NEXUS: Using Geo-fencing Services without revealing your Location

Michael Guldner, Torsten Spieldenner*, René Schubotz
German Research Center for Artificial Intelligence (DFKI)
(Saarbrücken Graduate School of Computer Science)*
66123 Saarbrücken, Germany
Email: {firstname.lastname}@dfki.de

***Abstract*—While becoming more and more present in our every day lives, services that operate on users' locations or location trajectories suffer from general fear of misappropriation of the transmitted location data. Several works have investigated of how to cope with this drawback. Respective systems claim location-privacy, i.e. keeping users' locations secret, by employing anonymisation techniques concerning a user's identity, or by obfuscating the transmitted location. These approaches lead to a degrade of quality-of-service and can be vulnerable to de-anonymisation attacks, or allow to learn at least the approximate location of a user. Focusing on the application domain of *geo-fencing*, we present as remedy a protocol that is based on homomorphic encryption of a user's location. The protocol provably provides full location-privacy by non-exposure of the users' location data, while producing exact geo-fencing results. We provide a detailed definition of the protocol, show its applicability in an actual geo-fencing application, and show that the resulting system fulfills all security properties we see for a location-privacy preserving system.**

***Keywords*-location-based service; geo-fencing; data privacy; location privacy**

I. Introduction

Location-based services [1], [2] are information services accessible with mobile devices through the mobile network. They utilise the ability to access, store and analyse real-time geographic information from mobile devices, and, in turn, provide service offerings such as orientation and localisation, navigation, search, identification or checking [3].

Despite considerable user uptake, e.g., in real-time social systems [4], [5] or travel-related applications [6], [7], and promising potentials in commerce [8] or smart cities [9], [10], widespread adoption of location-based services is impeded by specific concerns regarding location privacy [11] and general fear of misappropriation [12].

These qualms hold in particular for location-tracking services [13], i.e. location-based services that capture or predict users' location trajectories. One prime example of location-tracking services is geo-fencing, a class of location-based services that trigger actions or fire events whenever mobile devices enter or exit virtual perimeters set up for geographical areas known as geo-fences.

Clearly, a straight-forward approach to avoid misappropriation of users' location data, are client-based approaches that perform all processing of location information locally on the mobile device. But not only can entirely client-based approaches come with several drawbacks [14], they moreover require mobile devices to constantly perform computations against geo-fences locally, using processing power and draining battery of the device.

This paper is therefore concerned with designing a protocol for network- and cloud-based privacy-preserving geo-fencing services that provably does not leak any location information to any party that is involved with the geo-fencing evaluation.

Towards this end, we present *NEXUS* (Non-Exposure User location privacy System), a protocol for multi-party geo-fencing evaluation that employs an asymmetric, homomorphic encryption scheme to satisfy non-exposure of the users location, non-exposure of the geo-fences, and computational correctness of the geo-fencing evaluation. The presented approach has been prototypically implemented, being efficient enough to be employed in actual geo-fencing applications.

The remainder of this paper is organised as follows: Section II discusses the related literature on privacy preservation in location-based services. System assumptions, an adversarial model, as well as properties that a location-privacy preserving system should fulfill, are described in Section III. Our NEXUS protocol for location privacy-preserving geo-fencing evaluation is presented in Section IV, along with an algorithm that, based on the NEXUS protocol, correctly evaluates rectangular-shaped geo-fences; the fulfillment of the previously stated location-privacy properties is discussed subsequently. We conclude with summary and future work in Section VII.

II. Related Work

The available literature on location privacy preservation techniques is vast, and can be categorised into four working principles. We briefly outline the underlying key ideas and refer the interested reader to representative works in each category.

Regulation: Legal frameworks [15] governing collection, processing and distribution of individuals' location information have been established in most nations. However, regulation itself cannot prevent invasion of privacy, often lags behind new technology and innovations, and might stifle location-aware applications.

