STOP: a location spoofing resistant vehicle inspection system

Henrique F. Santos[1][0000−0003−1158−9378], Rui L. Claro[1][0000−0003−0176−2720], Leonardo S. Rocha[2][0000−0002−2608−1844], and Miguel L. Pardal[1][0000−0003−2872−7300]

[1] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal
{hfigueiredosantos,rui.claro,miguel.pardal}@tecnico.ulisboa.pt
[2] Universidade Estadual do Ceará, Brazil
leonardo.sampaio@uece.br

Abstract. An effort is being made by authorities worldwide to improve the safety of the transportation of goods while preserving efficiency. Vehicle inspections are important for safety but not very frequent. When they do happen, vehicles are selected on the roadside and authorities spend a long time retrieving the relevant information while the vehicle is stopped. In this paper, we present and evaluate STOP, a road transportation vehicle inspection support system with tamper-proof records to prevent location spoofing attacks. To the best of our knowledge, it is the first such system described in literature. The STOP system uses mobile devices and a central server to allow authorities to select and notify vehicles for inspection while retrieving the needed information to prepare the procedure beforehand. The location chain for each vehicle can be verified and signed by the inspectors. We implemented a prototype in the Android platform and tested it with real users. We evaluated the system's location retrieval accuracy, response times, and Bluetooth communication during inspection.

Keywords: Smart Mobility · Transportation · Mobile Applications · Location Spoofing Prevention · Location Proofs.

1 Introduction

The frequent inspection of road transportation can bring several positives outcomes, such as improved safety for drivers, vehicles, and goods, along with decreased environmental impact and significant savings. At an inspection site, an inspector orders incoming transportation vehicles to stop to conduct an inspection, with no previous knowledge of what these vehicles are transporting. Depending on the type or size of freight, the inspector has to adapt the procedure to the situation, possibly requesting assistance from colleagues. Naturally, these manual steps can take a long time. If the selection and notification of vehicles for inspection could be done beforehand, inspectors would then have additional time to prepare the inspection procedure until the vehicle arrives. This can improve

efficiency and reduce the duration of inspections. By leveraging location-based services (LBS), it is possible to enable location reporting of transportation vehicles to authorities. As such, it is possible to know the ongoing transportation and what vehicles are close to the inspection site. A simple mobile device with Internet connection can be used by the inspector to retrieve the documentation beforehand and to create a checkpoint. Additionally, inspectors can submit inspection outcome reports digitally.

In this paper, we present and evaluate STOP, a novel road transportation vehicle inspection support system using location proofs. Its main goal is to validate location chains, one for each vehicle, allowing information critical to the inspection process to be stored and validated in tamper-proof records.

The paper is organized as follows: Section 2 presents the background and related work; Section 3 presents the STOP system in detail; Section 4 presents the experimental evaluation that was done; and Section 5 completes the document with a summary of the contributions and opportunities for future work.

2 Background and Related Work

The location reporting of each vehicle enables the selection of vehicles and the consecutive preparation of inspections. Therefore the proposed system needs reliable location reporting. This section provides background on location systems, with an emphasis on systems that are able to provide location proofs.

2.1 GPS-based Location Systems and Applications

The Global Positioning System (GPS) is composed by a set of 31 operational satellites that emit radio signals that a GPS receiver can use to determine its position on Earth [1, 6]. The receiver locks to the signal of at least 4 satellites and calculates its position, taking into account the current time and the known coordinates of the satellites. Each GPS satellite continually broadcasts a signal that includes a pseudo-random code known to the receiver and a message that includes the time of transmission of the code and the satellite position at that time.

Location Tracking Systems A GPS tracker is a device that enables real time position tracking of attached objects [9]. This device continuously retrieves its location by retrieving satellite signals from GPS. Currently transportation companies use *fleet management systems* that receive and gather data from the trackers inside vehicles to present real time information of the vehicles to the users.These solutions allow companies to monitor their fleet, ensuring secure transportation and reporting the delivery to a client as it happens. The device transmits the collected information through Global System for Mobile Communications (GSM) cellular network to the servers of the provider, which is presented through a web portal or computer software.

Use of Location by Mobile Applications GPS location is widely used across the majority of mobile devices in use today. Two of the most common uses are road navigation and ride-sharing [4, 14]. These mobile applications rely on the location reported by devices to guide users to their destination for example.

