applied sciences

MDPI

Article

Analysing the Performance of a Trust-Based AODV in the Presence of a Flooding Attack

Ali Alzahrani and Nigel Thomas *

Newcastle University, Newcastle-upon-Tyne NE1 7RU, UK; a.a.a.alzahrani2@newcastle.ac.uk
* Correspondence: nigel.thomas@newcastle.ac.uk

Abstract: Mobile ad hoc networks (MANETs) are wireless multi-hop networks that do not rely on any fixed infrastructure, unlike traditional networks. Nodes in MANETs are formed dynamically and are free to move in any direction at variable speeds. The special characteristics of MANETs make them vulnerable to flooding attacks, which can have a negative impact on their performance. Moreover, due to their nature, employing solutions designed for traditional networks is not feasible. One potential solution to enhance the performance of MANETs in the face of network attacks is to implement trust management. This paper evaluates the performance of Ad hoc On-Demand Distance Vector (AODV) Routing in the presence of a flooding attack. We propose a direct trust management scheme to detect and isolate malicious nodes and implement this scheme on AODV. We name the modified protocol Trusted AODV (TAODV) and, finally, compare the performance of AODV and TAODV when both are under a flooding attack to measure the improvement achieved by our suggested scheme.

Keywords: MANET; AODV; trust management; performance

Citation: Alzahrani, A.; Thomas, N. Analysing the Performance of a Trust-Based AODV in the Presence of a Flooding Attack. *Appl. Sci.* **2024**, *14*, 2874. https://doi.org/10.3390/app14072874

Academic Editor: Andrea Prati

Received: 4 February 2024
Revised: 21 March 2024
Accepted: 25 March 2024
Published: 29 March 2024

1. Introduction

A mobile ad hoc network (MANET) is a decentralised, wireless, self-configured network that is formed dynamically by multiple mobile nodes without the use of any centralised administration or existing infrastructure [1–3]. MANETs do not rely on a pre-existing infrastructure, such as access points, routers, or servers. Instead of using routers to forward data through the network, each node participates in routing, forwarding the data from the source node to the destination node. A route in MANET is created when it is needed, in other words when a node wants to send data to another node in the network [4]. There are many routing protocols for MANETs, each of which has its own mechanism for creating the routes through the network [5].

The absence of a central administration leads to vulnerabilities that can be exploited to harm the network performance [6]. For example, uncooperative (malicious) nodes can join the network at any time, as can cooperative nodes, without any authentication or validation and launch a flooding attack. In the flooding attack, A malicious node sends a huge number of fake routing requests asking for routes to non-existent destinations. As a result, the network is flooded by fake requests that can consume its resources and lower its performance level.

A potential solution that could help assure the performance of MANETs in the face of a network attack is to use the idea of trust management. The principle of trust management in MANETs is that each node monitors the behaviour of its neighbouring nodes and tries to detect any malicious activities. Once a node identifies that a neighbouring node is malicious, it will categorise it as untrustworthy and avoid dealing with it in the future.

The notion of a security tradeoff with other aspects of a system, e.g., performance or reliability, has been widely explored, e.g., [7]. MANETs can be used to support many kinds of application, for example Internet of Things (IoT), sensor and vehicular networks and

recently cyber-physical systems [8]. Each new application area brings different demands and potentially different vulnerabilities. In this work we consider only the underlying MANET routing and focus on one kind of attack, however the approach we use could be applied to other protocols and other forms of malicious behaviour. Specifically, the aim of this paper is to show the impact of flooding on the AODV routing protocol and to explore whether a simple direct trust mechanism can improve performance.

2. Related Work

Since the AODV routing protocol was introduced, many schemes and solutions have been proposed to improve its performance in the presence of network attacks. Some of these schemes use techniques that require high computational power, such as cryptography. Although many of these solutions improve the performance of MANETs by detecting and isolating malicious nodes, the cooperative nodes suffer from the high computational load required. This may not fit the special characteristics of MANETs. Other solutions rely on the assumption that the first RREP packet received and the RREP packet with the highest sequence number are sent by a malicious node. This assumption is true in the case of black-hole and grey-hole attacks, but not in the case of selfish attack. Various solutions have been suggested to improve the performance of MANETs black-hole, grey-hole and selfish attacks and these are reviewed in the following subsections.

Several solutions and algorithms have been suggested to deal with flooding attacks. The flooding attack model is completely different from the black-hole, grey-hole and selfish attacks. In flooding, the attackers use the RREQ packet to launch the attack, unlike black-hole, grey-hole and selfish attackers, who use the RREP packet. They flood the network with RREQ packets to non-existing destinations to keep the nodes processing these packets for the longest possible time. The solutions previously outlined concentrated on the RREP packets to detect black-hole, grey-hole and selfish attacks. In contrast, flooding attack solutions need to focus on and analyse RREQ packets.

Jhaveri et al. [9] suggested an observation mechanism to calculate a trust value for neighbouring nodes. This mechanism aims to detect malicious nodes and isolate them. The trust value is calculated dynamically after every time interval using three parameters: (i) the RREP packet sequence number; (ii) the routing table sequence number; (iii) the number of replies received over the time interval. This algorithm is designed to detect and isolate malicious nodes in the route discovery phase. When sending the RREQ packet in the discovery phase, the source node broadcasts a list of malicious nodes. When the intermediate nodes receive this list, they update their routing tables. This trust-based mechanism was found to improve the packet delivery ratio (PDR) under black-hole and grey-hole attacks.

Sharma and Chauhan [10] implemented a trust-based distributed algorithm in the AODV protocol to detect and separate grey-hole attackers from other nodes. Each node in the network monitors the behaviour of its neighbours; when it senses the existence of a grey-hole node, it stores it in a grey-hole attacker list. To confirm the decision, it sends RREQ packets to the other neighbouring nodes to see if the suspected node will still behave maliciously. Based on the responses received from the neighbours, it can decide whether the suspected node is a grey-hole attacker or not. When it is confirmed that the suspected node is malicious, the source will avoid any future communication with it. This algorithm shows a better throughput in the presence of a grey-hole attack.

Vasantha et al. [11] proposed a trust-based mechanism to improve MANET performance under black-hole and grey-hole attacks. The mechanism detects malicious nodes by filtering their RREP packets at the source and intermediate nodes. The mechanism ignores the RREP packets with very high sequence numbers and also ignores the first RREP packet received. The malicious nodes will be added to a blacklist to prevent any communication with them in the future. The mechanism consists of two stages: prevention and detection. Both stages run while transmitting data between the source and destination nodes.

Jhaveri [12] proposed an algorithm to detect and separate multiple black-hole and grey-hole attackers during the routing discovery phase. They suggested modifying the functionality of receiving RREP packets. Each node in a MANET monitors the behaviour of its neighbouring nodes by analysing the RREP packet received. If an intermediate node detects a malicious node after receiving an RREP packet, it labels the RREP packet *do not consider* and marks the node as a *malicious node* in the routing table. The intermediate node will reverse the path, sending the RREP packet back towards the source node, which will update the routing tables of all nodes with the malicious node entry. The new route towards the destination node is selected by unmarked RREP packets.

