Survey on location authentication protocols and spatial-temporal attestation services

A.I. González-Tablas[1], K. Kursawe[2] and B. Ramos[1]

[1] Universidad Carlos III, Madrid (Spain)
Computer Science Department - SeTI
aigonzal,benja1@inf.uc3m.es

[2] Katholieke Universiteit Leuven,
Department Electrical Engineering - ESAT, COSIC
klaus.kursawe@esat.kuleuven.be

Abstract. A survey on location authentication protocols and spatial-temporal attestation services is presented. Several protocols and services with these objectives have been proposed during the last decade, but still there is a lack of understanding of the security properties they should provide and which security mechanisms are appropriate. We first define the goals and threat model of location authentication protocols, next they are described and analyzed against this model. Also, spatial-temporal attestation services are described and classified depending on their goal and kind of issued evidence.

1 Introduction

The development of location technologies and the increasing mobility of our communications have allowed the deployment of Location Based Services (LBS). In this context some applications will benefit of authenticating the location of certain entity (*location authentication*) while others would prefer to obtain an evidence about the spatial-temporal conditions of certain entity or document (*spatial-temporal attestation*). For example, a service provider may require that in order to grant access to a service, their clients must be located at some specific set of locations, or a shopping center may desire to grant privileges depending on the visiting history to the center. In another context, spatial-temporal attestation services can be used to notarize from where some data is being sent, where some document is signed or where a certain payment is done. Another application is to provide accountability to the tracking of entities or assets. During the last 10 years several location authentication protocols and spatial-temporal attestation services have been proposed. Still there is a lack of understanding of the security properties they should provide and which security mechanisms are appropriate. In this paper these issues are addressed: a comprehensive survey on these protocols and services is presented and at the same time its security is analyzed.

2 Location authentication protocols

2.1 Definitions, assumptions and threat model

The general setting for a location authentication protocol involves a *prover* (P) and a *verifier* of the location (V_{loc}). P is an entity which has some means for being located by a *positioning infrastructure* PI (see [HB01] for a survey on location systems) and that we assume to have an unique identification p. The verifier V_{loc} is presented with, or presumes beforehand, the purported location of the prover. Then, **location authentication** is defined as *the process whereby one party (V_{loc}) is assured (through acquisition of corroborative evidence) of the location of a second party (P) in a protocol, and that the second party has actually participated in the protocol (i.e., is active at, or immediately prior to, the time the evidence is acquired).* A set of *locating entities* LE, which are part of the positioning infrastructure, may collaborate with V_{loc} to authenticate the prover's location. V_{loc} may be also part of the positioning infrastructure. We define that a **location authentication protocol is secure** if *in all its executions run with an adversary $\mathcal{A}$, V_{loc} accepts the claim that the prover p is in location l at time t iff this statement is true.* The goal of an adversary $\mathcal{A}$ is that V_{loc} accepts claims on target tuples $\tau_t = (p_t, l_t, t_t)$ such that some or several of the elements of the tuple makes the statement 'p_t was in location l_t at time t_t' false.

We assume that provers are physical devices which know a secret s that allows to prove its identity p to other entities. This secret s is stored in a tamper-resistant module such that all the operations that use s are done inside this module and s cannot be leaked. However, the adversary $\mathcal{A}$ can manipulate other P's physical characteristics in order to subvert the protocol. Authenticating P's location does not provide guarantees about who is the user U that may be controlling it. Although some mechanisms to authenticate the proximity of the user to the device can be used, such as protecting s with some other secret known by the user, we assume that both are bound to each other.

We assume that the adversary $\mathcal{A}$ has under his control a set of compromised provers $\mathcal{P}^* \subset \mathcal{P}$ where $\mathcal{P}^* = \{p_1^*, \ldots, p_n^*\}$. The adversary can place these compromised provers in any location $l \in \mathcal{L}$ chosen by the adversary at any time $t \in \mathcal{T}$ and make them to execute a location authentication protocol with V_{loc}, to communicate securely between them using radio, sound or other mediums, or to capture, intercept or insert any message. Once an execution of a protocol has started, the adversary cannot move the compromised provers arbitrarily at his will if this movement is against the physic laws, but he can force them to not follow the steps of the protocol or to claim a different identity. The adversary can also record executions of the protocol run by provers under his control or by other provers, and use this information in later executions. The adversary may also want to know the whereabouts of provers which are not under his control, that is, $\mathcal{A}$ would like to attempt against the privacy of provers $p \notin P^*$. We are not going to analyze the protocols against this threat. In the same way, we will not consider denial of service attacks, even when it is very easy to run them successfully in the depicted scenario (e.g. jamming).