Policy-based approaches [16], [17] address the definition of trust-based mechanisms for proscribing certain uses of location information. These mechanisms are often highly complex and of questionable practicality for highly dynamic location-aware environments. More importantly, policy systems rely on extratechnological pressures to ensure privacy policies are adhered to.

Anonymisation approaches rely on the notation of k-anonymity, i.e. a node is made indistinguishable from at least $k-1$ other nodes, to prevent location privacy invasion. Popular techniques for achieving k-anonymity are location cloaking [18], [19], controlled flooding [20] or obfuscation [21], to name a few. Pseudonymisation [22], a variation of anonymisation, assigns persistent but non-identifying pseudonyms to individuals. Anonymisation is by far the most popular approach to (location) privacy preservation, however, it presents a barrier to authentication and personalisation, deliberately degrades quality-of-service, and exhibits vulnerabilities to de-anonymisation [23].

Secure multi-party computations (SMPC) enables parties to jointly compute an arbitrary agreed function of their private inputs with the computation results guaranteed to be correct. Prior works investigate multi-party computational geometry [24], [25], [26], privacy-preserving proximity detection in a two-party setting [27], privacy-preserving algorithms for determining fair multi-party rendez-vouz points [28] or nearest-neighbor queries [20]. Although closest to our approach, we are not aware of secure multi-party protocols for privacy-preserving geo-fencing.

III. System, Adversary and Requirements

In the following, we model a geo-fencing system and a privacy-invading adversary. Ensuing, the privacy properties that our solution should satisfy are outlined.

System Model. We assume the existence of a set of mobile nodes $\mathcal{M}$, a geo-fencing service $\mathcal{G}$, and a set of nodes $\mathcal{S}$ subscribed to geo-fencing events generated by $\mathcal{G}$.

Each node $M_i \in \mathcal{M}$ may periodically publish its location information in the form $(M_i, \mathcal{E}(\mathbf{L}))$ to the geo-fencing service $\mathcal{G}$, where $\mathbf{L} \in \mathbb{R}^2$ specifies M_i's current location and $\mathcal{E}(\mathbf{L})$ denotes a representation of $\mathbf{L}$ suitable for our concerns.

The service $\mathcal{G}$ manages a number of geo-fences, each of which is a tuple of the form $(\mathbf{F}, \mathcal{S}' \subseteq S)$. We denote a geo-fence's boundary with $\mathbf{F}$, and restrict ourselves to rectangles on the $\mathbb{R}^2$ plane, thus $\mathbf{F} = \{A, B, C, D\}$. The set $\mathcal{S}'$ contains the nodes subscribed to the geo-fencing events for $(\mathbf{F}, \mathcal{S}' \subseteq S)$ as generated by $\mathcal{G}$.

Upon receiving a tuple $(M_i, \mathcal{E}(\mathbf{L}))$, $\mathcal{G}$ determines if M_i is inside or outside of any eligible geo-fence $(\mathbf{F}, \mathcal{S}')$ by evaluating

$$f(\mathcal{E}(\mathbf{L}), \mathbf{F}) = \begin{cases} 1, & \mathbf{L} \in \mathbf{F} \\ 0, & \mathbf{L} \notin \mathbf{F} \end{cases} \tag{1}$$

using solely $\mathcal{E}(\mathbf{L})$ and $\mathbf{F}$.

Based on $f(\mathcal{E}(\mathbf{L}), \mathbf{F})$, $\mathcal{G}$ issues notifications to the subscribers in $\mathcal{S}'$. Hence, a subscriber $\mathrm{s} \in \mathcal{S}'$ is informed about whether or not a mobile node $M_i \in \mathcal{M}$ is within the geo-fence to which s was subscribed, however, s has no knowledge about the particular geo-fence itself.

Adversary Model. Attacking the system as modeled above, the adversary primarily intends to break a mobile node's location privacy. That is, the adversary will try to systematically and secretly record any M_i's current or past location for later use.