Bindra et al. [13] proposed an algorithm to detect black-hole and grey-hole attacks by adding extended data routing information to the routing table in the AODV protocol. The routing table exists in every node in the network and fields are added to detect malicious nodes and save a history of their previous malicious activities. Creating a historical record for each node's malicious activities is a way of addressing grey behaviour. This mechanism is built on the AODV protocol.

Marti et al. [14] proposed an idea called "watchdog", which is a direct trust management mechanism and a path rater. Each node in the network listens to the transmissions of the next node along the route to detect malicious behaviours. The path rater holds the trust values of the nodes, which range from 0 to 0.8, with neutral taking the value of 0.5. These values given to the nodes are updated continuously by 0.1 each 200 ms. The source node should be able to detect selfish nodes and avoid sending packets to them. This scheme has shown better performance in the presence of selfish attacks. However, it also increases the memory overhead as the watchdog mechanism needs to maintain the information collected from the packets.

Buchegger and Le Boudec [15] proposed an amended routing protocol called Cooperation Of Nodes - Fairness In Dynamic Ad hoc NeTworks (CONFIDANT). This improved protocol contains a trust management and reputation system. While monitoring the behaviour of the neighbouring node, the trust manager evaluates the node's activities and sends alarms to neighbouring nodes telling them about a malicious node. The malicious node is then isolated by all nodes that receive the alarm. This mechanism combines direct monitoring and a reputation system. The mechanism has shown better performance in the presence of black-hole attacks.

Michiarde and Molva [16] suggested a collaborative reputation scheme based on collaborative monitoring. This schema has a more complicated reputation system in which it is not only one node monitoring the behaviour of the next node, but many nodes monitoring the behaviour of the same nodes. The decision is made using the data collected by all nodes involved in the network. The decision here should be more accurate than using other methods because it is not made by an individual node. Each node has a reputation table and a watchdog mechanism. Each node monitors the behaviour of its neighbours, assigns trust values to them and saves those values in its trust table. The nodes share their trust values with each other. This mechanism has shown better performance in the presence of selfish attacks.

Pissinou et al. [17] proposed a secure MANET routing protocol based upon a direct trust mechanism establishing the trust level of the nodes along the route. This protocol stores trust values in the RREQ packet. When the source node wishes to initialise a route to the destination, it sends the RREQ packet to its neighbouring node. The intermediate node updates the trust values in the RREQ packet and forwards it to the next node. When the RREQ packet reaches the destination node, the destination node replies to the source by sending an RREP packet containing the level of trust of the route. Then the source can choose the route with the highest trust level rather than choosing the shortest route.

Balakrishnan et al. [18] proposed a reputation-based trust model called secure MANET routing with trust intrigue. The evidence of trustworthiness in this model is collected efficiently from direct interactions with the neighbouring nodes and through recommendations from other nodes. This mechanism contains two components: detection and reaction. In

the detection phase, evidence of the malicious behaviour of neighbouring nodes is collected and in the reaction phase, the source node uses this evidence to accept or reject a newly discovered route. Also, in the reaction stage, the source node can predict the future behaviour of a cretin node by using the evidence collected.

Wang et al. [19] proposed an improved trust-based version of the DSR routing protocol. Each node in the MANET has a trust table that stores the trust values of the neighbouring nodes. The amended DSR selects the routing path with the highest trust values and least delay, unlike the standard DSR, which chooses the routing path that has the lowest hop count. The trust values are calculated and updated individually by the nodes. A trust value ranges between −1 and +1, where −1 means distrust and +1 absolute trust. Implementing this mechanism in the DSR protocol showed a better PDR under selfish attacks.

Li et al. [20] proposed a trust algorithm based on packet forwarding ratio (PFR). PFR is calculated using the ratio of the number of packets forwarded to the number of packets received. Trust values are assigned based on the PFR results. The trust value of any node can increase or decrease based on its behaviour in forwarding packets. When the PFR increases, the trust value increases, and when it decreases, the trust value decreases. The trust value in this model is between 0 and 1, where 0 means distrust and 1 means absolute trust. A trust value between 0 and 0.5 indicates the node is malicious, 0.5 to 0.75 indicates the node is suspect, 0.75 to 0.9 suggests the node might be not entirely trustworthy and 0.9 to 1 means the node is trustworthy. Nodes with low trust values are isolated and not allowed to participate in new routes.

Yi et al. [21] suggested a scheme called Flooding Attack Prevention (FAP), which aims to monitor the RREQ packets received from neighbouring nodes. The FAP algorithm assumes that there is less of a risk of a node that sends fewer RREQ packets being a malicious node running a flooding attack. Thus, if a node sends fewer RREQ packets, it is assigned a higher priority for selection. When a node broadcasts a high rate of RREQ packets, the neighbouring nodes will observe this and reduce the responding priority for this node.

Yi et al. [22] added a fixed threshold to the FAP. The modified algorithm assumes that if the RREQ packets received by a neighbouring node exceeds the threshold, it is launching a flooding attack. Any RREQ packets from this node will be ignored in the future. The disadvantage of this modification is that if the attacker guesses the fixed threshold, it is possible simply to make the number of RREQ packets less than the threshold. This way a flooding attack will not be detected.

Song et al. [23] proposed a filtering mechanism that has two thresholds: the rate limit and the blacklist limit. The source node only accepts RREQ packets from nodes that have not reached the rate limit. After receiving the RREQ packet, the source node compares the sequence number of the packet with the one stored in the blacklist. If it is greater than the sequence numbers in the blacklist, the RREQ packet is discarded and the node will be added to the blacklist. The disadvantage of this mechanism is that the attacker can run the attack by reducing the flooding rate and sequence number if they discover the values of the thresholds.

Venkataraman et al. [24] proposed a trust management solution to mitigate the impact of flooding attacks on MANETs. The algorithm of the solution classifies neighbours based on their given trust values under three categories: friend, stranger and acquaintance. A friend node is trusted, a stranger is not trusted and an acquaintance is neutral. There are three transmission rate threshold values and each category has its own threshold value. When the source node receives an RREQ packet from a neighbouring node, it first checks its type and compares its transmission rate to the threshold value of its category. If the transmission rate is below the threshold value for the category, the RREP packet is processed; otherwise, the packet will be discarded and the node will be added to the blacklist.

Bandyopadhyay et al. [25] proposed a scheme that allows a limited number of RREQ packets to be sent by any source node in the network. The limit was set at 10 RREQ packets per second. This low rate helped prevent flooding the network with RREQ packets.

The disadvantage of this approach is that the malicious node can change the value of the allowed rate parameter because it has full access to it.

Khattak and Nizamuddin [26] provided a hybrid approach for preventing black-hole and grey-hole attacks. The proposed algorithm uses a hash function and timestamp base solution. The source node selects the second shortest route to the destination instead of the first. The probability of the existence of a black-hole attacker decreases for the second shortest route. This is because black-hole and grey-hole attackers send RREP packets to the source stating that they have the shortest route to the destination. Choosing the second shortest path is thus safer in the AODV protocol. The point of monitoring the RREP packets coming from the neighbouring nodes and ignoring the one with the shortest path is a direct trust mechanism for improving MANET performance.

