SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2010; **00**:1–9
DOI: 10.1002/sec

RESEARCH ARTICLE

Privacy-Protected Communication for Location-Based Services

Martin Werner

Mobile and Distributed Systems Group, Ludwig-Maximilians University Munich, Germany

ABSTRACT

Location-Based Services are emerging fast and the problems with privacy are growing with them. While a platform for Location-Based Services can provide the user with high-quality Location-Based Service browsing and powerful mechanisms to reduce the amount of location data transmitted such a platform is dangerous as it has to manage the location data of the users and the actual service usage. This aggregation of private data is a risk in itself. With this paper we want to show that it is possible to implement most Location-Based Services without such a platform and propose a mechanism enabling finegrained control of privacy for a Location-Based Service user. We make use of strong cryptographic techniques to enable a real trust relation between individuals and a weaker trust relation between an individual and a company.

KEYWORDS

Location-Based Services, Privacy

Received . . .

1. INTRODUCTION

Location-Based Services (LBS) are services which depend on the location and other context information (such as the time, weather, environment ...) of the user. This type of service is becoming more and more common for mobile use as most new cellular phones have a GPS chip enabling cheap usage of location information. The most important benefit of Location-Based Services is that a user of a mobile device only gets informations and services relevant for his position. This is very handy compared to searching for some service (e.g. italian food) or a specific information (e.g. weather) in a classical web search due to the difficulties of typing with an on-screen keyboard.

Currently there are platforms emerging which allow anyone to generate a Location-Based Service without any programming by describing the service in some specified form. One such example is Aloqa [1] which in essence is an intelligent Location-Based Service browser. For the Aloqa case the information is organized in channels which the user can subscribe. These channels include public transport information, restaurants, concerts, health-care services and many others.

All such platforms currently work as an intermediator in the sense of privacy. They collect - on a per user or per session basis - position data of users along with a description of their interests and then present such users with Location-Based Services which might attract them. This type of platform is very easy to implement and is easy to exploit commercially. The results of user profiling is of great commercial interest and the platform can provide advanced personalized advertising.

Though it is evident that the usage of any Location-Based Services always reveals private information to the provider of the service (e.g. some sort of location data) it is very dangerous to design systems which collect service data and user tracking data. The problem is not the absense of trust in the platform design or operator but that the data of such a platform will be of great interest to traders as well as criminal organisations and that there is a correlation of danger and success in the sense that once such a system is successfull and collects more and more users it will not be able to guarantee for the privacy protection of the data.

The main argument for the introduction of LBS-platforms is that the platform can be used to intelligently reduce the amount of data transmitted and hence to safe battery power for a longer uptime. While this is true for some complex LBS (e.g. k-next-neighbour tracking, mobile gaming) there is no need for a platform in a one-one LBS-connection (e.g. connections between the user and the actual service provider). The calculational power and intelligence of the mobile platform nowadays suffices to perform most of these traffic optimizations on the mobile device.

Prepared using secauth.cls [Version: 2010/06/28 v2.00]

As the usage of Location-Based Services always needs a trust relation, platform provider argue that the disclosure of location information to a specific known institution (the platform) can be acceptable. We want to propose alternative (though more complicated) ways to provide Location-Based Services in a manner which only needs a trust relation to the actual service itself (e.g. the coffee company or the actual friend I want to use a Friend-Tracker application with). This implies some restrictions on the type of service which can be provided in this way but still the introduction of some platform only for those services which really need a platform is possible.

The mechanism we have in mind uses strong cryptographic technology to protect the private data from collection by platforms or carriers. The main idea is based on the observation that a user typically uses only a very small subset of available Location-Based Services and only allows very few of them to proactively notify him. As a result we can allow us more complexity in the way we exchange location information.

The rest of the paper is organized as follows: In the following section we describe five scenarios which we have in mind. The first one is a Friend-Finder application which shall proactively notify the user about friends which are near (in a configurable distance). The second scenario is a Coffee Company which wants to promote his offerings with a coupon service spreading several location based discounts. The third scenario is a car-to-infrastructure communication application which wants to export information about hazardous driving situations in a way such that other cars can rely on it and issue warnings to the driver. This information shall be stored in a distributed infrastructure at a place which is physically near to the place of relevance. This type of information distribution is often called Geo-Casting. The fourth type of application is an application where the infrastructure guarantees for a messages geographic origin. The fifth example is a social online network whose messaging is based on the mechanisms of exchanging locagrams.