A) **Initialization.**
   1. V_{loc} generates uniformly at random k bits α_i.
   2. P generates uniformly at random k bits m_i.

B) **Commitment.**
   1. P commits to the k bits m_i using a secure commitment scheme protocol.

C) **Fast exchange.** This phase is run repeatedly k times for $i = 1 \ldots k$:
   1. V_{loc} starts a timer.
   2. $V_{loc} \rightarrow P : \alpha_i$
   3. $P \rightarrow V_{loc} : \beta_i = \alpha_i \oplus m_i$ (immediately after it receives α_i)
   4. V_{loc} stops the timer and measures the latency time λ_i.

D) **Commitment opening.**
   1. P opens its commitments on bits m_i to V_{loc}.

E) **Authentication and verification.**
   1. P builds $m = \alpha_1|\beta_1|...|\alpha_k|\beta_k$, signs this value m and sends the result to V_{loc}.
   2. V_{loc} verifies if the committed bits in step $B.1$ are the same as $\alpha_i \oplus \beta_i$. If this holds, V_{loc} computes m as P would have done and verifies P's signature on m. If this also holds, V_{loc} computes an upper-bound on the distance using the maximum of the measured latency times $max(\lambda_i)$ with $i = 1, ..., k$, and accepts if and only if P is close by.

Fig. 1. Brands-Chaum distance-bounding protocol [BC94]

2.2 Distance-bounding protocols

The goal of these protocols is to authenticate that the prover P is within some distance d_{lim} from some location l_0 where a locating entity LE or a verifier V_{loc} is placed. Without losing generality, the set of locations $\mathcal{L}_t$ that the adversary may target is defined as $\mathcal{L}_t = \{l_t : l_t \in \mathcal{L}, d(l_0, l_t) \leq d_{lim}\}$, where $d(\cdot, \cdot)$ is a function that returns the distance between two locations. Following we describe and analyze the distance-bounding protocols that currently exist in the literature.

Based on fast challenge-response exchanges Some distance-bounding protocols are based on the measurement of the round trip latency λ between P and V_{loc}. They are designed as interactive two-party protocols and the main assumption is that the signals used to transmit the exchanged messages have a constant propagation speed v, where $v = v_c \cong 3 \times 10^8 m/s$ if radio or optical signals are used and $v = v_s \cong 340 m/s$ if sound. The round trip latency is defined as $\lambda = t_{pp}(l_0, f(P, t_{run})) + t_{pc}(P) + t_{pp}(f(P, t_{run}), l_0)$, where $t_{pp}(l_1, l_2)$ is the propagation time between location l_1 and location l_2, $t_{pc}(A)$ the processing time of an entity A between the reception of a challenge and the transmission of its response and $f(p, t)$ returns the location of prover p at time t.

Brands and Chaum were the first that proposed a protocol falling in this category in [BC94] (see Figure 1). Their protocol and the ones in [ČBH03,Bus04] assume that the device has some hardware that performs the exchange in a fast manner over some dedicated communication channel. Then, they assume that the prover's processing time is negligible compared to the propagation time and an upper-bound of the distance between V_{loc} and P can be computed as $\delta = v \times \lambda/2 \geq d(l_0, f(p, t_{run}))$. Other proposals in [SSW03,WF03] assume that the device has a non-zero processing delay. Then, the upper-bound is calculated as $\delta = v \times (\lambda - t_{pc}(P))/2$. In [SSW03], responses are sent using sound while challenges use radio signals, then $\delta \cong v \times (\lambda - t_{pc}(P))$.