We assume that mobile nodes in $\mathcal{M}$ publish their locations correctly, and that they cannot be physically tracked by the adversary. Secondly, we assume $\mathcal{G}$ and nodes in $\mathcal{S}$ to be semi-honest, i.e. these nodes run the protocol exactly as specified but may try to learn as much as possible about the locations of nodes in $\mathcal{M}$. Finally, we consider the adversary to be computationally bounded.

Privacy Properties. The properties that we intend to provide for a geo-fencing system in the presence of privacy-invading adversaries are given as follows:

(P1) **Location Non-Exposition**. No party must be must able to obtain M_i's current or past location from interaction or observation.

(P2) **Location-privacy Preservation.** No party must be able to learn or deduce M_i's current or past location from interaction or observation.

(P3) **Computational Correctness**. Semi-honest parties are guaranteed to produce the correct outputs.

(P4) **Network Assistedness.** Since completely client-oriented approaches present several drawbacks [14], *some* information about a mobile nodes' locations must be published to a remote party to not leave all crucial computations to the clients (in our case, the mobile nodes).

To the best of our knowledge, no previous study has investigated properties (P1) - (P4) in our context. As outlined before, regulation and policy-based systems rely on extratechnological pressures to ensure privacy; anonymisation-based approaches certainly violate (P1) and (P3) and exhibit vulnerabilities with respect to (P2); SMPC approaches do not yet address geo-fencing services at all.

IV. Protocol Specification and Implementation

Obviously, encryption seems to be a suitable way to represent $\mathbf{L}$ to $\mathcal{G}$. The chosen encryption scheme needs to provide certain characteristics to allow evaluation of f solely based on $\mathcal{E}(\mathbf{L})$.

For this, in the following, we first provide some basics about cryptography and homomorphic encryption. Ensuing, we introduce our Non-Exposure User location privacy System (NEXUS) protocol for geo-fencing services, based on a homomorphic public key encryption scheme. Finally, we provide an overview of our prototype implementation and detail on how to perform geo-fence containment tests within the constraints of our chosen encryption system.

Preliminaries. A conventional encryption scheme consists of the following functions:

- **Key generation function** generates a symmetric encryption and decryption key ek or an asymmetric [29] public-private key pair (pk, sk) based on some parameter.
- **Encryption function** $\mathcal{E}_k(x)$ outputs a ciphertext from plaintext x using a key k.
- **Decryption function** $\mathcal{D}_k(x)$ outputs a plaintext from ciphertext x using a key k.

For any plaintext x, one has $\mathcal{D}_{ek}(\mathcal{E}_{ek}(x)) = x$ for symmetric schemes and $\mathcal{D}_{sk}(\mathcal{E}_{pk}(x)) = x$ for asymmetric encryption schemes.

In addition, *homomorphic encryption* [30], [31] features characteristics

$$\mathcal{E}_k(m_1) \odot \mathcal{E}_k(m_2) = \mathcal{E}_k(m_1 \bullet m_2) \qquad (2)$$

In other words, a homomorphic encryption scheme enables specific computations on ciphertexts and generates encrypted results which, when decrypted, match the results of operations performed on the respective plaintexts.

We will in the following show that homomorphic encryption is thus suitable to evaluate f in Equation (1) solely by the provided $\mathcal{E}_k(\mathbf{L})$, as required. As homomorphic encryption requires to encrypt all operands with the same key (cf. Equation (2)), the chosen encryption scheme must be a public key encryption scheme, as otherwise, parties that need to encrypt values during the process could use the symmetric key to decrypt $\mathcal{E}_{ek}(\mathbf{L})$.

Protocol. Our protocol is based on a public key infrastructure (PKI) and a suitable homomorphic encryption scheme. We do not rely on the geo-fencing service $\mathcal{G}$ acting as a trusted third party (TTP), but rather on the collaboration of $\mathcal{G}$ and a certificate authority and evaluation service $\mathcal{A}$ taking the role to provide network assistedness. Hence, we extend our system model as indicated in Figure 1. The protocol then is as follows:

① The certificate authority $\mathcal{A}$ generates a public-private key pair (pk, sk) and shares pk on demand with every other party.