Velloso et al. [27] suggested a recommendation exchange protocol (REP), which enables a node to exchange trust values with its neighbouring node. In this model, the trust values are calculated based on direct observation. The decision to isolate a specific node is made using the evidence collected individually and by recommendations received from neighbouring nodes. Each node in the network assigns a trust value to its neighbours of between 0 and 1. This model consists of two phases: the learning phase and the trust phase. The learning phase is responsible for collecting data and converting it into knowledge. The trust phase is responsible for using that knowledge to detect and isolate malicious nodes.

Yu et al. [28] defined a trust management system that provides a degree of assurance for the future behaviour of nodes based on the services previously received from the nodes. Yu et al. classified the trust and reputation management system into two major categories, namely individual-level trust and system-level trust. The individual-level trust mechanism allows the source node to initialise communication with the subject node, aggregate declarations from other nodes about prospective communications, evaluate the trustworthiness of potential interaction based on the past recorded data and make a trustworthiness decision on whether to interact with the subject node. The system-level trust mechanism concentrates on applying punishment based on the node's trustworthiness and reputation to improve the utility for nodes that are highly trustworthy.

Lee et al. [29] suggested adding two new packets to the AODV protocol: a route confirmation reply (CREP) packet and a route confirmation request (CREQ) packet. When an intermediate node receives an RREQ packet from the source node, it checks its routing table and sends an RREP packet back to the source and sends a CREQ packet to its next hop node towards the destination node. When the next-hop node receives the CREQ packet, it sends a CREP packet back to the source node if it has a fresh route to the destination node. When the source node receives the CREP and RREP packets, it validates the path to the destination by comparing the paths in the two packets. If both agree, the path is valid.

Al-Shurman et al. [30] proposed a solution requiring additional computation which results in an increased overhead. The source node stores the sequence numbers of the last packets sent and the last packets received in two separate tables. When the source node receives an RREP packet, it compares the sequence number in this packet with the stored numbers. If there is a match, the source node starts data transmission; otherwise, it will classify the RREP packet as malicious. The source node will send an alarm message to the other nodes to block the malicious node.

Raj and Swadas [31] designed a solution that adds an additional task for the nodes: checking the RREP packet before accepting it. When a node receives an RREP packet, it checks if the sequence number is higher than a threshold value. If the sequence number of the RREP packet is higher than the threshold, it is classified as malicious and the node that sent it is added to a blacklist. When a node has detected a malicious node, it sends an alarm packet to inform all neighbouring nodes. This solution has been found to increase the overhead.

Mistry et al. [32] suggested a solution that entails analysing the RREP packets received. When a source node receives the first RREP packet, it does not react to it immediately

but waits for some time to receive multiple RREP packets and then saves them in a table. The source node then analyses the stored RREP packets in the table and rejects those with very high sequence numbers. The nodes that sent these packets are marked as malicious. The source node will arrange the remaining RREP packets according to the sequence numbers of their destination nodes, and the one with the highest number will be selected. This mechanism increases the end-to-end delay as the source node has to wait some time to receive multiple RREP packets.

Yang [33] proposed a cluster-based trust mechanism where data are transmitted securely using encryption. The objective is to identify nodes which produce high volumes of traffic as anomalous, but measuring not only the number of packets, but also their quality. Comparison is made against Energy-Aware AODV and Secure AODV and shows a slight improvement in performance.

3. Ad Hoc On-Demand Distance Vector (AODV)

AODV is a reactive routing protocol which specifies the route between a source and destination at the time when the route is needed. Each node in the network maintains a routing table of routes to known nodes. However, as the network topology changes, these routes become invalid. In addition, new nodes may join the network which then need to build new routing routing tables. To handle this process, AODV includes two phases: route discovery and route maintenance.

Route discovery is the process by which a valid route between a source and destination pair is found [34]. The routing discovery process in AODV operates as follows:

1. A source node that has data to send to a destination node will first check its stored routing table. If the routing table contains a valid route, then data transmission begins; otherwise, the source initiates an RREQ packet. The RREQ packet contains: source IP, destination IP, source sequence number, destination sequence number and broadcast ID number.
2. When a node receives an RREQ packet, it checks its own stored routing table. If it has a valid route to the destination then it sends an RREP packet to the source. The RREP packet contains the following information: destination IP, destination sequence number, hop count, lifetime and source IP. If the node does not hold a valid route to the destination then it broadcasts the RREQ packet within its transmission range.
3. Once the source node has received a RREP packet, it starts transferring data using that path.

Route maintenance is used when routes become broken due to nodes moving out of range or leaving the network. If this happens, then the following steps are taken to recover the broken route.

1. The upstream node sends an RRER packet to notify the source that the route is no longer valid. An RRER packet contains: source IP, unreachable destination IP and destination sequence number.
2. All intermediate nodes on the route mark it as invalid.
3. On receipt of an RERR packet, the source starts a new route discovery process.

4. Flooding Attack

A flooding attack can be defined as sending a huge amount of multiple fake RREQs to random nodes aiming to consume the network resources and leading to DoS [22,35]. This is a common type of attack in all kinds of network, including traditional networks with fixed infrastructure.

In a flooding attack, the malicious node floods the network with RREQs to nonexistent destinations to keep the nodes busy processing the fake packets. The aim of this is to consume the bandwidth, node memory, node power and computational resources and to prevent normal operations in the protocol [36]. The malicious node in this type of attack exploits the vulnerabilities in the route discovery phase, allowing an unlimited number

of RREQs and accepting any RREQ without validation. In traditional networks, a way of improving performance in the case of this type of attack is to install a firewall in the hardware, such as servers and routers [37].

In a flooding attack, the attacker selects an IP address that does not exist in the network to extend the search process in the network and consume the nodes' resources [25]. If the attacker knows the scope of IP addresses in the network, the attacker will select IP addresses from outside the range. If the attacker does not know the scope, they will select random IP addresses in the hope that they do not exist. If the IP address selected is outside the domain, no node can answer the RREQs. The AODV protocol does nothing to detect fake destination IP addresses because of the nature of MANETs, which allow nodes to join and leave the network freely at any time.

After selecting the IP addresses, the attacker generates a huge number of RREQs for the void IPs without waiting for the RREPs to arrive. When a flooding attack is launched successfully in a MANET, the whole network will be flooded with the fake RREQs sent by the attacker. Both the bandwidth and the resources of the nodes can be exhausted at the same time, which can easily lead to DoS [38]. To give a simple example of shutting down a MANET, each node's capacity for storing the routing table is extremely limited and if the node receives a huge number of RREQs over a short period of time, the routing table will be full and the node will not be able to receive any more RREQs. Hence, the node will not be able to serve the real RREQs from cooperative nodes.

The flooding attack is a type of DoS attack that exploits the routing discovery process of reactive routing protocols such as AODV [25]. AODV already has a default mechanism to prevent flooding attacks at some level. However, this mechanism is vulnerable and it can be hacked by malicious nodes. The default mechanism of AODV has the following barriers to prevent a flooding attack:

1. Limiting the flooding rate.
   AODV has a method to reduce congestion in the network by limiting the number of RREQs a node can send per second. The number of RREQs per second allowed, RREQ_RATELIMIT, is 10 by default [39].
2. Limiting the number of routing request attempts.
   AODV limits the number of attempts to find a route made by a source node, by setting a threshold RREQ_RETRIES. When a node broadcasts an RREQ, it waits to get an RREP. If the RREP does not arrive within a given number of milliseconds, NET_TRAVERSAL_TIME, the node may try again and send another RREQ until the number of retries reaches the limit of RREQ_RETRIES, which has a default value of 2 [39].
3. Limiting the time an RREQ can live in the network.
   Time-to-live, TTL, is the maximum time allowed for an RREQ to live in the network before it is discarded. Each RREQ has a TTL_START value stored in its header. This value increases by TTL_INCREMENT each time the node tries to send a new RREQ and the total will be stored in the TTL. This continues until the the TTL set in the RREQ reaches the TTL_THRESHOLD, then the RREQ will be discarded [40].