Assuming that the adversary $\mathcal{A}$ controls a single prover p_i^* such that $f(p_i^*, t_t) = l_i^* \in \mathcal{L}_t$, then $\mathcal{A}$ may try to impersonate some prover $p_t = p_j \neq p_i^*$ (*imperson-*

ation attack). This would be possible if provers are not authenticated at any time during the execution of the protocol. Most of the distance-bounding protocols based on fast exchanges authenticate provers. On the contrary, in the protocol in [SSW03] provers are not authenticated (it is not considered a goal), therefore the impersonation attack does not make any sense. The protocol in [WF03] does not authenticate the prover, but the whole spatial-temporal certification protocol which uses it in a phase does, then preventing this attack.

With a single compromised prover $p_t = p_i^*$ such that $l_i^* \notin \mathcal{L}_t$, $\mathcal{A}$ may try to decrease the measured latency λ (*decreasing measured latency attack*) with respect to the one that should have been measured (note that trying to increase λ will not help $\mathcal{A}$ to get the claim accepted). First, $\mathcal{A}$ may try to send the response in advance of receiving the challenge from V_{loc}. To avoid this, the response in this kind of protocols is chosen such as it depends on the challenge and a value which P commits to previously, as in the protocol in Figure 1. If it can be assumed that the propagation speed of the signals used to exchange the messages has an upper-bound which no prover, including those controlled by $\mathcal{A}$, can exceed, then the probability for $\mathcal{A}$ guessing a response $r \in \{0,1\}^m$, and succeeding in the attack, is $1/2^m$. To increase the security of the protocol, several exchanges can be done. In the case of the proposal in [SSW03] the previous assumption does not hold, because the response is transmitted using sound. Then $\mathcal{A}$ may try to decrease λ by using a faster signal in some part of the trajectory.

In protocols in [SSW03,WF03] a non-zero processing time is assumed, then $\mathcal{A}$ may try to decrease measured latency λ by decreasing this time. To avoid this, in [WF03] it is proposed that this time is known by the verifiers and it is assumed that $\mathcal{A}$ cannot manipulate it. On the contrary, in [SSW03] it is assumed that $\mathcal{A}$ may tamper this time; an effective countermeasure is proposed based on decreasing d_{lim} dynamically depending on the declared processing time according to $d_{lim}(t_{pc}(P)) = d_{lim}(0) - t_{pc}(P) \times v$.

If $\mathcal{A}$ controls a single prover p_i^*, he may try the attack referred as *mafia fraud* in [BC94,Bus04] or *proxy attack* in [WF03]. Prover p_i^* impersonates V_{loc} in order that $p_t = p_k \notin \mathcal{P}^*$ run the protocol with p_i^* instead of with V_{loc}. It is assumed that $d(l_0, f(p_k, t_t)) > d_{lim}$ and $d(l_0, l_i^*) \leq d_{lim}$. The protocols in [BC94,WF03,ČBH03,Bus04] prevent this attack as the distance between p_k and p_i^* makes λ increase and V_{loc} will not accept the claim (assuming that the propagation speed has an upper-bound which cannot be exceeded). The protocol in [SSW03] would prevent this attack if $\mathcal{A}$ could not use signals which propagate faster than sound, but this is not assumed in the protocol. Anyhow, as in [SSW03] anyone can impersonate other provers, the proxy attack does not make any sense.

When the adversary controls at least two provers p_i^* and p_j^*, which is a reasonable scenario, then the attack referred as *collaborator attack* in [BC94] or *terrorist attack* in [Bus04] can take place. Then the target tuple $\tau_t = (p_i^*, l_t, t_t)$ is such that $l_j^* \in \mathcal{L}_t$ but $l_i^* \notin \mathcal{L}_t$. If the fast exchange phase is not bound to the identity of the entity who executes the protocol, it can be done by a different one. For example, in protocol in Figure 1 p_j^* may sit between p_i^* and V_{loc} and act as a transparent proxy between them in all the phases except in the Phase

1. $P \rightarrow V_{loc}$: P, R, L
2. $V_{loc} \rightarrow LE$: $K_s\{P, N\}$, $K_{V_{loc},LE}\{K_s\}$
3. $LE \overset{rc_\lambda L}{\rightarrow} P$: P, N
4. $P \rightarrow V_{loc}$: P, R, N
5. V_{loc} verifies that the token N received in step 4 is the same as the one it sent to LE in step 2.

