The Pennsylvania State University

The J. Jeffrey and Ann Marie Fox Graduate School

A NOVEL METHOD TO PREVENT SYBIL ATTACKS IN VANETS AND SHARE KNOWLEDGE OF MALICIOUS NODES

A Thesis in

Computer Science

by

Andrew Smith

Copyright 2024 Andrew Smith

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

December 2024

The thesis of Andrew Smith was reviewed and approved by the following:

Jeremy J. Blum
Associate Professor of Computer Science
Thesis Advisor

Bimal Ghimire
Assistant Professor of Computer Science

Sayed Mohsin Reza
Assistant Professor of Computer Science

Md Faisal Kabir
Assistant Professor of Computer Science

Sukmoon Chang
Associate Professor of Computer Science
Program Head

ABSTRACT

Vehicular ad-hoc networks (VANETs) have become increasingly important as cars are being equipped with higher levels of self-driving technology. Inter-vehicle communications can be utilized for self-driving or alerting drivers to road conditions. This can help to prevent traffic accidents or manage routing changes based on information provided by other vehicles. As VANETs and inter vehicle communication become more common, the need for stable and secure networks is needed. Routing algorithms for ad-hoc networks remain vulnerable to many attack types. This thesis outlines a new addition to the popular routing protocol Ad-hoc On-Demand Distance Vector (AODV) routing. AODV is extended to authenticate nodes using a combination of public key cryptography and location verification. This authentication is used to share information about identified malicious nodes in the network. A simulation is then modeled in NS-3 to validate the effectiveness of the reporting system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank Dr. Jeremy Blum for his guidance and patience throughout this process. Without whom I would never have known how to get started in picking a topic, let alone have anything to present. I am grateful to all the professors whom I had the privilege to study under during these last 4 years. Their patience with me as I jumped from topic to topic and could not find a topic I liked until I had finished all my other degree requirements is a testament to their dedication to teaching. I also owe my gratitude to Dr. Adil Alsuhaim, whom I have never met, but none-the-less would never have successfully figured out ns-3 without his YouTube videos.

Finally, I owe so many thanks to my wife, Danielle. She has encouraged me all along the way and has always believed I could complete this thesis, even when I was not. Without her I would have given up long ago. She took care of our 2-year-old and newborn sons for weeks so that I could spend my evenings finally completing this work. Her support in this means more to me than I can express in words. I should also thank our sons, Newton and Waldon. I constantly work to set a good example for them. Completing this thesis and degree is my way of showing them how to follow through on commitments and always strive for something greater.

Chapter 1

Introduction

This thesis proposes a routing algorithm for vehicular ad hoc networks that will prevent sybil attacks and issue reports of malicious nodes of any kind. This routing algorithm will also allow for additional changes to be made that would identify or prevent other attack types. This routing algorithm will be based on a well-known routing algorithm for ad-hoc networks called Ad-Hoc On-Demand Distance Vector routing.

Vehicular Ad-Hoc Networks

Vehicular ad-hoc networks (VANETs) are a type of mobile ad-hoc network (MANET) in which the nodes within the network are either vehicles, called an on-board unit (OBU), or roadside units (RSU). RSUs are stationary devices that are placed by government organizations to assist in the management of the VANET. RSUs are costly to install and would be needed in great numbers to cover the entire road network. OBUs are installed on vehicles and allow them to communicate over a Wi-Fi spectrum outlined in IEEE 802.11p. IEEE 802.11p is a standard for wireless access in vehicular environments (WAVE) that is meant to be used for vehicles within intelligent transportation systems (ITS). This standard is the basis for communication between vehicular nodes by the Dedicated Short-Range Communications (DSRC) protocol on the 5.9 GHz spectrum [1].

Traffic Safety

Traffic safety is a prominent issue due to the large number of traffic accidents, causing fatalities and other adverse effects [2]. VANETs in an ITS are a solution to this issue since they can allow vehicles to communicate with each other and either give drivers additional notice or be used in a self-driving system. Basic Safety Messages (BSMs) are a way for vehicles within VANETs to exchange pertinent information about the network such as traffic information or about their own state. BSMs typically contain current velocity and other information about the operation of the vehicle, they may optionally contain information about traffic conditions, roadway conditions, and accidents. It is important that these messages be delivered reliably.

Routing protocols

Many routing algorithms have been designed for MANETs and VANETs to allow for nodes to communicate with each other over larger distances than direct communications range [3]. AODV was chosen as a basis for this protocol due to its performance compared to other popular routing protocols [4]. These multi-hop routing algorithms utilize the vehicles within the VANET to forward messages when the destination is not within range for direct communication over the DSRC protocol. This differs from traditional routing protocols since the nodes forwarding messages are constantly moving and cannot be inherently trusted. AODV utilizes on-demand route requests to find routes between nodes and route replies to confirm those routes. It also utilizes periodic hello messages to keep track of neighbors. Routes have short lifetimes and are refreshed as they are used. There are several attack types that these routing algorithms are susceptible to.

Malicious Attacks

Some of the attacks that VANETs are most susceptible to are black hole attacks, sybil attacks, and Denial of Service (DoS) attacks. While there has been much research into each of these, sybil attacks are some of the most difficult to detect and mitigate. Sybil attacks are also extremely damaging to VANETs. This thesis focuses on creating a system that allows nodes to share information about malicious attacks. It includes the prevention of sybil attacks, to prevent the information sharing system from being abused. Sybil attacks are categorized as attacks in which a malicious node either steals another node's identity or fabricates multiple identities. Black hole attacks are simulated as part of the experiment performed. They are categorized as attacks in which a node fabricates responses that indicate it has a valid route to a node. Then the malicious node drops any packets being sent through it.

LESAP

LIDAR Enabled Sybil Attack Prevention AODV (LESAP) will utilize additional control messages to trade RSA keys with other nodes that can be identified using on board light detection and ranging (LIDAR) systems without utilizing a Central Authority (CA) or Roadside Units (RSU). This will allow for the signing of subsequent routing control messages. LESAP further includes a control message type, for reporting to other nodes the existence of malicious nodes in the network. This reporting system combined with the authentication system will prevent abuse of reports and allow malicious node detection schemes to be added on top of LESAP to prevent additional attack types.

Twelve simulations were run to compare LESAP with AODV. Simulations varied in the numbers of nodes, and with or without malicious nodes included. The malicious nodes were black hole nodes. The results showed that LESAP was able to spread the knowledge of the malicious nodes quickly. All nodes were able to blacklist all malicious nodes prior to messages being sent. Some metrics, however, suffered from the additional overhead in LESAP when compared to AODV, such as a lower packet delivery ratio and lower throughput.

The rest of the thesis is organized as follows. Chapter 2 will go over some related work in the space of AODV and modifications to harden it against a variety of attack types. Chapter 3 will go into detail on the design and setup of the LESAP routing protocol, as well as the major similarities and differences between LESAP and AODV. Chapter 4 will describe the experiment run and discuss the results of that experiment, showing an improvement in the effect of malicious nodes on throughput and packet delivery ratio when using LESAP. Lastly, Chapter 5 will conclude this thesis and briefly discuss some possible improvements in future work.

Chapter 2

Related Work

Ad-hoc routing protocols designed for VANETs are susceptible to several attacks as outlined in the survey on VANET attacks and countermeasures by Abuarqoub et al [5]. They categorize attacks into 3 main types, which are fabrication, routing, and botnet attacks. The main attack type targeted by this thesis is fabrication attacks, namely sybil attacks and impersonation attacks. Blackhole attacks, a type of routing attack, will also be briefly covered. The reporting system has been designed to allow for various black hole detection systems to utilize it. Routing attacks are simulated in the experiment. AODV is used as a base and modified as part of this work.

Ad Hoc On-Demand Distance Vector

Ad Hoc On-Demand Distance Vector (AODV) was created as a distance vector routing algorithm with a goal of decreasing the amount of control messages needed over Destination-Sequenced Distance Vector routing (DSDV) [6]. AODV uses "pure on-demand route acquisition" to maintain routes to only the node it needs to communicate with. It utilizes periodic hello messages to keep up to date routes to all its direct neighbors. AODV then uses a "broadcast route discovery mechanism" that is borrowed and modified from a routing algorithm called Dynamic Source Routing (DSR). AODV combines this with a destination sequence number borrowed from another routing algorithm called Destination-Sequenced Distance Vector routing (DSDV). Each node maintains its own sequence number counter. This counter increases each time a route request is issued to allow nodes to replace stale routes.

Route Request

When a node requires a route to another node it will broadcast a route request to all neighbors which includes source and destination addresses and sequence numbers, a hop count, and a broadcast id. Source address and broadcast id will uniquely identify a request so duplicates can be dropped. Neighbors will either respond with a route or forward the request to each of their neighbors, increasing the hop count by 1. The destination sequence number is used to determine if a route is fresher than an existing route. The source sequence number is used to determine the freshness of the corresponding reverse route to the source. Reverse paths to the source are saved by each node along the requests path to the destination. Once the request arrives at the destination or a node with an active route to the destination with a larger sequence number, a route reply is generated and sent only to the neighbor that forwarded the request.

Route Reply

The reply contains the source and destination addresses, the updated destination sequence number, the hop count, and a lifetime. The reply then travels back along the path the request took with each node saving the route to the destination as it passes. A node will forward the first reply it receives for a source and destination. It will only forward additional route replies if they have a larger destination sequence number, or the same destination sequence number but a lower hop count.

Route Maintenance

Route expiration times are used to purge each node’s routing table of old routes. When a reverse route is added during the request step, an expiration time is added to invalidate the route if a reply is not received before the expiration. An active route expiration is also used for nodes that have received and forwarded a route reply for that route request. Each time a route is used the active route expiration is extended. The routing table entries for routes contain the destination address, hop count, next hop address, destination sequence number, active neighbors for the route, and expiration time. The destination sequence number and hop count are used to update routes when a route request or reply containing one that is fresher is received. As nodes move, they can resend route replies to refresh routes or send periodic hello messages to keep track of neighbors. When a message cannot be forwarded to the next hop address, a route error message is sent to all neighbors to inform them of the route no longer being valid. This is so they can remove any routes to that destination that go through the node sending the error. Sequence numbers are not incremented for hello messages but are for route requests. Hello messages contain a TTL of 1 so they will not be forwarded. After a neighbor fails to send a hello message twice in a row, any routes with that node as the next hop are marked invalid and a route error message is sent to active neighbors using that route.

Improvements

Research has been done to improve AODV in various ways. Aggarwal et al propose a trust-based model that only utilizes routes with neighbors that have a sufficiently high trust score that is made up of how many RREQ, RREP, and data messages that have been handled for that node [23]. Pandey and Singh propose a modified AODV that considers signal strength and energy use when selecting routes to use [24]. AODV has been officially specified by the Internet Engineering Task Force (IETF) in RFC 3561 [25]. AODV routing was proven to be loop-free, meaning route requests will never create routes that go through a node twice [6]. AODV performs better than DSDV routing, making it well suited for MANETs and VANETs [6].

Sybil Attack Detection and Mitigation

Sybil attacks are characterized by a malicious node either stealing another node's identity and communicating as that node, or by fabricating multiple identities and using those multiple identities to launch other types of attacks on the network. Traditional methods of preventing sybil attacks are not available in VANETs. Therefore, sybil attack mitigation is still an open issue. This thesis attempts to prevent sybil attacks rather than detect and mitigate them. The novel system to prevent sybil attacks combines 2 types of prevention techniques, location verification and authentication.

Sybil attack detection and prevention comes in a few different forms. One way to protect against sybil attacks is to use authentication between nodes, utilizing some sort of public or shared key encryption. This allows other nodes to verify a node's identity. Another method is to perform some sort of location verification before communicating with other nodes, also verifying a node's identity. Using traffic data and information about the road network and traffic rules is another way to identify sybil nodes. This is done by noticing deviations in the reported traffic compared to what is expected.

