Internet of Vehicles and Autonomous Connected Car - Privacy and Security Issues

Joshua Joy
UCLA
Email: jjoy@cs.ucla.edu

Mario Gerla
UCLA
Email: gerla@cs.ucla.edu

***Abstract*—In the Intelligent Vehicle Grid, the car is becoming a formidable sensor platform, absorbing information from the environment, from other cars (and from the driver) and feeding it to other cars and infrastructure to assist in safe navigation, pollution control and traffic management. The Vehicle Grid essentially becomes an Internet of Things (IOT), which we call Internet of Vehicles (IOV), capable to make its own decisions about driving customers to their destinations. Like other important IOT examples (e.g., smart buildings), the Internet of Vehicles will not merely upload data to the Internet using V2I. It will also use V2V communications between peers to complement on board sensor inputs and provide safe and efficient navigation. In this paper, we first describe several vehicular applications that leverage V2V and V2I. Communications with infrastructure and with other vehicles, however, can create privacy and security violations. In the second part of the paper we address these issues and more specifically focus on the need to guarantee location privacy to mobile users. We argue on the importance of creating public, open "smart city" data repositories for the research community and propose privacy preserving techniques for the anonymous uploading of urban sensor data from vehicles.**

I. Introduction

Academic researchers are becoming increasingly interested in studying smart city behaviors, like pedestrians, drivers, and traffic, city resources (eg energy) and city environment (eg pollution, noise). These studies are based on Open Shared Data made available by several Smart City testbeds around the country. To this end, Open Data Science enables researchers to collect the data, process it with data mining and ML techniques and create accurate models that allow them to realistically validate smart city design methodologies. This requires the collection of data from sensors, cameras embedded in the "smart city" (eg, smart building, smart transport, smart instrumented crowd) to derive models of behavior, predict trends, optimize system management and detect the onset of attacks.

In the past, much research in these environments has been performed in simulation or in carefully controlled and expensive testbed systems. Increasingly, research addressing these challenges must be performed in more realistic environments. However, doing so poses serious questions, many of which relate to cybersecurity aspects of their work. Researchers will need to deploy their technologies in real vehicles, in real roads and cities (or, at least, on-campus roads used for general purposes), to demonstrate that they are not mere toys suitable for use in a simulated environment or highly controlled test track. The necessities of the research require testing in such uncontrolled environments, yet such environments pose serious risks to the validity of the experiments and the integrity and privacy of the data gathered by them.

In autonomous vehicles, some of the data available on the cars will be used to examine the behavior of drivers and their reactions to traffic conditions. These measurements will be helpful to reassure future customers that autonomous vehicles are in fact safe. For example, today most manufacturers expect the drivers to keep their hands on the steering wheel. Accordingly, they are developing an "early warning" system. When the system issues an alarm, the driver should be ready to take over because of the announced danger. At times, the cautious driver will step on the brakes before the alarm, even if it was not necessary. Other times, the driver intervenes after hearing the alarm, but the alarm was unjustified. Other times again the alarm is justified but the driver is not alert nor ready. An important measurement will be the false positive rate, that is the fraction of manual interventions that were not justified by traffic conditions. Measuring the autonomous/manual switchover behavior of drivers is extremely useful to design the early warning system in the car, i.e. the system that instructs the driver to take over the controls. The warning threshold must be tuned to the particular driver's "personality", say cautious or aggressive, else the driver will not use it! Thus, these measurements are used both to classify the driver and tune the system.

In principle, these experiments can be run in a simulator. Namely, the driver is in front of a "game type" simulator that replicates realistic road conditions. The autonomous system then captures these conditions and triggers the alarm, when it determines that the traffic situation (eg pedestrians in the crosswalk, bicycles competing for the road, chaotic traffic jams, etc) may pose serious liability risks or may overwhelm the self driving controls.

The simulator solves the data privacy issue since no real data is used. However, the simulator will have serious limitations. It will not be able to accurately reflect the noise that in real life distorts the signals (radio, video, LIDAR, etc) captured by the car. Moreover, the driver that sits at a simulator tends to behave in a very different way than the driver at the wheel in real traffic. Because of the above reasons, these driver behavior experiments must be carried out in vivo. The data collected in these experiments will be of interest to a large cross section of Smart City researchers, from robotics experts to traffic engineering and behavioral scientists.

The autonomous/manual switchover experiment is a rather complex experiment, requiring the collection of many signals, namely: position and speed of the vehicle in question and of all the surrounding vehicles (to assess traffic context, congestion, etc.); road map and road conditions; road signal states such as intersection signals; green waves; acceleration and brake interventions by the autonomous car and by the driver, respectively; the triggering of cockpit alarms; the degree of attention of the driver (eg, reading, watching movie, talking, hands on/off wheel and eyes on/off the road, etc). This data may be collected, say, every 100 ms (human reaction time) and packaged for delivery to the Measurement Collection Center. Using this data, an estimate of what would have happened without manual intervention can be generated a posteriori. The alarm threshold must be low enough for timely warning to the driver for safe intervention. It should not be too low otherwise the superfluous triggering of alarms will exasperate the driver!

Looking at the problem from another viewpoint, the collection of this data poses privacy challenges. If the collection system is always on, the auto dealer, say, will know intimately the commuting habits of the driver, raising serious privacy concerns. The issue then is to extract only the data that is strictly necessary to build the model and carry out the optimization, while fuzzing the unnecessary details. This anonymization issue will be addressed in the last section of the paper, namely, private data collection. The next section provides the background of the vehicular system evolution to IOV and Vehicular Cloud, the establishment of V2V/V2I communications standards and the emergence of vehicular applications that exploit these standards. As we shall see, it will be some of these applications that carry intrinsic security and privacy risks.

