International Journal of Systems and Software Security and Protection
Volume 9 • Issue 4 • October-December 2018

A Secure MANET Routing Protocol for Crisis Situations

Martin Gilje Jaatun, SINTEF Digital, Trondheim, Norway
https://orcid.org/0000-0001-7127-6694
Åsmund Ahlmann Nyre, HEMIT, Trondheim, Norway
Inger Anne Tøndel, SINTEF Digital, Trondheim, Norway

ABSTRACT

Emergency and rescue operations are often carried out in areas where the network infrastructure cannot be relied on for message exchange between first responders. Since a fundamental feature of a Mobile Ad Hoc Network is the ability to operate independently of existing infrastructure, it is deemed a well-suited solution to first responders scenarios. In this article, the authors describe a security extension to the OLSR routing protocol specifically designed for first responder scenarios. The proposed protocol provides node authentication and access control using asymmetric encryption and digital certificates, and also offers a secure group communication scheme. A link encryption scheme is devised to allow for efficient encryption of data even in broadcast mode, without the need for a network-wide shared key. By utilising pairwise symmetric keys for link confidentiality, the authors' solution is both efficient and scalable.

KEYWORDS

Admission Control, First Responders, MANET, OLSR, Security

1. INTRODUCTION

Emergency and rescue operations are often carried out in areas where the network infrastructure cannot be relied on for message exchange between first responders. Although one may argue that some network infrastructure (e.g. GSM/GPRS/UMTS, Wi-Fi, WiMax, satellite, etc.) exists in even the most deserted places, the cause of the emergency operation (e.g. fire, hurricane, explosion, etc.) may also affect the infrastructure. Furthermore, rural infrastructure may not have been dimensioned for the network load imposed by a large-scale emergency operation. Since a fundamental feature of a Mobile Ad Hoc Network is the ability to operate independently of existing infrastructure, it is deemed a well-suited solution to first responder scenarios.

The nature of emergency and rescue operations imply that providing information security is a prerequisite for MANETs to be used in such situations (Meissner, Luckenbach, Risse, Kirste, & Kirchner, 2002, Dearlove, 2004). Unlike the general-purpose MANET, a first responder MANET

DOI: 10.4018/IJSSSP.2018100102

must restrict access to the network such that valuable resources (e.g. bandwidth, battery lifetime, processing power, etc.) are not wasted on activities not related to the operation. Access control also enables node authentication and confidentiality of information by only allowing authorised nodes to send and receive information. With limited resources and a great emphasis on availability it is equally important that security mechanisms do not substantially affect the overall performance and throughput of the network.

Our main contribution in this paper is the design and specification of a new security extension to the Optimised Link State Routing (OLSR) protocol specifically tailored to first responder scenarios. Our protocol extension utilises digital certificates and asymmetric encryption for node authentication and symmetric key establishment. We also specify a new certificate extension to allow for distributed access control based on authorised node descriptions. To efficiently provide confidentiality, our protocol extension also includes a link encryption scheme utilising dynamically established symmetric keys between neighbouring nodes. By limiting the use of asymmetric encryption, our protocol extension is efficient.

The article is structured as follows: We start by giving an overview of relevant state of the art on MANET security (Section 2). We then outline relevant security requirements in Section 3. Next we present an overview of our proposed protocol extension in Section 4, before we detail our solution in Section 5. Finally, we discuss our contribution (Section 9) before concluding and outlining further work in Section 10.

2. BACKGROUND AND STATE OF THE ART

In this section we will present some existing MANET routing protocols (that typically do not offer any security), then present existing attempts to provide secure routing in MANETs. We will also say a few words on intrusion detection in MANETs, and close the section by relating what we have described to MANETs used in crisis situations.

2.1. Routing Protocols

Attempts to secure routing in MANETs have mostly been done by specifying extensions to the original unsecured routing protocols. We therefore will in the following give an overview of the main classification of MANET routing protocols, before we briefly outline the main characteristics of three concrete examples.

