BLAST: Blockchain-based Trust Management in Smart Cities and Connected Vehicles Setup

Farah Kandah[#], Brennan Huber[#], Amani Altarawneh[◇], Sai Medury[◇], Anthony Skjellum[#◇]
Computer Science and Engineering[#], SimCenter[◇], University of Tennessee at Chattanooga
Chattanooga, TN, USA

***Abstract*—Advancement in communication technologies and the Internet of Things (IoT) is driving smart cities adoption that aims to increase operational efficiency of infrastructure, improve the quality of services, and citizen welfare, among other worthy goals. For instance, it is estimated that by 2020, 75% of cars shipped globally will be equipped with hardware to facilitate vehicle connectivity. The privacy, reliability, and integrity of communication must be ensured so that actions can be accurate and implemented promptly after receiving actionable information. Because vehicles are equipped with the ability to compute, communicate, and sense their environment, there is a concomitant critical need to create and maintain trust among network entities in the context of the network's dynamism, an issue that requires building and validating the trust between entities in a small amount of time before entities leave each other's range. In this work, we present a multi-tier scheme consisting of an authentication- and trust-building/distribution framework designed with blockchain technology to ensure the safety and validity of the information exchanged in the system. Through simulation, we illustrate the tradeoff between blockchain mining time and the number of blocks being generated as well as the effect of the vehicle speed on the number of blocks being generated.**

Keywords*—*Connected vehicles, Trust management, Blockchain, Smart Cities

I. Introduction

The rapid growth of connected devices comprising the Internet of Things (IoT) is transforming traditional elements of city life. Among the key components contributing to such smart cities' initiatives in the intelligent transportation system (ITS) are connected vehicles [1], [2]. The cybersecurity requirements of smart cities are distinct from conventional and past security issues; moreover, they are constantly evolving because of new trends in technology and use cases. Because of the fundamentally untrusted environment, it is difficult for vehicles to evaluate the credibility of received messages. Therefore, there is a critical and timely need to ensure the trustworthiness of communication between vehicles based on source reputation (trust), and completeness (integrity and availability).

When it comes to managing trust, it is important to take the challenges facing the network as well the trust management approach itself into consideration. For instance, on one hand potential threats against trust and verification in connected vehicles can raise a challenge where attackers work on injecting the network with malicious vehicle data, compromising vehicles by distorting trust values of other nodes in the network, and/or raising/lowering the trust values of a specific vehicles to spoof certain changes in the network [3]–[6]. On the other hand, the network setup, which can undergo frequent topology changes and the high mobility characteristics of connected vehicles, can create additional challenges in which cryptographic solutions to data trust cannot perform as well as expected and can easily be overtaken by malicious control over authorized and authenticated users [7].

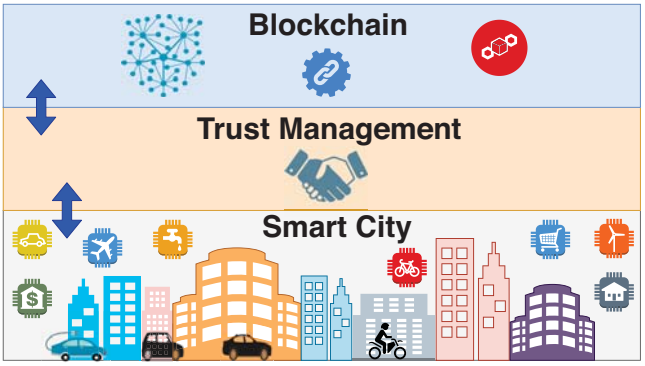

Figure 1: Conceptual view of the proposed approach

In this work, we propose an integrated system that is better able to build a distributed, tamper-proof, and consistent trust-based management approach capable of evaluating the trust among network entities than other approaches. This approach adds a dimension of assurance in which exchanged data has a quantitative metric of trustworthiness; this assurance will play a vital role in maintaining system safety. Distributed ledgers based on blockchain technology underly our approach to distributing data and maintaining trust.

The remainder of the paper is organized as follows: We discuss related work in Section II, followed by our motivations and contributions in Section III. We present our Blockchain-based Trust Management (BLAST) Approach in Section IV, followed by discussion on how our proposed approach mitigates the effect of the threat model in Section V. Our quantitative analysis is presented in Section VI. We conclude and discuss future directions in Section VII.

II. Related Work

Trust is based on the history of interactions (successful and/or unsuccessful) and the validity of the information exchanged between network entities [8]–[10]. Recently, the concept of managing trust in the network has received significant attention since it adds an additional security layer designed to ensure that the data being exchanged in the network is valid and originating from a trustworthy source [7], [11]. Several trust management schemes have been proposed including entity-based, data-based, and hybrid trust models [12]. The dynamic nature of connected vehicles requires a distributed system that allows vehicles to gather and share information germane to build trust in the network as they move from one place to another; this can be achieved through collaboration between the connected vehicles and fixed roadside units.