② Each M_i may periodically send information about its location $\mathbf{L}$ in the form $(M_i, \mathcal{E}_{pk}(\mathbf{L}))$ to $\mathcal{G}$.

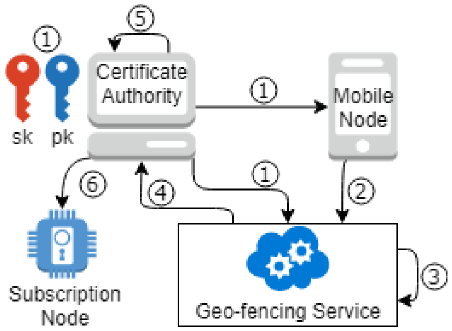

Figure 1. The actors and high level interactions in NEXUS.

③ $\mathcal{G}$ can not decrypt $\mathcal{E}_{pk}(\mathbf{L})$ or any results of homomorphic computations based on $\mathcal{E}_{pk}(\mathbf{L})$, but instead homorphically computes an (encrypted) intermediate result $\mathbf{R}$ as

$$\mathbf{R} = f_{\mathcal{G}}(\mathcal{E}_{pk}(\mathbf{L}), \mathbf{F}) \qquad (3)$$

with $f_{\mathcal{G}}$ being a function that homomorphically operates on $\mathcal{E}_{pk}(\mathbf{L})$, and $\mathbf{F}$ the boundary of a geo-fence $(\mathbf{F}, \mathcal{S}')$ as defined in Section III. $\mathcal{G}$ can encrypt $\mathbf{F}$ with pk for homomorphic operations, where necessary.

④ $\mathcal{G}$ passes $\mathbf{R}$, the ID of the mobile node M_i, and the set of subscription nodes $\mathcal{S}'$ that are to be invoked based on the evaluation result to $\mathcal{A}$.

⑤ $\mathcal{A}$ then evaluates $\mathbf{L} \in \mathbf{F}$ by an evaluation function $f_{\mathcal{A}}$, such that

$$\mathbf{L} \in \mathbf{F} \Leftrightarrow f_{\mathcal{A}}(\mathbf{R}) = \texttt{true}$$

$\mathcal{A}$ can decrypt $\mathbf{R}$ with sk, where necessary. By this, f as given in Equation (1) is computed as

$$f(\mathcal{E}_{pk}(\mathbf{L}), \mathbf{F}) = [f_{\mathcal{A}} \circ f_{\mathcal{G}}](\mathcal{E}_{pk}(\mathbf{L}), \mathbf{F})$$

⑥ $\mathcal{A}$ will invoke the respective subscriber $\mathbf{s} \in \mathcal{S}'$ with the result of the evaluation and the ID of the mobile node M_i.

Implementation. We base our implementation of above protocol on the *Paillier* encryption scheme. Implementation details concerning key generation, encryption, and decryption functions are found in [32]. Paillier features homomorphic addition:

$$\mathcal{E}_k(m_1) \oplus \mathcal{E}_k(m_2) = \mathcal{E}_k(m_1) * \mathcal{E}_k(m_2) = \mathcal{E}_k(m_1 + m_2), \qquad (4)$$

and pseudo-homomorphic multiplication of a ciphertext by an unsigned integer u:

$$\mathcal{E}_k(m)^u = \mathcal{E}_k(u * m), \qquad (5)$$

with $+$ and $*$ being arithmetic addition and multiplication. Moreover, using Equations (4) and (5), we can homomorphically subtract by adding a negated ciphertext.

$$\mathcal{E}_k(m_1) \ominus \mathcal{E}_k(m_2) = \mathcal{E}_k(m_1) \oplus \mathcal{E}_k(m_2)^{-1} = \mathcal{E}_k(m_1 - m_2) \qquad (6)$$

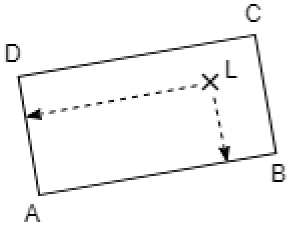

Figure 2. To check if a point is inside a rectangle, the point is projected perpendicular to the sides of the rectangle