Theoretically, these methods seem to work well to prevent flooding attacks as they limit the number of RREQs sent by any node in the network and do not allow any RREQ to live in the network forever. However, the malicious node can remove these restrictions by overriding the values of RREQ_RATELIMIT, RREQ_RETRIES and TTL_THRESHOLD. The parameters of this mechanism are accessible by the source node, which has full control to change their values. This allows the malicious node to remove the limitations on the number of RREQs allowed, the flooding rate and the period of time the RREQ can live in the network.

5. Trust Management

One of the essential goals for MANETs is to provide routing security in the network and ensure confidentiality, integrity, availability and anonymity [41]. The existence and

implementation of MANETs depend on the nodes' cooperative and trusting nature. There is a common assumption in existing MANET routing protocols that each node participating in the network is trustworthy [42]. This default assumption needs to change with the development of a mechanism that can distinguish a trustworthy node from a malicious node.

Trust can broadly be defined as a measure of subjective belief that an entity will perform an action and another will perform the promised work without the need to examine whether or not the work is done [43]. This definition contains two inspiring terms: measure and performance. This is how networks work; it is all about performance and the ability to measure it.

In MANETs, trust management is a reputational mechanism such that every node in the network observes and evaluates its neighbouring nodes' activities and tries to detect and isolate any malicious node [42]. It is a method of collecting the information about an entity needed to make a trust relationship decision about it [27]. Trust management can go further and obtain experiences from other entities. Thus, the trust decision can be made using information collected directly or using other nodes' experiences and recommendations.

Trust management has become crucial to solving many performance issues in MANETS. The principle of trust is derived from the social sciences and is used in many fields, such as economics, communications, business and computing and networking. In computer networks in general and in MANETs especially, trust entails believing that a node is cooperating with other nodes to forward the data through the network as expected without any disruption [44].

Ullah et al. [44] defined that the objectives of trust management in MANETs as follows:

1. To distinguish between trustworthy and malicious nodes.
2. To allow trustworthy nodes to participate in establishing routes between sources and destinations.
3. To isolate malicious nodes and prevent them from participating in the network

Due to the unique characteristics of MANETs, a trust management scheme must be distributive and self-organised. Also, it should consume fewer CPU cycles and battery, memory and bandwidth resources.

Trust management in MANETs comprises four steps: trust initialisation, information collection, trust calculation and decision making [44]. Each step has its own challenges and difficulties in MANETs.

5.1. Trust Initialisation

Trust initialisation happens when a node starts sending packets to its neighbouring node. In this step, a node in MANET has no previous interaction with its neighbour and has no background about its behaviour. This is a risky situation and most of the direct trust management algorithms proposed consider the default trust value to be neutral. For example, if the range of the trust value is between 0 and 1, the default value for the unknown node is 0.5.

5.2. Information Collection

In direct trust management, a node directly monitors the following node's behaviour and gathers information based on this observation. Direct observation is a powerful technique that can provide authentic information about a neighbouring node's behaviour. In indirect trust management, information is collected in two ways: through direct observation and by receiving recommendations from other nodes in the network.

5.3. Trust Calculating

The information collected about the behaviour of the neighbouring node is used to calculate a trust value for that node. The trust value can be updated over time to arrive at a more accurate judgment. Calculating the trust values can be as simple as counting the packets dropped by the node in question, or it can be more complicated and use maths functions.

5.4. Decision Making

Once the trust value of a specific node has been calculated, it can be used to decide whether the node is trustworthy or not. Based on this decision, the node in question must either be isolated or allowed to participate in creating routes. There are different levels of trust in more complex models.

6. Trusted AODV (TAODV)

Trusted AODV (TAODV) is very simple, which is one of its attractive features. No new messages are generated and the computational overhead is very low. Furthermore, as we employ direct trust management, there is no need for any additional separate authority to detect malicious nodes, which would go against the distributed ad hoc nature of MANETs.

In TAODV, each node maintains a list of known nodes in the network with a trust value for each node. A trust value can be calculated using any reputational measure, depending on the objective of the trust management scheme and in practice a node could maintain an number of different trust values in order to achieve different goals, or to counter different attacks. In this paper, we are specifically concerned with the impact of flooding attacks and so the trust value used here is specifically tailored to identify nodes which are attempting to launch flooding attacks. As discussed above, AODV has measures to limit the impact of flooding attacks but these can be bypassed by a malicious node. Specifically, a malicious node may attempt to create many long routes with to flood packets across the network. Thus, the trust value in AODV is based on identifying nodes that generate multiple RREQ requests in quick succession, or RREQ packets with long time to live (to create long routes). If such behaviour is found, then the trust value for the offending node will be set to zero by other nodes. If sufficient other nodes set the trust value to zero for a given node then that node will be unable to initiate routes on which to launch the flooding attack.

In particular, the proposed scheme can validate the values for the RREQ_RETRIES and TTL parameters in the intermediate node. Thus, instead of relying on the honesty of the source node, the intermediate node checks the values and if the TTL value is greater than the TTL_THRESHOLD value or the RREQ_RETRIES value is greater than 2/s, the source node will be given a trust value of zero. Thus, the intermediate node should be able to detect the malicious behaviour of the flooding attacker and isolate it.

To implement the scheme, a table needs to be added to each node to store the following information: (i) the IP address of the source node sending the RREQ; (ii) the number of RREQs received by the intermediate node; (iii) the trust value of the node. The number of RREQs received is zero by default and increases by one each time the intermediate node receives an RREQ. The number resets each second. The scheme works as follows:

1. The intermediate node checks the value of the TTL associated with the RREQ received. If the TTL value is greater than TTL_THRESHOLD, the RREQ will be discarded and the source node will be given a trust value of zero and isolated. The intermediate node can obtain the TTL_THRESHOLD on its own and compare it to the TTL of the RREQ received.
2. Each time the intermediate node receives an RREQ, it stores the IP address of the source node in the table and increases the number of RREQs received from this node by one. If the number of packets received from the same source node exceeds two packets within one second, the node is trying to flood the network. The source node will be given a trust value of zero and be isolated.

7. Simulation and Evaluation

The performance of the scheme is evaluated using the NS3 network simulator (version 3.33) and measured using the following metrics: throughput, PDR and end-to-end delay. We compare the performance of the plain AODV with the modified version, TAODV, when both are under flooding attack. The parameters used to run this simulation are shown in Table 1.

Table 1. Parameters used to evaluate performance of the TAODV protocol under flooding attack.