Previous work has proposed solutions for trust management implementation in Vehicle Ad Hoc Networks — Intelligent Transportation System (VANET-ITS) (e.g., [12]–[16]). The authors in [12] proposed a decentralized system, claiming that a centralized

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

system would be impractical for the growth that a VANET-ITS would require. Another drawback of a centralized system is the massive overhead that could be incurred if several vehicles should be communicating with the central node at once. By having several RSUs located throughout a city, each area within it can be divided roughly equally (spatially or based on load experience) and therefore the load will be reasonably balanced as well. Furthermore, the authors continue by discussing their proposal for trust-factor calculations. Each vehicle begins with a neutral value, and, as messages are passed between vehicles, the trust value will be updated depending upon the accuracy of the messages that were previously passed. The method of evaluating the accuracy of a message is based on the experiences other vehicles in the network have had with that message. The critical drawback to this approach is the scenario in which there are several malicious vehicles in the network and these vehicles collude to rate their messages as accurate. This scenario increases the malicious vehicles' trust factor in the network, thus decreasing the overall integrity of the system.

Blockchain is one of the recent, breakthrough technologies used in the financial industry. Blockchains create a consistent, tamper-proof distributed ledger that records financial trading without the need for a centralized bank [17]. Notable blockchains include Bitcoin [18] and Ethereum [19]. For instance, Ethereum's key features include decentralized control, availability, tamper-proof properties, and a consensus (mining) algorithm through proof-of-work (PoW) and, optionally, proof-of-stake (PoS) [20]. Tamper-proof is a key feature of blockchain technology. Any malicious nodes in the system would be unable to tamper with previous blocks because of the chain structure of a blockchain (subject to Byzantine limits). Later blocks depend on data collected from earlier blocks, and if any changes are made to such earlier blocks, this manipulation will create a disparity in the chain [17]. Some blockchain protocols support smart contracts, which are self-executing scripts; smart contracts are immutable and possible because of the tamper-proof blockchain feature [21].

In [12], the authors also proposed to use blockchain as a means of storing the trust factor of each vehicle. As vehicles exchange messages, these messages will be compiled into one block of data that will then be uploaded to a local RSU. A given RSU will then compete against other RSUs in the network via a joint Proof-of-Work (PoW) and/or Proof-of-Stake (PoS) to determine which will be elected as the miner. Using a joint PoW and PoS will evidently prevent RSUs that are in a high-demand area (with lots of ratings (stakes)) to always upload to the chain by also allowing smaller and less common RSUs also to upload their data to the blockchain. Although this work proposed the use of blockchain to support trust management in the network, it lacks the ability to verify the data being added to the block by RSUs, in which each RSU is programmed to accept any data provided (all data received is deemed accurate).

The authors in [22] proposed the solution of keeping only the last few blocks of the blockchain on each vehicle to save storage. While this solution will certainly require less storage capacity on vehicles, it will greatly increase the communications between vehicles and RSUs. This is so because, should a vehicle be unable to access all of the blockchain data and comes into contact with another vehicle whose trust is unknown, then an RSU query must be made to determine the trust factor of said vehicle. Querying the RSU each time a device comes into contact with a new vehicle can be a costly action. A method to eliminate such delays is necessary for a connected vehicle system to function in real-time.

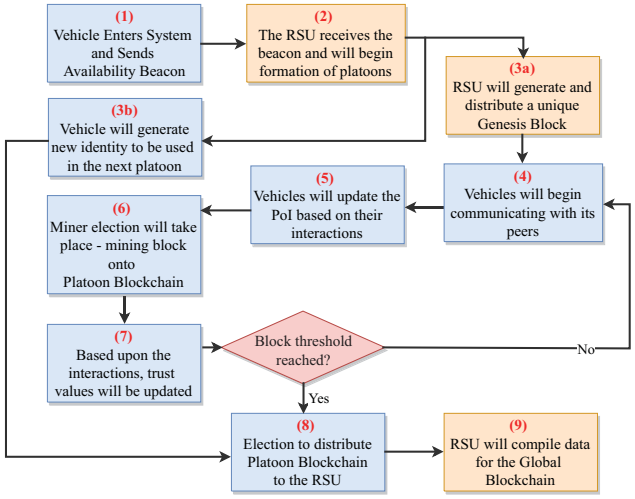

Figure 2: Overview of the proposed system

III. Motivations and Contributions

In a connected vehicle system, time is of the essence. Vehicles must be able to access the trust store, as well as record any updates in real-time. Since blockchain provides a decentralized ledger, it can offer significant improvements in terms of minimizing delays and supporting scalability and real-time aspects of dynamic, connected vehicle systems. However, we observed that while previous work implemented blockchain solutions to solve security aspects of a connected-vehicle system, these did little to ensure the integrity of the system and its real-time operation. Our aim is to create a trustworthy, connected-vehicle network through studying, designing, prototyping, demonstrating, and evaluating a cybersecurity layer that advances the next-generation cyber-physical systems against threats targeting the safety of human life and property. We summarize our contributions as follows:

- We constructed a Global Trust-based Blockchain database, to support integrity and availability, where vehicles can record their trust values. Once forming trust, any vehicle can access another vehicle's trust from anywhere, making it unnecessary to contact the vehicle directly to request its trust.
- We constructed a verification system, with a Blockchain that enables a quick consensus between platoons of vehicles thus minimizing the load generated by the data being transmitted to the Global Trust Database.
- We developed a methodology that efficiently verifies interactions between system entities, which helps the system avoid system attacks in which one entity would falsely report an interaction with another entity with which it has never come into contact, thus decreasing its trust.
- We developed a privacy approach which enables peers to remain pseudo-anonymous in a network while still participating in the communication of events in the system.

