International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 4| ISSN: 2456-3307

STAMP: Enabling Privacy-Preserving Position Proofs For Mobile Users

M. Lakshmipathi[1], J. Kiran[2]

[1]Sri Padmavathi Collage Of Computer Science & Technology,Tiruchanoor,A.P, India
[2]Assistant Professor, Sri Padmavathi Collage Of Computer Science & Technology,Tiruchanoor,A.P, India

ABSTRACT

Positionbased services are quickly becoming immensely popular. In addition to services based on users' current location, many potential services rely on users' location history, or their spatial temporal provenance. Malicious users may lie about their spatial emporal provenance without a carefully designed security system for users to prove their past locations. In existing, which includes an optimal method for the light setting and an approximate method for the heavy setting. The optimal method leverages vertex grouping and best-first pruning techniques to expedite the mining process. The approximate method can provide the performance guarantee by utilizing the greedy heuristic, and it is comprised of efficient updating strategy, index partition and workload-based optimization techniques. We propose an STP proof scheme named Spatial-Temporal provenance Assurance with Mutual Proofs (STAMP). STAMP aims at ensuring the integrity and non-transferability of the STP proofs, with the capability of protecting users' privacy. Most of the existing STP proof schemes rely on wireless infrastructure to create proofs for mobile users. However, it may not be feasible for all types of applications, we have presented STAMP, which aims at providing security and privacy assurance to mobile users' proofs for their past location visits. STAMP relies on mobile devices in vicinity to mutually generate location proofs or uses wireless APs to generate location proofs. Integrity and non-transferability of location proofs and location privacy of users are the main design goals of STAMP.

Keywords : Spatial-Temporal provenance Assurance with Mutual Proofs (STAMP), Malicious, privacy preserving

I. INTRODUCTION

STAMP requires only Single semi-trusted third party which can be embedded in a Certificate Authority (CA). We design our system with an objective of protecting users' anonymity and location privacy. No parties other than verifiers could see both a user's identity and STP information (verifiers need both identity and STP information in order to perform verification and provide services). Users are given the flexibility to choose the location granularity level that is revealed to the verifier. We examine two type s of collusion attacks: A user who is at an intended location masquerades s another colluding user and obtains STP proofs for . This attack has never been addressed in any existing STP proof schemes. Colluding users mutually generate fake STP proofs for each other. There have been efforts to address this type of collusion. However, existing solutions suffer from high computational cost and low scalability. Particularly, the latter collusion scenario is in fact the challenging Terrorist Fraud attack, which is a critical issue for our targeted system, but none of the existing systems has addressed it. We integrate the Bussard-Bagga distance bounding protocol into STAMP to protect our scheme against this collusion attack. Collusion scenario is hard to prevent without a trusted third party. To make our system resilient to this attack, we propose an entropy-based trust model to detect the collusion scenario. We implemented STAMP on the Android platform and carried out extensive validation experiments. The experimental results show that STAMP requires low computational overhead.

Most of the existing STP proof schemes rely on wireless infrastructure (e.g., WiFi APs) to create proofs for mobile users. However, it may not be feasible for all types of applications, e.g., STP proofs for the green commuting and battlefield examples certainly cannot be obtained from wireless APs. To target a wider range of applications, STAMP is based on a distributed architecture. Co-located mobile devices mutually generate and endorse STP proofs for each other, while at the same time it does not eliminate the possibility of utilizing wireless infrastructures as more trusted proof generation sources. In addition, in contrast to most of the existing schemes which require multiple trusted or semi-trusted third parties, STAMP requires only a single semi-trusted third party which can be embedded in a Certificate Authority (CA). We design our system with an objective of protecting users' anonymity and location privacy. No parties other than verifiers could see both a user's identity and STP information (verifiers need both identity and STP information in order to perform verification and provide services). Users are given the flexibility to choose the location granularity level that is revealed to the verifier.