We then review related work in this area and then describe a protoypical implementation in some detail giving hints on how to actually implement the discussed type of service.

2. SCENARIA

The following five scenaria show very different types of Location-Based Services. The Friend-Finder is a very private service where the location of users has to be tracked and exchanged permanently and in great detail. The second service only needs to exchange location information on service invocation. Furthermore the granularity of the location information is not important for the service to work. If the service is only presented with a coarse location (e.g. zip code) it will still work presenting the application with a list of possibly interesting locations which can then be checked locally - on the users phone - for their real distance and importance. The third service uses Locagrams for infratsructure-authenticated, unencrypted Location-Based Messaging for vehicle safety applications. The fourth service consists of the case of some trusted third party signing a location as the origin of a message. These will be cellular service providers which can affirm the origin up to network cell precision. The last service consists of an Online Social Network based on the communication principles of locagrams.

2.1. Friend Finder

Assume Bob and Alice are friends. They went to school together and now live in two different cities. They both work for big companies and travel very often. As a result they often find out when they meet that they have been in the same city at the same time and just missed a possibility to meet. They would like to have a Location Based Service notifying them when they are in the same city at the same time. But as they are careful about their privacy, they do not want to expose their location information to anyone else except each other.

2.2. Coffee Company

Assume a coffe company wants to advertise with location based discount coupons. They want to have a simple way to inform interested customers about discounts on their offerings. For simplicity they do not want to provide a Location-Based Service in several special ways for different Location-Based Service platforms but in a generic way through their web-page. They want to provide a web page which one can send his approximate location information to and get a list of active discounts for this area specification. This web page accepts most usual descriptions of locations such as the zip code, a GPS coordinate, a cityname or a street and of course a distance limit. With this generic setup the coffee company can simply use the existing webserver infrastructure and is ready for providing Location-Based Services. The coffee company can even advertise for their new service with standard tools such as QR-codes showing the URL of the Location-Based Service web page. If this web page is opened in a browser which does not provide the location information it will just show as a standard HTML page where you can search for local discounts in a classical web search. In this way, they instantly support any mobile device equipped with a webbrowser.

2.3. Geo-Casting Vehicles

Assume that some percentage of the cars driving around in a specific area have a connection to some cellular network. Assume that the cellular network provides a service to store some information at the base-stations covering a given area. Assume that a car security system generates warnings for other cars from hazardous situations. These warnings will be stored at the base-station and can be

Security Comm. Networks 2010; **00**:1–9 © 2010 John Wiley & Sons, Ltd.
DOI: 10.1002/sec
Prepared using secauth.cls

retreived by any other car. These cars can then inform the driver about counter-measures (e.g. speed reduction, alternative navigation) against coming into the same hazardous situation as the message publisher. This type of service of course has very high security and reliability needs.

2.4. Attestation of Message Origin

A Location-Based Service might want to have a location which can be trusted. This can be a Location-Based Games (e.g. a rally) or a security application which set access rights depending on the location of a device (e.g. disallowing log in to a corporate Wireless LAN unless a cellular network provider has attested that you a are in this area). With our framework it is possible to use the infrastructure provided by a cellular network provider to check whether the location of a location message is correct or not. In this case however three parties need access to the location information namely the originator, the cellular service provider and the destination. This implies a weaker privacy setting than the other applications have.

2.5. An Online Social Network Providing Protection of Informational Self-Determination

Online Social Networks have become very popular due to their property of keeping people in touch which do not communicate often. Through the usage of a social graph structure we can find many people in a social network which we know just because they communicate with our friends. The most important problem of online social networks is the fact that informational self-determination can not be provided. This is due to the fact that platform-based online social networks use the social data for advertising and market analysis and that some platforms are using search engines such as Google or Bing for advertising by keeping public as much data as possible. With the mechanisms explained in this paper and an extension dealing with permanent profile availability it is possible to construct a distributed online social network where the term friendship is equivalent to having exchanged a keypair and all data is strongly encrypted.

3. RELATED WORK

Many commercial Location-Based Services are arising today. Unfortunately the issues with privacy have been ignored in many cases. This is natural due to the fact, that most of the users do not know what data is exchanged and what data is stored in a non-anonymous way and hence accept applications for the individual service experience. We believe that the importance of privacy will grow in the near future when people realize that they are revealing very much information about themselves to a party that can not guarantee for the protection of the data from abuse.