IV. Blockchain-based Trust Management (BLAST)

The proposed design is a multi-layered system. At the lower level are the vehicles, which will be grouped into platoons based on proximity and speed. Vehicles in a platoon will be passing messages among themselves. Each vehicle will then analyze these messages to determine their validity. The upper level will consist of Road Side Units (RSUs). These RSUs are comprised of stationary hardware that will have non-trivial computational power and fast network

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

connections. RSUs will supervise a predetermined geographical sector of the network, in which all platoons in this sector will report.

The system interaction is presented in Fig. 2:

Step 1. Vehicles send a beacon message (M_{beacon}) and when they enter the geographical area under the purview of an RSU(s) indicating their availability and willing to participate in the system.

Step 2. Upon receiving the beacon messages, the RSU will begin forming the platoons based on the vehicles' availability and their proximity (Section IV-A).

Step 3. Upon identifying the platoons, the RSU will send the Genesis block (B_0) to the vehicles; the Genesis Block includes the most up-to-date trust values from the global blockchain (G_B) for each vehicle in the platoon. Note that each platoon will be given a unique B_0, which reflects information that is of interest to the platoon's members (see section IV-C3). Meanwhile, each vehicle will start creating its own ID mask to hide its identity and generate new identity to be used in the next platoon (see section IV-D).

Step 4. Vehicles will start interacting with their neighboring vehicles in the platoon and actions will be performed with a level of confidence based on the trust values given from the Genesis block (see section IV-B) or the previous Block depending on the size of Platoon Blockchain (P_B).

Step 5. Based on the system, each vehicle will begin updating the Proof of Interaction (PoI) based upon these interactions between vehicles (Section IV-B3).

Step 6. After a set of interactions, the platoon's members will choose a miner that will take care of creating a block into the Platoon Blockchain (P_B) as specified in the Mining and Verifying the Platoon Block sections (Sections IV-C4, IV-C5, and IV-C6).

Step 7. After a platoon block is formed and mined on the platoon blockchain, each vehicle will update the trust value of its neighboring vehicles according to the responses that were received based on the previous context.

Step 8. After generating a specified number of blocks in the platoon or based on the RSU's request to mine in the global blockchain, the platoon members will elect a leader that will send the platoon blockchain to be mined into the global blockchain (Section IV-C7).

A. Platoon formation and validation

Factors that affect signals being sent and received to and from an RSU, as well as those that are affecting the position and the replacement of RSU—such as the number and types of antennas (omnidirectional, directional), the average transmission range, signal strength (dBm), the size of the packets, the successful transmission ratio (which is the percentage of the arrived packets with a given packet size), and the number of vehicles within transmission range (which will limit the number of generated packets). Given these considerations, an RSU is responsible to form and validate a given platoon; that process starts with a 10 seconds timeout to listen for all the beacons from the vehicles that are joining its coverage range, then the number of different beacons representing various vehicles is at minimum of four cars. In this regard, the RSU applies the following procedure to form and validate the platoon, using the data sent in the beacon for each vehicle, which includes the vehicle's ID, speed, direction, and (x,y) coordinates:

Step 1. RSU starts a listening timeout for beacons and vehicle count.

Step 2. Each vehicle will announce its availability with a beacon message as given in Eqn.1:

$$M_{beacon} \leftarrow (V_{ID}, S, L_{(x,y)}, D, T_r), \tag{1}$$

where V_{ID} is the vehicle's identification, S is the vehicle speed, $L_{(x,y)}$, is the vehicle location, D is the vehicle direction, and T_r is the vehicle transmission range.

The RSU, in turn, will create a table based on the received beacon messages.

Step 3. For each vehicle in the system the RSU will calculate the following:

- The distance d_i in meters where $v_i \in$ RSU's transmission range (R) by using $v_i(x,y)$ coordinates and its direction and RSU(x,y) coordinates at its maximum coverage points in the same direction of the v_i:
$$d_i \leftarrow distance[v_i(x,y)\ and\ Max(R_{RSU})]. \tag{2}$$
- The distance d_j where v_i will be able to run the mining algorithm while it is still within the RSU's coverage limits; $d_j = v_i$'s speed $\times$ mining time (e.g., 10 seconds, which is adjustable according to the system's needs).
- RSU checks if $(d_j < d_i)$, then v_i is in the RSU range and can finish at least one mining round.

Step 4. For all cars, the RSU compares d_i for all vehicles and pick the one with the maximum distance to be the head of the platoon. If there is more than one with the maximum distance, the RSU still needs to pick only one car among them to be the head of the platoon.

Step 5. The RSU calculates the distance between v_i as the head of the platoon and all other cars within the range of RSU starting from it's neighbors and expanding to the neighbors of a neighbor based on the vehicle's transmission range, and limits this to the number of vehicles that could communicate in the given load channel [23].

Step 6. The RSU checks which vehicle will be fully connected with all its peers, group them all together as one platoon and send them all the same platoon's ID.

B. Trust Formation and Factor Generation

Each vehicle in the system will maintain a table of information that is continuously updated, which includes the vehicle ID, Vehicle's trust value, the platoon ID, the miner ID, and the context response (representing the incident being reported by the vehicle). This information will also be used as the transaction in the platoon blockchain.