II. Evolution from vehicles to IOV and the vehicular Cloud

The urban fleet of vehicles is rapidly evolving from a collection of sensor platforms that provide information to drivers and upload filtered sensor data (e.g., GPS location, road conditions, etc.) up the Internet, to a network of autonomous vehicles that exchange their sensor inputs among each other in order to optimize several different utility functions. One such function, and probably the most important for autonomous vehicles, is prompt delivery of the passengers to destination with maximum safety and comfort and minimum impact on the environment. We are witnessing today in the vehicle fleet the same evolution that occurred ten years ago in the sensor domain from Sensor Web (i.e., sensors are accessible from the Internet to get their data) to Internet of Things (the components with embedded sensors are networked with each other and make intelligent use of the sensors). In the intelligent home, the IOT formed by the myriad of sensors and actuators, that cover the house internally and externally, can manage all the utilities in the most economical way, with maximum comfort to residents, with virtually no human intervention. Similarly, in the modern energy grid, the IOT consisting of all components large and small can manage power loads in a safe and efficient manner, with the operators now playing the role of observers. In the vehicular grid, the Internet of Vehicles (IOV) is more complex than the smart home and smart energy grid IOTs. In fact there are many different “Things” in the IOV. Namely:

1) External sensors (GPS, cameras, lidars etc)
2) Internal automotive sensors and actuators (brakes, steering wheel, accelerator, etc)
3) Internal cockpit sensors (driver’s state of health, alertness, tone of voice, health sensors like the Ford heart monitor seat, etc)
4) The Driver’s messages (tweets, Facebook, other crowd-sourced info, etc) are also measurable sensor outputs that characterize the state of the system and of the driver.
5) Vehicle’s beacons, alarms report on the Vehicle state; say, position, key internal parameters, possible dangers, etc.

This complex picture (of sensors and stakeholders) tells us that IOVs are different from other IOTs. What sets them apart from other IOTs (and requires more “things”) are the following properties/characteristics:

1) **Mobility**:
   a) IoVs Must manage mobility and wireless bottleneck
   b) They must guarantee motion privacy
2) **Safety critical Applications**
   a) This implies low latency requirements
3) **V2V**:
   a) V2V is critical for safety, low latency apps (eg, platoons)
4) **Attacks**:
   a) Security and DDoS attacks (from hackers and form malicious agents) are made possible by V2V.

In the vehicular network, like in all the other IOTs, when the human control is removed, the autonomous vehicles must efficiently cooperate to maintain smooth traffic flow in roads and highways. Visionaries predict that the self-driving vehicles will behave much better than human drivers, handling more traffic with lower delays, less pollution and better driver and passenger comfort. However, the complexity of the distributed control of hundreds of thousands of cars cannot be taken lightly. If a natural catastrophe suddenly happens, say an earthquake, the vehicles must be able to coordinate the evacuation of critical areas in a rapid and orderly manner. This requires the ability to efficiently communicate with each other and also to discover where the needed resources are (e.g., ambulances, police vehicles, information about escape routes, images about damage that must be avoided, etc.). Moreover, the communications must be secure, to prevent malicious attacks that in the case of autonomous vehicles could be literally deadly since there is no standby control and split second chance of intervention by the driver (who meantime may be surfing the web).

All of these functions, from efficient communications to distributed processing over various entities, will be provided by an emerging compute, communications and storage platform specifically designed for vehicles—the *Vehicular Cloud*. The Vehicular Cloud is justified by several observed trends:

1) Vehicles are becoming powerful sensor platforms
   a) GPS, video cameras, pollution, radars, acoustic, etc
2) Spectrum is becoming scarce => Internet upload of all the sensor outputs expensive and besides infeasible
3) More data is cooperatively processed by vehicles rather than uploaded to Internet:
   a) road alarms (pedestrian crossing, electr. brake lights), platoon coordination signals, intersection announcement, etc
4) Distributed Surveillance (video, mechanical, chemical sensors)
   a) Must be locally supported, deployed
5) Protection from DDoS attacks, must be done locally, via the Vehicular Cloud

To support the above functions, the mobile Vehicle Cloud provides several basic services, from routing to content search, through standard, open interfaces that are shared by all auto manufacturers.

III. Emerging Applications

A number of applications have emerged in recent years, leveraging V2I and V2V

- Safe Navigation
- Crash prevention; platoon stability; shockwaves
- Content Download/Upload
- News, entertainment, location relevant info download; ICN
- Video upload (eg remote drive, Pic-on-wheels, accident scene, etc)
- Sensor Data gathering
- Forensics; driver behavior; traffic crowdsource; ICN
- Privacy preserving data analysis
- Intelligent Transport
- Efficient routing to mitigate congestion/pollution
- Vehicle Autonomy
- Autonomous, self driving vehicles, etc

We will review some of these application next, with focus on autonomous vehicle impact and security privacy concerns.

A. V2V for Safe and Efficient navigation

Safe navigation generally includes these features:

- Forward Collision Warning,
- Intersection Collisions
- Traffic shockwaves
- Platooning (e.g., trucks)
- Advisories about road perils such as“Ice on bridge”, “Congestion ahead”?

Figure 1 illustrates the use of DSRC beacons to support safe distance keeping within platoons. Automakers, however, have been reluctant to use V2V or V2I to implement safety procedures for customers. The major reason is liability in case of accidents. Say, suppose the car in front maliciously tells my car to brake because of a sudden obstacle, while there is no obstacle. My car stops and is hit by the car behind. I am technically at fault.