In our implementation, every position is a two dimensional vector consisting of a longitude and latitude value, which we indicate by indices lon and lat. We determine if a position $\mathbf{L}$ is inside a rectangle by testing perpendicular projection (see Figure 2) as follows:

$$0 \leq \overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AB} \leq \overrightarrow{AB}^2 \wedge 0 \leq \overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AD} \leq \overrightarrow{AD}^2 \qquad (7)$$

Upon receiving a tuple $(M_i, \mathcal{E}_{pk}(\mathbf{L}))$, $\mathcal{G}$ computes for a rectangular geo-fence $\mathbf{F} = \{A, B, C, D\}$ the terms of Equation (7) as

$$\mathbf{R} = \{\overrightarrow{AB}^2, \overrightarrow{AD}^2, \mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AB}), \mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AD})\},$$

constituting $f_{\mathcal{G}}$, cf. Equation (3).

$\mathcal{G}$ has to calculate $\mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AB})$ and $\mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AD})$ homomorphically due to encrypted $\mathbf{L}$. For this, $\mathcal{G}$ encrypts $\mathbf{F}$ with the shared public key pk. One can show that with Equations (4), (5) and (6), definition of dot product, and component-wise computation, it holds that

$$\mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}}_{lat}) = \mathcal{E}_{pk}(\mathbf{L}_{lat}) \ominus \mathcal{E}_{pk}(A_{lat}),$$

similar for longitude components, and with that

$$\mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AB}) = \mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}}_{lat})^{\overrightarrow{AB}_{lat}} \oplus \mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}}_{lon})^{\overrightarrow{AB}_{lon}},$$

and similar for $\mathcal{E}_{pk}(\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AD})$.

The calculated values $\mathbf{R}$ are then sent to the certificate authority and evaluation service $\mathcal{A}$ along with the node ID M_i and the subscription nodes $\mathcal{S}'$ of the actions to trigger. $\mathcal{A}$ decrypts the encrypted results with the secret key sk, evaluates Equation (7) as $f_{\mathcal{A}}$, and invokes the according subscribers.

Performance evaluation. For our performance evaluation deployment, we implemented $\mathcal{G}$ and $\mathcal{A}$ as Python applications with Flask[1] as backend Web-server. Same holds for the prototype client that we used for testing. Parties communicate with HTTP requests. For the homomorphic cryptography, we used the n1analytics/python-paillier Python library[2]. On a workstation (Linux, i5-2520M at 2.50GHz, 16GB RAM), we measured evaluation times of an average 323 milliseconds (1000 executions, network lag not included) per evaluation of $f_{\mathcal{A}} \circ f_{\mathcal{G}}$ with a 2048-bit Paillier key. The measured time is in the ranges suitable for an execution that is perceived ”uninterrupted” by users [33]. Thus, we consider NEXUS practicable for actual application.

V. Discussion and Results

The goal must be, that by our architecture and protocol, the security properties (P1) to (P4) can never be violated by a semi-honest adversary.

Adversaries, in our presented architecture the geo-fencing service $\mathcal{G}$, certificate authority and evaluation service $\mathcal{A}$, and subscriptions $\mathcal{S}$, could try to obtain M_i's secret location $\mathbf{L}$ directly (violating (P1)), which would require them to be in possession of both the encrypted location $\mathcal{E}_{pk}(\mathbf{L})$ and the secret key sk to decrypt L.

Or they could try to approximate $\mathbf{L}$ by observing evaluation of geo-fences $(\mathbf{F}, \mathcal{S}')$. For this, they would either need to learn about both $\mathbf{F}$ and the outcome of $f(\mathcal{E}_{pk}(\mathbf{L}), \mathbf{F})$ (Equation (1)). Or they need to know about the set $\mathcal{S}' \subseteq \mathcal{S}$ that is registered to a geo-fence $(\mathbf{F}, \mathcal{S}')$. Observing then if any $\mathrm{s} \in \mathcal{S}'$ was invoked as a result of an evaluation would allow to deduct the evaluation outcome against a geo-fence boundary $\mathbf{F}$.