1) Trust Formation: Previous trust-based schemes of which we are aware have been based solely on the history of successful and unsuccessful communications [13]–[16]. While message validation is an essential component in such systems, it is critical for vehicles in the system to know which to trust; therefore, in our design, we consider three types of trust:

a) Direct trust ($D_t(V_u,V_v)$) is established between vehicle u and vehicle v that are within each other's direct transmission range (one hop apart).
b) Indirect trust ($I_t(V_u,V_w)$) is established between vehicle u and vehicle w based on neighbor-of-neighbor connection (two hops apart).
c) Reputational trust ($R_t(V_u, V_v, V_w)$) can be formed between vehicle u and its directly connected vehicle v based on the information gathered from node w.

Integrating these three forms of trust will allow for all vehicles in the system to obtain a real-time trust factor for any given vehicle without the delay of having to calculate its own trust factor for each vehicle. This approach will also ensure consistency among vehicles, since all vehicles in the network will be able to use the most up-to-date and accurate trust factor for any given vehicle in the system.

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

2) Trust Factor Generation and Modification: Trust factor calculations are critical for a trust management system to work properly. A scale of $[0,1]$ will be used, in which all vehicles that first enter the system are given a neutral trust value of 0.5. As messages are passed between vehicles and the validity of each message is analyzed, the trust factor will dynamically adjust to better reflect the vehicle's likelihood of producing a future trustworthy message.

Upon receiving a message from a vehicle in the platoon, all other vehicles will need to evaluate whether the received message is valid and accurate. Great importance will be attached if the other vehicles in the platoon are currently experiencing an event or situation to which the message alludes. This will aid at moments where a vehicle with a high trust is reporting conflicting information from the rest of the group; the system will not dismiss the group's consensus even if each member of the group has a significantly lower trust factor.

Upon achieving accuracy consensus from the group, a formula to modify initial vehicle trust must be utilized. The proposed formula will apply each vehicle's trust as the weight of the calculation.

3) Proof-of-Interaction by Means of a Hash Matrix: For a trust management system to be both scalable and decentralized, there needs to be an efficient way to validate that interactions between vehicles have actually occurred. This is necessary to provide evidence for as to why a particular trust factor modification has been made. The proposed solution, proof-of-interaction (PoI), is executed via a predetermined hash matrix that is calculated and confirmed by each car in the system. Each interaction between vehicles will correspond to a unique hash; therefore, when evaluating if a specific interaction has taken place, it can quickly be verified against the matrix.

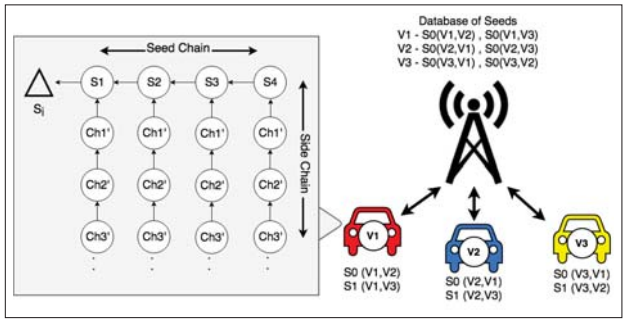

Figure 3: Proof of Interaction (PoI)

Upon forming a platoon, the RSU in range will generate key pairs (seeds) that will be distributed to the vehicles in the platoon. As a vehicle moves between platoons, new seed pairs will be constructed. Each vehicle will receive a seed for each other vehicle contained in the platoon. For instance, in our example shown in Fig. 3, vehicle v1 will receive two key pairs (v1, v2) and (v1, v3), and similarly for vehicles v2 and v3. To maintain a consistent interaction history between the vehicles and to allow each vehicle to authenticate the others, all vehicles will construct a separate one-way hash matrix where each matrix consists of a seed chain and side chain. The side chains are designed in a way so that they will be previously set to allow for a specific number of interactions between pairs of vehicles before the need to regenerate the chain. When this maximum number of interactions is reached, both vehicles will communicate that they need to advance the chain to have more interactions. To avoid tampering with the seed, the seed chain will be formed as a one-way hash chain to create multiple seeds from which the side chains can be derived. Therefore, if one vehicle is reporting an interaction with another vehicle, it can be quickly verified by analyzing the two corresponding hashes in the matrix. This process will be performed by a vehicle once and as soon as the vehicle receives the seed to avoid excess computational overhead on the vehicle.

C. Secure Decentralized Trust Database

Our design incorporates two levels of Blockchains. At the vehicle level, a Platoon Blockchain (P_B) will be formed for each platoon. And, at the RSU level, the Global Blockchain (G_B) will be formed based on the platoon's Blockchain, where RSUs begin mining to add the data (platoon's Blockchain) onto the global Blockchain. The concept underpinning the global Blockchain is to form a global trust management system that provides vehicles with a global view of the trust in the system, which will benefit the vehicles as they transit from one area to another. The platoon Blockchain is formed to store a localized trust consensus, which will only be accessible by members of said platoon. This limited accessibility will ensure that the data being sent to the RSU is valid and will be loaded onto the main Blockchain with confidence.