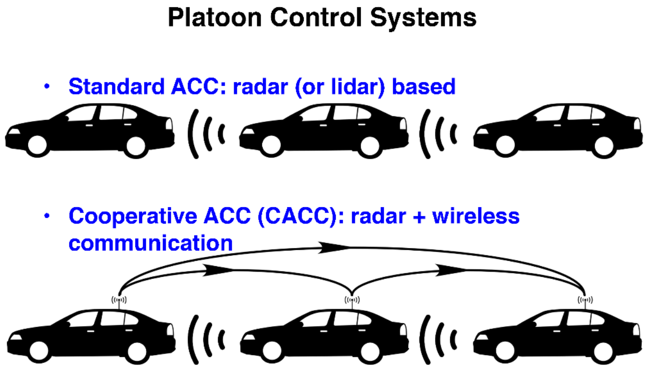

Platoon Control Systems
Standard ACC: radar (or lidar) based
Cooperative ACC (CACC): radar + wireless communication

Figure 1: **Platoon Control Systems.** Cooperative adaptive cruise control.

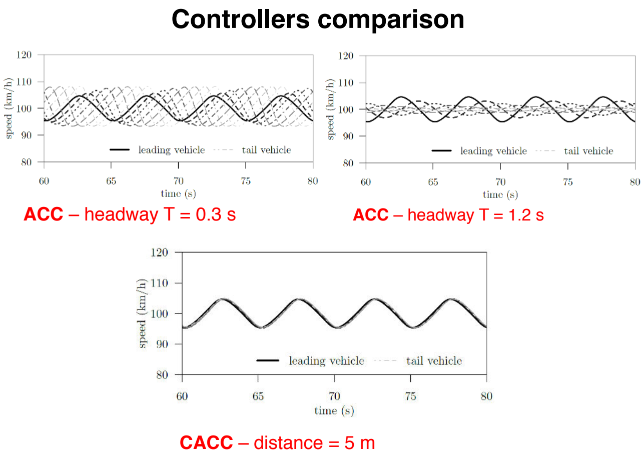
More fundamentally, the autonomous vehicle manufacturers today relay only on their own sensors (e.g., Acoustic, Laser, Lidar, Visual Communications, Video Cameras, GPS, accelerometer, etc. – the GOOGLE car has over 200 sensors on board). If their sensors are faulty, they accept the liability. In view of this attitude, we should ask the question: why should manufacturers need V2V to Protect Self Driving Vehicles and all the advanced cruise control vehicles, since they have all the sensors at their disposal to prevent crashes?

Unfortunately, for the self-driving car manufacturers, the future contradicts this assumption. Consider these points: Thousands of autonomous vehicles will share the road in 5-10 years. Advance platooning will be necessary to justify the cost of these cars. Very high speeds with small car intergaps will be required to efficiently utilize highways. Isolating the autonomous cars in “sensor cocoons” without car to car communications will cost them the ability to platoon efficiently. It will require them to maintain 40m gaps between cars (as depicted in Figure 1) with serious efficiency and safety consequences! Bottom line: Autonomous vehicles will be forced to adopt V2V.

B. Intelligent navigation - from Dash Express to WAZE

Dash Express revolutionized the navigator business in 2008 by exploiting Time and Speed crowdsensing by its customers. Namely, cars periodically submit Time and Speed reports. Routing instructions to cars are updated using customers reports. See Figure 3. Current Navigators are mostly based on the same crowdsourcing model. For example WAZE (by Google) is implemented in the Cloud and is accessed via V2I (DSRC, WIFI or LTE).

The Centralized, Cloud based Navigator Server allows for many advanced features like: optimization of routes; Mini-



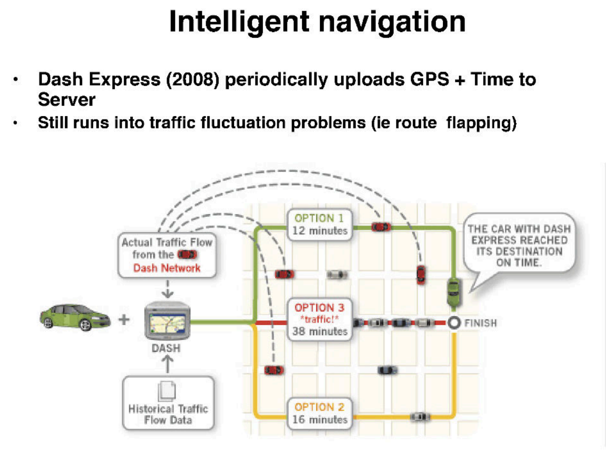
Controllers comparison

ACC – headway T = 0.3 s

ACC – headway T = 1.2 s

CACC – distance = 5 m

Figure 2: **CACC vs ACC.** Controllers comparison.



Intelligent navigation

- Dash Express (2008) periodically uploads GPS + Time to Server
- Still runs into traffic fluctuation problems (ie route flapping)

Actual Traffic Flow from the Dash Network
DASH
Historical Traffic Flow Data
OPTION 1 12 minutes
OPTION 3 "traffic!" 38 minutes
OPTION 2 16 minutes
FINISH
THE CAR WITH DASH EXPRESS REACHED ITS DESTINATION ON TIME.

Figure 3: **Dash Express.** Intelligent navigation.

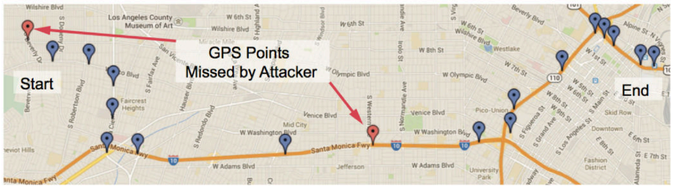


Figure 4: **Waze Attacks [1].** ***Privacy Attack***: Waze allows remote customers to view current traffic in an arbitrary window. By moving window, attacker tracks victim. ***DDoS Attack***: the malicious customer impersonates multiple WAZE vehicles in small area, simulating traffic bottleneck.

mization of pollution (eco routing); Traffic flow balancing; Arrival time control on preferred routes; Traffic and congestion control. However, centralized traffic management cannot react promptly to local traffic perturbations (WAZE has a reaction time of 10-15min). Thus, a doubled parked truck in the next block; a recent traffic accident; a sudden queue of traffic on the preplanned route forces me to wait up to 15 min before I find the cause with WAZE. The Internet based Navigator Server cannot micro-manage traffic for scalability reasons.

