Ninth Workshop Mobile Computing Systems and Applications, 2008.
5. W. Luo and U. Hengartner. Proving your location without giving up your privacy. In ACM HotMobile, 2010