The privacy threats of Location-Based Services have been brought to public attention such as in the EU directive (2002/58/EC) [6] which essentially requests the explicit consent of a user before the position data is allowed to be processed. In practice such a law does not help much because a one-time acceptance of a checkbox during installation is enough to allow some platform to track and store any private data of its users. As it is not easy to construct better law it is important to inform people about the real danger that lies in using such Location-Based Services.

The case study [2] identified the following three important design issues which have to be addressed for good Location-Based Services:

- It is essential that a system provides the user with real-time information about their level of privacy. The basic questions are who gets to know which private information about my position and context.
- Location-Based Services should enable easy short-term deactivation.
- Location-Based Services are more likely to be adopted in closed environments (co-workers or even bigger groups such as the students of a campus)

As most Location-Based Services share similar privacy concerns researcher have proposed several Privacy Enhancing Technologies (PETs) for LBS. Examples range from basic switches disabling the transmission of location information to more sophisticated systems such as area-based filter rules or mechanism related to k-anonymity. In the paper [3] the basic question whether PET's are used by people using LBS is answered positive. The bad news from this research is, that all PET's which need constant awareness of the users fail in practice [3, chapter 6]. Hence we conclude that it is important to protect as much of the location data as possible as it could be accidentally exposed (e.g. due to forgetting to disable a tracker). We will try to support this with mandatory strong end-to-end encryption.

Many algorithms have been proposed which reduce the privacy problem in very specific cases. For example in [10] the authors discuss three protocols for enabling location privacy. They use a distributed computation scheme based on homomorphic encryption to enable the exchange of the distance of two users without exposing location information to each other or to a third party. Of course this type of algorithm can solve the location problem up to intrinsic attacks such as using the protocol with a set of wrong positions to track down the real position of a user based on the knowledge of proximity decisions for several locations. This type of approach can reach a higher level of privacy than exchanging location information as we do. But this approach does not enable general Location-Based Services and is limited to proximity detection. We want to emphasise that it is possible to use this type of anonymisation protocols on top of our general architecture where needed.

Another strategy towards privacy in Location-Based Services is protecting the content of the information request from a platform by the usage of private queries. In [7] the authors explain how private queries can be used to enhance privacy in Location-Based Services. However their basic assumption is that all interesting information is held in a central database and that it suffices to protect the query for this database as well as the answer from eavesdropping by the platform or the network. This is of course a nice component for a Location-Based Service platform, but many Location-Based Services do not use location information of a central database and deploying such databases in a distributed manner is computationally too expensive. Moreover the degree of anonymity of the query is of course related to the size of the database which makes this type of anonymisation difficult in a distributed setting.

In general there are many approaches which protect privacy of a special type of information by using multi-party computation or advanced encryption schemes. A general privacy risk in this setting is that the intended service itself can be attacked. For example if one manages to have a completely secure proximity detection algorithm which reduces the position information of two people to the one-bit information of being in a specific distance to each other this service can be easily fed with fake locations (which can be selected randomly or even based on social information such as places where the user is often) to aggregate back this one-bit information to a location. With our framework we address this issue by proposing a mechanism where a cellular network provider signs the message origin which makes such attacks very expensive and hard to conduct.

Geo-Cast has been used as a term for a mechanism to send a message to a given geographic area. Usually Geo-Cast refers to a layer 3 protocol. In RFC 4291 [8] a specific IPv6 address space has been assigend to geographic based unicast addresses. In RFC 2009 [9] several mechanisms to enable Location-Based Routing and Geo-Cast have been proposed and discussed. There has however not been much work on Geo-Casting at higher layers. We believe, that the introduction of an alternative and more complex routing technology in the Internet will not happen for good reasons and that Geo-Casting will be done using overlay networks and overlay routing. The reason for this are that the usage of a location description for routing in itself is a big privacy problem and that distributed denial of service attacks will become more severe as they do not attack an abstract thing such as an IP-address which can be blackholed and replaced by another address but a real infrastructural component making it impossible to recover from such an attack in an acceptable amount of time.

With our generic framework for the exchange of location data we are able to support all of these issues and hence raise the acceptability of such Location-Based Services to a higher degree than current Location-Based Services.