Authentication

A variety of approaches use authentication mechanisms to detect and prevent sybil attacks. Chang, et al propose a method of utilizing RSUs and public key encryption to create a chain of authorized messages as vehicles travel through the network [7]. RSUs will change key pairs periodically and utilize a central authority to manage that list of keys. They then use this chain of authorized messages to compare against other nodes and determine if there are groups of nodes with identical paths through the network, which they claim is unlikely and therefore a sign of sybil nodes. A major downside to this method is when road networks are utilized during rush hour and many routes taken are similar or identical between vehicles. Gaikwad and Ragha propose a technique to identify sybil nodes by examining the path proposed by each neighbor from the sender to destination [8]. They utilize RSA signatures on hashes of messages to secure the network, much like the solution proposed in this thesis. The downside to this approach is that there could be many false positives when a group of nodes are all between 2 nodes on the most efficient path from sender to destination, causing them all to appear as sybil nodes. Additionally, this technique requires the sender to be aware of the entire path hop by hop to the destination, while in AODV and other ad-hoc routing protocols only the next hop is known. Kumar, et al proposed a method using RSA encryption to secure messages between sender and destination [9]. They need to transmit the key through a multi hop route, so any node on that path could be a malicious actor and change the public key with the intention of intercepting and replacing messages as they come through encrypted. Khalil and Azer propose a similar encryption technique that utilizes special OBUs and RSUs [10]. This technique requires a government mandated OBU to be installed on all vehicles with a private key included. The algorithm involves RSUs having access to all registered keys in all OBUs and being able to authenticate and issue a network key for use while within the vicinity of the RSU. This solution poses several issues, OBUs would be vulnerable to compromise. They would be installed on personal vehicles and available for access by anyone with access to that vehicle. It also requires a lot of communication with central authorities, introducing possible latency in the network. Liu et al proposes a scheme to share secrets between an RSU and a vehicle while direct communication is available and use that to encrypt messages later when multiple hops are required [11]. Unfortunately, the system only covers messages between RSUs and vehicles, limiting its usability. RSUs may not always be available and utilizing them for all communications between nodes poses a risk of increased latency, especially when RSUs are placed far away from one another. Mohanadevi and

Selvakumar propose a system to disseminate RSA keys between every node in the network [12]. They utilize the black hole detection scheme from another paper and follow it up by sending RSA keys around through nodes that were not flagged as malicious. There are many levels of RSA encryption included and nodes are required to forward signatures from other nodes. This would likely lead to increasingly large message sizes. Particularly when the network grows large.

Position Verification

All position verification methods utilize Received Signal Strength Indicator (RSSI) to locate neighbors and verify their position. RSSI suffers from being easy to trick by manipulating transmission power of a radio signal. Han et al propose a method of utilizing ultrasound (US) in addition to radio frequency (RF) to locate nodes when communicating with them [13]. US is a method of locating objects by bouncing a high frequency sound wave off them. This method was used to prevent both the need for RSUs and relying solely on RSSI. Unfortunately, the range available for US is very small at about 50 meters. This poses a large problem for usability, particularly at high speeds. Garip et al propose INTERLOC a RSSI based localization method to detect sybil attacks [14]. INTERLOC uses a radio propagation model to estimate the position of other vehicles. The neighbors of that vehicle then pool their knowledge and create a small area where it is possible for the vehicle to be located. This is then compared to detect sybil nodes. This method requires that nodes work together to compare data therefore it is susceptible to another malicious node interfering with its operation. Yao et al propose utilizing RSSI to locate and track other vehicles and a method of tracking position changes that protects against sybil nodes modifying their signal strength to spoof their location [15]. Their simulation was limited to 5 vehicles and does not contain mitigation techniques but is a promising method.

Traffic models

Quervedo et al propose a machine learning (ML) algorithm to detect sybil nodes [16]. They attempt to train a model using movement data of real nodes to use with a movement matrix that is generated based on the movement of other nodes. This technique is a hybrid between position verification and traffic model since it requires position data to use with a model trained on movement or traffic data. This model performs well but it is unclear if nodes in the network would have access to the information needed for inputs in the model. Additionally, ML models typically require specialized equipment and a lot of computational resources that may not be available in vehicles. Ayaida et al propose a traffic model-based approach that keeps track of neighbors and compares their reported position and speed to known traffic models and actual road conditions [17]. Lohar et al further expands on this algorithm replacing the speed calculation with one that uses an average [18]. The extension improves the performance of the original algorithm. This system requires knowledge of the road system that may not be available and would likely need to come from RSUs that may not be installed. This method detects sybil attacks but does not necessarily identify the sybil nodes themselves.

Black Hole Attack

Black hole attacks also pose an issue in the functioning of VANETs. There are some promising techniques to detect these types of malicious nodes [5]. These techniques are performed by singular nodes and malicious nodes are identified and blacklisted on a node-by-node basis. This thesis attempts to provide a system for nodes to share the discovery of these malicious nodes throughout the network. This is done in a way, combined with the sybil attack prevention, to prevent any abuse of the reporting system by malicious nodes.

Dangore and Sambare propose a method of black hole node detection in which nodes keep track of the number of times a neighbor sends a route reply vs the number of time that node sends data through a node [19]. If that ratio is out of bounds set by the algorithm, then the node is marked as a black hole node. This system works well but could be circumvented by a grey hole node that forwards just enough packets to keep the ratio within bounds. Yasin and Zant propose a black hole node detection system that uses baited messages sent periodically [20]. This system periodically sends out a route request for a node that does not exist. It determines a node is a black hole node if it responds with a route reply. It uses a low TTL for the request so that they are not forwarded across the entire network. Yadav et al propose a system to detect black hole nodes by tracking and averaging the hop count of route responses [21]. When a hop count outside of the threshold is met then that node is marked as a black hole node because it is proposed that the outliers are artificially created responses by black hole nodes. Tobin et al proposes a similar system that compares multiple paths and determines black hole nodes based on the feasibility of paths not being fabricated based on the average path statistics [22]. This method also includes a blockchain and accusation or reports of malicious nodes being sent to neighbors. There is not much security around the accusation, and they could be abused by a malicious node. All the black hole node detection systems described could be included in the algorithm outlined in the thesis to improve overall performance. In addition, Tobin et al could be improved by the report system outlined in this thesis, which would prevent a malicious node from abusing the reporting system with false reports.

Chapter 3

Methodology

In this thesis a new protocol named LIDAR Enabled Sybil Attack Prevention Ad-hoc On-demand Distance Vector Routing (LESAP) is discussed. It is built on top of Ad-hoc On-demand Distance Vector (AODV) routing. LESAP prevents sybil attacks from occurring, allows for nodes to report the existence of any type of malicious node to the rest of the network, and prevents malicious nodes from taking advantage of that reporting system.

LESAP

To prevent sybil attacks LESAP needs to authenticate and verify the identity of each node. To create a reporting scheme LESAP includes 3 control message types in addition to the control message types included in AODV. To prevent the abuse of the reporting system by malicious nodes, LESAP requires verification of neighboring nodes and multiple reports before blacklisting nodes.

AODV was chosen as the routing protocol to modify and compare against due to its performance compared to other well-known routing protocols for ad hoc networks [3][4].

Sybil Attack Prevention

To prevent sybil attacks, a node needs to tie other nodes to singular identities. Traditional models require a Central Authority to validate and authorize nodes [7][11]. Within an ad hoc network this becomes more difficult to manage. One option is to use Roadside Units (RSU) as those central authorities, as in several recent papers [18][10][11][7]. This requires municipalities to deploy RSUs on all roadways and maintain the hardware and software, particularly its security. Therefore, LESAP is designed with an authentication scheme that does not require RSUs or any Central Authority. Many cars today are outfitted with LIDAR systems. More cars are likely to include LIDAR in the future, the National Highway Traffic Safety Administration (NHTSB) has issued a mandate to have automatic emergency braking systems installed on all cars by 2029 [26]. LIDAR is an easy way for vehicle manufacturers to meet those requirements. LIDAR sensors today have ranges of 400 to 500 meters [27][28][29][30]. Therefore, LESAP will assume a LIDAR range of 500 meters. LESAP uses a sybil attack prevention system based on the capability of LIDAR systems to identify objects at a distance.

The sybil attack prevention system in LESAP will also utilize RSA 2048-bit encryption to encrypt control messages. Encrypted signatures will be included on the AODV Route Request, AODV Route Reply, and LESAP Report control messages. To distribute public keys to neighbors, LESAP includes 2 additional control message types, Need Key and Send Key. LESAP utilizes an existing feature of AODV to trigger the authentication process.

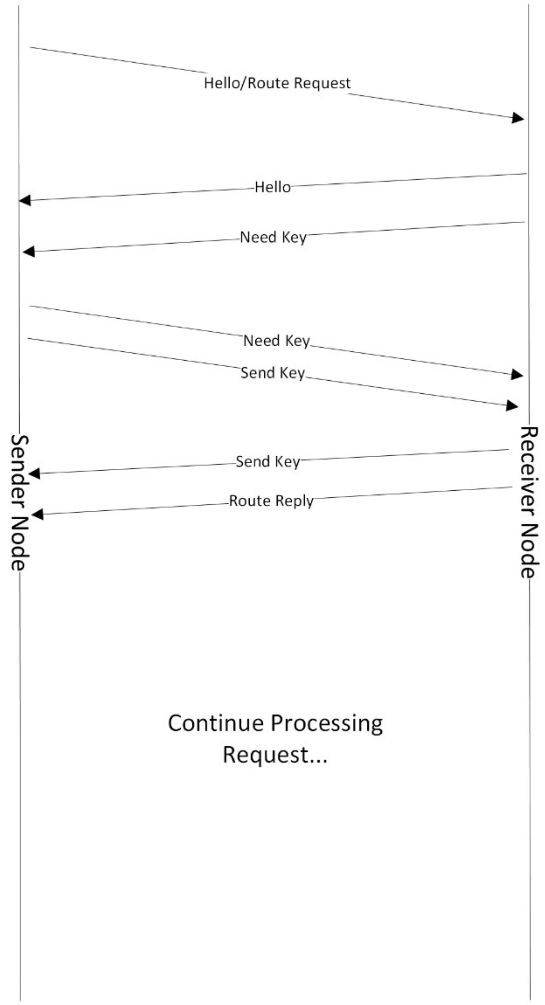

Figure **3-1**: Diagram of authentication process.

Each node maintains a LIDAR neighbors table the contains the nodes that it has verified via LIDAR and their RSA public key. Hello messages are generated and broadcast to all neighbors at the Hello interval which is set to 1 second. A Hello message is a particular type of Route Reply in which the origin and destination are the same and are the sending node's address. When a node sends a Hello message, the receiving node will check if it has that node in its LIDAR neighbors table and has an RSA public key for it. If it does, then it will increase the timeout on that record. If it does not then it will determine if it can see the node via LIDAR, this happens by checking the distance between nodes in the mobility model within ns-3. If the node is within the LIDAR distance, set to 500m by default, then a response will be created.

The receiving node will respond with its own Hello message and a Need Key message. Upon receiving a Need Key message, a node will do the same LIDAR distance check and respond with a Send Key message that includes the public key if the sender is within LIDAR distance. Figure 3-1 shows the authentication process between two nodes, triggered by either a route request or a hello message.

When a node receives a Send Key message it will check the reported location for a node on the mobility model, simulating a LIDAR system. If a node is found, then it will check to see if any other node it knows about should be in that same location. If it finds a collision it will mark the new node as malicious and blacklist it. It will then begin sending reports about that node at the next Hello interval. This system will prevent a node from stealing any other node's identity or creating multiple identities since there must be a location with a vehicle to validate. Control messages that are meant to have a signature but without one will be dropped. Hello messages are exempt from the need for a signature due to their use in beginning the key exchange process.

Malicious Node Reporting

In addition to the prevention of sybil attacks, LESAP includes a system for disseminating knowledge on the existence of malicious nodes of any type. When a node knows about a malicious node, that node will be included on its blacklist. This means that the route to that node is marked as unidirectional. LESAP includes a reports table to keep track of reports of malicious nodes and whether they are blacklisted. Reports on the nodes that are blacklisted are broadcast to all neighbor nodes at the Hello interval which is set to 1 second be default.

Preventing Abuse of Reporting System

To prevent a new type of attack, one in which a malicious node where to report normal nodes as malicious through the new reporting scheme, LESAP requires that multiple reports come in over a set amount of time before blacklisting. Within the simulation LESAP permanently blacklist nodes that the node has received 2 reports about being malicious. Therefore, multiple nodes must report that node for it to be blacklisted. The threshold by default is 2 reports so 2 or more reports are required to blacklist a node based on reports. The collision detection and any other malicious node detection system that could be combined with LESAP directly blacklist nodes without needing to meet the report threshold. But if the node has not witnessed the malicious activity itself then the report threshold is required. This will prevent malicious nodes from unilaterally reporting other nodes as a new method of DDOS attacks or some other attack scheme.