Enter distributed traffic management! The distributed approach is a good complement of centralized supervision, see Leontiadis et. al. [2]. In the referenced paper the distributed, totally crowdsourced scheme “CATE: Comp Assisted Travel Environment” is introduced. In a nutshell, Vehicles crowd source traffic information and build traffic load data base:

1) estimate traffic from own travel time;
2) share it with neighboring vehicles (with V2V in an ad hoc manner)
3) dynamically recompute the best route to destination

Interestingly, both Centralized and Distributed navigation systems lead to security issues. In a separate paper we describe the vulnerability of the distributed scheme to BOTNET attacks launched by compromised cars [xyz]. The compromised cars manage to propagate false information and lure honest cars in a major traffic bottleneck in a couple of minutes! The Centralized Navigator protects from BOTNETs, but exposes customers to Privacy attacks, as described below.

C. Security Problem: Privacy violations in V2I communications

The Centralized Navigators, have security problems and can lead to Communication Privacy Violations. In fact, with centralized navigators, Cars upload their position, velocity and intended destination to the Navigator. For example:

- WAZE delivers vehicle position and traffic conditions to GOOGLE traffic
- UBER vehicles upload passenger and vehicle status to UBER Server
- LTE providers can trilaterate and localize the vehicles as they connect to the Internet

The collected data can be used by the Navigation servers to track users and discover their habits and favorite hot spots. Naturally, Service Providers like GOOGLE, UBER and Cellular Companies are committed to protect customer privacy. However, privacy guarantees have been often broken in the past (intentionally or by mistake). Examples of privacy violations with WAZE are shown in Figure 4. In the Waze Privacy Attack, Waze allows remote customers to view current traffic in an arbitrary window. By moving the window, the attacker tracks the victim. In the Waze DDoS Attack, the malicious customer impersonates multiple WAZE vehicles in a small area, simulating traffic bottleneck.

In the remainder of this paper, we focus on the Privacy violation issue, a problem common to all applications that upload mobile data from IOT or IOV to Servers in the Cloud. We formally define the problem, introduce an efficient, scalable solution, Haystack, and evaluate it on the Southern California PEMS vehicle sensor loop data from CALTRAN.

IV. Related Work

Differential privacy [3], [4], [5], [6] has been proposed as a mechanism to privately share data such that anything that can be learned if a particular data owner is included in the database can also be learned if the particular data owner is not included in the database. To achieve this privacy guarantee, differential privacy mandates that only a sublinear number of queries have access to the database and that noise proportional to the global sensitivity of the counting query is added (independent of the number of data owners).

Distributional privacy [7] is a privacy mechanism which says that the released aggregate information only reveals the underlying ground truth distribution and nothing else. This protects individual data owners and is strictly stronger than differential privacy. However, it is computationally inefficient though can work over a large class of queries known as Vapnik-Chervonenkis (VC) dimension, which is one measurement of learning theory in regards to machine learning.

Zero-knowledge privacy [8] is a cryptographically influenced privacy definition that is strictly stronger than differential privacy. Crowd-blending privacy [9] is weaker than differential privacy; however, with a pre-sampling step, it satisfies both differential privacy and zero-knowledge privacy. These mechanisms do not add noise linear in the number of data owners and rely on aggressive sampling, which negatively impact the accuracy estimations.

The randomized response based policies [10], [11], [12], [13] satisfies the differential privacy mechanism as well as stronger mechanisms such as zero-knowledge privacy. However, the accuracy of the randomized response mechanism quickly degrades unless the coin toss values are configured to large values (e.g., greater than 80%).

V. Architecture

A. System Model

We now describe the system model. The system proceeds in epochs. Every epoch t each data owner i generates personal data (e.g. current location) on a device under their control (e.g., a smartphone or vehicle). There is a personal data stream $\mathbf{X} = \{X_{t,i}\}$ where $0 \leq t < T$.

The data owner's personal data $X_{t,i} = \{x_{t,i,l}\}$ where $0 \leq l < L$ is a bit vector of length L, where L is the number of queries (e.g., number of locations to monitor). While simple, binary classification can be extended to real-valued functions [14].

Next, each data owner generates a privatized (possibly randomized) view given their personal data $\mathbf{Z} = \{Z_{t,i}|X_{t,i}\}$ where $0 \leq t < T$. Each entry $Z_{t,i}$ is a random variable in t, and is independently distributed across each i entry though not identically distributed.

The contributions from data owners at epoch t are aggregated to learn statistics about the population. We model the privatized data collected at each epoch as a statistical database as a vector $\mathbf{Z_t} = (Z_{t,1}, ..., Z_{t,i})$ where each entry i has been contributed by a data owner. Each i row is protected due to the cryptographic private write and the aggregator does not know which data owner wrote each i entry.

B. System Goals

The goals for our system as follows. The protocol does not require a restart if a mobile node (e.g., smartphone or vehicle) enters or leaves. Messages received are processed in the same epoch in which they are received, regardless of whether they are at the beginning or end of the epoch.

In addition, the analysts are deemed to be reputable, e.g., Department of Transportation, National Institutes of Health, or Centers for Disease Control. Each analyst may own or contribute an aggregation server. Aggregation servers may also be contributed by privacy watchdog groups such as the Electronic Frontier Foundation (EFF).

C. Threat Model

The population (database) size or total number of participating data owners is not published or released. This mitigates auxiliary attacks whereby an adversary can utilize the database size (number of participants) to deduce if a particular individual is included [15], [16].