Navigation applications have also been used in the transportation sector [11]. Every carrier wants to decrease route times and reduce costs with fuel consumption and vehicle maintenance. Therefore it is important to dynamically change routes according to traffic information. The use of a mobile application provides a low cost integration with any road route navigation system through mobile data.

Security Despite being widely used, GPS is not considered fully secure [12, 13]. A GPS spoofing attack aims to deceive GPS receivers by broadcasting incorrect signals. These are structured to resemble a set of normal GPS signals and they can be modified to cause the receiver to estimate its position where desired by the attacker. Inexpensive GPS spoofing devices are available in the market [7], therefore an attacker can easily purchase such devices. It is then possible to deceive mobile devices running road navigation applications [17], air drones [8], ships [16] and working vehicles [3].

2.2 Location Certification

A *location proof*, as defined by Saroiu and Wolman, is a mechanism to allow mobile devices to prove their location to applications and services [15]. The authors considered that a component of an existent wireless infrastructure such as Wi-Fi Access Points and cellular towers can issue metadata containing location information. A device can request a location proof from the infrastructure and this proof can be sent to applications with the intent of proving the location of the mobile device. There have been several systems that allow the creation of location proofs, namely, Saroiu and Wolman's work, APPLAUS [18], CREPUSCOLO [2] and SureThing [5]. In these systems, a Prover broadcasts a location proof request through wireless communication to nearby devices. The witness creates a proof and signs it with its private key. The proof contains the observation made by the witness that can also contain additional data, such as specific secret code sequences being transmitted at the location, and pictures from a surveillance camera, that further prove that the prover device was at that location at the time. The Location Proof Server can later verify the proof.

Zhu and Cao proposed a location proof system called APPLAUS using only Bluetooth enabled mobile devices [18], using five entities: *Prover*, the mobile device who collects proofs from neighbors, *Witnesses*, untrusted mobile devices that generate location proofs, *Location Proof Server*, to store proofs, *Certificate Authority*, to store and validate public keys, and *Verifier*, that verifies submitted proofs. The system does not use an existent wireless infrastructure. It uses pseudonyms for each Prover and Witness to prevent device tracking.

Canlar et al. [2] created CREPUSCOLO to address both the *neighbor-based* type of proof-based solutions, where nearby mobile devices create proofs, and

the *infrastructure-based* type, where location proofs are acquired from trusted infrastructure elements, such as Wi-Fi Access Points. The system uses the same entities of APPLAUS with the addition of the *Token Provider*, a trusted entity placed at a strategic location that generates a proof, called *token*, that may contain an object, such as a picture from a surveillance camera, that proves the device was at that location. Location proofs are exchanged and created like in APPLAUS, with the addition of a nonce in the proof request and in the associated location proof, to prevent replay attacks. The Token Provider is used to mitigate attacks where one device may broadcast messages from another device located at a different site and therefore witnesses may create proofs of the prover located at a different place.

SureThing [5] aims to provide correct location proofs to other applications and services, indoors or outdoors, using as motivation improving the APPLAUS and CREPUSCOLO works. It uses multiple entities similar to the ones in the two previous works presented, *Prover*, *Witness*, *Verifier* and *Certification Authority*, and it also uses geographical coordinates, Wi-Fi fingerprinting and Bluetooth beacons as location proof techniques. Ferreira and Pardal introduced two methods for collusion avoidance, to prevent colluding devices to create incorrect location proofs. The *Witness Redundancy* mechanism forces the Prover to gather proofs from more than one Witness and chooses the number of witnesses according to the level of service possible. Each proof has a different trust value according to the number of witnesses used. *Witness Decay* ensures that if a Prover is getting proofs from the same Witness, they gradually become less valuable and the Verifier will not validate the location if the Prover can not gather proofs with enough value.

3 The STOP system

We present a road transportation inspection support solution named STOP: Secure Transport lOcation Proofs. Its main goal is to provide and register the accurate location information for inspectors and drivers, by using mobile devices. STOP has security mechanisms to prevent and mitigate malicious intents. The system is owned by an *Authority* responsible for the rules for vehicle selection and goods inspection. It audits the system and validates every procedure. It also keeps the history of each participant, and can use it to handle exceptions, like equipment or inspector failures.