II. ALGORITHM

A. Preliminaries

Location Granularity Levels: We assume there are n granularity levels for each location, which can be denoted by $L_1,L_2,L_3,\ldots..L_N$ where L_1 represents the finest location granularity (e.g., an exact Geo coordinate), and represents the most coarse location granularity (e.g., a city). Hereafter, we refer to location granularity level as location level for short. When a location level L_X is known, we assume it is easy to obtain a corresponding higher location level L_y where y>x .

B. Cryptographic Building Blocks

STAMP uses the concept of commitments to ensure the privacy of provers. A commitment scheme allows one to commit to a message while keeping it hidden to others, with the ability to reveal the committed value later. The original message cannot be changed after it is committed to.

One-way hash functions have the similar binding and hiding properties as commitment schemes. However, for privacy protection purpose, we do not use hash functions because they are vulnerable to dictionary attacks. An adversary who has a full list of possible inputs could run an exhaustive scanning over the list to crack the input of a hash function.

$M_1 \| M_2$	Concatenation of messages M_1 and M_2
K_u^+	Public key of user u
K_u^-	Private key of user u
$E^K(M)$	Encryption of message M with key K
$H(M)$	One-way hashing of message M
$C(M,r)$	Commitment to message M with nonce r

C. List of notations

Distance Bounding: A location proof system needs a prover to be securely localized by the party who provides proofs. A distance bounding protocol serves the purpose. A distance bounding protocol is used for a party to securely verify that another party is within a certain distance. Different types of distance bounding protocols have been studied and proposed. A most popular category is based on fast-bit-exchange : one party sends a challenge bit and another party replies with a response bit and vice versa. By measuring the round-trip time between the challenge and the response, an upper bound on the distance between the two parties can be calculated. This fast-bit-exchange phase is usually repeated a number of times.

D. Protocol

Overview: Our protocol consists of two primary phases: STP proof generation and STP claim and verification. When a prover collects STP proofs from his/her co-located mobile devices, we say an STP proof collection event is started by the prover. An STP proof generation phase is the process of the prover getting an STP proof from one witness. Therefore, an STP proof collection event may consist of multiple STP proof generations. The prover finally stores the STP proofs he/she collected in the mobile device. When a prover encounters a verifier (the frequency of such encounters is specific to the application scenarios) and he/she intends to make a claim about his/her past STP to the verifier, the STP claim and verification phase takes place between the prover and the verifier. A part of the verification job has to be done by CA. Therefore, communication between the verifier and CA happens in the middle of the STP claim and verification phase.

E. STP Proof Generation

Prover: Suppose a prover wants to start an STP proof collection event at time t, the prover first broadcasts an

STP proof request to other nearby mobile devices and waits for responses.A PReq is constructed as follows:

$$PReq = C(ID_p, r_p)|L_1|t$$

Where IDp is the prover's ID,rp is a random nonce generated by the prover for the commitment to IDp , and L1 is the lowest level of the current location.

III. CONCLUSION

In ths paper, the main concept is providing the security for gurantee to mobile user proofs for their locations. STAMP stands for Spatial-Temporal provenance Assurance with Mutual Proofs. It uses wireless APs to generate location proofs. Transferability and non-transferability of location proofs are the main designs of STAMP. To detect P-W collusion, we proposed an entropy-based trust model to evaluate the trust level of claims of the past location visits.

IV. REFERENCES

[1]. J. Shi, R. Zhang, Y. Liu, and Y. Zhang, "PriSense: Privacy-preserving data aggregation in people-centric urban sensing systems," in Proceedings of the 30th IEEE International Conference on Computer Communications (InfoCom '10), San Diego, CA, March 2010.

[2]. M. Langheinrich, "Privacy in uniquitous computing," in Ubiquitous Computing, J. Krumm, Ed. Chapman & Hall, CRC Press, 2009.

[3]. M. Gruteser and B. Hoh, "On the anonymity of periodic location samples," in Proceeding of the 2nd International Conference on Security in Pervasive Computing, Boppard, Germany, April 2005, pp. 179–192.

[4]. A. Kapadia, D. Kotz, and N. Triandopoulos, "Opportunistic sensing: Security challenges for the new paradigm," in Proceedings of the First International Conference on Communication Systems and Netwprls (COMSNETS '09), Bangalore, India, January 2009, pp. 1–10.