Data owners establish an independent TLS connection to each aggregator to transmit the shares confidentially. The TLS connection is long-lived to amortize the connection setup.

Aggregation parties may try to collude, though we assume there is at least one honest aggregation party which does not collude (e.g., a privacy watchdog like the EFF). Data owners may try to collude with the aggregation servers, though we assume there are at least two honest data owners who do not collude with the aggregators. Of course the larger the number of honest data owners, the stronger the privacy guarantees are for each data owner.

Aggregators are expected to be available and online, so we do not consider denial of service attacks whereby data owners are not able to transmit their responses. We assume aggregators do not corrupt the messages though they can attempt to read all messages. Privacy holds as long as there is at least one honest aggregator who does not collude.

D. Query

Queries are of the form of counting queries such as "What is the distribution of vehicles across New York City and how fast are the vehicles traveling?". The query is a two dimensional matrix where the first dimension is a list of locations and the second dimension is a range of vehicle speeds. Each vehicle truthfully responds by selecting the coordinate which matches their current location and speed. That is, the value of a single entry of the matrix is incremented by one. This results in a query of sensitivity of only *one*. More generally, a d-dimensional query can be supported to learn more complex distributions.

VI. Haystack Privacy

We now introduce the notion of Haystack Privacy. The main goal is to scale the the number of participants in order to strengthen privacy yet maintain accuracy.

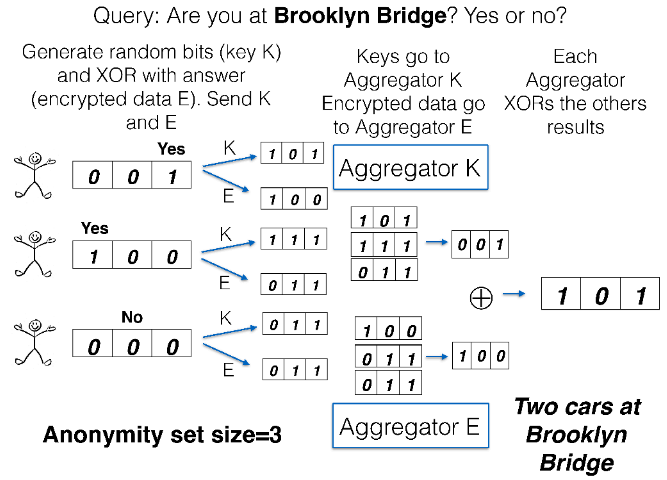

Query: Are you at **Brooklyn Bridge**? Yes or no?
Generate random bits (key K) and XOR with answer (encrypted data E). Send K and E
Keys go to Aggregator K Encrypted data go to Aggregator E
Each Aggregator XORs the others results
Yes
0 0 1
K
1 0 1
E
1 0 0
Aggregator K
Yes
1 0 0
K
1 1 1
E
0 1 1
1 0 1
1 1 1
0 1 1
0 0 1
⊕
1 0 1
No
0 0 0
K
0 1 1
E
0 1 1
1 0 0
0 1 1
0 1 1
1 0 0
Anonymity set size=3
Aggregator E
Two cars at Brooklyn Bridge

Figure 5: Each data owner selects an index uniformly at random to write their location status. The information-theoretic private write is protected as long as there is at least one honest aggregator which does not collude. The aggregators share their results to compute the final aggregate output.

In Haystack, data owners locally privatize their personal data independent of other data owners or centralized services. Second, data owners blend with, and are indistinguishable, from at least c crowds, where each crowd is composed of at least k data owners. Finally, data owners perform a cryptographic private write (similar to Figure 5) such that no aggregator knows where in the data structure (which row in Figure 6) a data owner wrote to. Effectively each data owner can be thought of as "hiding in a haystack".

The key contribution of Haystack Privacy is that everyone in the population participates. For example, suppose we crowdsource vehicle densities across New York City using the query in Figure 6. A data owner (vehicle) begins by answering the query "Am I at Brooklyn Bridge?". Prior work using the Laplace mechanism [6], [4] would have everyone at Brooklyn Bridge answer truthfully. Then, a small amount of privacy noise is added to protect privacy. In Haystack Privacy, all data owners respond to the query as seen in Figure 6. A small fraction of those *not* at Brooklyn Bridge will respond "Yes, I'm at Brooklyn Bridge". A small fraction *at* Brooklyn Bridge will respond "No, I'm not at Brooklyn Bridge". Both cases provide plausible deniability and are controlled by two different Bernoulli trials specified in the query. To estimate the aggregate count, the expected value of the privacy noise due to the Bernoulli trials is calculated and removed. One observation is the number of people at Brooklyn Bridge is fixed. While the number of people in any locale (e.g., Brooklyn Bridge) may be fixed, the inclusion of inputs from people not at that location enables us to leverage the law of large numbers to ensure that the estimated privacy noise approaches the expected value and thus preserves accuracy. In Section §VII, we show that, for our use case, Haystack Privacy preserves accuracy. That is, increasing participation improves the privacy yet maintains accuracy.

A. Mechanism

In the Haystack Privacy mechanism, each data owner independently and individually privatizes their data before privately writing to a cloud service which aggregates all the responses. The privatization is performed by a series of Bernoulli trials which randomizes the truthful answer in a particular way such that the final aggregation is able to calculate the expected value over the population as a Binomial distribution and then remove the expected value of the noise to approximate the ground truth value.