Additional detection systems

LESAP is designed to allow for other malicious node detection systems to be added on top of it. Such as a baited blackhole detection as Yasin and Zant propose since this can be down at the application level rather than in the routing protocol [20]. The application would then need to perform a blacklist operation at the routing protocol level. Dangore and Sambare propose a system to keep track of the number of times a node forwards messages and sends route replies [19]. They compare these values to detect black hole nodes, this could be integrated into LESAP by utilizing the lidar neighbor table and blacklisting detected nodes. Yadav et al proposes a system to detect black hole nodes based on hop count of returned routes [21]. This could be integrated into LESAP in a similar way by adding this functionality to the routing table and subsequently blacklisting detected nodes.

Diagram

Communications between nodes in LESAP require that nodes be within LIDAR distance and communications range to trade RSA public keys and validate position. Figure 3-2 demonstrates this on a three-lane roadway.

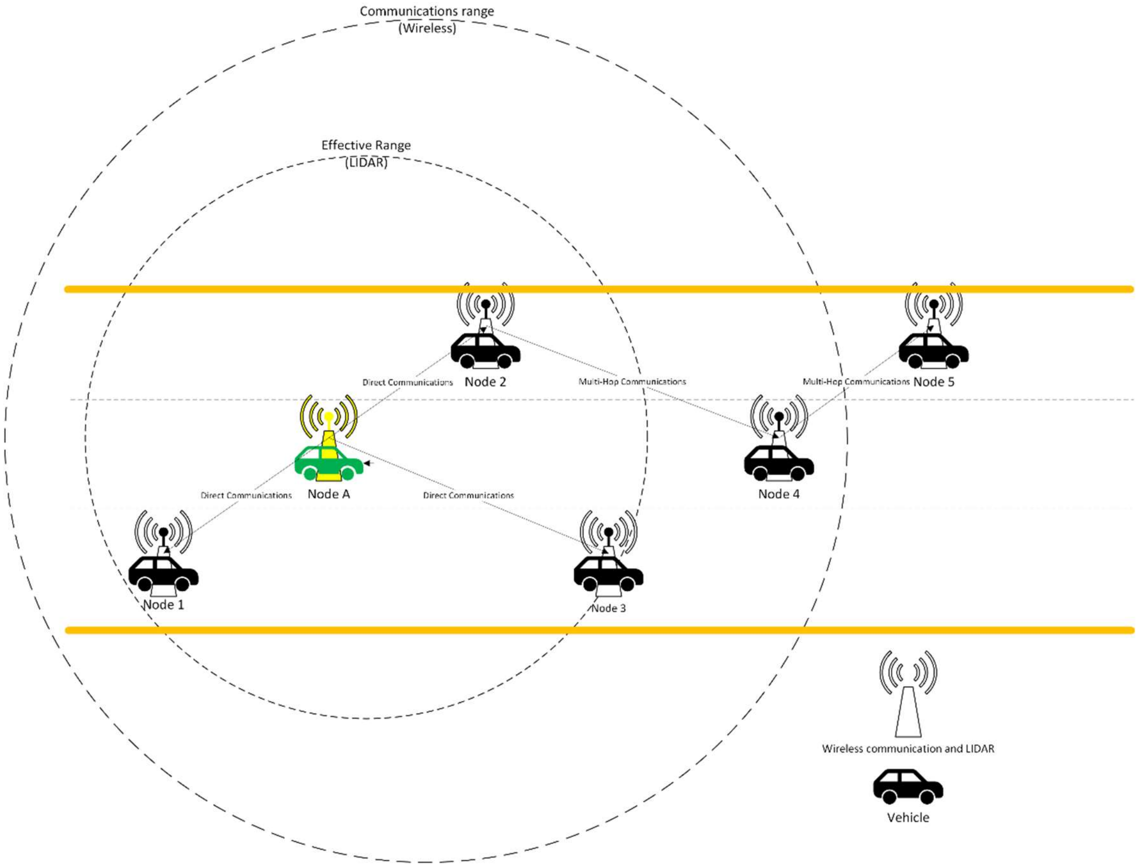

Figure **3-2**: Diagram of communications on a three-lane roadway.

Figure 3-2 shows a network of 6 vehicles from the perspective of Node A. Node A can trade public keys and validate nodes 1, 2, and 3 on LIDAR. Therefore, Node A can send messages directly to those vehicles. Even though Node 4 is within communications range, communications between Node A and Node 4 will be ignored since they cannot validate each other on LIDAR. Therefore, Node A must send communications to Node 4 via a multi-hop route through Node 2. Node 5 is neither in communications range or LIDAR range, therefore, Node A must use a multi-hop route to send messages to Node 5 as well.

Chapter 4

Results

This experiment has been set up to compare Ad Hoc On-Demand Distance Vector Routing (AODV) with then new extension to AODV called LIDAR Enabled Sybil Attack Prevention Ad Hoc On-Demand Distance Vector Routing (LESAP).

Experiment Setup

This experiment includes twelve different simulations. The simulation is repeated over both the AODV algorithm and my LESAP algorithm, it is run with and without malicious nodes, and it is run on simulations of a 3-lane highway in one direction containing 25, 50, and 75 cars. Therefore, the 12 simulations where as follows, AODV with 25 nodes; AODV with 20 nodes and 5 malicious nodes; AODV with 50 nodes; AODV with 40 nodes and 10 malicious nodes; AODV with 75 nodes; AODV with 60 nodes and 15 malicious nodes; LESAP with 25 nodes; LESAP with 20 nodes and 5 malicious nodes; LESAP with 50 nodes; LESAP with 40 nodes and 10 malicious nodes; LESAP with 75 nodes; LESAP with 60 nodes and 15 malicious nodes.

Three different lanes are created by altering the position of nodes on the y-axis by 5 or 10 meters from the base position of 20. This is done by taking the modulus of the node index by 3. The position in the lane of travel is accomplished in the same way using the x-axis. A starting position on 20 is modified by the node index times 40, therefore creating nodes that are each 40 meters ahead of the previous one on the x-axis. The combination of these 2 methods results in nodes in 3 lanes of travel, each node following behind another by 120 meters and 40 meters behind the node in the next lane. Velocity is accomplished by using a starting velocity of

30 m/s and adding the modulus of the node index by 3. This gives lanes of travel at speeds 30m/s, 32 m/s, and 34 m/s. Figure 4-1 shows this setup with speeds and distances between nodes.

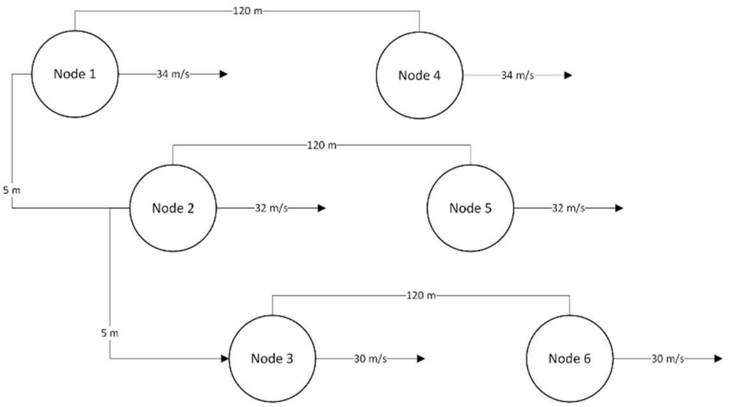

Figure **4-1**: Simulation setup with speeds and distances between nodes.

In ns-3, a script was created using the manet-routing-compare.cc script as a basis, to run simulations in ns-3 but modified to use the constant velocity mobility model with the positions and velocities set as specified above. It was further modified to compare the 12 simulations discussed rather than a variety of different routing algorithms. The OnOff application in ns-3 was used to instruct nodes to send messages to certain other nodes at an interval of 10 per second. Nodes send messages 10 times per second to the nodes that were 5 and 10 ahead of them in the list of nodes. Messages were sent between seconds 50 and 150 of the simulation, which lasted 180 seconds in total. These messages reflect basic safety messages and are 128 bytes each. They are UDP messages therefore the total size of the packets is 136 bytes. This allowed for random messaging to be written based on easily constructed logic. In the simulations with malicious nodes, every 5th node was malicious, every 10th node was a blackhole node and the rest were greyhole nodes which dropped packets 10% of the time rather than all the time. The next 4 nodes begin the simulation with that node on their blacklists and will send out those reports to other nodes at the hello interval which is set to 1 second. Once a node receives 2 reports about a malicious node from different other nodes, it will add that node to its blacklist. Reports and blacklists can also be generated by a node sending its key and failing the collision detection. Figure 4-2 shows the setup with the inclusion of malicious nodes at every 5th node.

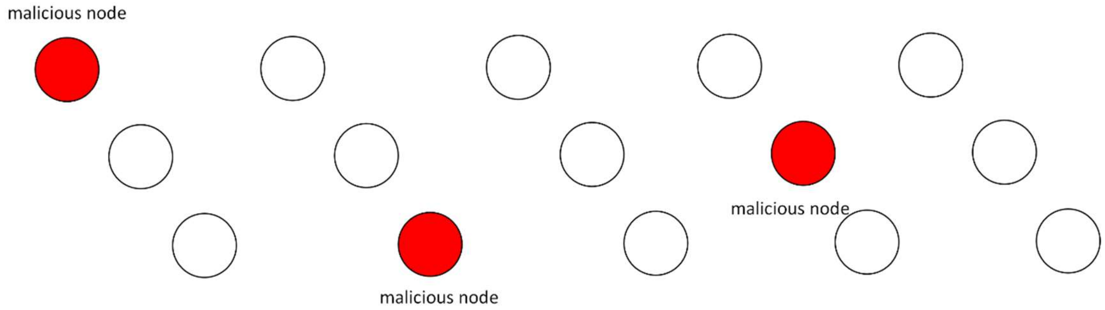

Figure **4-2**: A representation of the simulation setup with malicious nodes.

Many different files were logged to give a variety of ways to inspect the results of the simulation. Logging was added to each place in both algorithms where a packet might be forwarded, dropped, or otherwise handled. When messages are received at their destination was also logged. This is the logging that is being used to report my results the results being compared between AODV and LESAP. Namely, packet delivery ratio, end-to-end delay, normalized routing load, and throughput. Some code that would modify the behavior of nodes when they are set as malicious was also needed, this code was added to both AODV and LESAP. In the simulations without malicious nodes, nodes were simply not flagged as malicious in the routing protocol code. Pcap files of the entire simulation were also logged. Flow monitor files were also added, these can be used with the flow monitor tool in ns-3. Lastly, a log of each node's blacklist was added, at the end of the simulation and as nodes were blacklisted or suspected. This will be used to report on the effectiveness of the reporting system.

Limitations

The experiment design is limited in its ability to fully evaluate LESAP regarding performance in a real-world environment. The mobility model used is a constant velocity model in which the nodes do not change speed or direction. This is not likely to occur in a real-world highway scenario, cars could change speed, the road could have turns, and cars could enter and exit the highway. The map used is a simulated 3-lane highway moving in one direction without ramps or interchanges. In a real-world scenario, there may be some communication between nodes going in the opposite direction, as well as nodes entering, exiting, or passing the highway in question. The simulation map placed the nodes at exact intervals from one another, this is also unlikely to occur in a real-world highway scenario for an extended period. The LIDAR system also assumed all objects within 500 meters could be seen. This included any objects that may have been obstructed by other objects. Ns-3 does not support nodes entering and exiting the simulation at times other than the start and stop times, therefore the simulation could not be designed to account for this possibility. Additionally, the RSA encryption was approximated, no encryption was done but messages were padded by the size of the key or the encrypted message for validation.

Deploying LESAP in a live environment, whether a closed track or on public roadways, would allow for additional pieces to be fully tested. LIDAR object detection and RSA encryption could be fully tested. Nodes could enter and exit the simulation as needed and true sybil and blackhole attacks could be performed by some nodes. This would give a much better picture of the effectiveness of the algorithm if deployed.

Simulation Results

To measure the effectiveness of the malicious node reporting system, 3 metrics will be used. The number of false positives in the collision detection system will be shown for each simulation. This will measure the number of normal nodes that are flagged as having collided with another node which indicates that a node is pretending to be another node that we already know about. The ratio of nodes that were able to blacklist the malicious nodes will be calculated, this will show the effectiveness of requiring the 2 reports of nodes, particularly with only a small number of nodes knowing about the malicious node in the first place since only 4 nodes start each simulation knowing about each malicious node. A calculation of the ratio of nodes that were able to authenticate a single report and suspect a malicious node of being malicious will also be used to show the cost of the 2-report limit set. This will differentiate the effectiveness of receiving 1 report vs multiple reports in the system.