In the following, we show that by employing the protocol as described in Section IV, none of these attacks are possible, and by this, the security properties (P1) to (P4) as stated in Section III are always fulfilled:

Location Non-Exposition (P1): The geo-fencing service $\mathcal{G}$ only receives the encrypted location $\mathcal{E}_{pk}(\mathbf{L})$ from a mobile node M_i. $\mathcal{G}$ is not in possession of the private key sk to decrypt the data, nor does it need to decrypt any value by exploiting the homomorphic characteristics of the Paillier system as described in IV. $\mathcal{G}$ can thus at no point obtain $\mathbf{L}$ directly. The Authority service $\mathcal{A}$, though in possession of the private key sk, never receives any encrypted location $\mathcal{E}_{pk}(\mathbf{L})$ from neither M_i nor $\mathcal{G}$. Thus, $\mathcal{A}$ does at no point obtain information about $\mathbf{L}$. Subscribers in S receive neither sk nor the encrypted location $\mathcal{E}_{pk}(\mathbf{L})$. (P1) is by this always satisfied.

Location-privacy Preservation (P2): With $\mathcal{G}$ defining the geo-fences $(\mathbf{F}, \mathcal{S}')$, it is clearly in knowledge about both $\mathbf{F}$ and $\mathcal{S}'$. $\mathcal{G}$ is not in possession of sk and by this can not decrypt the evaluation result and invoke any of the subscriptions $\mathcal{S}'$ based on it. $\mathcal{G}$ can thus neither obtain, nor observe the result of an evaluation for a geo-fence $(\mathbf{F}, \mathcal{S}')$.

$\mathcal{G}$ does not disclose information about geo-fence boundaries $\mathbf{F}$ to any party, and in particular no information about to which geo-fence $(\mathbf{F}, \mathcal{S}')$ subscriptions $\mathrm{s} \in \mathcal{S}'$ are

[1] http://flask.pocoo.org/

[2] https://github.com/n1analytics/python-paillier

registered. $\mathcal{A}$ by this can never observe for which geo-fence $(\mathbf{F}, \mathcal{S}')$ it decrypted the evaluation result. To retrieve $\mathbf{L}$ from the decrypted values $\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AB}$ and $\overrightarrow{A\mathbf{L}} \cdot \overrightarrow{AD}$, $\mathcal{A}$ would need knowledge about the geo-fence boundary $\mathbf{F} = \{A, B, C, D\}$, which is never disclosed by $\mathcal{G}$.

Subscribers in $\mathrm{s} \in S$ receive the result of an evaluation, but do not have knowledge about the geo-fence boundary $\mathbf{F}$, as $\mathbf{F}$ is not disclosed by $\mathcal{G}$. By this, a subscriber can not observe the location of a mobile node by observing its own invocation.

None of the transmitted data or computational intermediate results are stored during the process. By this, (P2) is always fulfilled.

Computational Correctness (P3): None of the parties willingly provides wrong data during the process, as we assume all participants to be semi-honest. Correctness of computations during the evaluation is moreover ensured by the homomorphic characteristics of the chosen Paillier encryption scheme. We have shown the correct computation of a containment check of $\mathbf{L}$ against a rectangle based on the Paillier homomorphism in Section IV. (P3) is by this always fulfilled.

Network Assistedness (P4): Mobile nodes do not receive geo-fences from $\mathcal{G}$ and by this can not perform geo-fencing evaluation computations on their own. They need to provide *some* information to a third party, in this case, $\mathcal{E}_{pk}(\mathbf{L})$ to $\mathcal{G}$, which performs the evaluation with the help of $\mathcal{A}$. As shown above, information exchanged during this process is not sufficient for any other party to violate (P1) to (P3), and (P4) is always fulfilled.