4. LOCAGRAMS

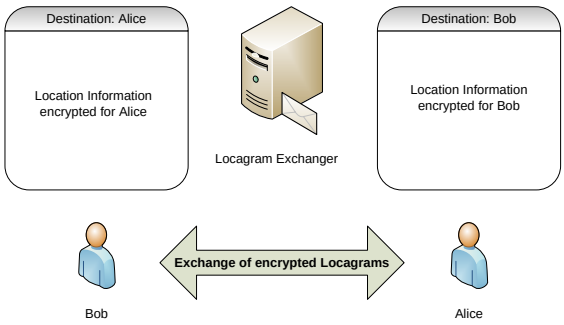

Figure 1. Communication is done using a common known locagram exchanger

A system which can provide users with full control about their privacy and the way they export their location information to a Location-Based Service automatically needs a trust relation between the endpoints of communication, namely between the user of a Location-Based Service and the Service itself. We decided to allow the usage of modern technologies such as strong encryption and some sort of microblogging to enable anonymous information exchange to the maximum extent possible. The basic Location-Based Service communication is done via so called Locagrams which stands for Location-describing Telegrams. These shall be short messages which are constructed as depicted in figure 2 from the following data:

1. General Header
2. Destination Identifier
3. Location Description
4. Time-To-Live
5. Source Identifier
6. Distance bound
7. URL for further communication
8. Additional Information (Payload, limited in length)
9. Signature

where either 3. to 8. or 4. to 8. can be encrypted by some public key encryption and everything can be cryptographically signed - the message authentication code being stored in 9.. The *General Header* specifies the cryptographic algorithms and whether the location description is encrypted or not. In this way everything can be protected against modification and all information except the destination description can be encrypted.

These Locagrams shall be used as the basic messaging system of a Location-Based Service. It is of course expensive to use Locagrams due to the complex encryption and the administrative complexity of sharing the needed keying material and exchanger addresses for each message exchange. For many services however it is not too expensive to do this. In cases where there is much

Security Comm. Networks 2010; **00**:1–9 © 2010 John Wiley & Sons, Ltd.
DOI: 10.1002/sec
Prepared using secauth.cls

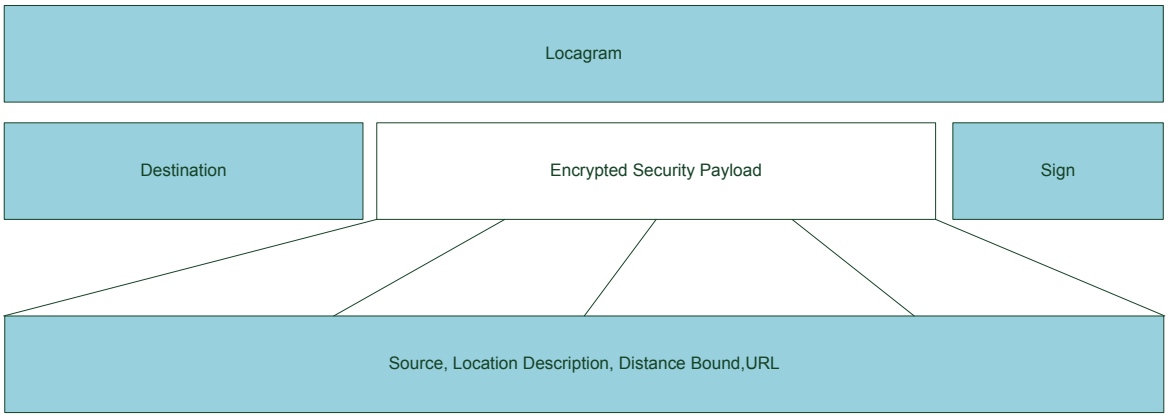

Figure 2. The general structure of a locagram

communication demand, Locagrams shall be used to exchange a shared secret and a new communication channel specification (e.g. IP address, protocol and port information) to reduce the computational impact of our design.

Figure 1 illustrates how locagrams shall be exchanged. They are stored on a server (Locagram Exchanger) and can be downloaded by users in an anonymous fashion. There is no need to introduce sessions or passwords or any identification here unless for scalability reasons or to prevent abuse of the exchanger. Techniques to do this protection are beyound the scope of this paper, but we want to note that the system is highly distributed and hence more difficult to attack then most common systems.

4.1. General Header

The General Header describes some general options for this Locagram. It specifies the cryptographic algorithms being used, which of the fields are encrypted, which type of message authentication code has been used and the length of the *Additional Information* field.

4.2. Destination Identifier

The Destination Identifier should be any unique identifier identifying the person which shall receive this locagram. It can but need not include information to contact the destination. Possible choices include a cryptographic public key, an account name or some synonym.