The system uses *pseudonyms* instead of the real identities of the participating entities as it does not need this information to operate.

3.1 Inspection Process

A transportation starts with a company registering the freight information with the competent authorities. A carrier or the company itself performs the transportation, which can be inspected by authorities at any point of the route. The

on-board device retrieves its location and uploads it at a system-defined rate. The process is finished when the goods are delivered to the reported receiver.

An inspector arrives at an inspection site, starts the application, logs in and creates a *checkpoint*. The inspector defines the *selection range*, a perimeter from inspection sites where all vehicles inside are considered for selection. The *selection rule* is applied, and, for example, a vehicle is chosen at random from inside the selection range. At this time, the on-board device of the selected vehicle retrieves the checkpoint information, which is presented to the driver. When the vehicle arrives at the checkpoint, the Transport device communicates with the Inspect device and inspection starts. The inspector checks the system records and the vehicle and freight documentation. The inspector can register additional information in form of text, pictures or audio. When all of the inspection information is complete, it can be reviewed and approved.

The *Location Chain* needs to be valid. The chain represents the positions of the vehicle during the transportation of goods, in chronological order. A location chain item is either a *Location Point* or *Location Proof*, as illustrated in Figure 1. Both contain the signature of the previous item. A local copy of the location chain is kept by the Transport device so that the system can operate even when an Internet connection is not available.

A location point contains the geographic coordinates retrieved by the transporter device GPS, at a time point of the trip. A location proof contains the geographic and time coordinates retrieved by an inspector device at a checkpoint along with the additional collected evidence.

The location chain is protected by the chain of signatures. Each item signs the previous one, including the signature. This way, it is possible to verify if the previous item is modified or missing, providing protection against record tampering. It is also possible to check whether the location data from the previous items is consistent with the inspection being actually carried out on site. The location tracking and the inspection data is intertwined, and, as a result, both are strengthened: the location points have to be consistent with the itinerary until the inspection, and the inspection data is reinforced to have happened at the time and place, following the itinerary.

3.2 Localization

The STOP system uses the *Google Play services location* API, which allows to program constant location retrieval. We use this to obtain the most accurate location positions possible for small time intervals (1 second). These intervals are still subject to fluctuations, due to battery optimization or poor connectivity of the device.

Device localization has changed in recent Android versions. Location retrieval is no longer tied only to GPS tracking, as devices also use additional information from nearby Wi-Fi networks, from GSM networks and other device sensors[3].

[3] https://policies.google.com/technologies/location-data

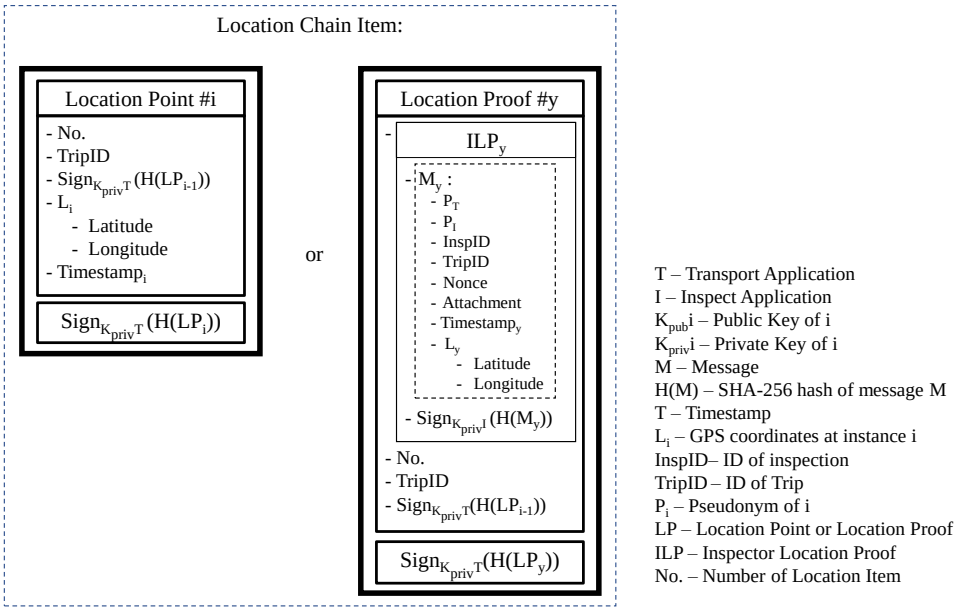

Fig. 1. Types of location chain item.