MANET routing protocols perform route discovery either proactively or reactively. Proactive route discovery protocols utilize beacon messages, i.e. messages that are transmitted periodically, to inform other nodes of current routes in the network. Thus, whenever a node needs a route to a destination, it is already available, and no additional delay is introduced. The problem with this approach is that control data overhead may be significant due to the periodic flooding of routing information, particularly for dense networks and networks with few transmissions. Routing tables may be quickly outdated for high mobility networks. MANET protocols based on reactive route discovery do not utilize any periodic dissemination of routing information, but instead flood the network for a route to a destination whenever this is needed by the node. Thus, there is no control data overhead as long as the network is idle, and consequently the risk of congesting the network with such control data is reduced. However, if a link in an established route breaks, the entire route discovery process must be re-initiated, which may cause a significant delay in packet delivery. In networks with little node movement, this will rarely happen, and hence the overhead is greatly reduced compared to the proactive approach. There are several factors that need to be considered to determine which of the two approaches are better, including node movement, network density, area size (average hop-count), bandwidth, network load, etc.

The Destination Source Routing (DSR) protocol (Johnson, Hu, & Maltz, 2007, Johnson & Maltz, 1996) is a reactive protocol where the entire route to the destination is listed in each packet. Route

discovery is done through broadcasting route request messages containing the destination address. The request is propagated through the network with all intermediate nodes adding their address to the route stored in the packet, until either the destination or a node with a route to the destination is reached. A route reply is then sent either using the reverse path of the request, or preferably piggybacked on a new route request to the initial sender. Piggybacking is considered better since links may be asymmetric and hence the reversed route may not be valid. Route maintenance is performed either actively through the reception of link-layer acknowledgements or passively through detecting the receiving node's retransmission in promiscuous mode. Detected link errors, i.e. missing acknowledgements, result in the transmission of a link error message to the sender. Similar to route reply, this may either be done through the reverse path of the current route or preferably piggybacked on a route request to the sender. To improve efficiency, DSR also allows nodes to utilize promiscuous mode to discover routes and errors handled by adjacent nodes.

Ad hoc on-demand distance vector routing (AODV) (Perkins & Royer, 1999, Perkins, Belding-Royer, & Das, 2003), is a reactive protocol similar to DSR. AODV however does not carry the entire path in the packet header, instead each intermediate node independently computes the optimal next-hop for the given destination. Route discovery is performed by flooding route requests (RREQ) in the network to reach either the destination or an intermediate node with a valid route to the destination. The next-hop in the reverse path, i.e. the node from which the RREQ was received, is recorded by every intermediate node. Upon reaching the destination (or another node with a valid route) a route reply (RREP) message is unicast back along the the recorded reverse path. Intermediate nodes receiving a RREP record the forward path, i.e. the node from which the RREP was received. Timers are associated with the routing table entries such that invalid or unused routes are removed after a predefined period of time. AODV is said to be "a pure on-demand route acquisition system" (Perkins & Royer, 1999), meaning that unless nodes lie on an active path (i.e. route), they do not have to maintain or advertise any routing information.

The Optimized Link State Routing (OLSR) protocol (Jacquet et al., 2001, Clausen & Jacquet, 2003) is a proactive protocol that actively maintains routes to all destinations in the network by periodically transmitting control information. Local link sensing is achieved by broadcasting HELLO messages containing every one-hop link known to the node. The receiver is then able to compute its two-hop neighbour set, which in turn allows it to create a Multi-Point Relay (MPR) set. The MPR set is formed such that it includes the least number of one-hop neighbours such that every two-hop neighbour can be reached. The protocol specifies that only neighbours belonging to the MPR set are allowed to forward control messages on behalf of a node. Thus, the cost of flooding control packets in the network is considerably reduced. Topology information beyond the two-hop neighbours already known using HELLO messages, is distributed using Topology Change (TC) messages. Every node maintains a MPR Selectors set containing all nodes that have selected it as MPR. Every node with a non-empty MPR Selectors set must periodically flood the network (using MPR) with TC messages containing at least every node in the MPR Selectors set. One may extend the TC messages to include additional nodes and also create suboptimal MPR sets, however at the cost of increased overhead and consequently reduced performance.