Fig. 2. Kindberg-Zhang distance-bounding protocol [KZ01b]

C, which p_i^* would execute by itself given that p_i^* has communicated p_j^* the bits m_i. Protocols in [BC94,WF03,SSW03,ČBH03] are vulnerable to this attack (however note that this attack does not make any sense in [SSW03] again). The proposal in [Bus04] solves this problem by binding the secret s to the fast exchange phase. In Bussard's protocol the response depends also on s in such a way that this dependency can be proved without revealing it by using proof of knowledge protocols. This protocol is secure (with some probability) if it can be assumed that the signals' speed has an upper-bound that cannot be exceeded.

Based on token broadcast Other distance-bounding protocols are based on broadcasting some token N through a set of short-range beacons playing the role of LE. Protocols proposed in [KZ01b,Mic03] are of this kind. In this setting it is assumed that the token can only be received if $d(l_0, f(p, t_{run})) < d_{lim}$, d_{lim} determining the end of LE's transmission range. Then, knowing N is assumed to be a proof of having been close to LE. As Kindberg and Zhang discuss in [KZ01b] this assumption can be reasonably held in certain scenarios (e.g., if infrared or ultrasound signals are used and the region is delimited with walls).

If the adversary controls a single prover p_i^* such that $d(l_0, l_i^*) > d_{lim}$, he may try to guess N (*guessing attack*). To prevent this attack, tokens should be random nonces, and at the same time should depend on the area and the broadcast time to prevent *reuse attacks*. Protocols in [KZ01b,Mic03] prevent these attacks.

As in the previous setting, $\mathcal{A}$ may try to perform *impersonation attacks*, to prevent this attack, some kind of prover authentication is needed. However, protocols falling in this category do not agree with this approach. Protocol in [Mic03] does not authenticate provers during execution (one of its main goals is prover anonymity). Kindberg and Zhang in [KZ01b] claim that entity authentication or anonymity issues are orthogonal to the location authentication problem, and therefore they do not consider this issue in their protocols (see protocol in Figure 2). As in our model impersonation attacks are relevant, prover authentication should be required.

Proxy attacks may also be run in this setting. In a first version $\mathcal{A}$ will target a tuple $\tau_t = (p_i^*, l_t, t_t)$ such that $d(l_0, l_i^*) > d_{lim}$. The attack involves a prover $p_k \notin \mathcal{P}^*$ such that $d(l_0, l_k) \leq d_{lim}$ and p_i^* sits between p_k and V_{loc}, acting as a transparent proxy between them and playing the role of V_{loc} to p_k. The protocol in [Mic03] is vulnerable to this attack as it does not authenticate provers. Protocol in [KZ01b] could prevent this attack if V_{loc} authenticated provers and kept a registry binding broadcast tokens with specific prover requests, or if this binding were done within the token and its authenticity preserved (assuming that honest provers would not accept tokens not addressed to them). A second

version of the proxy attack is that one where the prover p_i^* sits between LE and p_k, acting again as a transparent rely between them. $\mathcal{A}$ targets the tuple $\tau_t = (p_k, l_t, t_t)$ where $d(l_0, l_i^*) \leq d_{lim}$ but $d(l_0, l_k) > d_{lim}$. This attack would not be detected even if tokens were bound to provers in an authentic manner. A possible countermeasure suggested in [KZ01b] is that V_{loc} measures the response time to verify if it corresponds to the expected distance between V_{loc} and p_k; then similar techniques to the ones presented in the previous section will be applied. Another countermeasure might be that LE used unforgeable RFID schemes and that tokens were bound to each detected prover.

If $\mathcal{A}$ controls at least two provers p_i^* and p_j^* then *collaborator attacks* may take place. Target tuple would be $\tau_t = (p_i^*, l_t, t_t)$ where $d(l_0, l_i^*) > d_{lim}$ and $d(l_0, l_j^*) \leq d_{lim}$. All the considerations made before for proxy attacks are relevant, but in this case p_j^* will collaborate in the attack (e.g. accepting tokens not addressed to it). Again, a possible countermeasure would be that LE authenticated provers in the acceptance area and that tokens were bound to them.

2.3 Absolute positioning protocols

The goal of these protocols is to authenticate P's absolute position with some resolution. These protocols usually rely on triangulation techniques. If the target prover is $p_t = p_i^*$ such that $f(p_i^*, t_t) = l_i^*$, then $\mathcal{L}_t = \{l_t : l_t \in \mathcal{L}, l_t \neq l_i^*\}$. Following the two kind of protocols falling in this category are described.