1) Platoon Blockchain (P_B): Because of the high overhead associated with Blockchains, we propose the use of a smaller, less computationally demanding Blockchain that stores trust values of vehicles in a platoon (Section IV-C5). A platoon will be able to add data onto its Blockchain quicker than it could onto the global Blockchain. This approach better supports the real-time requirement of the connected vehicle system.

The use of a less computationally demanding algorithm allows for blocks to be generated and mined quickly, which in turn allows the dynamic nature of connected vehicles to be unaffected by the high overhead associated with adding blocks to a chain. Furthermore, with the dynamic nature of the system, vehicles will be passing in and out of platoons, as well as in and out of local RSU zones. To upload the platoon chain onto the global Blockchain, the elected miner will be required to push the platoon chain to an RSU upon entering a new RSU zone. The RSU will be required to verify that the platoon chain is valid and will then proceed to perform the calculations necessary to create a new block and upload this to the global Blockchain. Upon uploading, the platoon will "forget" the previous platoon chain and begin constructing a new one. Because an index block is also necessary and must be provided by a trusted source, we propose that the RSU will respond to the uploaded platoon chain with the index block (genesis block), which will contain the trust values for all of the vehicles in the group (Section IV-C3). This arrangement allows all vehicles to have a quick reference point for the trust factors of other platoon members, thus decreasing overhead of having to query the RSU to obtain this information.

By using a platoon Blockchain between groups of vehicles, and along with the tamper-proofing nature of the Blockchain in general, we seek to prevent any malicious node from modifying contents of the Blockchain prior to data being distributed to all nodes in the network. Furthermore, with each group of vehicles working on their own data sets and platoon chains, platoon Blockchains will rapidly allow data to be uploaded to the global Blockchain, which ensures the real-time nature that is required by the system.

2) Global Blockchain (G_B): This global Blockchain will store trust factors of all vehicles in the system. It will obtain these values from the numerous platoon Blockchains that will be uploaded to the RSUs. As an RSU receives data from the platoon Blockchains, it will begin to mine a new block. When an RSU is attempting to mine a block to add to the global Blockchain, the data will likely be several

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

platoon Blockchains from the many platoons that the RSU has come into contact with recently. The global Blockchain will be considered the source of the trust management in the system, because of the tamper-proof concept discussed earlier. Upon mining a block, the RSU will broadcast the newly created block to maintain consistency throughout the network.

3) Genesis Block: The genesis block will be provided by an RSU and will contain a given vehicle's list of peers in the platoon and their associated trust values. When a new platoon is formed, the RSU will begin by creating this genesis block. The first step will be to collect the most up-to-date trust value of every vehicle in the associated platoon, where the RSU will use the most recent block from the global blockchain. The next most critical portion of the genesis block will be the data used to identify and communicate between vehicles. Thus, the genesis block will contain a map of trust values to the identity of each vehicle in the platoon. Finally, the genesis block will contain the pool of contexts in which the platoon members will report and the current context which is the questions that the vehicles will be currently reporting to each other.

4) Platoon mining: The most important issues to consider when implementing the blockchain technology for a basic IoT network are the device's limited computational power and its communication capabilities. The consensus mechanisms in general require a high computational energy, thus using proof of work (POW) [24] is not an option because of its demand of intensive computational power unless the difficulty level is changed. On the other hand, proof of stake (POS) [25] eliminates the need to a high-power consumption but gives more freedom for the miner with highest stake to take advantage of the system with no penalty for any bad behavior.

In our system we chose the lightweight mining protocol (LWM) [26] with Practical Byzantine Fault Tolerance (PBFT) [27] as a consensus mechanism. The LWM protocol is fast with no high power computational demand, and with a random miner selection. Also, PBFT tolerates 1/3 plus one of the number of vehicles to act maliciously in order to reach consensus (for platoons with at least 4 vehicles). In BLAST, the platoon size is considered smaller than for other IoT systems, because the size depends on the number of the vehicles communicating with each other to reach consensus on a certain event in a particular area, so the network extensive communication will not be affected much. In BLAST, all vehicles IDs are known for all the RSUs, thus any vehicle joins in mining is known beforehand.

In the LWM algorithm, each miner generates a random number and a hash using SHA-3, then miners broadcast the hash first with all other miners [26]. Thus, none of the miners know what the chosen random number is initially; later, miners share the random numbers plus each miner calculates the sum of these numbers, and the sum mod the number of miners, where the result remains random and is hard to predict. This resulting random number represent the order of the miner as in the initial list of miners in the genesis block, thus the miner is selected to mine the next block.

5) Lightweight Mining Algorithm: During each timeout before a new LWM cycle begins, each vehicle will broadcast its response to the current context questions. All vehicles will build a table containing the "Transaction" data described above. After the timeout has occurred, LWM will begin to elect a miner and begin mining. During this process the elected miner will perform a set of operations that include; the calculations for aggregating transactions to calculate the 'correct' answer to the context. This will be done through combining responses that are true (+1) and false (-1). If the end value is positive then the 'correct' answer is true and, respectively, if the end value is negative then it is false. There is the potential of getting a value that is equal to 0, but the chances are low because of the variety of trust values. The mining vehicle is in charge of updating trust values of each vehicle according to the answer to the context.