Parameter	Value	Unit
Simulator	NS-3.33	-
Packet size	512	byte
Simulation time	100	s
Simulation area	1200 × 1200	m
Number of nodes	50, 80, . . ., 200	-
Node speed	1	m/s
MAC protocol	802.11b	-
Transmission range	250	m
Total of simulation runs	10	-
Malicious nodes	5	-
Attack type	flooding	-
Flooding rate	50	packet/s
Routing protocol	AODV, TAODV	-

7.1. Performance Based on Number of RREQs Varying the Number of Nodes

The number of RREQs sent by nodes in the network can be used as a metric to measure the success of the scheme. The scheme should discard or at least reduce the number of fake RREQs by blocking the malicious nodes. That will lead to a reduction in the total number of RREQs living in the network.

Figure 1 shows that the number of RREQs increases with an increase in the number of nodes in both AODV and TAODV. This is expected because having more nodes means more RREQs will be generated. The addition of the mechanism reduces the number of RREQs in TAODV by blocking the malicious nodes and stopping them from flooding the network with more fake RREQs. For example, if the number of nodes is 100, the number of RREQs is 51,000 in the case of TAODV and 57,000 in the case of AODV. Thus, the scheme was able to prevent 6000 fake RREQs in 100 s.

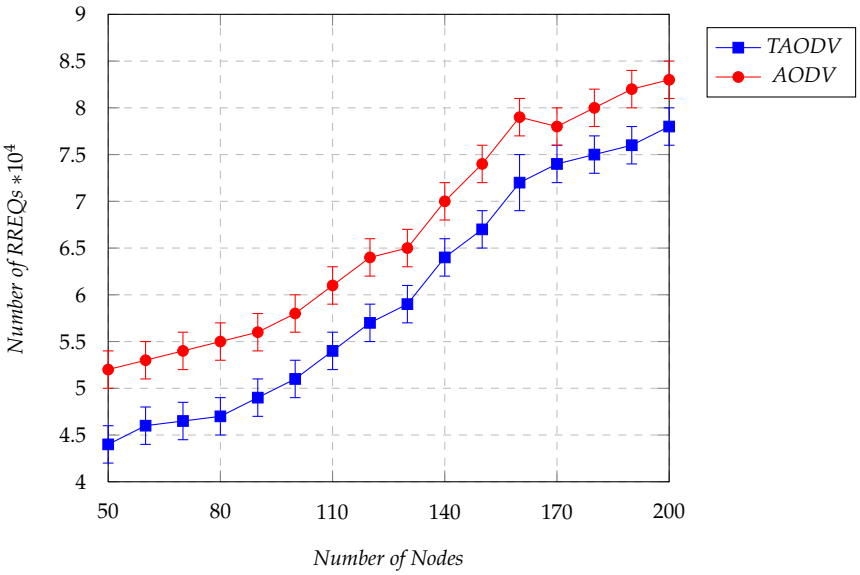

Figure 1. Number of RREQs in TAODV vs. AODV (95% CI).

7.2. Performance Based on Number of RREQs Varying the Number of Malicious Nodes

This simulation was run varying the number of malicious nodes from 5 to 25 with a constant number of 200 nodes and a flooding rate of 50 packets per second.

Figure 2 shows that increasing the number of malicious from 5 to 25 increases the number of RREQs from 85,000 to 380,000 in the AODV case, whereas in TAODV, the highest

number of RREQs is fewer than 190,000 when the number of malicious nodes reaches the highest value of 25.

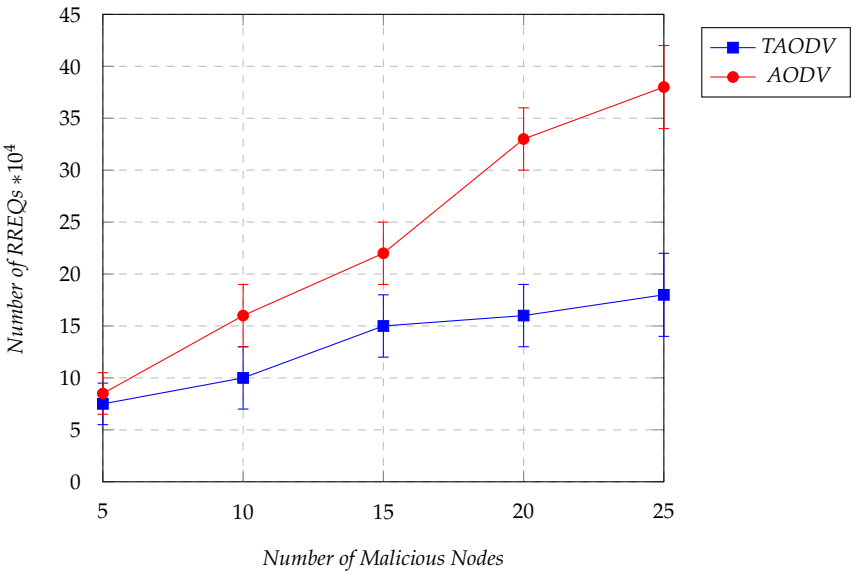

Figure 2. Number of RREQs in TAODV vs. AODV under flooding attack (95% CI).

7.3. Performance Based on Number of RREQs Varying the Flooding Rate

The simulation was run varying the flooding rate from 50 to 100 packets per second with a constant number of 200 nodes and 5 malicious nodes. The flooding rate is the number of fake packets sent by a malicious node per second. Figure 3 shows that in AODV, increasing the flooding rate leads to a rapid increase in the number of RREQs, whereas the number of RREQs in TAODV seems more stable. The reason for this is that any node reaching the limit of permitted RREQs will be isolated regardless of the flooding rate.

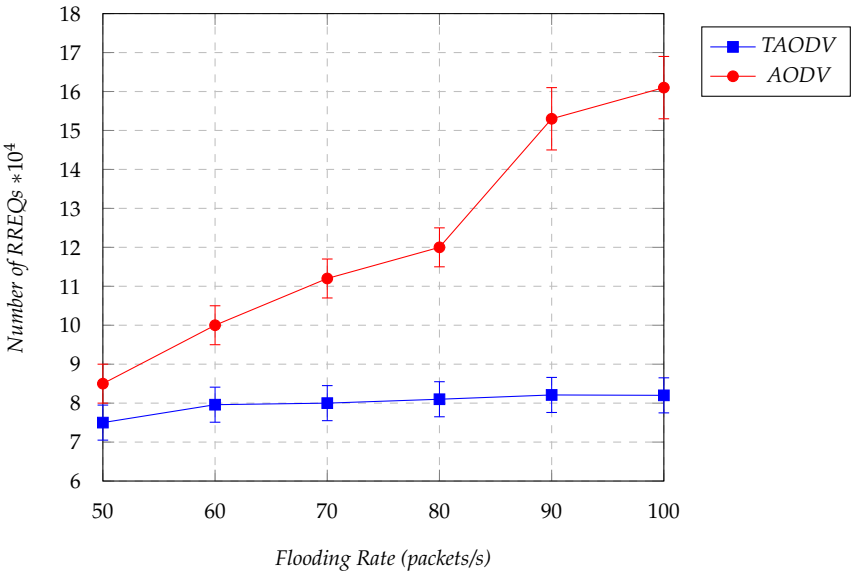

Figure 3. Number of RREQs in TAODV vs. AODV under flooding attack (95% CI).