VI. Future Work

We have in this paper limited ourselves to rectangular geo-fence shapes. For future work, we plan to investigate how to evaluate containment of a users' position in arbitrarily shaped geo-fences within the possible mathematical operations as imposed by the encryption schemes.

Geo-fences are defined independently. This allows for parallel evaluation of multiple geo-fences. We plan to do further research on highly scalable large-scale distributed setups of several instances of geo-fencing and evaluation services, employing capabilities of latest Infrastructure-as-a-Service and virtual container management systems.

We moreover see in the proposed solution a promising approach to realise general purpose validation tasks in distributed IoT environments, similar to works on homomorphic privacy preserving multi party computations as for example presented in [34]. In fact, while we focused on the geo location validation as use-case in this paper, we see promising application opportunities for every kind of numerical computations or validations, like for example factory process sanitary monitoring in an Industrie 4.0 scenario, general sensor analysis in a Smart City environment, and other.

VII. Conclusion

In this paper, we have presented NEXUS (Non Exposure User location privacy System), a novel protocol for geo-fencing systems that ensures location privacy based on non-location exposure by employing a homomorphic encryption scheme.

In the paper, we have presented the following contributions to the topic:

Based on a list of properties for location-privacy preserving geo-fencing schemes, we have developed a protocol for a geo-fencing system that, unlike existing approaches, does not rely on anonymisation or obfuscation of the user. Instead, it utilises characteristics of homomorphic encryption. This allows to perform the evaluation entirely on encrypted location data. We have shown that by our protocol, provably, the correct result of the geo-fencing evaluation can be computed. The location of the client is kept entirely secret to all parties involved with the evaluation process.

The presented protocol was implemented in a prototype application for rectangular geo-fences based on the Paillier encryption system [32]. The prototype performed the computation in time ranges that allow for an actual applicable system.

Acknowledgement

The work in this paper was funded by financial means of the German Federal Ministry for Economic Affairs and Energy (BMWi) in the project GUIDED AL under the support code 01MD16010C, and by the Federal Ministry of Education and Research of Germany in the project Hybr-iT under support code 01IS16026A.

References

[1] K. Virrantaus, J. Markkula, A. Garmash, V. Terziyan, J. Veijalainen, A. Katanosov, and H. Tirri, "Developing gis-supported location-based services," in *Web information systems engineering, 2001. Proceedings of the Second International Conference on*, vol. 2. IEEE, 2001, pp. 66–75.

[2] J. Schiller and A. Voisard, *Location-based services*. Elsevier, 2004.

[3] T. Reichenbacher, *Mobile cartography: adaptive visualisation of geographic information on mobile devices*. Verlag Dr. Hut München, 2004.

[4] Z. Cheng, J. Caverlee, K. Lee, and D. Z. Sui, "Exploring millions of footprints in location sharing services." *ICWSM*, vol. 2011, pp. 81–88, 2011.

[5] J. Lindqvist, J. Cranshaw, J. Wiese, J. Hong, and J. Zimmerman, "I'm the mayor of my house: examining why people use foursquare-a social-driven location sharing application," in *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 2011, pp. 2409–2418.

[6] C.-C. Yu and H.-P. Chang, "Personalized location-based recommendation services for tour planning in mobile tourism applications," in *International Conference on Electronic Commerce and Web Technologies*. Springer, 2009, pp. 38–49.

[7] M. Pedrana, "Location-based services and tourism: possible implications for destination," *Current issues in Tourism*, vol. 17, no. 9, pp. 753–762, 2014.

[8] G. Heinemann and C. Gaiser, "Study: Status and potential of location-based services," in *Social-Local-Mobile*. Springer, 2015, pp. 155–185.

[9] D. Singh and N. Ratan, "Location based services: Adding another dimension to smart cities." [Online]. Available: https://www.pwc.in/assets/pdfs/publications/2015/location-based-services-adding-another-dimension-to-smart-cities.pdf

[10] C. Wang, "Location based services and location based behavior in a smart city," Ph.D. dissertation, Université de Lyon, 2016.