4.3. Location Description

The location description should be a textual representation of the current location. It could contain one of the following information:

- WGS84-coordinate (possibly obtained from GPS or some coarse network localization)
- zip code
- address description(either complete or only a city name)

4.4. Time-To-Live

The Time-To-Live field contains an integer specifying the duration (in seconds) that a locagram exchanger is requested to keep a locagram. This field must be used by the locagram exchanger as the maximum time to keep a locagram. In this way a basic deactivation of the software leads to the removal of all location information within the time specified in this fields.

4.5. Source Identifier

The Source Identifier shall be the same type of identification as the destination identifier except that it shall describe the source of the message. In this way it is possible for the destination to answer to locagrams with another locagram.

4.6. Distance Bound

A distance bound is introduced to describe the area for which the locagram is relevant. This is currently stored as a string and might contain either a floating point number in a predefined unit or a string containing a well-known unit string (e.g. ”1 km”).

4.7. URL for Further Communication

This field can contain any URL. We think of web URL's for allowing enterprises to export Location-Based Services in a simple way through their webpage and special values such as

- about:return indicating that the same locagram exchanger shall be used to answer to this locagram with another locagram
- phonecall:number indicating that in case of relevance the user should be prompted to call the given phone number
- sms:number indicating that the locagram can be answered with a SMS

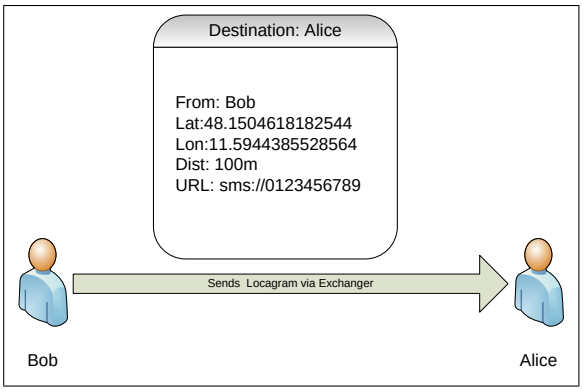

Figure 3. A Locagram example for the Friend-Finder

4.8. Additional Information

Here the actual service on top of our concept can put some information. This information shall be either small (for the Friend-Finder application there is actually no additional information) or a specification of an alternative channel possibly including symmetric keying material, IP-addresses or URL's. The size of the Locagram and the question whether the actual information is given directly or by reference has strong implications on the speed and frequency with which Locagrams can be exchanged.

We want to stress that this field shall only used for application data if the overhead of specifying an out-of-band information channel is higher than the actual information or if the overhead of reading out such an out-of-band information source is too high. An example where this will happen is given by beaconing applications in vehicular area networks which are extremely time-critical and hence have to include the application information in the first message.

4.9. Signature

The Signature field contains a cryptographical message authentication code for the complete Locagram protecting it from changes.

5. THE USAGE OF LOCAGRAMS

Now we want to describe how the information which can be exchanged using Locagrams can be used to implement the types of Location-Based services we described earlier in section 2.

5.1. A Friend-Finder with Locagrams

For the Friend-Finder locagram implementation we use a server where we can store locagrams and recieve locagrams by identifier. This server of course has to be protected from spamming and denial of service attacks by some technology. There are very different ways to achieve something like that and as it depends very much on the actual implementation and the network location of the Locagram Exchanger, we do not want to go into detail with this.

Now the basic functionality of the Friend-Finder is the following: The Friend Finder of Alice has to be configured with the following data:

- Identifiers of Alice and Bob
- A public key of Bob
- The key pair whose public key was given to Bob for communication
- A Locagram Exchanger address

The key exchange is done via SMS or via displaying QR-Barcode on the displays when the two friends meet. In the future we could also make use of NFC for key exchange. It is of course also possible to use any other mechanism to exchange keys, but we believe that our proposal is a very good mixture of usability and security.

Now Bob activates his Friend-Finder. As Bob knows Alice (e.g. Alice's public key) Bob sends a locagram for Alice containing his current position and a given configured distance bound where Bob wants Alice to inform him. A Locagram example for this situation is given in figure 3. The URL is set such that Alice is notified if near enough to Bob and told to send an SMS to Bob in reaction to a proximity event. In this way, she is able to contact Bob in the way Bob prefers: Proactively via SMS. Bob could alternatively have set the URL to the Locagram Exchanger in the Locagram's URL field. Then Bob's Friend-Finder would have to regularly check for incoming locagrams and notify him if a locagram answer is incoming within the given distance bound.