6) Verify the Platoon Blockchain: When vehicles have gathered enough data and produce a sufficient number of blocks, the associated platoon blockchain will be offloaded to the RSU so that the data can be extracted and mined onto the Global Blockchain (G_B). When the RSU receives a platoon blockchain, it will begin the verification steps to ensure no malicious activity or incorrect data is placed onto the global blockchain. The RSU will iterate through the blockchain to verify each aspect. First, it will verify the 'correct' answer to the context was calculated correctly, it will do this through the mining verification algorithm presented above. The next step will be to calculate each vehicle's trust value correlating to the response provided to the context. Such that if a vehicle reported False to a context that was calculated to be True, the trust value of this vehicle would be decremented according to the trust value update calculation described above. Lastly, the RSU will verify that the properties of the blockchain are intact (e.g., that each entry is signed correctly and the calculated hashes are correct). When all aspects of the platoon's blockchain has been verified, the RSU will take the final trust value results from the verification process and begin to add the data onto the global blockchain.

7) Global Mining: When mining on the global Blockchain, every RSU will be competing to add a block onto the global Blockchain. As RSUs are receiving platoon Blockchains, they will immediately begin trying to calculate a nonce that will put the new hash below the predetermined threshold. When a nonce is discovered, the RSU will broadcast the newly created block to all other RSUs in the system to be verified. Having RSUs compete to upload data instead of dividing the work evenly further helps the system in real time. Also, because each RSU is implemented on localized hardware, they will all have comparable computational power, which is where the scheme will be consistent among all RSUs and competition will be fair. This consistency makes it infeasible for any given RSU to take over the block mining process unless it is physically compromised by an adversary installing advanced processing power while avoiding detection.

D. Dynamic ID generation

As vehicles are identified by their IDs, we established a way to mask their IDs every time they participate in a new platoon, which makes it harder for a malicious vehicle to know or predict an ID and claim it for itself. The idea is to generate a dynamic ID for a given vehicle during their existence in the system, where public and private keys will be used to maintain security interactions between vehicle and RSU, dynamic ID generation procedure is taking place in any time during the current platoon in order to have the ID's mask ready to be used in the next platoon:

Step 1. Each vehicle will indicate its existence in the RSU range using a beacon message that includes its ID using

$$RSU \leftarrow P(ID)_{RSU}.$$

Step 2. The RSU will respond to each vehicle with a hash function (H) and a list of all vehicle's trust (sub-ledger ($SL(v_i,v_j,v_k,...,v_m)$) within its range that will form the platoon:

$$v_i \leftarrow P(SL,H_i)_{V_i}$$

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

Step 3. Upon receiving the information, each vehicle will select a subset of its neighbors' IDs to form its new ID using the hash function provided by the RSU such as:

$$ID'(v_i) \leftarrow H(ID(v_i), sub_{ID} \in (v_j, v_k, ...)).$$

If platoon size is k vehicles, then the number of all possible subsets that each vehicle could generate is 2^{k-1}. In our system, the smaller platoon size is formed with at least four vehicles, which is the highest probability for a malicious vehicle to predict the chosen set; the number of subsets could be generated among three vehicles is $2^3 = 8$ possibilities, then the probability for a malicious vehicle to predict the new mask ID is 0.125 (which is the highest probability of prediction with the smallest platoon size that has four vehicles). Even in this case, only immediate impersonation is likely, since platoons evolve over time; a would-be impersonator's chance would be highest at the first transaction, before vehicles come and go.

Step 4. Each vehicle will send the subset of IDs that was selected to generate its new ID back to the RSU, in order for the RSU to keep track of the new ID for that specific vehicle.

$$RSU \leftarrow P(sub_{ID} \in (v_j, v_k, ...)).$$

Step 5. As the vehicle exits the platoon and enters another platoon, it will start using the new ID, which will be populated by the RSU to avoid the impersonation and cloning the ID later. The RSU and the specific vehicle will be the only entities that will store the ID changes.

V. Discussion

In this section, we discuss how our proposed work mitigates the effects of a set of threats, comprising our threat model.

Threat 1. Black Hole Miners: In the event that a vehicle refuses to compute a block to add onto the platoon blockchain, all other vehicles in the platoon will wait for a predetermined period of time before considering the miner to be malicious. When this timeout happens, the mining election process will reset. Therefore, at most the system will be delayed by a short period of time before continuing with the process, thus having minimal overall impact on function.

Threat 2. Bad Mouthing: When reported values are pushed to the RSU and added onto the blockchain, all other nodes in the system will verify the contents of the block to ensure that the trust values were not modified by more than 10%. This restriction will limit the overall effect of a single malicious node on other nodes in the system. Furthermore, because of vehicle mobility in the system, the chances of having a prolonged effect of bad mouthing on a single vehicle is low.

Threat 3. Malicious Miners: With a malicious RSU (M_R) in the system, additional protocols will be needed. As proposed earlier, a deduction of at most 10% of a vehicle's trust will be assessed in particular scenarios. When an RSU mines a block onto the Global Blockchain, all other RSUs will run a quick calculation to ensure that the data meets all predetermined conditions (including the 10% maximum change per trust) to ensure the validity of the mined block. If a discrepancy is found, then an RSU will report it, triggering a flag that alerts all nodes of the invalid block. When this flag is triggered, the invalid block will be ignored and the previous block's data will be used in its place. At most, this process may delay the system by two mining cycles because the M_R will mine an invalid block, which occupies a cycle, while reporting the invalid block will also take a mining cycle.