To compare AODV with LESAP, 4 main metrics will be used. The throughput of data packets meant to simulate Basic Safety Messages (BSM) in the network will be measured. The packet delivery ratio is the ratio of sent packets to packets received at the destination. This will be reported on to compare LESAP and AODV. Normalized routing load will be calculated for each algorithm and each simulation. Normalized routing load is the number of routing packets sent for each delivered data packet. Lastly, the end-to-end delay of BSM data packets in the network as well.

Nodes Identified (blacklisted)

The focus of LESAP is to create a system to report nodes as malicious to other nodes. LESAP requires at least 2 reports before a node will blacklist a reported node. The blacklist of each node is logged at the end of the simulation and the number of nodes blacklisted through the reporting system for each simulation is shown on Table 4-1.

Table **4-1**: Results of reporting of malicious nodes.

Normal Nodes	Malicious Nodes	Blacklisted nodes across the network at start of simulation	Blacklisted nodes across network at end of simulation	Total possible blacklisted nodes (# malicious nodes * # normal nodes)	Performance
20	5	20	100	100	100%
40	10	40	400	400	100%
60	15	60	900	900	100%

Table 4-1 shows the number of blacklisted nodes across all nodes in the network at the end of the simulation. Performance at the end of the simulation was 100% in all simulations. This means that every normal node knew about and had blacklisted every malicious node.

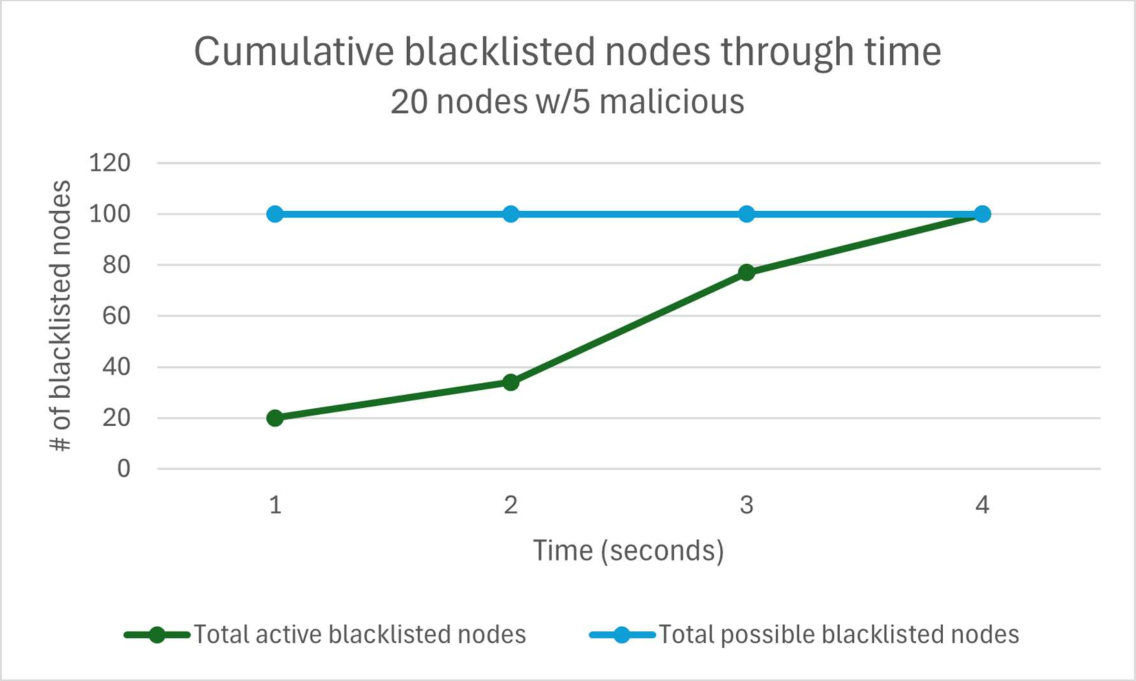

Figure **4-3**: The number of blacklisted nodes over the 20-car simulation by time.

The number of nodes blacklisted through time in the 20-car simulation with 5 malicious nodes is shown in Figure 4-3. The total possible number of blacklisted nodes is 100. If each of the

20 nodes had blacklisted each of the 5 malicious nodes there would be 100 blacklisted nodes in total. The LESAP simulation with 20 nodes showed that the protocol could successfully blacklist all the malicious nodes in under 4 seconds. This indicates that the reporting setup was able to effectively spread the blacklist reports through the network. Each malicious node was on the blacklists of 4 other nodes at the beginning of the simulation and each node needed to receive 2 reports of a malicious node to blacklist it.

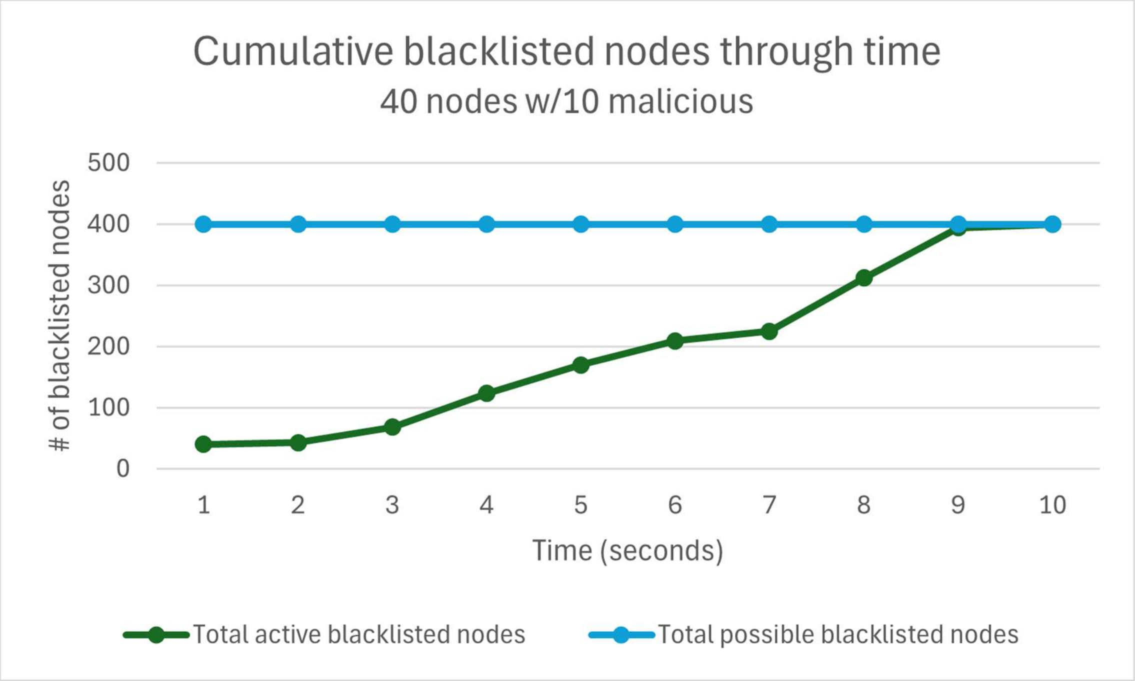

Figure **4-4**: The number of blacklisted nodes over the 40-car simulation by time.

The number of nodes blacklisted through time in the 40-car simulation with 10 malicious nodes is shown in Figure 4-4. The total possible number of blacklisted nodes is 400. This simulation showed that the protocol could successfully blacklist all the malicious nodes within 10 seconds, it did not take much longer than the previous simulation. This indicates that the reporting setup was able to effectively spread the blacklist reports through the network.

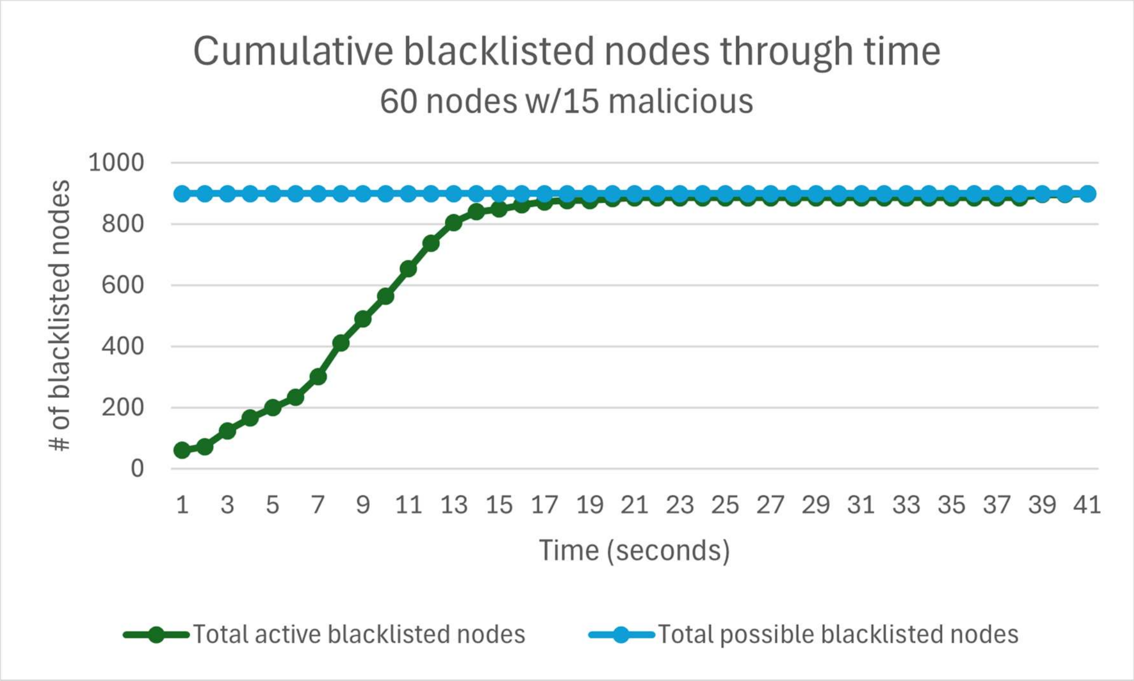

Figure **4-5**: The number of blacklisted nodes over the 60-car simulation by time.

The number of nodes blacklisted through time in the 60-car simulation with 15 malicious nodes is shown in Figure 4-5. The total possible number of blacklisted nodes is 900. This simulation showed that the protocol could successfully blacklist all the malicious nodes within 41 seconds. This indicates that the reporting setup was able to effectively spread the blacklist reports through the network but took 4 times as long as the 40-car simulation and 10 times as long as the 20-car simulation. Future work would be needed to evaluate this further with a variety of different numbers of nodes starting with blacklist values and to simulate malicious nodes that could be detected by the collision detection scheme and then spread organically.

Nodes Suspected (reports but not necessarily blacklisted)

The focus of LESAP is to create a system to report nodes as malicious to other nodes. LESAP requires at least 2 reports before a node will blacklist a reported node. This is to prevent a malicious node from abusing the reporting system. A node suspects another node as being malicious but does not blacklist it when it only has 1 report. The list of suspected nodes is logged at the end of the simulation and includes the blacklist in its number, this is shown on Table 4-2.

Table **4-2**: Results of reporting suspicions of malicious nodes.

Normal Nodes	Malicious Nodes	Suspected nodes across the network at start of simulation	Suspected nodes across network at end of simulation	Total possible suspected nodes (# malicious nodes * # normal nodes)	Performance
20	5	20	100	100	100%
40	10	40	400	400	100%
60	15	60	900	900	100%

Table 4-2 shows the number of suspected nodes across all nodes in the network at the end of the simulation. Performance at the end of the simulation was 100% in all simulations. This means that every normal node knew about and had at least 1 report of each malicious node.