Based on simultaneous fast challenge-response exchanges As these protocols are designed as the simultaneous execution of several distance-bounding protocols based on fast exchanges run by P and several LE, then, the analysis presented in the previous section for distance-bounding protocols based on fast exchanges can be applied to this setting. Given this, let's assume that $\mathcal{A}$ controls one single prover p_i^*, that the speed of the exchanged signals cannot be exceeded by any prover and that some countermeasures have been applied to prevent manipulation of processing times if devices are assumed to have any. Then $\mathcal{A}$ may try to prove being in another location $l_t \neq l_i^*$ by delaying prover's answers. Čapkun and Hubaux prove in [ČH04] that if the prover lies within the triangle with vertices each LE, it cannot prove successfully being at another location than where it actually is. A prover can always prove to be further from one of the LE but then, if it lies within the mentioned triangle, it must prove to be closer to at least other of the LE, which is not possible under the assumptions.

Based on authenticated ranging Some location authentication protocols are based on signals broadcast by global navigation satellite systems (GNSS) such as GPS. In these systems several satellites orbiting around the earth transmit continuously signals L_i; satellites play the role of LE. The positioning principle is based on measuring the time of flight from a satellite LE_i to the prover P, which allows to compute their range or distance; several ranges can be used to calculate P's absolute position by triangulation. This method needs that

satellite and receiver clocks are synchronized, but usually there exists some bias or offset in the receiver clock respect to system time (satellite clocks are much more stable and precise); therefore to calculate the prover's position (latitude, longitude, height) the bias must be solved and at least four measurements are needed. In this section it is assumed that provers are GNSS receivers with added functionalities such as communication capabilities.

A first approach to authenticate P's location at time t would be that the device calculated its position $f(p,t)$ using the received navigation signals and sent a spatial-temporal report containing the tuple $(p, f(p,t), t)$ to V_{loc}. If these reports are not protected, $\mathcal{A}$ can intercept them and send faked ones instead (*report manipulation attack*). To avoid this, message authentication should be provided as it is suggested in [GW99,PWK04].

Even if reports were authenticated, $\mathcal{A}$ might try to manipulate provers in order to transmit false reports (*device manipulation attack*). If $\mathcal{A}$ controls one prover p_i^* located in l_i^* at time t_i^*, he may force p_i^* to send forged reports $\tau_t = (p_t, l_t, t_i)$ such that $p_t \neq p_i^*$, $l_t \neq l_i^*$, $t_t \neq t_i^*$ (if reports can be sent at a later time $t_j > t_i^*$) or a combination of these. To avoid this threat in [PWK04] it is proposed to use tamper resistant receivers such that they only output authenticated spatial-temporal reports calculated with received navigation signals, and which can check its integrity status and send reports on it.

Anyway, even if these assumptions can be held, $\mathcal{A}$ does not need to tamper provers to make them generate false reports. This is possible because satellite signals can be easily synthesized or manipulated with the appropriate software and fed to the device (*signal manipulation attack*). Anyhow, the price of these simulators or its hiring is high and in several applications it may not be worth compared to the benefit that $\mathcal{A}$ may obtain. To avoid these attacks the authentication of the broadcast signals should be guaranteed. One approach is to use symmetric encryption, as in one of the GPS signals where spreading code encryption with a symmetric secret key is used. Other approach considers that satellites broadcast some unpredictable information which would be recorded by P and forwarded to V_{loc}. This approach is somehow used in [MMZ+97], where small errors such as satellite orbit errors and ionospheric errors are used as unpredictable information. However, Kuhn points out in [Kuh04] that with this mechanism anyone who were able to verify the correction of the unpredictable information could also spoof the signal by including this information on a synthesized signal or transforming the signal according to it; further research should be done to check if this kind of attack could be detected in this case. A last approach to provide authentication to broadcast signals is based on asymmetric cryptography. For example, the proposal by Kuhn in [Kuh04] uses digital signatures to provide protection against signal synthesis attacks and also selective delay attacks; in this case undetectable hidden markers are inserted in the signal at unpredictable times and, after some time, signed information that allows markers verification is broadcast.