Threat 4. Colluding Attacks: This attack can be mitigated with several different protocols in place. First, the system allows no greater than a 10% change to a trust value in one mining cycle. It will take prolonged effort from the colluding entities to have a substantial effect on the trust value of its neighbors or itself to be reflected in the global blockchain. Furthermore, as soon as a malicious vehicle leaves the colluding group, it will only take one mining cycle to lose a large portion of its trust value. When all other vehicles are not malicious, the distributed consensus will discover that the malicious vehicle is attempting to add false data to the blockchain and the trust factor will be reduced accordingly.

VI. Quantitative Analysis

To measure the performance of our proposed approach, we considered a connected vehicle network with n vehicles in an urban environment. Our network is set up in such a way so that a platoon is formed by spawning a number of vehicles that are fully connected with each other to form this platoon. Each vehicle in the system will be defined by a vehicle ID, moving direction and speed (vehicle velocity (m/s)).

Through our evaluation, we chose the **average number of blocks** as our first performance metric, which is defined as the number of blocks that a vehicle is able to add to the Platoon's Blockchain before losing its connection to its peers in the platoon. The use of this performance metric will lead us to determine the ideal number of blocks that can be generated in the platoon taking into consideration the time it takes to upload to the global blockchain and the vehicles' speed.

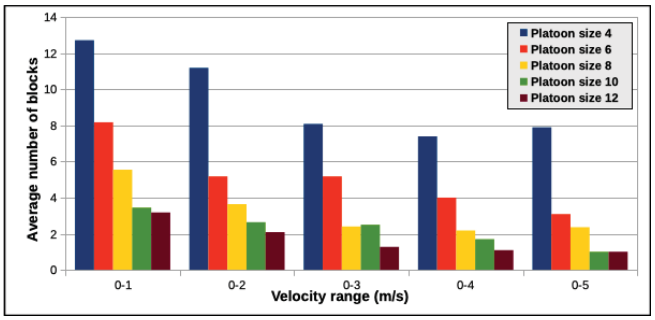

Figure 4: Transmission Range of 50m

Our first evaluation is presented in Fig. 4. We considered a transmission range of 50 meters, where vehicles within range will be grouped into a platoon and will start transmissions and platoon's blockchain formation. To perform platoon changes, we set the speed (velocity) ranges of 1, 2, 3, 4, and 5 *m/s*. It can be seen that when the velocity of vehicles is at or within 1 *m/s* of each other, a platoon size of four is able to complete approximately 13 blocks before breaking the platoon, where one of the vehicles move outside of the 50 meter range of another vehicle. With increasing platoon size (number of vehicles in the platoon), it can be seen that it becomes harder to maintain the platoon, which will result in decreasing the number of blocks being generated.

We chose the **time between miner election** as our second performance metric, which is defined as the time when vehicles would be passing messages back and forth awaiting the next miner election to take place. To measure this quantity, the same platoon sizes were used, the same transmission ranges, and the velocity of the vehicles was set to having a maximum difference of 3 *m/s*. The variable in this case is the time before the next miner election was modified from 4, 6, 8, and 10 seconds per run.

Our results with a transmission range of 50 meters is presented in Fig. 5. It can be seen that with this short transmission range, the additional delay introduced by spending more time waiting on vehicles to pass messages became too large and the platoons were consistently averaging below two blocks per platoon cycle. With these results, it can

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

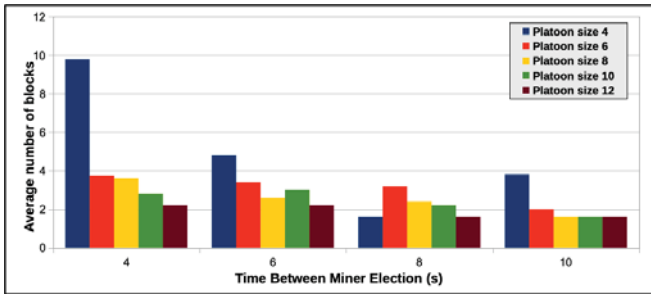

Figure 5: Block Time with Transmission Range of 50m

readily be concluded that decreasing the time between mining cycles is necessary because this enables the platoon to mine more blocks onto the platoon blockchain which aids in security from modification.

VII. Conclusion and Future work

Crucial to the flourishing of advanced societies, smart cities are an emerging opportunity to enhance safety and well being of humanity. Transportation plays a significant part in this process and to ensure safety, trust must be built between connected autonomous vehicles, providing safe transportation to these connected communities and their members. Blockchain, as technology, provides a mathematically verifiable framework for trust building. We presented a blockchain-based trust management scheme aiming to create a trustworthy environment for connected autonomous vehicles. Through the discussion, we argued that our proposed scheme can enhance the field of connected vehicles and can be further applied to different fields in smart cities settings. We provided a data architecture that supports secure trust, platooning concepts, and we performed quantitative simulation experiments to derive the ability to communicate reliably as a function of platoon dynamics. These results point to tradeoffs of secure block generation for local and global trust calculations vs. mobility.