7.4. Throughput Performance Varying the Number of Nodes

Figure 4 shows that during the simulation, the throughput of TAODV was better than AODV operating under flooding attack. The throughput improved as a result of isolating the five malicious nodes and preventing them from flooding the network with fake RREQs.

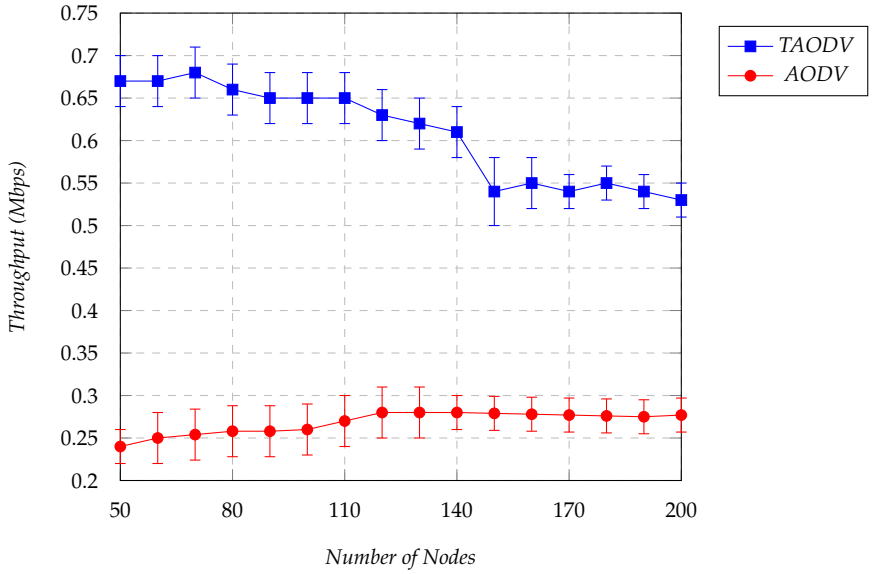

Figure 4. Throughput of TAODV vs. AODV (95% CI).

7.5. Throughput Performance Varying the Number of Malicious Nodes

The simulation was run varying the number of malicious nodes from 5 to 50 with a constant number of 200 nodes and a flooding rate of 50 packets per second.

Figure 5 shows that increasing the number of malicious nodes from 5 to 25 sharply reduces the throughput of AODV from 0.27 Mbps to around 0.054 Mbps. In contrast, there was a slight decrease in the TAODV case.

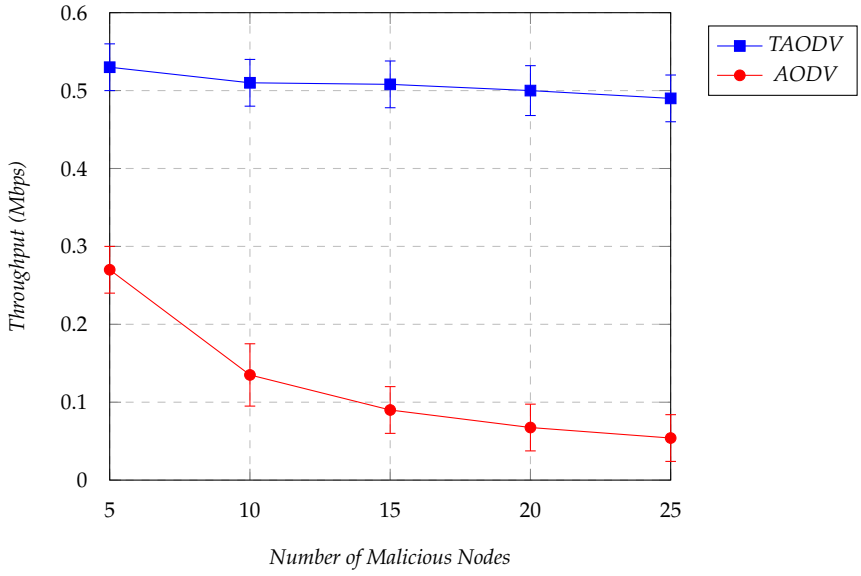

Figure 5. Throughput of TAODV vs. AODV under flooding attack (95% CI).

7.6. Throughput Performance Varying the Flooding Rate

The simulation was run varying the flooding rate from 50 to 100 packets per second with a constant number of 200 nodes and 5 malicious nodes.

Figure 6 shows that increasing the flooding rate from 50 to 100 packets per second reduces the throughput of AODV from 0.27 Mbps to less than 0.15 Mbps. In TAODV, the throughput looks more stable, with a slight decrease.

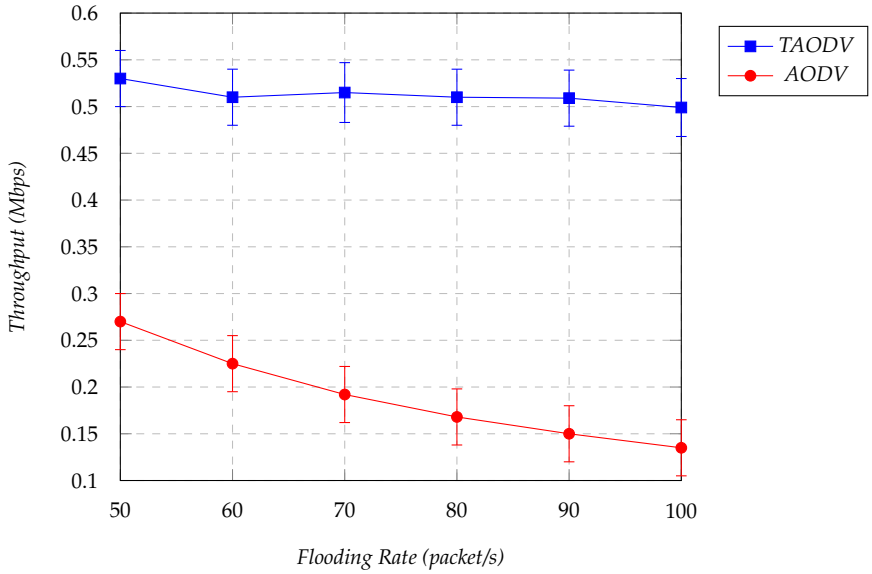

Figure 6. Throughput of TAODV vs. AODV under flooding attack (95% CI).

7.7. The Overhead

End-to-end delay is one of the QoS metrics that can be used to measure the overhead. QoS should be always measured when adding any kind of mechanism to a routing protocol. It is important to determine the trade-off between the cost and the improvement achieved.

Figure 7 shows that the more work added to the AODV protocol, the greater the effect on the end-to-end delay metric. End-to-end delay is a very sensitive metric and is likely to be affected by adding any load to the protocol. The simulation shows that the end-to-end delay increased by around 0.2 s in TAODV, which seems slight compared to the improvement in the throughput metric and the reduction in the number of fake RREQs.