$\mathcal{A}$ may try to run in this setting both variants of *proxy attacks*. The first version (where $l_i^* \neq l_t$) is not possible in [PWK04] as devices are assumed to

output authenticated reports, but the second version of the attack would make the device to calculate a wrong position unless it could detect that the signals had been forwarded. In [MMZ+97] the first version of the attack might be prevented if latency measurements or similar countermeasures were carried out, because reports are apparently not bound to a specific receiver. The second version of the attack would not easily succeed as devices must prove to be at a fixed set of positions, the forwarding of the signals and its feeding to the device will make V_{loc} to fail in the calculation of its position with a high probability. This last situation would happen also in [MMZ+97] if $\mathcal{A}$ tried to carry out a *collaborator attack*, which would not be possible in [PWK04] as trusted devices are assumed.

3 Spatial-temporal attestation services

Similar to the definition for non-repudiation services in [ISO97], we define **spatial-temporal attestation services** as *those services that generate, collect, maintain, make available and validate evidences concerning either the spatial-temporal conditions of an entity either of the spatial-temporal conditions under which a transformation or action is made by some entity on certain data.* A trusted third party (TTP), the *spatial-temporal evidence generator* (G_e), is in charge of generating the evidences, and probably also collects, maintains and makes them available. Another TTP may exist, the *spatial-temporal evidence verifier* (V_e), if the evidences cannot be verified by any party by itself. We assume that the generator of the evidence G_e, before certifying the spatial-temporal conditions of the *subject* of the evidence, delegates the verification of these conditions to some entity V_{loc}, which should execute a location authentication protocol.

Assuming that spatial-temporal attestation services rely on secure location authentication protocols, the goal of a spatial-temporal attestation service is to provide unforgeable, non-transferable and verifiable spatial-temporal evidences on tuples $\tau = (p, l, t)$ such that it is true that the subject p was in location l at time t. The goal of an adversary $\mathcal{A}$ is to obtain evidences on tuples $\tau_t = (p_t, l_t, t_t)$ such that some or several of the elements of the tuple makes the statement 'p_t was in location l_t at time t_t' not true.

Following, a classification of spatial-temporal attestation services is presented depending on their specific goal and which kind of evidence they issue. Most of the spatial-temporal attestation services existing in the literature use well known evidence generation mechanisms such as digital signatures, secure seals or authenticator tokens. Therefore, a security analysis as the one developed in Section 2 is not presented in this case.

Spatial-temporal certification services. A first kind of spatial-temporal attestation services are those that have as main goal to provide evidences on the spatial-temporal conditions of a subject. A first group between these services provide evidences that may be certificate-like or credential-like. The proposals in [ZKK01,WF03] fall within the certificate-like category while the one proposed in [Bus04] can be classified as credential-like.

Other authors in [GTRR03] suggest to link certificate-like spatial-temporal evidences, as it is done in linked time-stamps schemes, to provide accountability to the temporal order of the evidences. Some of the protocols presented in [ČBH03] also provide some kind of spatial-temporal evidence but the location is not explicitly included in the evidence, they are more like temporal authenticators of the encounters between entities than proper spatial-temporal evidences.

A second group between spatial-temporal certification services provides ticket-like evidences or short-term credentials, such as the protocol in [Mic03]. Another protocol that falls into this approach is presented in [NNT03], but in this case the ticket, which is more similar to an authenticator than to a proper credential, can be used only a limited number of times.

Spatial-temporal stamping services. A second kind of spatial-temporal attestation services are those that have as main goal to provide evidences about the spatial-temporal conditions under which some document exist or a transformation is made by some subject on this document. In this case, G_e issues spatial-temporal stamps, which bind the document or its transformation with the spatial-temporal conditions. One of the more interesting transformations is to sign some data, which can be useful for example if some payment is done or some contract or attestation is signed. The only proposals that really fall under this approach are the ones presented in [KZ01a,LSBP03].

4 Conclusions and open issues

The expectations raised by recently proposed location authentication protocols and spatial-temporal attestation services are very promising. Although several protocols and services with these objectives have been proposed in the last decade, there is a lack of a framework that comprises them and that helps to analyze its security. In this paper we have surveyed existing location authentication protocols and spatial-temporal attestation services, their goals have been stated and its security has been analyzed against a proposed threat model.