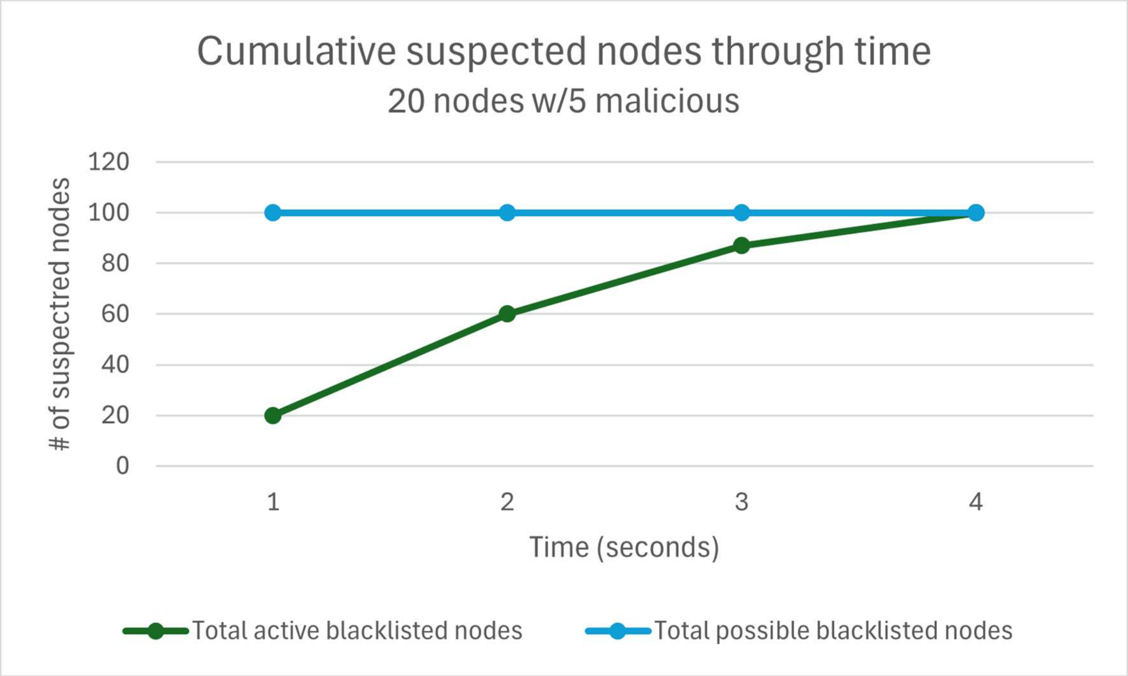

Figure **4-6**: The number of suspected nodes over the 20-car simulation by time.

The number of nodes suspected through time in the 20-car simulation with 5 malicious nodes is shown in Figure 4-6. The total possible number of suspected nodes is 100. These numbers include any node that received a report of a malicious node including those in which there was only 1 report of that node being malicious. The simulation shows that taking away the 2-node limit would not increase the spread rate of blacklisted nodes. Only requiring 1 report would open another line of attack by malicious nodes. A node might try to implement a Denial of Service like attack by broadcasting fake reports of all the other nodes it knows about.

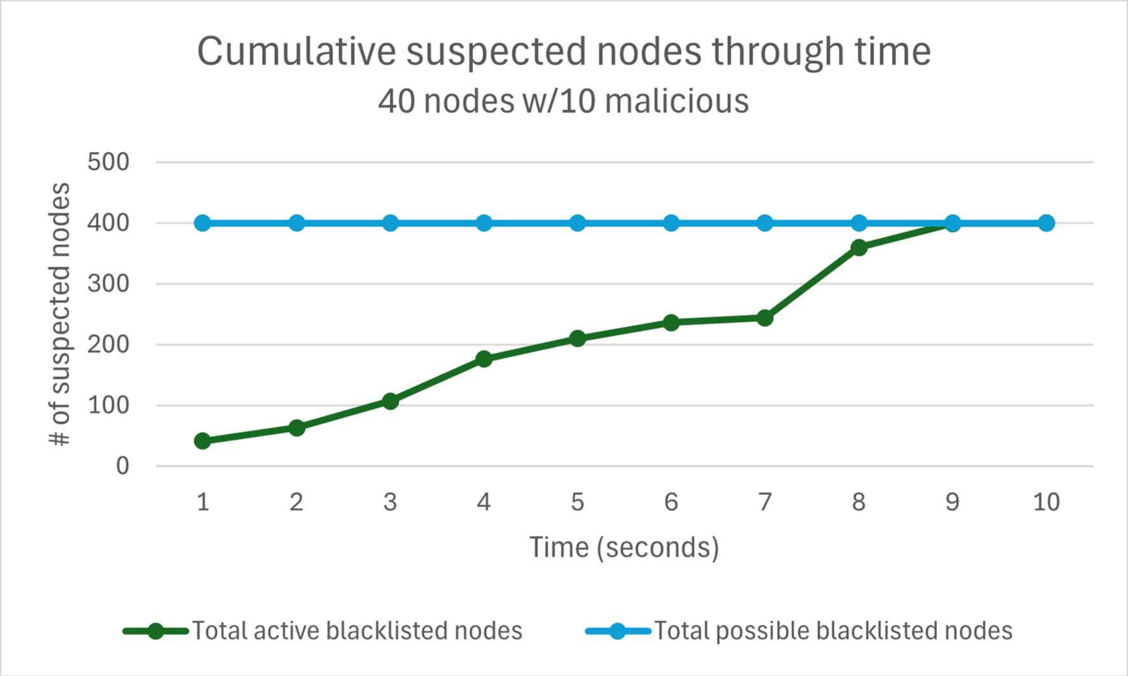

Figure **4-7**: The number of suspected nodes over the 40-car simulation by time.

The number of nodes suspected through time in the 40-car simulation with 10 malicious nodes is shown in Figure 4-7. The total possible number of suspected nodes is 400. The simulation shows that taking away the 2-node limit would not increase the spread rate of blacklisted nodes.

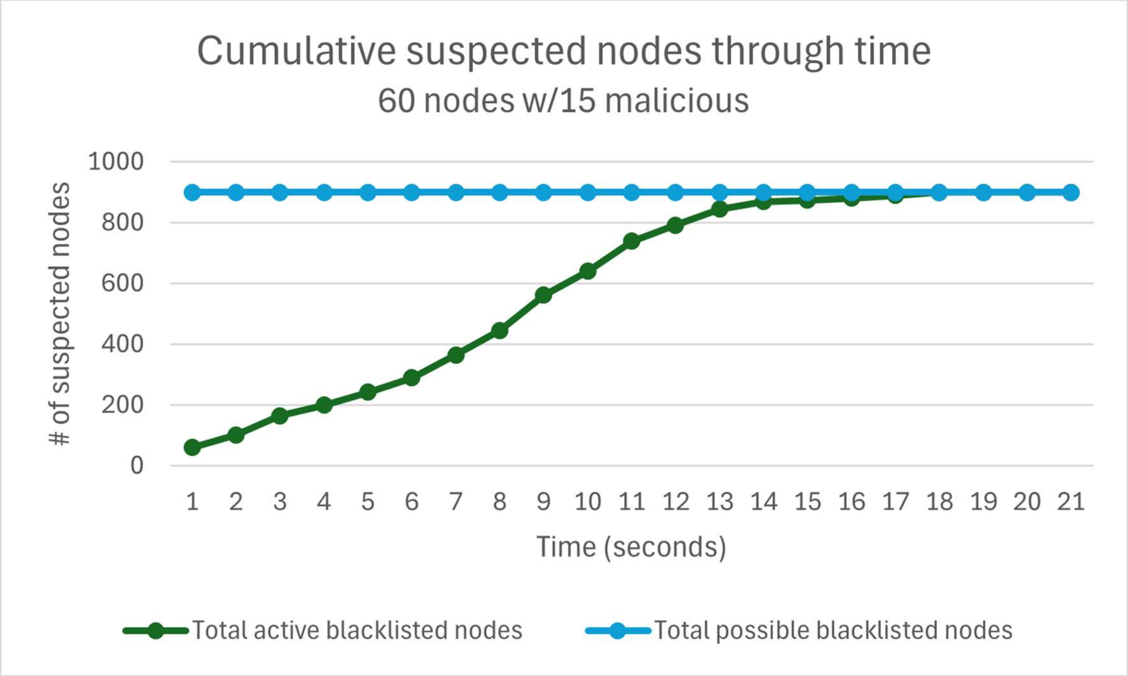

Figure **4-8**: The number of suspected nodes over the 60-car simulation by time.

The number of nodes suspected through time in the 60-car simulation with 15 malicious nodes is shown in Figure 4-8. The total possible number of suspected nodes is 900. The simulation shows that taking away the 2-node limit would increase the spread rate of blacklisted nodes by half, or 20 seconds. The blacklisted values still spread quickly so it may not be advantageous to remove the limit. If another method of preventing Denial of Service attacks were implemented on top of LESAP then it might help to include this provision for larger networks. Otherwise, the protection provided by the 2-node rate outweighs the time it takes to spread knowledge of blacklisted malicious nodes.

Risk of False Positives

LESAP does a quick collision test when authenticating with another node. It tries to determine if the node is claiming to be in a position in which the routing protocol already knows there is a node. There is a risk of false positives in this system, but the experiment did not have any across all simulations. Although the collision detection system is very simplistic and could likely be improved by using another object tracking algorithm built for use with LIDAR. Pairing the routing protocol with a LIDAR object tracking algorithm should improve the protocol's ability to prevent sybil attacks. As is currently configured, a sybil node could possibly fool this system by driving erratically and re-authenticating with neighboring nodes before each change in trajectory. Although the sybil node could still be limited by lowering the active timeout on the LIDAR neighbor table.

Throughput

Throughput measures the ability of the network to transmit data. It is a measure of the amount of data that can be transmitted in a network over time. This is measured below in bytes per second.

Table **4-3**: Throughput for each simulation without malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Throughput (Bytes/Sec)	Total Packet Size (Bytes)
AODV	75	0	10117.41	1814912
LESAP	75	0	4228.97	760960
AODV	50	0	7127.07	1279488
LESAP	50	0	2163.21	389376
AODV	25	0	3198.93	574080
LESAP	25	0	2781.24	498688

Table 4-3 shows the throughput for each simulation without malicious nodes. As can be seen the throughput is greatly affected by the additional requirements of the LESAP protocol. This decrease in overall throughput is due to the limitation on communicating with only nodes that can be identified on LIDAR and keys shared as well as the large control packet sizes. LESAP does better in the simulation with the least cars. The decrease in throughput is very small in that simulation compared to the others.

Table **4-4**: Throughput for each simulation with malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Throughput (Bytes/Sec)	Total Packet Size (Bytes)
AODV	60	15	9151.29	1646848
LESAP	60	15	4224.71	760448
AODV	40	10	6549.56	1177728
LESAP	40	10	2625.34	471936
AODV	20	5	2740.00	492672
LESAP	20	5	3128.43	561024

Table 4-4 shows the throughput for each simulation with malicious nodes. As can be seen in the table above the throughput decreases in the presence of malicious nodes. But it also decreases due to the additional requirements of the LESAP protocol. While LESAP decreases the throughput overall, it does show less decrease in the presence of malicious nodes. IN the simulation with the least number of nodes LESAP achieves a higher throughput that AODV and nearly the same as AODV without malicious nodes. The malicious nodes being simulated in this experiment are certainly damaging to the network's stability, but certain attack types could cause issues that are not necessarily seen in this data. LESAP does prevent nodes from utilizing multiple identities and performing a sybil attack on the network, which is not simulated here. A sybil attack could take many forms, including the modification of messages or the dropping of certain messages only. Those attacks would also be prevented through the reporting system. The signing of messages and reporting of malicious nodes adds more utility to the simulation than can be shown in throughput alone. LESAP is designed so that it can be combined with other modifications to AODV that will prevent or detect other attacks or improve performance. Other methods may also affect the throughput both negatively and positively.