The main goal is to be able to increase the population participating such that the data owners blend with each other to provide privacy protection. As we control the randomization, we construct a mechanism whereby the error introduced by the sampling variance cancels out. We construct our mechanism as follows. Each data owner responds *twice* for the same query, though flips a single term to allow for the error cancelation. *YES* refers to the population that is at a particular location and conversely *NO* are those that are not.

$$YES_A\ Privatized\ Value = \begin{cases} \perp_1 & \text{with probability } \pi_{\perp_1} \\ \perp_1 & \text{with probability } \pi_Y \\ \perp_2 & \text{with probability } 1-\pi_{\perp_1}-\pi_Y \end{cases} \tag{1}$$

$$NO_A\ Privatized\ Value = \begin{cases} \perp_1 & \text{with probability } \pi_{\perp_1} \\ \perp_1 & \text{with probability } \pi_{\perp_N} \\ \perp_2 & \text{with probability } 1-\pi_{\perp_1}-\pi_{\perp_N} \end{cases} \tag{2}$$

$$YES_B\ Privatized\ Value = \begin{cases} \perp_1 & \text{with probability } \pi_{\perp_1}-\pi_Y \\ \perp_2 & \text{with probability } \pi_Y \\ \perp_2 & \text{with probability } 1-\pi_{\perp_1} \end{cases} \tag{3}$$

$$NO_B\ Privatized\ Value = \begin{cases} \perp_1 & \text{with probability } \pi_{\perp_1}-\pi_{\perp_N} \\ \perp_2 & \text{with probability } \pi_{\perp_N} \\ \perp_2 & \text{with probability } 1-\pi_{\perp_1} \end{cases} \tag{4}$$

The expected values are as follows:

$$\begin{aligned} E[\perp_{1_A}] &= (\pi_{\perp_1}+\pi_Y)\times YES + (\pi_{\perp_1}+\pi_N)\times NO \\ &= \pi_{\perp_1}\times YES + \pi_Y\times YES + \pi_{\perp_1}\times NO + \pi_N\times NO \\ &= \pi_{\perp_1}\times TOTAL + \pi_Y\times YES_A + \pi_N\times NO \\ &= \pi_{\perp_1}\times TOTAL + \pi_Y\times YES_A + \pi_N\times TOTAL - \pi_N\times YES \end{aligned} \tag{5}$$

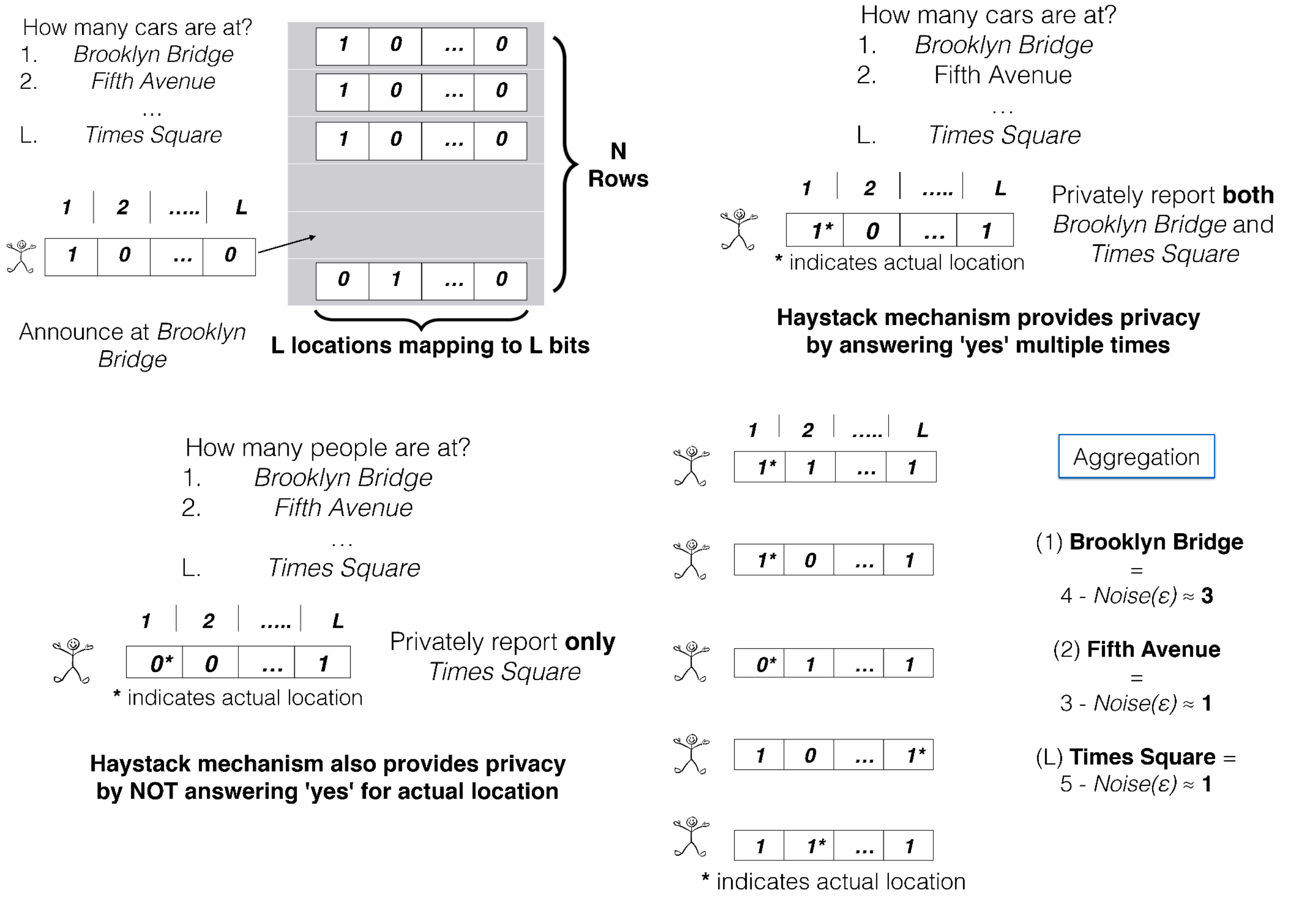


Figure 6: **Haystack mechanism.** Each data owner selects a row uniformly at random from an $N \times L$ matrix, where N is the number of rows (greater than the number of data owners), and performs a cryptographic private write. A crowdsourced regional query maps each location to a bit. The query specifies two Bernoulli trials to privately respond. Each privatized response is protected by reporting multiple locations and by sometimes NOT reporting the actual location. Aggregation is performed for each location. The final aggregate output is calculated by subtracting the expected value of the privacy noise due to the Bernoulli trials.