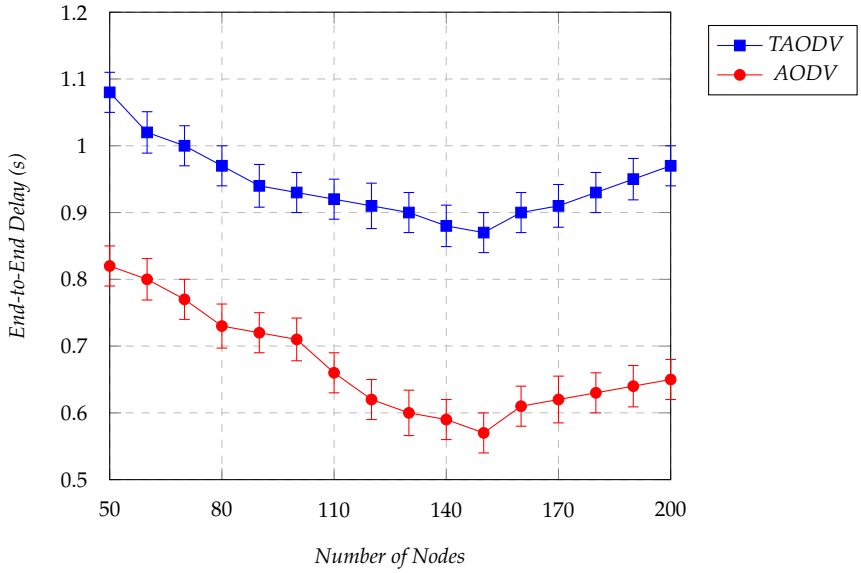

Figure 7. End-to-end delay of TAODV vs. AODV under flooding attack (95% CI).

8. Conclusions

The aim of this paper is to show the impact of flooding on the AODV routing protocol (which is already designed to deter flooding) and to demonstrate how a very simple direct trust mechanism (which requires no new operations in the protocol and has a low overhead) can significantly improve performance.

In this paper we have explained what MANETs are and how the routing protocols AODV works. It has described the vulnerabilities that AODV suffers from, which can be exploited by malicious nodes to run flooding attacks, leading to negative effects on performance. The paper then introduced trust management as a promising principle that could solve many issues in MANETs. Then proposed a direct trust management scheme to detect and isolate the flooding attack, which improved the performance of the network in the presence of the attack. The proposed scheme shows a considerable improvement in the performance of MANETs, but with some overhead. The increase in overhead was expected as adding any tasks to the routing protocol will likely consume more resources such as the memory for example.

There are clearly other means to counter flooding attacks and other forms of attack against MANET routing, as discussed in Section 2. The advantage of the TAODV scheme proposed here is its simplicity and, hence, ease of implementation and low overhead. Despite this simplicity, TAODV is shown to be effective in countering flooding attacks. The authors are not aware of any works which show a comparison between different trust-based approaches or other mechanisms for defending against attacks. An empirical comparison of the performance of different approaches to defending against flooding attacks in AODV-based MANETs would be a substantial but worthwhile task, but one which lies beyond the scope of this current paper.

Direct trust management has been used to identify malicious behaviour in many other different contexts in order to defend against different forms of attack. The mechanism used in this paper is only intended to identify a flooding attack, but other approaches to calculate a trust value could easily be added to counter other attacks against AODV. Furthermore, the approach used here could be adapted for use in other network routing protocols and other forms of trust management, indirect or global, could be used.

Author Contributions: Conceptualization, N.T. and A.A.; Methodology, N.T. and A.A.; Formal analysis, N.T. and A.A.; Investigation, N.T. and A.A.; Writing—original draft, A.A.; Writing—review & editing, N.T.; Project administration, N.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sarkar, S.K.; Basavaraju, T.G.; Puttamadappa, C. *Ad Hoc Mobile Wireless Networks Principles, Protocols, and Applications*, 2nd ed.; CRC Press: New York, NY, USA, 2016.
2. Wu, S.H.; Sheu, J.P.; King, C.T. Unilateral wakeup for mobile ad hoc networks with group mobility. *IEEE Trans. Mob. Comput.* **2012**, *12*, 2507–2517. [CrossRef]
3. Younes, O.; Thomas, N. Modelling and performance analysis of multi-hop ad hoc networks. *Simul. Model. Pract. Theory* **2013**, *38*, 69–97. [CrossRef]
4. Roy, R.R. *Handbook of Mobile Ad Hoc Networks for Mobility Models*; Springer: New York, NY, USA, 2011.
5. Chen, Z.; Zhou, W.; Wu, S.; Cheng, L. An adaptive on-demand multi-path routing protocol with QoS support for high-speed MANET. *IEEE Access* **2020**, *8*, 44760–44773. [CrossRef]
6. Wang, H.; Wang, Y.; Han, J. A security architecture for tactical mobile ad hoc networks. In Proceedings of the 2nd International Workshop on Knowledge Discovery and Data Mining, Moscow, Russia, 23–25 January 2009; pp. 312–315.
7. Abdelshafy, M.A.; King, P.J.B. AODV and SAODV under attack: Performance comparison. In *Ad-Hoc, Mobile, and Wireless Networks, Proceedings of the 13th International Conference, ADHOC-NOW 2014, Benidorm, Spain, 22–27 June 2014*; Springer: Cham, Switzerland, 2014.
8. Ju, Y.; Yang, M.; Chakraborty, C.; Liu, L.; Pei, Q.; Xiao, M.; Yu, K. Reliability–Security Tradeoff Analysis in mmWave Ad Hoc–based CPS. *ACM Trans. Sens. Netw.* **2024**, *20*, 1–23. [CrossRef]
9. Jhaveri, R.H.; Patel, S.J.; Jinwala, D.C. A novel approach for grayhole and blackhole attacks in mobile ad hoc networks. In Proceedings of the 2nd International Conference on Advanced Computing and Communication Technologies, Rohtak, India, 7–8 January 2012; pp. 556–560.