2.2. Secure MANET Routing

Ariadne (Hu, Perrig, & Johnson, 2005) is a secure on-demand routing protocol based on DSR. It provides three ways of authenticating routing messages; using pairwise shared secret keys, using pairwise shared secret keys combined with broadcast authentication or using digital signatures. If shared keys or digital signatures are used then the routing message is authenticated by appending a Message Authentication Code (MAC) or digital signature for each intermediate node. The protocol also proposes the use of the Timed Efficient Stream Loss-tolerant Authentication (TESLA) broadcast authentication mechanism (Perrig, Canetti, Tygar, & Song, 2002) for intermediate hop authentication and shared secret for endpoint authentication. The TESLA mechanism utilizes

reversed hash chains and delayed key disclosure to provide authentication of routing messages. The protocol requires loosely synchronised clocks and a delay of at least the network round-trip time to guarantee that the message has been received by all nodes before the key is disclosed. Ariadne provides both integrity and authentication of routing information, however non-repudiation can only be guaranteed when using digital signatures, since MACs can also be calculated by the recipient, and are impossible for others to verify.

The Secure Routing Protocol (SRP) (Papadimitratos & Haas, 2002) is designed as an extension to DSR or the inter zone part of the Zone Routing Protocol (ZRP) (Haas, 1997). The protocol relies solely on symmetric key cryptography for authenticated route discovery, assuming that shared secret keys have already been established between the source and destination nodes. A MAC based on the shared key is appended to route requests in order to allow the destination to authenticate the originator. However, intermediate nodes and the recorded route are not authenticated. Additionally, route error messages do not contain any verification and hence can be forged by adversaries. The protocol provides authentication and integrity, but introduces some serious issues for the availability.

The Secure AODV routing protocol (SAODV) (Zapata & Asokan, 2002) utilizes hash chains for authenticating mutable data in route request messages. However, for non-mutable data the protocol uses only digital signatures. A node requesting a route to a destination generates a random seed for the hash chain and computes the maximum hash chain value by repeated hashing of the seed until reaching the maximum hop count. The signature on all fields but the seed and hop count is appended to the message. Intermediate nodes verify the signature and that the maximum hash chain value is reached after hashing the received seed (max_hop_count-hop_count) times. If verification holds, the hop count is stepped and the seed is updated by hashing it. In order to allow intermediate nodes to respond with a RREP whenever it holds a valid route in its route cache, the double signature scheme is proposed. Route error messages do not use the hash chain mechanism, but are instead digitally signed. Since it is not considered relevant which node initially started the error message, the signature is replaced for each hop, rather than appended. The protocol provides authentication for end nodes, but not for intermediate, allowing adversaries on the path to forge their identity. The hash chain mechanism guarantees that malicious nodes cannot reduce the hop count value, but may increase it or omit updating it.

Authenticated Routing for Ad hoc Networks (ARAN) (Sanzgiri et al., 2005) is a signature-based extension to the AODV routing protocol, providing secure route discovery. Route requests are signed by the originator of the request and propagated throughout the network. Intermediate nodes will, upon receiving the request, verify the signature and the sequence number before adding their signature and forwarding it to their neighbours. The destination validates all signatures and creates a signed route reply message including the sequence number and source of the request. The reply is sent back to the source along the reverse path of the request, where intermediate nodes verify and sign it in the same manner as the request. Link failures are detected and reported using routing error messages, which are signed by the reporting entity and propagated through the network. No intermediate node signs the error message. The proof-of-concept implementation and subsequent testing indicates that the protocol increases the delay for route setup by several orders of magnitude. The tests done on the protocol show that even with fairly powerful laptops, the ARAN protocol using 1024 bits RSA keys are approximately 23 times slower than the unsecured AODV protocol (Sanzgiri et al., 2005).

The Secure Link State Protocol (Papadimitratos & Haas, 2003) is a secure proactive routing protocol employing a similar strategy as SAODV for message authentication. Link State Updates (LSUs) are digitally signed by the originating node, with all mutable fields excluded. The mutable fields are instead governed by a hash chain, which does not allow reduction in the hop count. By specifying a maximum hop count, the protocol can be used as the intra zone part of ZRP (Haas, 1997) Only end-nodes are authenticated, such that intermediate nodes may spoof their identity without being revealed.

The Secure Transmission Protocol (STP)(Papadimitratos & Haas, 2006) utilises symmetric key encryption for reliable end to end authentication of data transmission. Messages are split up and sent on disjoint routes, and missing packets result in resending and updated routing information. Symmetric keys are assumed to be established in advance. As pairwise shared secrets do not scale well, Puzar et al. (Pužar, Plagemann, & Roudier, 2008) suggest a solution where every node in the network shares the same key. Mechanisms are defined that result in periodic key changes, but during key re-selection the network is in an inconsistent state unable to route messages.