[11] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive computing*, vol. 2, no. 1, pp. 46–55, 2003.

[12] J. E. Dobson and P. F. Fisher, "Geoslavery," *IEEE Technology and Society Magazine*, vol. 22, no. 1, pp. 47–52, 2003.

[13] L. Barkhuus and A. K. Dey, "Location-based services for mobile telephony: a study of users' privacy concerns." in *Interact*, vol. 3. Citeseer, 2003, pp. 702–712.

[14] M. Duckham and L. Kulik, "Location privacy and location-aware computing," *Dynamic & mobile GIS: investigating change in space and time*, vol. 3, pp. 35–51, 2006.

[15] L. Ackerman, J. Kempf, and T. Miki, "Wireless location privacy:law and policy in the u.s., EU and japan-ISOC member briefing #15." [Online]. Available: https://www.isoc.org/briefings/015/

[16] L. Cranor, *Web privacy with P3P*. " O'Reilly Media, Inc.", 2002.

[17] R. Bellis, A. Cooper, and R. Sparks. IETF working group on geographic location/privacy. [Online]. Available: https://datatracker.ietf.org/wg/geopriv/about/

[18] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1st international conference on Mobile systems, applications and services*. ACM, 2003, pp. 31–42.

[19] C.-Y. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*. ACM, 2006, pp. 171–178.

[20] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, "Private queries in location based services: anonymizers are not necessary," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 2008, pp. 121–132.

[21] M. Duckham and L. Kulik, "A formal model of obfuscation and negotiation for location privacy," in *International conference on pervasive computing*. Springer, 2005, pp. 152–170.

[22] T. Rodden, A. Friday, H. Muller, and A. Dix, "A lightweight approach to managing privacy in location-based services." [Online]. Available: http://eprints.lancs.ac.uk/12967/

[23] S. Gambs, M.-O. Killijian, and M. N. del Prado Cortez, "De-anonymization attack on geolocated data," *Journal of Computer and System Sciences*, vol. 80, no. 8, pp. 1597–1614, 2014.

[24] M. J. Atallah and W. Du, "Secure multi-party computational geometry," in *Workshop on Algorithms and Data Structures*. Springer, 2001, pp. 165–179.

[25] S.-D. Li and Y.-Q. Dai, "Secure two-party computational geometry," *Journal of Computer Science and Technology*, vol. 20, no. 2, pp. 258–263, 2005.

[26] Y.-L. Luo, L.-S. Huang, and H. Zhong, "Secure two-party point-circle inclusion problem," *Journal of Computer Science and Technology*, vol. 22, no. 1, pp. 88–91, 2007.

[27] G. Zhong, I. Goldberg, and U. Hengartner, "Louis, lester and pierre: Three protocols for location privacy," in *International Workshop on Privacy Enhancing Technologies*. Springer, 2007, pp. 62–76.

[28] I. Bilogrevic, M. Jadliwala, V. Joneja, K. Kalkan, J.-P. Hubaux, and I. Aad, "Privacy-preserving optimal meeting location determination on mobile devices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 7, pp. 1141–1156, 2014.

[29] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[30] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," *Foundations of secure computation*, vol. 4, no. 11, pp. 169–180, 1978.

[31] C. Fontaine and F. Galand, "A survey of homomorphic encryption for nonspecialists," *EURASIP Journal on Information Security*, vol. 2007, no. 1, p. 013801, 2007.

[32] P. Paillier *et al.*, "Public-key cryptosystems based on composite degree residuosity classes," in *Eurocrypt*, vol. 99. Springer, 1999, pp. 223–238.

[33] S. K. Card, G. G. Robertson, and J. D. Mackinlay, "The information visualizer, an information workspace," in *Proceedings of the SIGCHI Conference on Human factors in computing systems*. ACM, 1991, pp. 181–186.

[34] R. Cramer, I. Damgård, and J. B. Nielsen, "Multiparty computation from threshold homomorphic encryption," in *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2001, pp. 280–300.