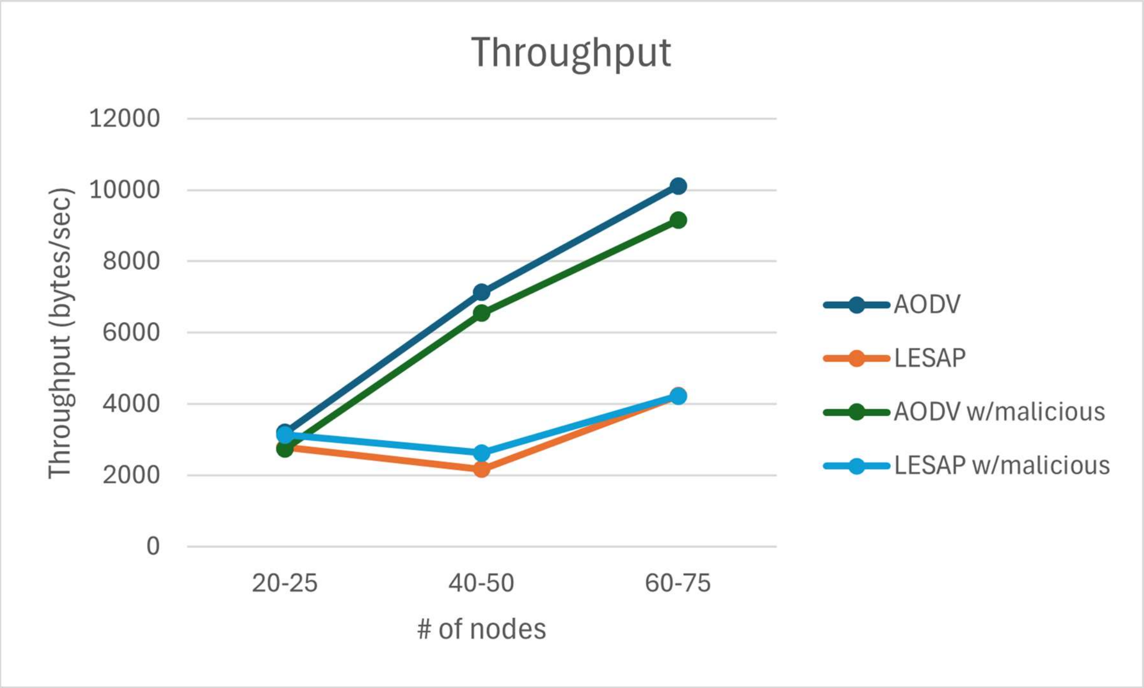

Figure **4-9**: Total throughput over all simulations.

Figure 4-9 shows total throughput across all simulations. As can be seen in the table above the throughput is constant in simulations with less nodes but diverges as more nodes are added. LESAP shows a decrease in the throughput overall in the larger simulations but does show an increase in the presence of malicious nodes. LESAP does not see a change in throughput in the largest simulation. This is because of the large message sizes in LESAP, the network is at maximum load regardless of the malicious nodes and therefore cannot improve beyond where it is at baseline.

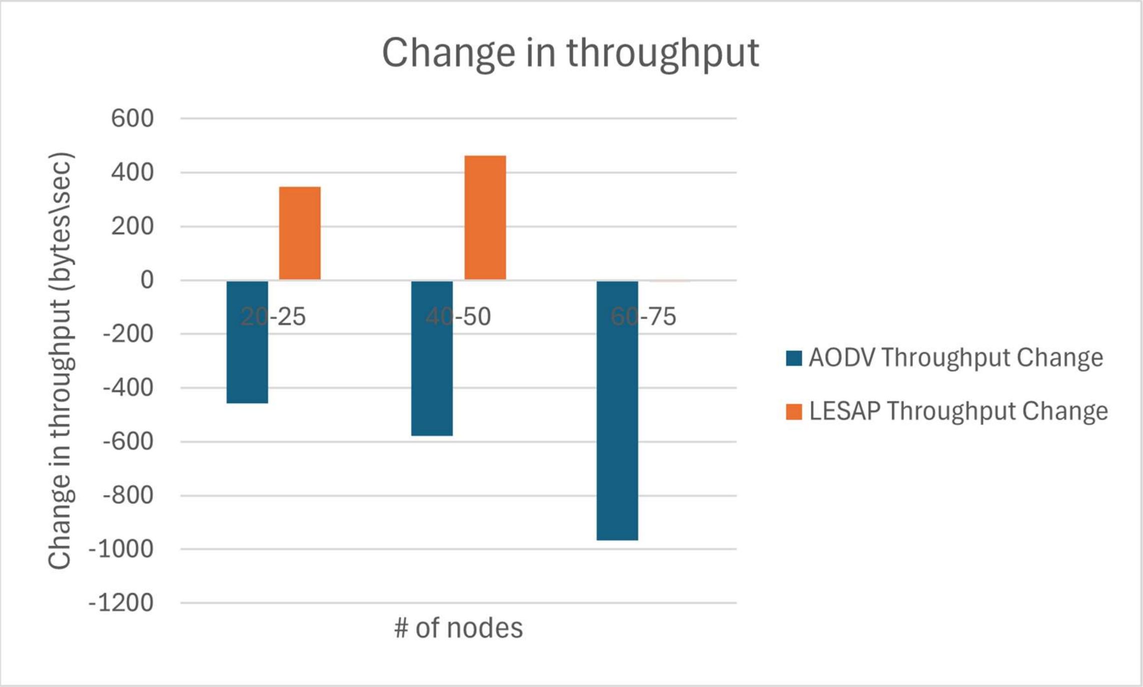

Figure **4-10**: Change in throughput with the addition of malicious nodes.

Figure 4-10 shows the change in throughput when malicious nodes are introduced into the simulations. As can be seen in the table above the throughput is affected by the presence of malicious nodes. AODV sees a large decrease in throughput when malicious nodes are included but LESAP sees an increase in throughput in the smaller two simulations when malicious nodes are introduced. In the simulations used, cars were sufficiently close to one another that there were multiple paths available. Therefore, the malicious nodes were not able to fully deny message transmission meaning the decrease in AODV would likely be much greater if that were not the case. While LESAP decreases the throughput overall, it does show an increase in the presence of malicious nodes. The largest LESAP simulation did not see a change in throughput due to the large control messages taxing the network regardless of the malicious nodes.

End-to-End Delay

End-to-end delay is a measure of the average amount of time it takes a data message to reach its destination. End-to-end delay is measured in seconds. End to end delay shows much of the same trends as throughput.

Table **4-5**: End-to-End delay for each simulation without malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Avg End-To-End Delay (Seconds)
AODV	75	0	0.0798
LESAP	75	0	0.4336
AODV	50	0	0.0728
LESAP	50	0	0.5524
AODV	25	0	0.0369
LESAP	25	0	0.0329

Table 4-5 shows the end-to-end delay for each simulation without malicious nodes. End to end delay shows much of the same trends as throughput. End-to-end delay increases greatly in LESAP except in the simulation with the least number of nodes. In the 25-car simulation LESAP performs better on this metric than AODV. LESAP is limited by the large size of its control messages but in cases with less nodes this does not matter. This seems to indicate that LESAP is particularly well suited to smaller clusters of nodes within close proximity to one another.

Table **4-6**: End-to-End delay for each simulation with malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Avg End-To-End Delay (Seconds)
AODV	60	20	0.0791
LESAP	60	20	0.3211
AODV	40	10	0.0747
LESAP	40	10	0.4761
AODV	20	5	0.0289
LESAP	20	5	0.0276

Table 4-6 shows the end-to-end delay for each simulation with malicious nodes. End to end delay shows much of the same as throughput. End-to-end delay increases in LESAP except in the simulation with the least number of nodes. Although the increase in delay is less in these simulations than it is without malicious nodes. In the 20-car simulation LESAP performs better on this metric than AODV. LESAP is limited by the large size of its control messages but in cases with less nodes this does not matter.

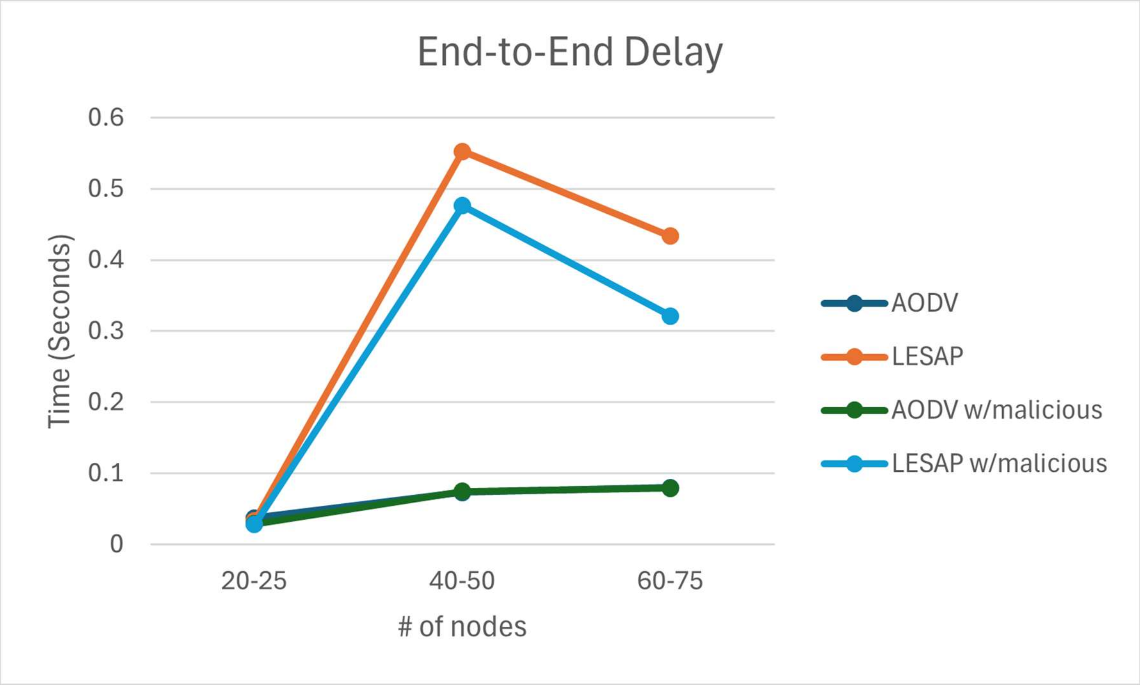

Figure **4-11**: Total end-to-end delay over all simulations.

Figure 4-11 shows total end-to-end delay across all simulations. End to end delay shows much of the same as throughput. AODV is not largely affected due to the closeness of the nodes in the simulation. LESAP sees an improvement in delay in the presence of malicious nodes though. LESAP does show the same delay and throughput as AODV in the 20-car and 25-car simulation.

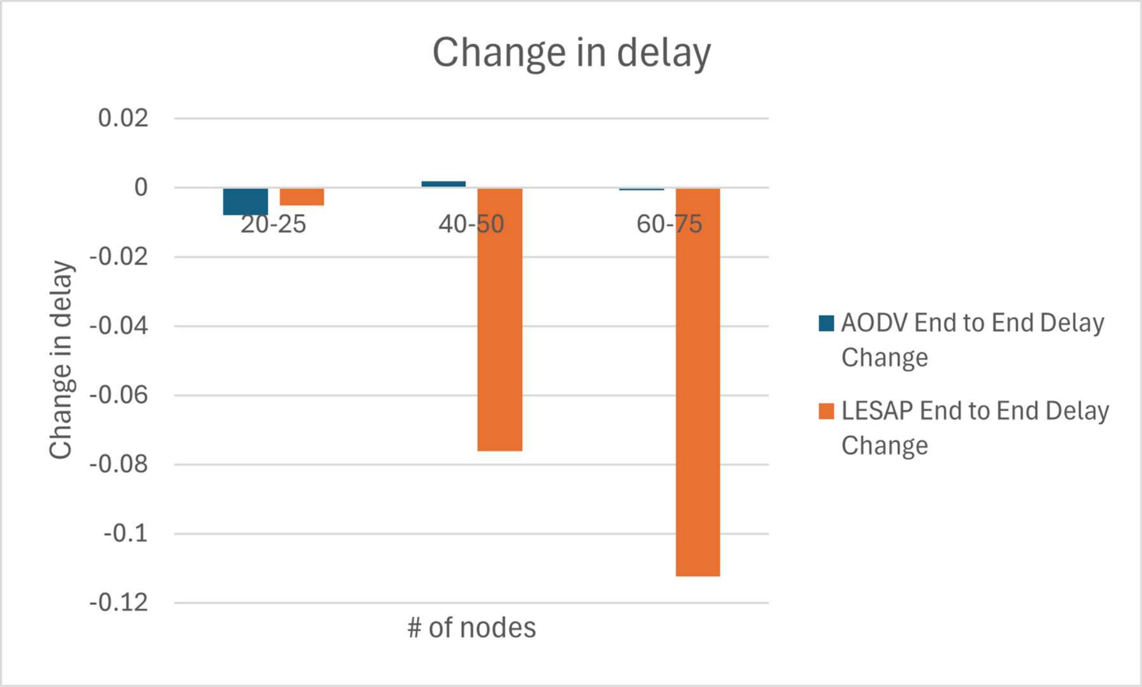

Figure **4-12**: Change in end-to-end delay with the addition of malicious nodes.