$$\begin{aligned}
E[\perp_{1_B}] &= (\pi_{\perp_1} - \pi_Y) \times YES + (\pi_{\perp_1} - \pi_N) \times NO \\
&= \pi_{\perp_1} \times YES - \pi_Y \times YES + \pi_{\perp_1} \times NO - \pi_N \times NO \\
&= \pi_{\perp_1} \times TOTAL - \pi_Y \times YES - \pi_N \times NO \\
&= \pi_{\perp_1} \times TOTAL - \pi_Y \times YES - (\pi_N \times TOTAL - \pi_N \times YES) \\
&= \pi_{\perp_1} \times TOTAL - \pi_Y \times YES - \pi_N \times TOTAL + \pi_N \times YES
\end{aligned} \tag{6}$$

$$\begin{aligned}
E[\perp_{2_A}] &= (1 - \pi_{\perp_1} - \pi_{\perp_Y}) \times YES + (1 - \pi_{\perp_1} - \pi_N) \times NO \\
&= (1 - \pi_{\perp_1}) \times TOTAL - \pi_Y \times YES - \pi_N \times NO \\
&= (1 - \pi_{\perp_1}) \times TOTAL - \pi_Y \times YES - \\
&(\pi_N \times TOTAL - \pi_N \times YES) \\
&= (1 - \pi_{\perp_1}) \times TOTAL - \pi_Y \times YES - \\
&\pi_N \times TOTAL + \pi_N \times YES
\end{aligned} \tag{7}$$

$$\begin{aligned}
E[\perp_{2_B}] &= (1 - \pi_{\perp_1}) \times YES + \perp_Y \times YES + \\
&(1 - \pi_{\perp_1}) \times NO + \perp_N \times NO \\
&= (1 - \pi_{\perp_1}) \times TOTAL + \pi_Y \times YES + \pi_N \times NO \\
&= (1 - \pi_{\perp_1}) \times TOTAL + \pi_Y \times YES + \\
&\pi_N \times TOTAL - \pi_N \times YES
\end{aligned} \tag{8}$$

We should now be able to subtract either pairs of expected values and solve for YES. Either $E[\perp_{1_A}] - E[\perp_{1_B}]$ or $E[\perp_{2_A}] - E[\perp_{2_B}]$. In order to solve the system of equations, the count $\pi_N \times TOTAL$ is revealed. Note this value only includes responses from the NO population and not the YES population. We then eliminate this value from both systems of equations and solve for YES.

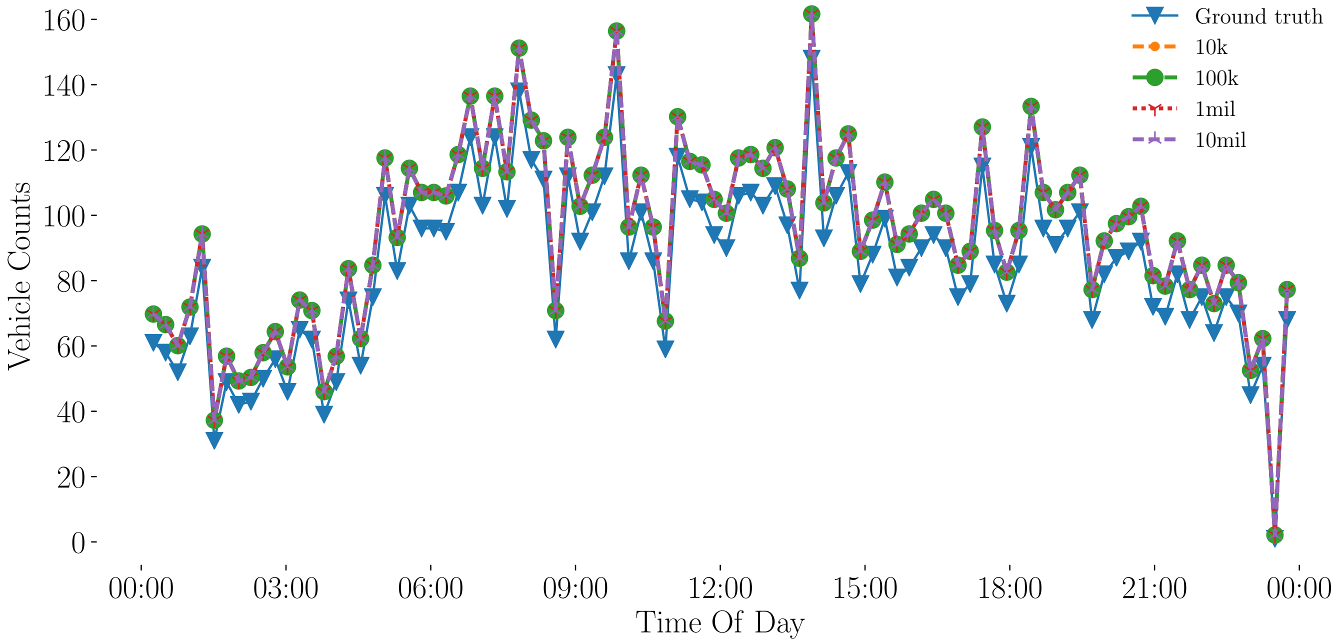


Figure 7: **Accuracy.** Ground truth versus privatized vehicle counts with a 95% confidence interval. The population not at the station being monitored (i.e., *No* population) increases by expanding the query to include additional vehicles not at the particular location.