We discuss the impact of the usage of multiple sources of localization in our evaluation, in Section 4.1.

3.3 Architecture

The STOP system is structured in three tiers: Presentation, Logic and Data, as shown by Figure 2. This allows for integration of new components such as different storage systems and user interfaces.

The main components of the system are the Central Ledger, the Transport and Inspect mobile applications. The *Central Ledger* is a central server that receives transportation and inspection records. All communication with the Central Ledger is done through a Representational State Transfer (REST)ful Application Programming Interface (API). A detailed description of the interface was done in *OpenAPI* description language format. The records are kept in a database for concurrency control, load balancing, and increased availability, with multiple servers.

The *Transport* mobile application runs on a mobile device inside of the vehicle transporting the reported goods in a device with an active Internet connection during the transportation process.

The *Inspect* mobile application is used by the inspector on a mobile device at an inspection location. After a vehicle is selected, the application presents the respective transportation information for the inspector to analyze while the vehicle reaches the checkpoint. The application communicates with the device

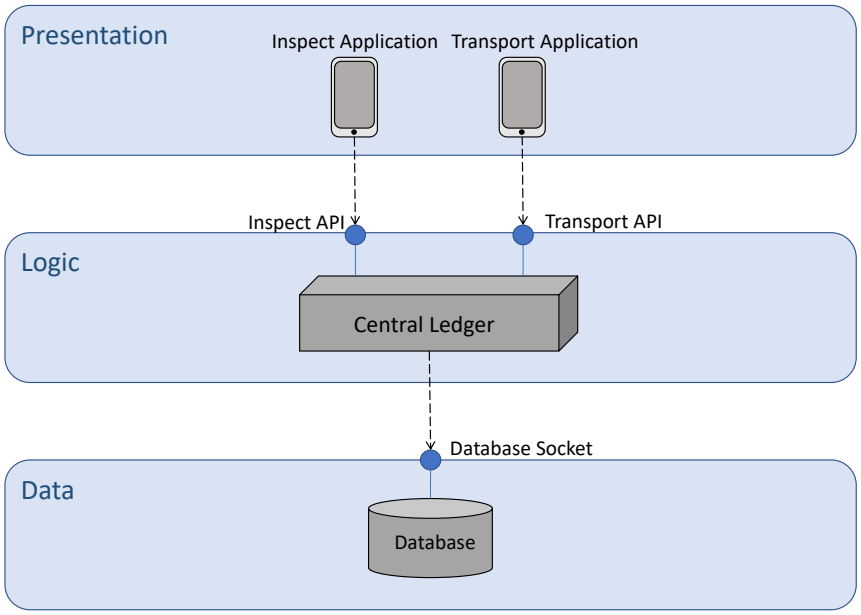

Fig. 2. STOP system architecture.

inside of an inspected vehicle via short-range communication. A location proof is generated at the end of the inspection procedure. The proof contains pseudonyms of the *Transport* and *Inspect* devices, a trip identifier, and a random nonce generated by the Central Ledger for the occasion. This proof can replace any paper report done by the inspector, as it proves the inspection was conducted and contains the relevant evidence.

3.4 Communication protocol

The *remote* communication between the applications and the Central Ledger is done through the provided REST API web service via cellular network. This API uses standard HTTP over TLS[4] to protect the messages [10].

The Central Ledger acts, effectively, as a Certification Authority (CA) for the public keys. An external CA can also be used.

The *local* communication between devices is done using Bluetooth. As a close proximity communication protocol, it is ideal for the inspection process, and acts as a location spoofing countermeasure.

Figure 3 shows the interaction when a vehicle is selected for inspection. The *Inspect* and *Transport* devices obtain the public key certificate of the other device from the central ledger, along with a nonce and a pseudonym for each device. This is necessary to encrypt the Bluetooth communication between these devices and to prevent replay, eavesdropping and tampering attacks.