Figure 4-12 shows the change in end-to-end delay when malicious nodes are introduced into the simulations. AODV is not largely affected by malicious nodes in this simulation set-up. LESAP can be seen to improve delay in the presence of malicious nodes though. Except in the smallest simulation in which its delay is better than AODV without malicious nodes already.

Packet Delivery Ratio

Packet delivery ratio (PDR) is the ratio of data packets sent to data packets received at their end destination. Packet delivery ratio is a very similar measurement to throughput in this simulation since all data packets are the same size.

Table **4-7**: Packet Delivery Ratio for each simulation without malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Total Sent Packets	Total Received Packets	Packet Delivery Ratio
AODV	75	0	18600	14179	0.76
LESAP	75	0	18600	5945	0.32
AODV	50	0	12400	9996	0.81
LESAP	50	0	12400	3042	0.25
AODV	25	0	6200	4485	0.72
LESAP	25	0	6200	3896	0.63

Table 4-7 shows the packet delivery ratio for each simulation without malicious nodes. LESAP sees a decrease in the PDR over AODV. This is directly related to the throughput and the same pattern emerges. LESAP does much better in the simulation with less cars. This is the cost of authentication, since control messages are so large.

Table **4-8**: Packet Delivery Ratio for each simulation with malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Total Sent Packets	Total Received Packets	Packet Delivery Ratio
AODV	60	15	18600	12866	0.69
LESAP	60	15	18600	5941	0.32
AODV	40	10	12400	9201	0.74
LESAP	40	10	12400	3687	0.30
AODV	20	5	6200	3849	0.62
LESAP	20	5	6200	4383	0.71

Table 4-8 shows the packet delivery ratio for each simulation with malicious nodes. The packet delivery ratio in AODV is reduced by the existence of malicious nodes but the same pattern emerges that was seen in throughput. LESAP sees a decrease in the PDR over AODV, though not as much with malicious nodes in the simulation. LESAP had a better PDR in the 20-

car simulation than AODV. LESAP performs better in simulations with less nodes. LESAP also prevents nodes from utilizing multiple identities and performing a sybil attack on the network.

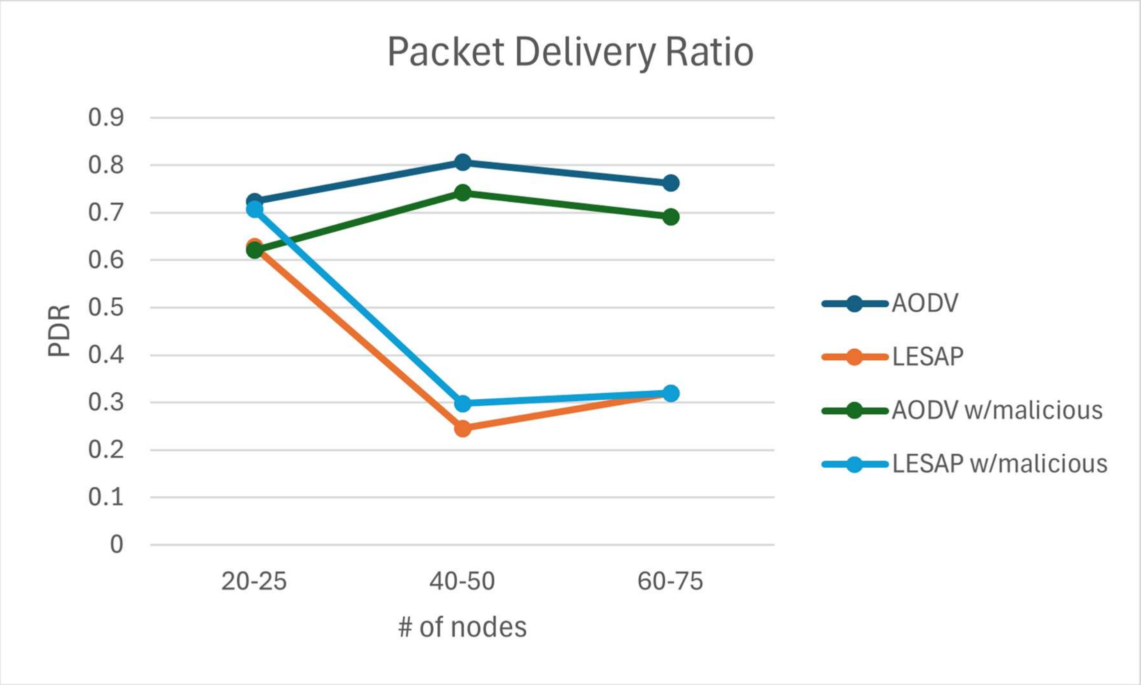

Figure **4-13**: Total packet delivery ratio over all simulations.

Figure 4-13 shows total packet delivery ratio across all simulations. As can be seen in the chart above the PDR the same in LESAP with malicious nodes as it is in AODV without them and vice versa. LESAP sees a decrease in PDR overall in the larger simulations but does show an increase in the presence of malicious nodes, except in the largest simulation. The network in the largest simulation with LESAP is saturated with traffic due to the control messages and therefore we do not see a difference with and without malicious nodes.

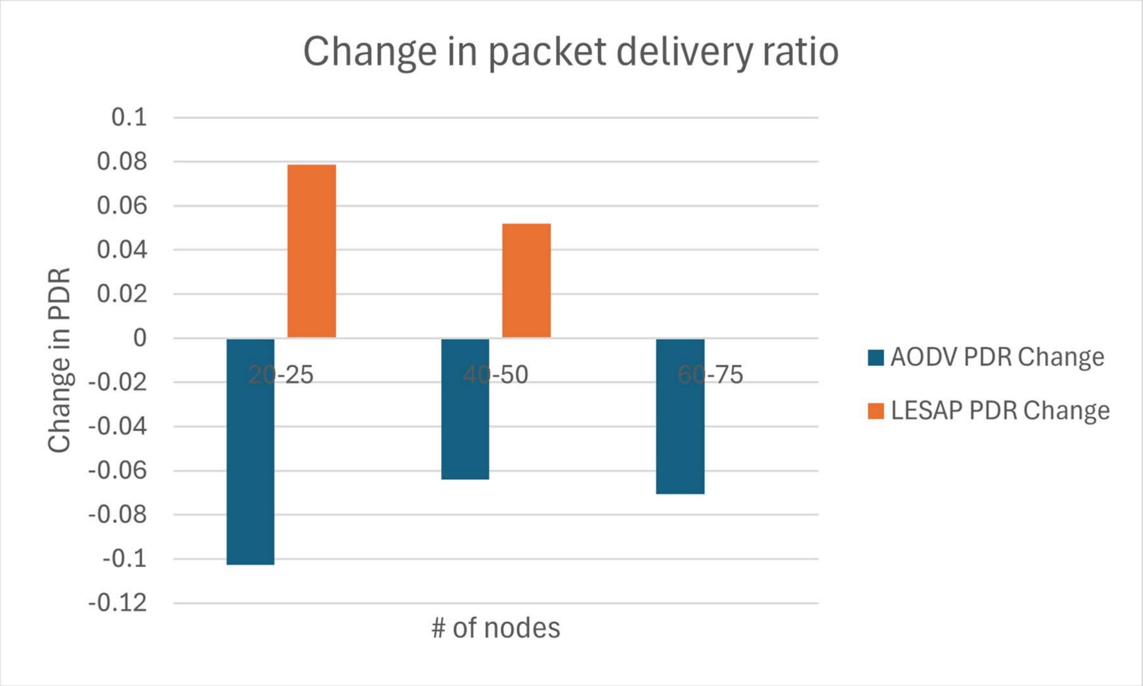

Figure **4-14**: Change in packet delivery ratio with the addition of malicious nodes.

Figure 4-14 shows the change in packet delivery ratio when malicious nodes are introduced into the simulations. Apart from the largest simulation, LESAP sees an increase in PDR with the addition of malicious nodes. AODV sees a larger decrease in PDR with the existence of malicious nodes. This follows the same pattern as throughput and for the same reasons outlined in that section.

Normalized Routing Load

Normalized routing load (NRL) is a measure of the number of control packets that are needed per delivered data packet. This measure can show how efficient the routing algorithm is at transferring data.

Table **4-9**: Normalized Routing Load for each simulation without malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Number of Control Packets	Number of Data Packets	Normalized Routing Load
AODV	75	0	2609431	14179	184
LESAP	75	0	1195015	5945	201
AODV	50	0	1127699	9996	112
LESAP	50	0	751670	3042	247
AODV	25	0	247336	4485	55
LESAP	25	0	314817	3896	80

Table 4-9 shows the normalized routing load for each simulation without malicious nodes. Normalized routing load in AODV slowly increases as more nodes are added. In LESAP the 50-car simulation has a larger overhead than the simulations with either more or less cars. This indicates that there may be a peak in control packet need around that size. LESAP tends to have a higher normalized routing load in general. This is due to the additional authentication messages traded by nodes identified via LIDAR. It is important to note that some of the control packets in LESAP are much larger than they were previously in AODV. This is because they need to be signed using RSA encryption and require a minimum key size of 2048 bytes and therefore a minimum size of 2048 bytes.

Table **4-10**: Normalized Routing Load for each simulation with malicious nodes.

Routing Protocol	Number of Normal Nodes	Number of Malicious Nodes	Number of Control Packets	Number of Data Packets	Normalized Routing Load
AODV	60	15	2020867	12866	157
LESAP	60	15	1988748	5941	334
AODV	40	10	1054009	9201	114
LESAP	40	10	1470575	3687	398
AODV	20	5	194822	3849	50
LESAP	20	5	409972	4383	93

Table 4-10 shows the normalized routing load for each simulation with malicious nodes. With malicious nodes, like without them, normalized routing load in AODV slowly increases as more nodes are added. In LESAP the 40-car simulation has a larger overhead than the other simulations. LESAP tends to have a higher normalized routing load in general. Some nodes are being added to blacklists, the report messages about those nodes are also control messages under LESAP. Reports are broadcast out to all neighboring nodes once per second, and this adds many mode control packets.

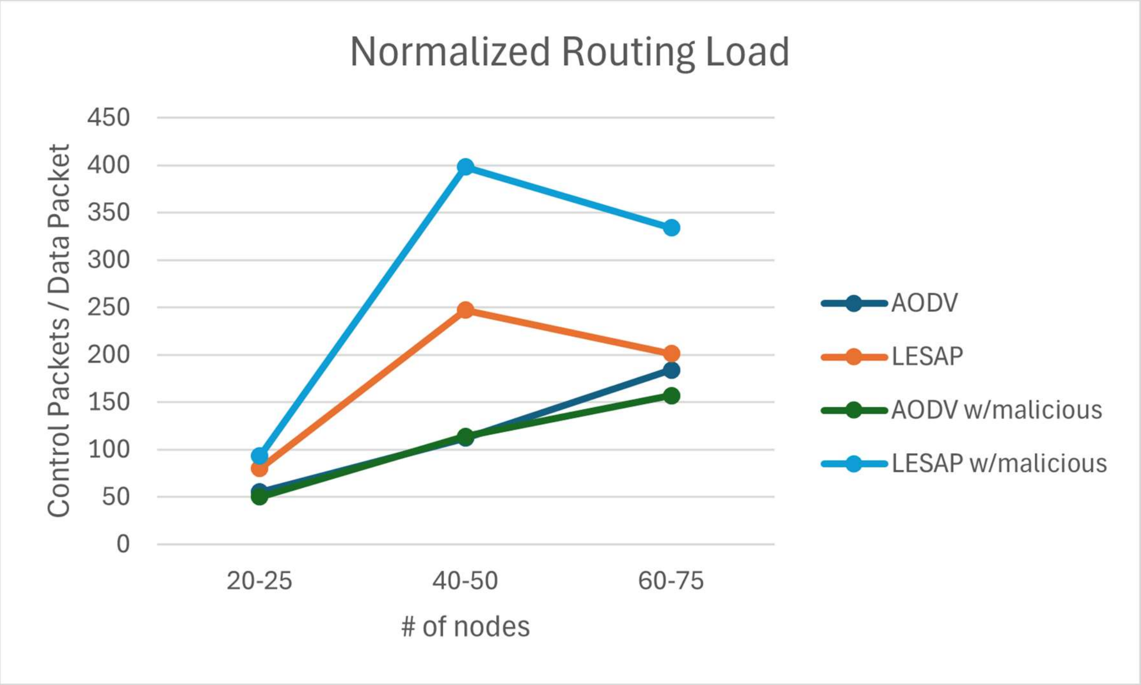

Figure **4-15**: Total normalized routing load over all simulations.