[5]. S. Spiekermann and L. Cranor, "Engineering privacy," IEEE Transactions on Software Engineering, vol. 35, no. 1, January 2009.

[6]. K. Shilton, "Four billion little brothers?: Privacy, mobile phones, and ubiquitous data collection," Communications of the ACM, vol. 52, no. 11, pp. 48–53, 2009

[7]. 3. G. Avoine and A. Tchamkerten. An efficient distance bounding RFID authentication protocol: Balancing false-acceptance rate and memory requirement. In Proceedings of Information Security, volume 5735, pages 250–261, 2009.

[8]. A. Bay, I. C. Boureanu, A. Mitrokotsa, I.-D. Spulber, and S. Vaudenay. The Bussard-Bagga and Other Distance-Bounding Protocols under Attacks. In the 88th China International Conference on Information Security and Cryptology (Inscrypt 2012), 2012.

[9]. M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In Proceedings of the 1st ACM conference on Computer and communications security, CCS '93, pages 62–73, New York, NY, USA, 1993. ACM.

[10]. I. Boureanu, A. Mitrokotsa, and S. Vaudenay. On the Pseudorandom Function Assumption in (Secure) Distance-Bounding Protocols. In A. Hevia and G. Neven, editors, Progress in Cryptology – LATINCRYPT 2012, Lecture Notes in Computer Science, pages 100– 120. Springer, 2012.

[11]. L. Bussard and W. Bagga. Distance-bounding proof of knowledge to avoid real-time attacks. In Security and Privacy in the Age of Ubiquitous Computing, IFIP TC11 20th International Conference on Information Security (SEC 2005), May 30 - June 1, 2005, Chiba, Japan, pages 223–238. Springer, 2005.

[12]. N. Chandran, V. Goyal, R. Moriarty, and R. Ostrovsky. Position based cryptography. In S. Halevi, editor, Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings, volume 5677 of Lecture Notes in Computer Science, pages 391–407. Springer, 2009.

[13]. H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. The Annals of Mathematical Statistics, 23(4):493–507, 1952.

[14]. C. Cremers, K. B. Rasmussen, and S. Capkun. Distance hijacking attacks on distance bounding protocols. In ˇ IEEE Symposium on Security and Privacy, pages 113–127, 2012.

[15]. G. Kapoor, W. Zhou, and S. Piramuthu. Distance bounding protocol for multiple RFID tag authentication. In C.-Z. Xu and M. Guo, editors,

Proceedings of the 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing - Volume 02 – EUC'08, pages 115–120, Shanghai, China, December 2008. IEEE Computer Society.

[16]. C. H. Kim and G. Avoine. RFID distance bounding protocol with mixed challenges to prevent relay attacks. In Proceedings of the 8th International Conference on Cryptology and Networks Security (CANS 2009), volume 5888, pages 119–131, 2009.

[17]. C. H. Kim, G. Avoine, F. Koeune, F. Standaert, and O. Pereira. The swiss-knife RFID distance bounding protocol. In International Conference on Information Security and Cryptology – ICISC, Lecture Notes in Computer Science. Springer-Verlag, December 2008.

[18]. J. Munilla and A. Peinado. Distance bounding protocols for RFID enhanced by using void-challenges and analysis in noisy channels. Wireless Communications and Mobile Computing, 8:1227–1232, November 2008

[19]. J. Munilla and A. Peinado. Security Analysis of Tu and Piramuthu's Protocol. In New Technologies, Mobility and Security – NTMS'08, pages 1–5, Tangier, Morocco, November 2008. IEEE Computer Society. [20] J. Munilla and A. Peinado. Attacks on a Distance Bounding Protocol. Computer Communications, 33:884–889, 2010.

[20]. K. B. Rasmussen and S. Capkun. Realization of RF distance bounding. In ˇ Proceedings of the 19th USENIX conference on Security, USENIX Security'10, pages 25–25, Berkeley, CA, USA, 2010. USENIX Association.