[4] https://tools.ietf.org/html/rfc8446

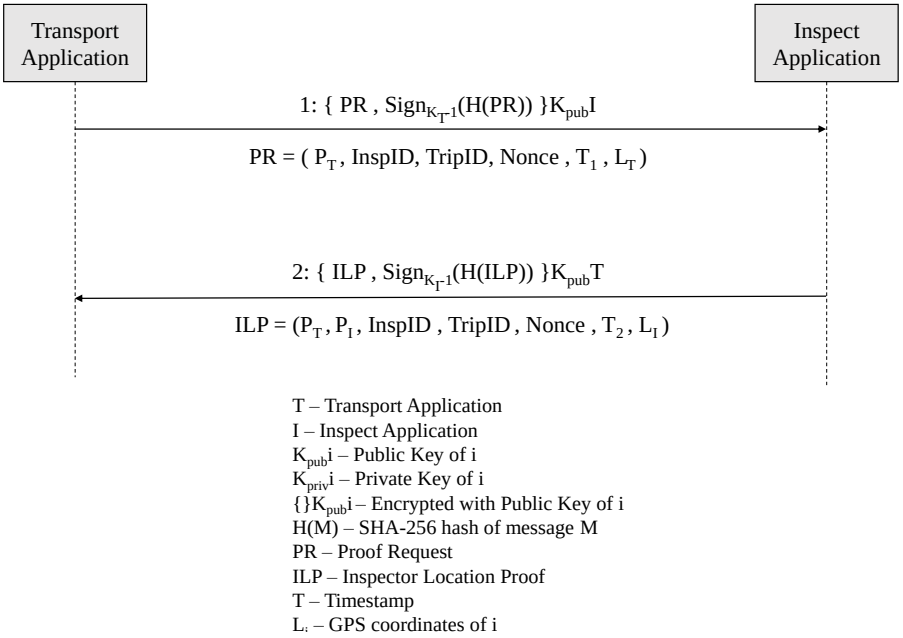

Fig. 3. Inspection protocol.

When the vehicle arrives to the checkpoint, the *Transport* application starts searching for the Bluetooth device announcing as device name the pseudonym of the device of the inspector. When found, the transporter device starts the communication by broadcasting a proof request. The broadcast message is encrypted with the public key of the inspector to guarantee that it can only be decrypted by the inspector. The broadcast message contains the proof request, represented in the figure as PR, and the signature of the hash of the proof request, made with the private key of the transporter, to guarantee that the proof request was created by the transporter. The proof request contains pseudonyms of the devices, the identifiers of the inspection and trip, the nonce generated by the central ledger, the timestamp of the transporter device and its GPS coordinates.

When the inspector device receives a message from a device with the pseudonym of the transporter device, it validates if it is a proof request and, if correct, notifies the inspector to conduct the inspection. When the inspection is done, the outcome is reported in a message containing the proof, represented in the figure as proof, signed by the inspector. The message is encrypted with the public key of the transporter. The message is then sent through the established Bluetooth socket to the transporter device. The inspector device additionally sends a copy of the proof to the central ledger. The transporter device receives the proof, decrypts and validates it, adds the signature of the previous location item and sends it to the central ledger. If the transporter device did not receive the proof after successfully sending a proof request, it will request the central ledger to

produce a new nonce and pseudonym for that inspection. Messages with the same nonce, pseudonyms and identifiers are rejected as possible replay attacks.

Every message or object requires a digital signature to be considered authentic. A signature is computed by calculating the hash value of the object with the *SHA-256* algorithm. It is then encrypted with the RSA algorithm using the private key of the device that created the message.

4 Evaluation

The evaluation of the system focused on the following subjects:

- Are the location coordinates retrieved from Android mobile devices accurate enough for the system procedures?
- What are the best parameters for the selection of vehicles for inspection?
- Is the designed interaction protocol suitable for Bluetooth communication in an inspection scenario?

4.1 Location Accuracy

As the system uses the latest reported location from the on-board device of a vehicle, it is important to determine if mobile devices are capable of retrieving accurate location points. We set out two courses done with the STOP Transport application with different users. Course I was done using a mobile device inside of a automobile. Course II was done with 3 groups of two users, each one with a mobile device and each group traveling in a different bus. Having the users traveling through Course II in groups of two allowed us to assess possible discrepancies between devices performing the same route.