Figure 4-15 shows normalized routing load across all simulations. Apart from the 40-car and 50-car simulations, LESAP has a normalized routing load very similar to AODV. The normalized routing load in the largest simulation does increase for LESAP with the introduction of malicious nodes, this is due to the large number of report messages generated and the lower packet delivery ratio. LESAP does not require many more control messages in the simulation with the least cars. Once again, this indicates that LESAP performs best in smaller groupings of cars.

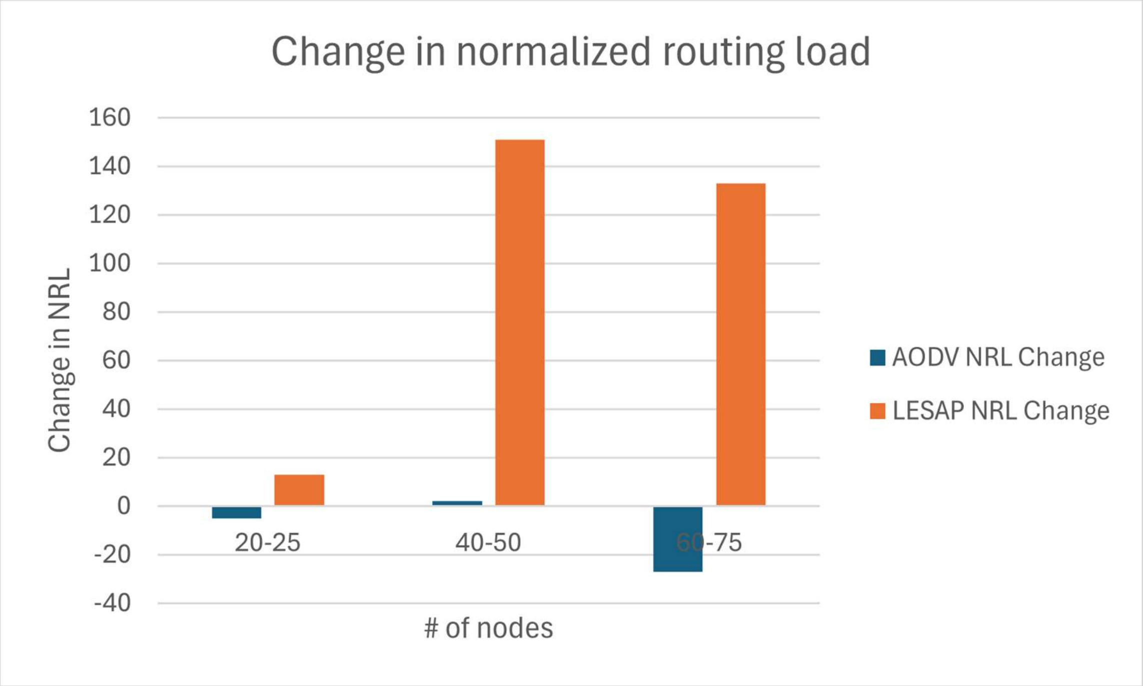

Figure **4-16**: Change in normalized routing load with the addition of malicious nodes.

Figure 4-16 shows the change in normalized routing load when malicious nodes are introduced into the simulations. Normalized routing load keeps steady in AODV when malicious nodes exist but increases under LESAP. In the 20-car simulation LESAP does not see much of an increase. The effect of the additional authentication and reporting message causes the NRL to increase, particularly when the simulation is larger.

Overall, the performance of LESAP shows a decrease in some important metrics. But the reporting system is very successful in informing all nodes throughout the network about malicious actors quickly. And the authentication of nodes which prevent multiple identities and sybil attacks is worth the performance cost.

Chapter 5

Conclusion

This thesis proposed a new authentication and reporting extension to AODV that would authenticate nodes using LIDAR to prevent sybil attacks, LIDAR Enabled Sybil Attack Prevention AODV (LESAP). Furthermore, the reporting system would allow for nodes to effectively blacklist malicious nodes and let other nodes know about it to universally blacklist misbehaving nodes. LESAP utilized LIDAR to guarantee that a node existed in space and was not claiming multiple identities during the route discovery process. The reporting system was hardened against attacks as well, requiring multiple reports combined with authenticating nodes to prevent a new type of malicious attack on the reporting system itself.

The experiment was set up to test this new algorithm. A 3-lane highway in one direction was simulated. The 3 lanes of travel had cars evenly dispersed with 40 meters between them. Cars in the same lane were all 120 meters apart and malicious nodes were evenly dispersed within the lanes of traffic. This was used to set up simulations using AODV and LESAP, with and without malicious nodes being included. This created a total of 12 simulations that were run using network simulator 3 (ns3). The results show LESAP can quickly spread information about malicious nodes through the network.

In the results, it is shown that while there is a trade-off in performance via throughput and packet delivery ratio the LESAP protocol improves on its performance when malicious nodes are present. Messages are authenticated and attacks by malicious nodes can be prevented throughout the network once detected. The authentication prevents sybil attacks entirely and with the addition of other detection algorithms can prevent all other attacks through the reporting scheme. In smaller clusters of vehicles, LESAP performed close to the level of AODV on all metrics.

In the future, this algorithm could be improved to try to decrease overhead by using Diffie-Hellman key exchange to exchange a shared secret to use a symmetric key algorithm, this would have the tradeoff of requiring each node to store a separate shared key for each neighbor it authenticates with but would decrease the size of certain control messages by a little less than 75%. A more promising extension would be to use a more sophisticated collision detection and/or object tracking algorithm to decrease false positives and build on the sybil attack prevention. Some additional malicious node detection systems, focusing on black hole and DoS detection could be included in different configurations to simulate a comparison of them that includes this reporting and authentication system. Additionally, The LIDAR functionality could be built out more thoroughly within ns3 to more accurately simulate what could be detected via LIDAR in a real-world scenario.

References

[1] "IEEE Standard for Information Technology--Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks--Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Redline," *IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016) - Redline*, pp. 1–7524, Feb. 2021.

[2] National Center for Statistics and Analysis, "Summary of motor vehicle traffic crashes: 2022 data," *National Highway Traffic Safety Administration*, Sep. 2024.

[3] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, 1998.

[4] S. Kumar Kaushik, K. Chahal, S. Singh, and S. Dhariwal, "Performance evaluation of mobile ad hoc networks with reactive and proactive routing protocols and mobility models," *Int. J. Comput. Appl.*, vol. 54, no. 17, pp. 28–35, 2012.

[5] A. Abuarqoub, A. Alzu'bi, M. Hammoudeh, A. Ahmad, and B. Al-Shargabi, "A survey on vehicular ad hoc networks security attacks and countermeasures," in *Proceedings of the 6th International Conference on Future Networks & Distributed Systems*, 2022.

[6] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[7] S. Chang, Y. Qi, H. Zhu, J. Zhao, and X. Shen, "Footprint: Detecting Sybil attacks in urban vehicular networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 6, pp. 1103–1114, 2012.

[8] V. Gaikwad and L. Ragha, "Mitigation of attack on authenticating identities in ad-hoc network," in *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, 2017.

[9] I. V. Ravi Kumar, G. Rajitha, and B. Nancharaiah, "A secure handshaking AODV routing protocol (SHS-AODV) with reinforcement authentication in MANET," in *Advances in Sustainability Science and Technology*, Singapore: Springer Nature Singapore, 2022, pp. 99–111.

[10] M. Khalil and M. A. Azer, "Sybil attack prevention through identity symmetric scheme in vehicular ad-hoc networks," in *2018 Wireless Days (WD)*, 2018.

[11] H. Liu, Y. Chen, H. Tian, T. Wang, and Y. Cai, "A novel secure message delivery and authentication method for vehicular ad hoc networks," in *2016 First IEEE International Conference on Computer Communication and the Internet (ICCCI)*, 2016.

[12] Mohanadevi and S. Selvakumar, "An effective encryption mechanism for security issues in adhoc networks," *ECS Trans.*, vol. 107, no. 1, pp. 17465–17474, 2022.

[13] S. Han, D. Ban, W. Park, and M. Gerla, “Localization of Sybil nodes with electro-acoustic positioning in VANETs,” in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, 2017.

[14] M. T. Garip, P. H. Kim, P. Reiher, and M. Gerla, “INTERLOC: An interference-aware RSSI-based localization and sybil attack detection mechanism for vehicular ad hoc networks,” in *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, 2017.

[15] Y. Yao, B. Xiao, G. Yang, Y. Hu, L. Wang, and X. Zhou, “Power control identification: A novel Sybil attack detection scheme in VANETs using RSSI,” *IEEE J. Sel. Areas Commun.*, vol. 37, no. 11, pp. 2588–2602, 2019.

[16] C. H. O. O. Quevedo, A. M. B. C. Quevedo, G. A. Campos, R. L. Gomes, J. Celestino, and A. Serhrouchni, “An intelligent mechanism for Sybil attacks detection in VANETs,” in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020.

[17] M. Ayaida, N. Messai, S. Najeh, and K. Boris Ndjore, “A macroscopic traffic model-based approach for Sybil attack detection in VANETs,” *Ad Hoc Netw.*, vol. 90, no. 101845, p. 101845, 2019.

[18] D. Lohar, S. K. Panda, S. Padhi, and S. K. Nayak, “An efficient Sybil attack detection approach for vehicular ad-hoc networks,” in *2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC)*, 2022.

[19] M. Y. Dangore and S. S. Sambare, “Detecting and overcoming blackhole attack in AODV protocol,” in *2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies*, 2013.

[20] A. Yasin and M. Abu Zant, “Detecting and isolating black-hole attacks in MANET using timer based baited technique,” *Wirel. Commun. Mob. Comput.*, vol. 2018, no. 1, 2018.

[21] S. Yadav, M. C. Trivedi, V. K. Singh, and M. L. Kolhe, “Securing AODV routing protocol against black hole attack in MANET using outlier detection scheme,” in *2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON)*, 2017.

[22] J. Tobin, C. Thorpe, and L. Murphy, “An approach to mitigate black hole attacks on vehicular wireless networks,” in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, 2017.

[23] A. Aggarwal, S. Gandhi, N. Chaubey, and K. A. Jani, “Trust based secure on demand routing protocol (TSDRP) for MANETs,” in *2014 Fourth International Conference on Advanced Computing & Communication Technologies*, 2014.

[24] P. Pandey and R. Singh, “Decision factor based modified AODV for improvement of routing performance in MANET,” in *Communications in Computer and Information Science*, Cham: Springer International Publishing, 2022, pp. 63–72.

[25] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” Jul. 2003.

[26] N. Media, “NHTSA finalizes key safety rule to reduce crashes and save lives,” *NHTSA*. [Online]. Available: https://www.nhtsa.gov/press-releases/nhtsa-fmvss-127-automatic-emergency-braking-reduce-crashes. [Accessed: 24-Sep-2024].

[27] “Aeva advances safety and performance for automated driving with industry-first ultra long range detection for dark objects – Aeva,” *Aeva.com*. [Online]. Available: https://www.aeva.com/press/aeva-advances-safety-and-performance-for-automated-

driving-with-industry-first-ultra-long-range-detection-for-dark-objects/. [Accessed: 24-Sep-2024].

[28] B. Spencer, “Argo AI Lidar to help realise ride-hail AVs,” *ITS International*. [Online]. Available: https://www.itsinternational.com/its4/its5/news/argo-ai-lidar-help-realise-ride-hail-avs. [Accessed: 24-Sep-2024].

[29] S. Rangwala, “The LiDAR range wars - mine is longer than yours,” *Forbes*, 27-May-2021. [Online]. Available: https://www.forbes.com/sites/sabbirrangwala/2021/05/27/the-lidar-range-wars-mine-is-longer-than-yours/. [Accessed: 24-Sep-2024].

[30] *Autonews.com*. [Online]. Available: https://www.autonews.com/mobility-report/self-driving-startup-aeva-says-its-sensor-can-detect-vehicles-over-500m-away?utm_source=daily&utm_medium=email&utm_campaign=20210506&utm_content=article9-headline. [Accessed: 24-Sep-2024].