In the near future, we plan to develop a testbed to validate the proposed scheme practically, since it is crucial to determine the time needed to receive a message and to determine the trustworthiness of connected vehicles in a smart city setup. Our future work will demonstrate a successful system construction that would allow analysis of real-world environments. Also, we plan to develop and apply a novel approach to evaluating autonomous vehicle Blockchain security, robustness, and reliability using proven analytical methods for assessing system trustworthiness and resistance to intruder attacks. Queueing theory will be employed to identify the average waiting time an autonomous vehicle waits before discarding a request for information confirmation from other resources.

Acknowledgements

The authors acknowledge support from the University of Tennessee at Chattanooga. Research reported in this publication was supported by the 2019 Center of Excellence for Applied Computational Science and Engineering grant competition (CEACSE). This work was performed with partial support from the National Science Foundation under Grants Nos. 1812404, 1821926, and 1642133. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Sunilkumar S. Manvi and Shrikant Tangade. A survey on authentication schemes in vanets for secured communication. *Vehicular Communications*, 9:19 – 30, 2017.
[2] Autonomous Vehicles. Self-driving vehicles enhanced legislation. http://www.ncsl.org/research/transportation/autonomous-vehicles-self-driving-vehicles-enacted-legislation.aspx. Accessed on: June 2017.
[3] Y. Ren, V. I. Zadorozhny, V. A. Oleshchuk, and F. Y. Li. A novel approach to trust management in unattended wireless sensor networks. *IEEE Transactions on Mobile Computing*, 13(7):1409–1423, July 2014.
[4] Xin Kang and Yongdong Wu. A trust-based pollution attack prevention scheme in peer-to-peer streaming networks. *Computer Networks*, 72(Supplement C):62 – 73, 2014.
[5] Asad Amir Pirzada and Chris McDonald. Establishing trust in pure ad-hoc networks. In *Proceedings of the 27th Australasian Conference on Computer Science - Volume 26*, ACSC '04, pages 47–54, Darlinghurst, Australia, Australia, 2004. Australian Computer Society, Inc.
[6] Zhaoyu Liu, A. W. Joy, and R. A. Thompson. A dynamic trust model for mobile ad hoc networks. In *Proceedings. 10th IEEE International Workshop on Future Trends of Distributed Computing Systems, 2004. FTDCS 2004.*, pages 80–85, May 2004.
[7] C. A. Kerrache, C. T. Calafate, J. Cano, N. Lagraa, and P. Manzoni. Trust management for vehicular networks: An adversary-oriented overview. *IEEE Access*, 4:9293–9307, 2016.
[8] M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized trust management. In *Proceedings 1996 IEEE Symposium on Security and Privacy*, pages 164–173, May 1996.
[9] Li Xiong and Ling Liu. Building trust in decentralized peer-to-peer electronic communities. 2002.
[10] Li Xiong and Ling Liu. Peertrust: supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):843–857, July 2004.
[11] W. Li, H. Song, and F. Zeng. Policy-based secure and trustworthy sensing for Internet of Things in smart cities. *IEEE Internet of Things Journal*, 5(2):716–723, April 2018.
[12] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung. Blockchain-based decentralized trust management in vehicular networks. *IEEE Internet of Things Journal*, pages 1–1, 2018.
[13] J. Jiang, G. Han, F. Wang, L. Shu, and M. Guizani. An efficient distributed trust model for wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 26(5):1228–1237, May 2015.
[14] Jiangwen Wan, Xiang Zhou, Xiaofeng Xu, and Renjian Feng. A trust evaluation algorithm for wireless sensor networks based on node behaviors and d-s evidence theory. 2011.
[15] G. Zhan, W. Shi, and J. Deng. Design and implementation of TARF: A trust-aware routing framework for wsns. *IEEE Transactions on Dependable and Secure Computing*, 9(2):184–197, March 2012.
[16] I. Butun, S. D. Morgera, and R. Sankar. A survey of intrusion detection systems in wireless sensor networks. *IEEE Communications Surveys Tutorials*, 16(1):266–282, First 2014.
[17] G. Zyskind, O. Nathan, and A. '. Pentland. Decentralizing privacy: Using blockchain to protect personal data. In *2015 IEEE Security and Privacy Workshops*, pages 180–184, May 2015.
[18] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.
[19] Dr Gavin Wood. Ethereum: a secure decentralised generalised transaction ledger. 2016.
[20] Alex Mizrahi Meni Rosenfeld Iddo Bentov, Charles Lee. Proof of activity: Extending bitcoins proof of work via proof of stake. 2014.
[21] K. Christidis and M. Devetsikiotis. Blockchains and smart contracts for the Internet of Things.
[22] Shiho Kim. Chapter two - blockchain for a trust network among intelligent vehicles. 111:43 – 68, 2018.
[23] Rijkswaterstaat. Rsu placement guidelines. *MAPtm*, 2017.
[24] Satoshi Nakamoto et al. Bitcoin: A peer-to-peer electronic cash system. 2008.
[25] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151:1–32, 2014.
[26] Richard R Brooks, KC Wang, Lu Yu, Jon Oakley, Anthony Skjellum, Jihad S Obeid, Leslie Lenert, and Carl Worley. Scrybe: A blockchain ledger for clinical trials. In *IEEE Blockchain in Clinical Trials Forum: Whiteboard challenge winner*, 2018.
[27] Mehrdad Salimitari and Mainak Chatterjee. An overview of blockchain and consensus protocols for iot networks. *arXiv preprint arXiv:1809.05613*, 2018.