10. Sharma, B. A distributed cooperative approach to detect gray hole attack in MANETs. In Proceedings of the Third International Symposium on Women in Computing and Informatics, Kochi, India, 10–13 August 2015; pp. 560–563.
11. Vasantha S.V.; Damodaram, A. Bulwark AODV against black hole and gray hole attacks in MANET. In Proceedings of the IEEE International Conference on Computational Intelligence and Computing Research, Madurai, India, 10–12 December 2016.
12. Jhaveri, R.H. MR-AODV: A solution to mitigate blackhole and grayhole attacks in AODV based MANETs. In Proceedings of the International Conference on Advanced Computing and Communication Technologies, Rohtak, India, 6–7 April 2013; pp. 254–260.
13. Bindra, G.S.; Kapoor, A.; Narang, A.; Agrawal, A. Detection and removal of co-operative blackhole and grayhole attacks in MANETs. In Proceedings of the International Conference on System Engineering and Technology, Bandung, Indonesia, 11–12 September 2012.
14. Marti, S.; Giuli, T.; Lai, K.; Baker, M. Mitigating routing misbehavior in mobile ad hoc networks. In Proceedings of the 6th International Conference on Mobile Computing and Networking, Boston, MA, USA, 6–11 August 2000; pp. 255–265.
15. Buchegger S.; Le Boudec, J.-Y. Performance analysis of the confidant protocol (cooperation of nodes: Fairness in dynamic ad-hoc networks). In Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing, Lausanne, Switzerland, 9–11 June 2002; pp. 226–236.
16. Michiardi, R.; Molva, P. A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In Proceedings of the IFIP TC6/TCII Sixth Joint Working Conference on Communication and Multimedia Security, Portoroz, Slovenia, 26–27 September 2002; pp. 107–121.
17. Pissinou, T.M.K.; Ghosh, N. Collaborative trust-based secure routing in multi hop ad hoc networks. In *Networking 2004: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications, Proceedings of the Third International IFIP-TC6 Networking Conference, Athens, Greece, 9–14 May 2004*; Springer: Berlin/Heidelberg, Germany, 2004; Proceedings 3, Volume 3042, pp. 1446–145l.
18. Balakrishnan, V.; Varadharajan, V.; Lucs, P.; Tupakula, U.K. Trust enhanced secure mobile ad-hoc network routing. In Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops/Symposia, Niagara Falls, ON, Canada, 21–23 May 2007; Volume 1, pp. 27–33.
19. Wang, C.; Yang, X.; Gao, Y. A routing protocol based on trust for MANETs. In Proceedings of the 6th Annual International Conference on Grid and Cooperative Computing, Beijing, China, 30 November–3 December 2005; Volume 3795, pp. 959–964.
20. Li, X.; Jia, Z.; Zhang, P.; Wang, H. A trust-based multipath routing framework for mobile ad hoc networks. In Proceedings of the 7th International Conference on Fuzzy Systems and Knowledge Discovery, Yantai, China, 10–12 August 2010; Volume 2, pp. 773–777.
21. Yi, P.; Dai, Z.; Zhong, Y.; Zhang, S. Resisting flooding attacks in ad hoc networks. In Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC), Las Vegas, NV, USA, 4–6 April 2005; Volume 2, pp. 657–662.
22. Yi, P.; Hou, Y.; Zhong, Y.; Zhang, S.; Dai, Z. Flooding attack and defence in ad hoc networks. *J. Syst. Eng. Electron.* **2006**, *17*, 410–416.
23. Song, J.-H.; Hong, F.; Zhang, Y. Effective filtering scheme against rreq flooding attack in mobile ad hoc networks. In Proceedings of the 7th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'06), Melbourne, VIC, Australia, 3–6 December 2006; Volume 3326.
24. Venkataraman, R.; Pushpalatha, M.; Khemka, R.; Rao, T.R. Prevention of flooding attacks in mobile ad hoc networks. In Proceedings of the International Conference on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, 23–24 January 2009; pp. 525–529.
25. Bandyopadhyay, A.; Vuppala, S.; Choudhury, P. A simulation analysis of flooding attack in MANET using NS-3. In Proceedings of the 2nd International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace and Electronic Systems Technology, Chennai, India, 28 February–3 March 2011.
26. Khattak, H.; Nizamuddin. A hybrid approach for preventing black and gray hole attacks in MANET. In Proceedings of the 8th International Conference on Digital Information Management (ICDIM), Islamabad, Pakistan, 10–12 September 2013; pp. 55–57.
27. Velloso, P.B.; Laufer, R.P.; Cunha, D.D.O.; Duarte, O.C.M.; Pujolle, G. Trust management in mobile ad hoc networks using a scalable maturity-based model. *IEEE Trans. Netw. Serv. Manag.* **2010**, *7*, 172–185. [CrossRef]
28. Yu, H.; Shen, Z.; Miao, C.; Leung, C.; Niyato, D. A survey of trust and reputation management systems in wireless communications. *Proc. IEEE* **2010**, *98*, 1755–1772. [CrossRef]
29. Lee, S.; Han, B.; Shin, M. Robust routing in wireless ad hoc networks. In Proceedings of the International Conference on Parallel Processing Workshop, Vancouver, BC, Canada, 21–21 August 2002; pp. 73–78.
30. Al-Shurman, M.; Yoo, S.M.; Park, S. Black hole attack in mobile ad hoc networks. In Proceedings of the Annual Southeast Conference, Huntsville, AL, USA, 2–3 April 2004; pp. 96–97.
31. Raj, P.N.; Swadas, P.B. Dpraodv: A Dynamic Learning System Against Blackhole Attack in AODV Based MANET. *IJCSI Int. J. Comput. Sci. Issues* **2009**, *1*, 54–59.
32. Mistry, N.; Jinwala, D.C.; Zaveri, M. Improving AODV protocol against blackhole attacks. In Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS), Hong Kong, China, 17–19 March 2010; pp. 1–5.
33. Yang, H. A Study on Improving Secure Routing Performance Using Trust Model in MANET. *Mob. Inf. Syst.* **2020**, *2020*, 8819587. [CrossRef]
34. Misra, S.; Goswami, S. *Network Routing: Fundamentals, Applications, and Emerging Technologies*; Wiley: Hoboken, NJ, USA, 2017.

35. Patel, M.; Sharma, S.; Sharan, D. Detection and prevention of flooding attack using SVM. In Proceedings of the 2013 International Conference on Communication Systems and Network Technologies, Gwalior, India, 6–8 April 2013; pp. 533–537.
36. Choudhury, P.; Nandi, S.; Pal, A.; Debnath, N.C. Mitigating route request flooding attack in MANET using node reputation. In Proceedings of the IEEE 10th International Conference on Industrial Informatics, Beijing, China, 25–27 July 2012; pp. 1010–1015.
37. Ding, P.; Tian Z.; Zhang H.; Wang Y.; Zhang L.; Guo S. Detection and Defense of SYN Flood Attacks Based on Dual Stack Network Firewall. In Proceedings of the 1st International Conference on Data Science in Cyberspace (DSC), Changsha, China, 13–16 June 2017; pp. 526–531.
38. Moudni, H.; Er-Rouidi, M.; Mouncif, H.; El Hadadi, B. Performance analysis of AODV routing protocol in MANET under the influence of routing attacks. In Proceedings of the 2016 International Conference on Electrical and Information Technologies (ICEIT), Tangiers, Morocco, 4–7 May 2016; pp. 536–542.
39. Perkins, C.; Belding-Royer, E.; Das, S. Ad Hoc On-Demand Distance Vector (AODV) Routing (RFC3561), 2003. Available online: https://www.rfc-editor.org/info/rfc3561 (accessed on 24 March 2024). [CrossRef]
40. Kaur, G.; Jain, V.K.; Chaba, Y. Prevention of Flooding Attacks in Mobile Ad Hoc Networks. In Proceedings of the 2nd International Conference on Wireless Intelligent and Distributed Environment for Communication, Milan, Italy, 11–13 February 2019; Springer: Cham, Switzerland, 2019; pp. 193–201.
41. Sharma, S.B.; Chauhan, N. Security issues and their solutions in MANET. In Proceedings of the 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management, Greater Noida, India, 25–27 February 2015; pp. 289–294.
42. Thorat, S.A.; Kulkarni, P.J. Design issues in trust based routing for MANET. In Proceedings of the Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Hefei, China, 11–13 July 2014; pp. 1–7.
43. Pirzada, A.A.; McDonald, C. Trust establishment in pure ad-hoc networks. *Wirel. Pers. Commun.* **2006**, *37*, 139–168. [CrossRef]
44. Ullah, Z.; Islam, M.H.; Khan, A.A. Issues with trust management and trust based secure routing in MANET. In Proceedings of the 13th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 12–16 January 2016; pp. 402–408.