2.3. Intrusion Detection

Given the lack of network perimeters and the open collaborative nature of mobile ad hoc networks it is hard to define what actually constitutes a network intrusion. Commonly, intrusions are viewed as malicious behaviour aimed at disrupting or degrading network performance.

The WATCHERS protocol (Bradley, Cheung, Puketza, Mukherjee, & Olsson, 1998) was proposed to enable detection of disruptive nodes in the network. The idea is to use conservation of flow, i.e. what comes in must come out, to detect misbehaving nodes. Every node monitors its neighbours and measures the amount of dropped packets, misrouted packets, etc, by listening to the communication of adjacent nodes and comparing received packages to the transmitted ones. If metrics exceed a predefined threshold, the corresponding node is considered malicious and the link to it dropped. The protocol has been criticised for its assumptions on the reliability of wireless communication (Hughes, Aura, & Bishop, 2000), since there are numerous valid reasons for dropping a packet.

A similar detection and prevention scheme were proposed by Marti et al. (Marti, Giuli, Lai, & Baker, 2000) where a *watchdog* is used to detect misbehaving nodes and a *pathrater* is used to compute paths avoiding the detected nodes. Designed for the DSR protocol, the watchdog mechanism utilizes promiscuous mode and knowledge of the path to the destination to assert whether the neighbour node actually forwards packets as expected. A counter is increased whenever a routing misbehaviour is detected, ultimately blocking the node if the counter reaches a predefined threshold. Unlike the WATCHERS protocol, watchdog and pathrater are protocol specific so as not to rely solely on the conservation of flow as a detection mechanism.

The COllaborative REputation mechanism (CORE) (Michiardi & Molva, 2002) like the previous protocols also utilizes a watchdog mechanism and additionally includes a reputation system. The reputation system specifies three different types of reputation; subjective, indirect and functional. Subjective reputation is based on direct observation through the watchdog mechanism operating in promiscuous mode. Indirect reputation is based on received reputation metrics from other nodes, while functional reputation indicates the reputation for a particular functionality (e.g. packet forwarding). To prevent denial-of-service attacks by malicious broadcasting of negative ratings for benign nodes, indirect reputation may only take positive values. Unlike the *watchdog/pathrater* approach described above, CORE does not exclude malicious nodes from routes, but rather encourages cooperation in order to receive network services.

The DSR protocol extension CONFIDANT (Cooperation Of Nodes: Fairness In Dynamic Ad-hoc NeTworks) (Buchegger & Boudec, 2002) consists of a monitor, a trust manager, a reputation system and a path manager. The monitor is similar to the watchdog mechanism and performs local detection of misbehaviour. The trust manager is responsible for distributing ALARM messages regarding malicious behaviour to nodes belonging to a friends list. It also computes trust levels of received information such that weighting may be employed for rating changes. The reputation system provides a quality rating of participating nodes, based on local and received information. Sufficient evidence must be gathered before a decision is made and it must have been gathered over a long enough time to rule out coincidence. The path manager is responsible for rating the active paths in the network and to react to paths containing malicious nodes (e.g. delete the path).

CONFIDANT is similar to the watchdog/pathrater approach, but additionally creates incentives for correct behaviour of nodes by refraining from forwarding packets on behalf of misbehaving nodes.

The CONFIDANT protocol proposes the use of a trust manager to share its ratings with the other nodes in the network. Route selection is done according to a trust metric such that the most trusted path is selected. If there is more than one path with highest trust rating, the shortest is selected.

The strategy by Wang et al. (Wang, Lamont, Mason, & Gorlatova, 2005) is to use protocol specific properties for sanity checking routing updates. For the OLSR protocol, the use of multi-point relays (MPRs) allows some checking of the originating node. For example; If node A advertises a link to node B, then node A must be an MPR of node B. Thus, node B can perform a sanity check of the received information by comparing the originator to its set of MPRs. Wang et al. (Wang et al., 2005) further propose for B to broadcast (through its MPRs) a message to invalidate the advertised link, so that other nodes will refrain from using it. There are several such properties that may be used to verify the correctness of the advertised information. The article does not discuss other reasons for such incoherence, such as latency in TC updates, link failures, etc, nor what actions should be taken upon receiving an invalidation of a link. Labelling the originator as malicious would introduce the possibility for malicious nodes to emit invalidations randomly to its MPR nodes and thereby convince the network that the benign node is malicious. If the check was performed by any adjacent node to B (i.e. in B's HELLO set) or any of B's MPRs, a majority vote could be used to guarantee the correctness of the invalidation.