There are still some open issues that should be further studied such as to analyze the efficiency of the protocols and services, the privacy they provide or how they may defend against denial of service attacks. The results of this work may be applied to analyze the security of the standardized positioning techniques in the context of mobile telephone networks.

5 Acknowledgment

The authors would like to thank the anonymous referees for their useful comments and suggestions. First author wants to thank Karel Wouters from K.U. Leuven for sharing several discussions about the collaborator scenario. First and third authors are partly supported by "Dirección General de Investigación del M.E.C." under contract SEG2004-02604: *'CERTILOC: Digital CERTIfication service for LOCation information'*.

References

BC94. S. Brands and D. Chaum. Distance-bounding protocols. In *Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, pages 344–359. Springer-Verlag New York, Inc., 1994.

Bus04. L. Bussard. *Trust Establishment Protocols for Communicating Devices*. PhD thesis, Institut Eurécom, Télécom Paris, 2004.

ČBH03. S. Čapkun, L. Buttyán, and J. P. Hubaux. SECTOR: Secure Tracking of Node Encounters in Multi-hop Wireless Networks. In *1st ACM Workshop on Security in Ad Hoc and Sensor Networks*, October 31, 2003.

ČH04. S. Čapkun and J. P. Hubaux. Securing position and distance verification in wireless networks. Technical Report EPFL/IC/200443, EPFL, May 2004.

GTRR03. A. I. González-Tablas, B. Ramos, and A. Ribagorda. Path-Stamps: A proposal for enhancing the security of location tracking applications. In *Ubiquitous Mobile Information and Collaboration Systems Workshop*, 2003.

GW99. E. Gabber and A. Wool. On location-restricted services. *IEEE Network*, November/December 1999), 1999.

HB01. J. Hightower and G. Borriello. A Survey and Taxonomy of Location Systems for Ubiquitous Computing. Technical Report UW-CSE 01-08-03, University of Washington, 2001.

ISO97. ISO/IEC 10181-4. Information technology - OSI - Security frameworks in open systems - Part 4: Non-repudiation framework, 1997.

Kuh04. M. Kuhn. An asymmetric security mechanism for navigation signals. In *6th Information and Hiding Workshop*, 23-25 May 2004.

KZ01a. M. Kabatnik and A. Zugenmaier. Location stamps for digital signature: A new service for mobile telephone networks. In *Networking - ICN 2001, First International Conference*, LNCS 2094. Springer, 2001.

KZ01b. T. Kindberg and K. Zhang. Context authentication using constrained channels. Report HPL-2001-84. Technical report, HP Labs Tech., 2001.

LSBP03. A. Lakshminarayanan, V. Singh, F. Bao, and K. P. Prabhu. Patent WO 03/007542. Method for certifying location stamping for wireless transactions, 2003. Publication date: 23/01/2003.

Mic03. N. Michalakis. Location aware access control for pervasive computing environments. Master's thesis, MIT, 2003.

MMZ+97. P. F. MacDoran, M. B. Mathews, F. A. Ziel, K. L. Gold, S. M. Anderson, M. A. Coffey, and D. E. Denning. Patent WO 97/13341. Method and Apparatus for Authenticating the Location of Remote Users of Network Computing Systems, 1997. Publication date: 10/04/1997.

NNT03. K. Nakanishi, J. Nakazawa, and H. Tokuda. LEXP: Preserving user privacy and certifying the location information. In *Proc. of the 2nd Workshop on Security in Ubiquitous Computing (UBICOMP 2003)*, October 2003.

PWK04. O. Pozzobon, C. Wullems, and K. Kubik. Secure tracking using Galileo services. In *Proc. of the 2004 Intl. Symposium on GNSS/GPS*, 2004.

SSW03. N. Sastry, U. Shankar, and D. Wagner. Secure verification of location claims. In *Proc. of the 2003 ACM workshop on Wireless security*. ACM Press, 2003.

WF03. B. R. Waters and E. W. Felten. Secure, Private Proofs of Location. TR-667-03. Technical report, Princenton, Computer Science, January 2003.

ZKK01. A. Zugenmaier, M. Kreutzer, and M. Kabatnik. Enhancing applications with approved location stamps. In *Proc. of IEEE Intelligent Network 2001 Workshop (IN2001)*, 2001.