Upon visualizing the reported location points throughout the different courses, it is possible to detect some anomalies, but overall location points are close to the real trajectory. One of the performed courses contains a section inside of a tunnel and the mobile device that performed this course did not report any location point in this section. Figure 4 shows this anomaly, as the sections of the course that do not contain red dots are the sections inside the tunnel.

Another performed course has tall buildings in its surroundings which is known to affect GPS signal. Upon visualizing the several reported user trajectories in this course, we noticed moments where the location coordinates reported were in buildings. Although we cannot confirm it, we suspect, as Android also uses Wi-Fi fingerprint for location retrieval, that the devices might have detected known SSIDs and BSSIDs of Wi-Fi networks in these buildings. With a poor GPS connectivity, the devices might have calculated their positions inside of the building, taking into account the Wi-Fi networks detected.

Although visual analysis helps recognizing and understanding some issues, it does not gives us the overall accuracy levels of the reported location points. Therefore we have performed calculations on the retrieved location information of the devices. Table 1 shows the average distance between the reported and the exact trajectories of each user.

Fig. 4. Location detection issues inside of a tunnel.

User	**No. Points**	**Average distance (m)**
A	1244	4.64
B	1673	5.57
C	832	7.32
D	1375	8.19
E	1376	8.94
F	1820	18.97
G	1885	7.51

Table 1. Location retrieval accuracy results.

User A performed a course that was primarily highway courses with occasional city sections, while the rest of the users performed the same city course. The average distance of user A is lower than 5 meters, which we consider tolerable as the vehicle was mainly traveling between 90Km/h and 120Km/h and the city sections of the course were not surrounded by tall buildings and did not include narrow roads. With the rest of the users, we conclude that accuracy in a complete city environment is not as good as in a highway. Vehicle speeds are lower but the average distance was higher. All users of this course, except user F, had an average distance to the real trajectory between 5 and 9 meters. User F reported that his device may have a GPS malfunction because previous usages of navigation applications showed incorrect location positions. We conclude that this malfunction justifies the substantial average distance to the real trajectory, as user F always traveled with user G and this user had an overall average similar to the other users.

This accuracy assessment allows us to determine where the optimal location for a inspection site is. Inspectors should assess if the area inside the selected inspection selection range is not surrounded by tall buildings and does not include narrow roads. To our knowledge, heavy road vehicle inspections often occur in location that fulfills this requirement, as most of these vehicles do not travel in a constant city environment.

4.2 Vehicle Selection

We consider that the parameters defined in our architecture and by the Authority user should be evaluated as they influence the selection procedure. As vehicles will be traveling at different speeds and we want to have an efficient application, we want to assess if a fixed location retrieval rate should be implemented or not, taking into consideration that a higher location retrieval rate requires more processing from the mobile device and Central Ledger. The highest location retrieval rate possible will ensure the system has the most recent location of each vehicle, however it will demand more processing from the components. Before assessing this parameter, we wanted to confirm if the location retrieval rates defined in the Android implementation were in fact being fulfilled. Figure 5 illustrates the reported location retrieval rates. The horizontal axis represents the number of the reported location point and the vertical axis represents the time interval the location point took to be retrieved.

For all users, which had a 1 second rate set, there were some points with a substantial interval, however most of the points are in the exact 1 second mark. This showcases why the average rate is above one second but the percentage of points that have not fulfilled the set rate is minimal. We presume that a substantial location retrieval interval occurs when the GPS signal is not satisfactory, the device cannot use mobile data or the device is optimizing the battery consumption.

Results show that it is possible to have a one second retrieval rate, therefore we conclude that we can rely on the location retrieval rate defined on Android systems. However as mentioned, having a one second retrieval rate would create