This is a very simple approach which can be optimized in many ways. The first optimization would be to let Alice answer to Bob's locagram with a locagram of it's own position and distance information such that Alice and Bob have the possibility to estimate the time until they will update each other with location information. If the distance between Alice and Bob is big, it is not important for the Friend Finder to exchange location information. Another important optimization at this point is to start with Locagrams with very coarse location information and only if they do not conflict to send finer locagrams. In this way we can even safe battery power by using some localization mechanism which is more battery efficient than enabling the GPS receiver.

To allow even more privacy, Alice and Bob could configure HTTP proxies to exchange locagrams with the locagram server or even setup their own locagram server on their private home page. It is of course also possible to use existing internet technologies such as a microblog (e.g. Twitter) or the Internet Relay Chat for the exchange of locagrams.

If the Friend-Finder is implemented in this way, we enhance the privacy of Alice and Bob in several ways. The first enhancement is, that no one except Alice and Bob

Security Comm. Networks 2010; **00**:1–9 © 2010 John Wiley & Sons, Ltd.
DOI: 10.1002/sec
Prepared using secauth.cls

can get any location information as it is strongly encrypted per default. Another enhancement is, that all information is kept physically on Alice and Bob's devices. So no one will have any interest in collecting locagram data. Moreover the system does not have a central element (such as a platform) but can use any distributed data exchange mechanism (Microblog, Internet Relay Chat, ...).

As this is a very trivial example of a Location-Based Service which unlike many other services relies only on the peer-to-peer exchange of information which allows the estimation of proximity, it is easy to propose an algorithmic aid for reducing the semantic information for this very specific case. However this is not a contradiction to using Locagrams for exchange of protocol information.

5.2. The Coffee-Company Location Based Voucher System

But how can a coffee company export a Location-Based Service within this framework? The coffee company exports its Location-Based Service using a web page which takes location information, a user identification and public key and in answer to the request sends a list of possibly applicable locagrams.

Assume Bob wants to get a coffee. He then actiates the Coffee-LBS for its actual position. Bob has configured his Location-Based Service browser to export only the name of the city to the coffee-company. Thus the application starts with mapping its actual GPS-coordinate to city names (which Bob - as he really likes privacy - has downloaded a list for). Bob then provides the Coffee-Shop's locagram page with this city name and in turn gets many locagrams as answers. Which locagram is the most applicable for him now can be checked locally on his phone and a map application is opened showing the coffee shops and vouchers.

The main advantage of such an approach is, that the data is exchanged only between the actual service provider and the service user. And due to the integration of web publishing for Locagrams, we achieve a simpler integration into existing infrastructures.

5.3. Geo-Casting with Locagrams

Geo-Casting refers to mechanisms to send a message to a specific geographical area. There have been several mechanism proposed. Most of them either change the routing technology to do Location-Based Routing or construct Location-Based overlay networks on top of the Internet infrastructure. With our approach a more stable and simple mechanism could be implemented. Assume that vehicles have a configured list of Locagram Exchangers which are responsible for some geographical area. Then these vehicles could exchange safety messages using Locagrams and this Locagram Exchanger infrastructure in an authenticated, distributed and transparent way.

In the Car2X domain applications are usually classified into one of the following three classes: Beaconing applications, messaging applications and infotainment applications. Beaconing applications periodically exchange short messages over short distances (preferably using ad-hoc technology) which are relevant only for a very small amount of time. For this class of applications the overhead of Locagram encryption and Exchanger invocation is too high. Messaging applications exchange information about driving situations (e.g. aqua-planing, congestion etc.) which are relevant for a considerable timeframe. This is actually the area where Locagrams should be used in Car2X communication. The infotainment class of applications does not have time constraints and is usually the domain of classical Internet applications.

5.4. Authenticated Geographical Source

For some applications it could be very interesting to have a trusted third party guarantee the fact that a message originated from a specific geographic area. We refer to this as authenticating the geographical source of a message. Applications for this are manyfold. It could be used to make it more difficult to spam a Location-Based System with incorrect positions and hence attack for example secure proximity detection algorithms. We can also imagine that a connection to a Wireless LAN is only allowed if the mobile phone of the user proofs that the user is near to the physical access point. This would make it much more difficult to attack a Wireless Network even with stolen credentials. The only possible attack would be to steal the phone or to abuse the access rights while the users phone is near. Another application of this type of service could be the limitation of debit cards in areas where a registered phone is not able to proof presence.