VII. Evaluation

We evaluate the Haystack mechanism over a real dataset rather than arbitrary distributions. We utilize the California Transportation Dataset from magnetic pavement sensors [17] collected in LA\Ventura California freeways [18]. There are a total of 3,220 stations and 47,719 vehicles total. We assign virtual identities to each vehicle. Each vehicle announces the station it is currently at.

Figure 7 compares the Haystack mechanism to the ground truth data over a 24 hour time period with a confidence interval of 95%. We select a single popular highway station that collects and aggregates vehicle counts every 30 seconds. (For reasons of illustration we graph a subset of points for readability). We assign virtual identifiers and have every vehicle at the monitored station truthfully report “Yes” while every other vehicle in the population truthfully reports “No”. The Haystack Privacy mechanism then privatizes each vehicle’s response. Traffic management analyzing the privatized time series would be able to infer the ebbs and flow of the vehicular traffic.

The coin toss probabilities are fixed with parameters $\pi_{\perp_1} = 0.15$, $\pi_{Yes} = 0.20$, $\pi_{No} = 0.25$. The additional blending population (those not at the particular location) starts with 10 thousand and increases to 10 million vehicles.

VIII. Conclusion

In this paper we have demonstrated that data can be privately collected into a common open data vehicular database to be shared amongst multiple collaborators. We introduce the concept of Haystack Privacy, which strengths in privacy strength as more data owners participate yet maintains accuracy. We believe this is a new direction in open data vehicular research.

References

[1] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, “Defending against sybil devices in crowdsourced mapping services,” in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys 2016, Singapore, June 26-30, 2016*, R. K. Balan, A. Misra, S. Agarwal, and C. Mascolo, Eds. ACM, 2016, pp. 179–191. [Online]. Available: http://doi.acm.org/10.1145/2906388.2906420

[2] I. Leontiadis, G. Marfia, D. Mack, G. Pau, C. Mascolo, and M. Gerla, “On the effectiveness of an opportunistic traffic management system for vehicular networks,” *IEEE Trans. Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1537–1548, 2011. [Online]. Available: http://dx.doi.org/10.1109/TITS.2011.2161469

[3] C. Dwork, “Differential privacy,” in *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, ser. Lecture Notes in Computer Science, M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, Eds., vol. 4052. Springer, 2006, pp. 1–12. [Online]. Available: http://dx.doi.org/10.1007/11787006_1

[4] C. Dwork, F. McSherry, K. Nissim, and A. Smith, “Calibrating noise to sensitivity in private data analysis,” in *TCC*, 2006.

[5] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, “Our data, ourselves: Privacy via distributed noise generation,” in *EUROCRYPT*, 2006.

[6] C. Dwork and A. Roth, “The algorithmic foundations of differential privacy,” *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211–407, 2014. [Online]. Available: http://dx.doi.org/10.1561/0400000042

[7] A. Blum, K. Ligett, and A. Roth, “A learning theory approach to noninteractive database privacy,” *J. ACM*, vol. 60, no. 2, pp. 12:1–12:25, 2013. [Online]. Available: http://doi.acm.org/10.1145/2450142.2450148

[8] J. Gehrke, E. Lui, and R. Pass, "Towards privacy for social networks: A zero-knowledge based definition of privacy," in *Theory of Cryptography - 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28-30, 2011. Proceedings*, ser. Lecture Notes in Computer Science, Y. Ishai, Ed., vol. 6597. Springer, 2011, pp. 432–449. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-19571-6_26

[9] J. Gehrke, M. Hay, E. Lui, and R. Pass, "Crowd-blending privacy," in *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, ser. Lecture Notes in Computer Science, R. Safavi-Naini and R. Canetti, Eds., vol. 7417. Springer, 2012, pp. 479–496. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-32009-5_28

[10] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[11] J. A. Fox and P. E. Tracy, *Randomized response: a method for sensitive surveys*. Beverly Hills California Sage Publications, 1986.

[12] B. G. Greenberg, A.-L. A. Abul-Ela, W. R. Simmons, and D. G. Horvitz, "The unrelated question randomized response model: Theoretical framework," *Journal of the American Statistical Association*, vol. 64, no. 326, pp. 520–539, 1969.

[13] A. C. Tamhane, "Randomized response techniques for multiple sensitive attributes," *Journal of the American Statistical Association*, vol. 76, no. 376, pp. 916–923, 1981.

[14] N. H. Bshouty and V. Feldman, "On using extended statistical queries to avoid membership queries," *Journal of Machine Learning Research*, vol. 2, pp. 359–395, 2002. [Online]. Available: http://www.jmlr.org/papers/v2/bshouty02a.html

[15] K. Chaudhuri and N. Mishra, "When random sampling preserves privacy," in *Advances in Cryptology - CRYPTO 2006, 26th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2006, Proceedings*, ser. Lecture Notes in Computer Science, C. Dwork, Ed., vol. 4117. Springer, 2006, pp. 198–213. [Online]. Available: http://dx.doi.org/10.1007/11818175_12

[16] D. Kifer and A. Machanavajjhala, "Pufferfish: A framework for mathematical privacy definitions," *ACM Trans. Database Syst.*, vol. 39, no. 1, p. 3, 2014. [Online]. Available: http://doi.acm.org/10.1145/2514689

[17] "California Department of Transportation," http://pems.dot.ca.gov/. [Online]. Available: http://pems.dot.ca.gov/

[18] "Google's Waze announces government data exchange program with 10 initial partners," http://www.dot.ca.gov/cwwp/InformationPageForward.do. [Online]. Available: http://www.dot.ca.gov/cwwp/InformationPageForward.do