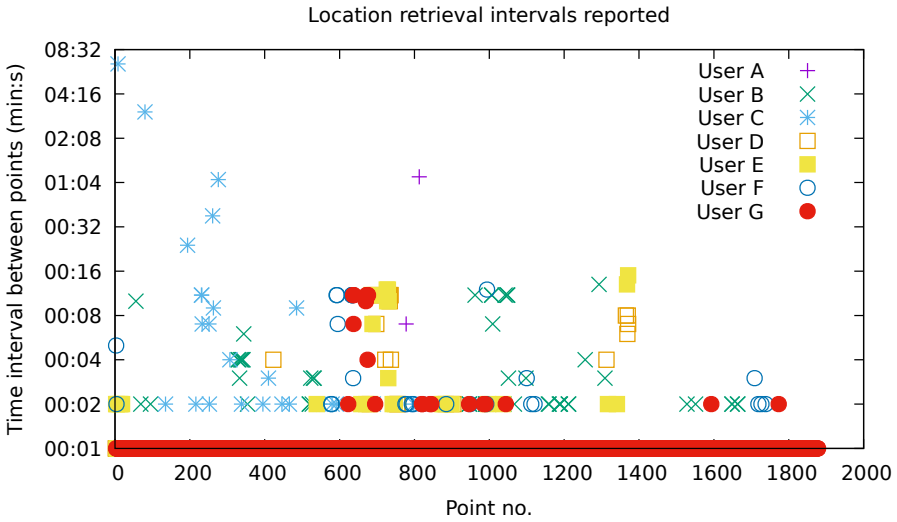

Fig. 5. Location retrieval intervals reported.

a considerable demand from the device and Central Ledger, despite guaranteeing that the system would have the most possible up-to-date location. We suggest that the location retrieval rate should be variable considering the speed of the vehicle. The device would constantly change its location retrieval rate to adapt to the speed at which the vehicle is moving. Speed can be calculated with the already retrieved points or with a specialized Android location tool-kit method[5].

We also performed inspection selection tests simultaneously with 6 users. Two Inspect users were at one checkpoint each and the distance between the two checkpoints was higher than the defined inspection selection range of 500 meters. The six Transport users started the course and the Inspect users were at the corresponding checkpoint, requesting an inspection every minute until the request was fulfilled. Inspection protocols would be performed with the two devices side-by-side.

Out of the inspections performed, there was an occasion where a user that had just been inspected was again selected for inspection. The issue occurred because the user was stopped due to traffic near the checkpoint, therefore he was eligible for selection due to the defined rule in the prototype. One improvement that could prevent this situation is to establish a minimum selection range, i.e., vehicles too close to the checkpoint would not be considered for inspection and there would not be any risk of a vehicle being selected and not being able to stop on time. The rest of the inspections performed did not have any anomalies.

[5] https://developer.android.com/reference/android/location/Location#getSpeed()

4.3 Bluetooth Inspection Interaction

We replicated an inspection area with a metal container similar to ones that carry goods in transportation vehicles. A Samsung Galaxy S9 device running Android 8 was used as the Transport device and a Nokia 8 device running Android 9 was used as the Inspect device. Both devices have Bluetooth 4.0. We positioned the Transport device in front of the container and proceeded to request an inspection in the Inspect device. The Transport device was selected.

In a typical inspection scenario, an inspector might move around the container and our architecture considers that a Bluetooth connection is maintained during this procedure. However a metal container might interfere with the Bluetooth connection. Therefore we performed several movements around the container to test if the connection was maintained.

The inspector was able to walk around the container and approve the inspection near the Carrier user. This procedure was done successfully 3 times. This did not happen when the inspector would stop for more than 5 seconds behind the container, the connection would be lost. Therefore we conclude that the Bluetooth inspection protocol cannot consider that a Bluetooth connection is fully maintained during an inspection process, while the inspector moves to perform the inspection. A possible change to the protocol would be to divide it in two phases. After ending the inspection procedure, the inspector heads towards the driver and approves the inspection to send the proof.

4.4 Discussion

We evaluated important features of Android devices used for our prototype, specifically location retrieval and Bluetooth communication. We concluded that in a highway course location points are accurate. Inside tunnels, however, devices cannot retrieve location information because they cannot receive signal from the GPS satellites. In a city course we concluded that GPS signal strength varies and the device may report location points outside of roads for example because of the obstructions caused by buildings, for example. The system will operate better on roads outside of cities or in locations without GPS obstacles.

Regarding the location retrieval rate, we found the results to be satisfactory as the Android devices were able to report most of the location points at the defined location rate. We suggest a variable location retrieval rate for better device optimization.

Upon testing the initial selection rules implemented, we proposed that the selection rule should be composed of maximum and minimum inspection selection range and a estimated time of arrival with a route planning procedure. This allows vehicles to be notified on time and guarantees that a selected vehicle does not need to change its route to reach the checkpoint.