A cellular network provider usually knows a coarse position of each device when it has active connections. Now the main question is, how should a cellular network provider authenticate the message origin of a Location-Based Service message? We identify the following two possibilities for this:

- Deploy Locagram Exchanger for each cell (possibly on each eNodeB) which takes a Locagram with unencrypted position information and forwards this locagram to a specified external Locagram Exchanger
- Deploy a Location Proof Interface in the cellular network, where a user can let the cellular provider sign a coordinate with a well-known certificate

The first mechnaism is obviously more complex than the second one. Providing and managing a service on each cellular network cell is an expensive task. Nevertheless if the demand of short and local information exchange keeps growing, this can keep away much time-critical traffic from the backend as for example in the area of car safety application messaging. Moreover the problem in this area is, that highly accurate position information is being

stored for every client (so per-client memory is needed) for some time whereas the second service can work without additional memory.

The second mechanism we have in mind is that we can get a proof of our position to be correct from an interface. In this case every base station would have to cryptographically sign a challenge consisting of a random part and our location description with the private key of a well-known key pair of the service provider in the case that the location is correct. We would then add this signed data in our Locagram's *Additional Data* field and the receiver of the locagram could then (after having decrypted the locagram) check whether he wants to trust the sign or not.

This type of service of course weakens the privacy enhancement of Locagrams in the way that another party knows the position information, namely the cellular service provider. But as this service provider only gets to know the part of information which he actually needs (at least in the second case) this is the best possibility with regards to privacy.

5.5. Secure Online Social Network Based on Locagrams

An online social network consists of an overlay network over the Internet where a network connection shall be equivalent to having a friendship in real life. With a thorough analysis of the requirements for having a friendship in real life we found out [4] that the most important properties of real communication behaviour and real friendship which are not provided in available online social networks are informational self-determination and a strong trust relation. Usually people accept a friendship in an online social network based on the name and picture of the user being correct. Moreover the social network user does not have the possibility to define access rights to digital assets for each of his friend independently. Though other work indicated that such complex privacy settings would not be used in the private area it is a must for business usage of social networks. For details on how this type of social network can be constructed using Locagrams and how to deal with the advanced communication patterns in such a setting we refer to [4] and [5] which explain the principles of the VEGAS Online Social Network system.

6. PROTOTYPE

The usage of locagrams as described before is a very generic approach to implementing a Location-Based Service. We did not and do not want to specify the usage of one specific communication system. For our prototypical implementation we decided to use a web-page as well as an email system as two possible locagram exchanger implementations and some string as the destination identification. If one would use the public key as the identification one could extend the locagram exchanger to check the identity of the requesting person. For readability we however decided to keep the names Bob and Alice (or anything else you configure).

It is possible for the user to generate a new key-pair for every invocation of the coffee-shop service and hence being very private. It would be nice if the locagrams would not have to be stored somewhere but transported directly to the clients. We did not implement something like that, because at the time being cellular service providers (at least in germany) do not allow much more types of traffic than HTTP-client requests. We also hope that at some day a cellular service provider could allow the direct exchange of locagrams via SMS or similar service in a cheap way.

Our prototype consists of four components: A *Locagram Exchanger* implemented as a web page, a *LBS Service* consisting of some Java classes implementing the basic LBS functionality and a *Friend-Finder LBS* and a *Coffee-Company LBS web-page*. The Locagram Exchanger and the Coffee-Company LBS web-page have been implemented in PHP while the LBS Service and the Friend-Finder have been implemented in Java using a basic and simple RSA implementation for encryption. As the Java Crypto API is not fully supported on all mobile platforms we used our demonstrative implementation of RSA for the cryptographic operations.

6.1. The LBS-Service

The LBS service is implemented as a Java class which holds a list of different locagram exchanger URLs and some basic configuration for polling locagram exchanger.

The LBS-Service exports the basic functionality of

- addition and removal of locagram exchangers
- query/poll locagram exchangers for new locagrams
- generation and exchange of public keys via SMS and QR-Code
- a storage for public keys

This set of function suffices for many Location-Based Services to be realized. Note, that anonymisation or cooperation protocols are to be implemented on top of this service.