As a result of the Bluetooth experiments, we redesigned the protocol to be divided in two phases, with separate Bluetooth connections, one for the start and another for the completion of the inspection.

5 Conclusion

This paper described the architecture, implementation and evaluation of the STOP system. The system uses the location from on-board mobile devices to track incoming vehicles to inspection sites and location proofing to digitally certify the location chain and the inspection data. The evaluation of the prototype provided insights regarding the feasibility of this type of system and the location retrieval features of Android devices.

Acknowledgements

This work was supported by national funds through FCT, Fundação para a Ciência e a Tecnologia, under project UIDB/50021/2020 and through project with reference PTDC/CCI-COM/31440/2017 (SureThing).

References

1. Bajaj, R., Ranaweera, S.L., Agrawal, D.P.: GPS: location-tracking technology. Computer **35**, 92–94 (2002)
2. Canlar, E.S., Conti, M., Crispo, B., Di Pietro, R.: CREPUSCOLO: a Collusion Resistant Privacy Preserving Location Verification System. In: 2013 International Conference on Risks and Security of Internet and Systems (CRiSIS) (2013)
3. CBS: N.J. Man In A Jam, After Illegal GPS Device Interferes With Newark Liberty Operations (2013), https://newyork.cbslocal.com/2013/08/09/n-j-man-in-a-jam-after-illegal-gps-device-interferes-with-newark-liberty-operations/
4. eMarketer: Maps and Navigation Apps: Discovery, Exploration Features Open Up Ad Opportunities (2018), https://www.emarketer.com/content/maps-and-navigation-apps
5. Ferreira, J., Pardal, M.L.: Witness-based location proofs for mobile devices. In: 17th IEEE International Symposium on Network Computing and Applications (NCA) (11 2018)
6. GPS.gov: Gps space segment (2019), https://www.gps.gov/systems/gps/space/
7. Hill, K.: Jamming GPS Signals Is Illegal, Dangerous, Cheap, and Easy (2017), https://gizmodo.com/jamming-gps-signals-is-illegal-dangerous-cheap-and-e-1796778955
8. Humphreys, T.: Statement on the vulnerability of civil unmanned aerial vehicles and other systems to civil GPS spoofing. Tech. rep., The University of Texas at Austin (2012)
9. Hynes, M., Miller, B., Barrett, M.: GPS Tracker US Patent (2003)
10. Krawczyk, H., Paterson, K.G., Wee, H.: On the security of the TLS protocol: A systematic analysis. In: CRYPTO 2013: Advances in Cryptology. pp. 429–448 (2013)
11. Loten, A.: Life on the Road Gets a Little Easier as Truckers Adopt Digital Technology (2019), https://www.wsj.com/articles/life-on-the-road-gets-a-little-easier-as-truckers-adopt-digital-technology-11559727001
12. Narain, S., Ranganathan, A., Noubir, G.: Security of gps/ins based on-road location tracking systems. In: 2019 IEEE Symposium on Security and Privacy (SP) (2019)

13. Onishi, H., Yoshida, K., Kato, T.: Gnss vulnerabilities and vehicle applications. In: 2016 13th Workshop on Positioning, Navigation and Communications (WPNC) (2016)
14. Ridester: Inside the Ridesharing Revolution: 2018 Edition (2018), https://www.ridester.com/2018-rideshare-infographic/
15. Saroiu, S., Wolman, A.: Enabling new mobile applications with location proofs. In: ACM (ed.) Proceedings of the 10th workshop on Mobile Computing Systems and Applications. p. 9 (2009)
16. The University of Texas at Austin: UT Austin Researchers Successfully Spoof an $80 million Yacht at Sea (2013), https://news.utexas.edu/2013/07/29/ut-austin-researchers-successfully-spoof-an-80-million-yacht-at-sea/
17. Zeng, K., Shu, Y., Liu, S., Dou, Y., Yang, Y.: A Practical GPS Location Spoofing Attack in Road Navigation Scenario. In: ACM Workshop on Mobile Computing Systems and Applications (HotMobile) (2017)
18. Zhu, Z., Cao, G.: Applaus: A privacy-preserving location proof updating system for location-based services. In: IEEE Conference on Computer Communications (INFOCOM) 2011 (2011)