6.2. The Friend-Finder

The Friend Finder is implemented as an Android application and works just as described in the basic scenario. On startup, it shows a menu where you can either start a key exchange via SMS or QR-Code or run the Friend-Finder. Once running, the Friend-Finder contacts the configured Locagram Exchanger and requests locagrams using the LBS-Service. Once it receives a locagram, it will check, whether the distance and position apply and then will inform the user. It also contains a list of friends which are regularly informed about the position by sending out locagrams.

Security Comm. Networks 2010; **00**:1–9 © 2010 John Wiley & Sons, Ltd.
DOI: 10.1002/sec
Prepared using secauth.cls

6.3. The Coffee-Company LBS Web Interface

The Web Interface for the Coffee-Company LBS Web Interface is implemented as a website which needs the following text variables and returns a status code to the client requesting a service.

- **ident** which is the identification to be used as the destination of the locagram
- **pubkey** which is the public key of the client requesting the locagrams
- **position** which is some description of the position as explained before
- **distance** which is the distance limit for which locagrams shall be received
- **locagramexchanger** which is the URL of the locagram exchanger to be used

This website will then send the locagrams via the given locagramexchanger or - if the locagram exchanger is not given - on the webpage itself. It can return status codes which might indicate that it refuses to encrypt data (e.g. for scalability reasons), that there were no results to the given search or that other error conditions occured. In absence of errors the locagrams will be transmitted to the given locagram exchanger where the user can download them and analyze them further.

6.4. VEGAS - Secure Online Social Network

The basic mechanism of encryption and exchangers discussed in this article have been used as the basis for a privacy-friendly Online Social Network as described in section 5.5. This network incorporates mobile devices and stationary devices and uses email and an FTP server as primary forms of locagram exchangers. The exchange of keys is done as described before by exchaning information visually from display to display. VEGAS has been implemented in a student project and is still growing in functionality and implementation. We are proud to see the basic idea is very successfull in this area.

7. OUTLOOK

With this paper we have presented a novel approach to protecting privacy in Location-Based Services which does remove the need of a trusted party intermediating between the service provider and the service user. This is important because a market-place for Location-Based services which manages the tracking of clients as well as the service discovery can collect too much private information. It is of course clear, that a Location-Based Service user does expose location information but it does not make sense to expose this information to anyone else but the entity providing the actual service.

This approach can be extended to cover almost any type of Location-Based Service and reduces the amount of private data exchange to an absolute minimum. It is even possible to generate a new key-pair for every invocation of e.g. the Coffee-Shop-Service. It now depends on the user to decide who gets what information in what granularity. With this framework it is even possible to implement Location-Based Services in closed high-security environments.

Especially in situations where the usage of GPS and network does not imply problems with power consumption (e.g. in a car with a navigation system) the usage of locagrams can very efficiently protect the privacy of the user against e.g. a gas station operator.

REFERENCES

1. Aloqa - always be a local, 2010.
2. L. Barkhuus. *Privacy in Location-Based Services, Convern vs. Coolness*. Proceedings of Workshop paper in Mobile HCI, 2004.
3. Thorben Burghardt, Erik Buchmann, Jens Müller, and Klemens Böhm. *Understanding User Preferences and Awareness: Privacy Mechanisms in Location-Based Services*, volume 5870. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
4. Michael Dürr and Martin Werner. Re-socializing online social networks. accepted for publication in the IEEE International Symposium on Social Computing and Networking 2010, 2010.
5. Michael Dürr and Kevin Wiesner. A privacy-preserving social p2p infrastructure for people-centric sensing. accepted for publication in the proceedings of KiVS 2011, 2010.
6. Directive 2002/58/ec of the european parliament and of the council, 2002.
7. Gabriel Ghinita, Panos Kalnis, Ali Khoshgozaran, Cyrus Shahabi, and Kian L. Tan. Private queries in location based services: anonymizers are not necessary. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 121–132, New York, NY, USA, 2008. ACM.
8. R. Hinden and S. Deering. IP Version 6 Addressing Architecture. RFC 4291 (Draft Standard), February 2006.
9. T. Imielinski and J. Navas. GPS-Based Addressing and Routing. RFC 2009 (Experimental), November 1996.
10. Ge Zhong, Ian Goldberg, and Urs Hengartner. Louis, lester and pierre: Three protocols for location privacy. In *Proceedings of Seventh Privacy Enhancing Technologies Symposium (PET 2007)*, pages 62–76, 2007.