Otrok et al. propose a different strategy for intrusion detection that greatly reduces power consumption of participating nodes (Otrok, Mohammed, Wang, Debbabi, & Bhattacharya, 2008). The idea is to let nodes in a cluster elect one single node to perform intrusion detection on behalf of the others in a collaborative game, maximising the security for the network as a whole. In order to mitigate the risk of having a misbehaving node performing the intrusion detection a set of checkers are simultaneously elected to verify correct behaviour. By sampling the communication, the checkers collaboratively decide through majority vote whether the elected node is misbehaving. For this approach to be valid, at least half of the checkers must be benign in order to guarantee that no benign node is blocked from the network. Although the approach is favourable in terms of energy consumption, networks of highly mobile nodes may force constant re-elections of both intrusion detection nodes and checkers. While obviously degrading performance and throughput of the network, this may also hamper detection of misbehaving nodes as it is impossible to gather sufficient information for making a decision before a re-election is done.

Another approach to reduced energy consumption is for each node to only have its intrusion detection mechanism running a portion of the time, as suggested by Marchang and Tripathi (Marchang & Tripathi, 2007). They develop a game theoretic approach to model how the defender and attacker choose the percentage of the time the defence and attack will be running, respectively. By assuming different detection rates, the game is simulated to show the impact of reduced monitoring.

2.4. Relating State-of-the-Art to Crisis Situations

In the previous sections we have given an overview of preventive and reactive security mechanisms tailored for use in MANETs. The next step would be to identify the missing parts (if any), in order to provide secure MANETs, and thus we need to map each of the protocols to whether they provide authentication, confidentiality, integrity, authentication and non-repudiation.

For reactive protocols aimed at detecting misbehaving nodes, there is typically no cryptographic support that enables confidentiality, authentication and non-repudiation. Integrity could be supported by observing neighbours' retransmissions, however the key property of such protocols is availability. By detecting and reacting upon misbehaving nodes the probability of correct functioning of the network is improved. Thus, when identifying whether the protocols meets the security goals, we have only included the preventive protocols. Table 1 summarizes how the various protocols meet the security goals. Note that the availability property is considered satisfied if the protocol improves denial-of-service resistance and does not imply that it will resist all attacks. Also, the non-repudiation property is not considered satisfied when using hash-chains or symmetric key MACs for message

Table 1. Comparison of proposed secure MANET protocols

Protocol	Availability	Confidentiality	Integrity	Authentication	Non-Repudation	Assumptions
Ariadne	Yes	No	Yes	Yes	No	Established PKI or shared secret keys
SRP	Yes	No	Yes	Yes	No	Established shared secret keys
SAODV	Yes	No	Yes	Yes	Yes	Established PKI
ARAN	Yes	No	Yes	Yes	Yes	Established PKI
SLSP	Yes	No	Yes	Yes	Yes	Established PKI

authentication. Hash chains only provide temporal evidence, since after key disclosure anyone can create authentic messages. MACs on the other hand are not verifiable to anyone but the entities that share the secret key, and do not provide evidence as to which of these entities initiated the message. What is perhaps most noteworthy is the fact that none of the protocols provide any confidentiality of routing information. For general purpose MANETs with free access, confidentiality may seem unnecessary. However, for closed networks such as military, rescue or crisis management MANETs, it may be vital that outsiders cannot identify network participants and also are unable to build a network map. Thus, for such applications of MANETs, there should be a protocol to provide this. Note also that all protocols either rely on an established MANET-wide PKI or pairwise shared secret keys. Although there exist numerous key management and key sharing schemes (Zhou & Haas, 1999, Ramkumar & Memon, 2005; Saxena, Tsudik, & Yi, 2007), this is not trivially achieved, especially for open commercial applications areas such as a conference venue.

Because of the problems with network wide keys, we do not believe the approach by Puzar et al. (Pužar et al., 2008) to be the best solution for MANETs. Still Puzar et al. specifically address emergency and rescue operations, and many of their ideas fit well within this setting; they rely on pre-existing certificates to be in place, all certificates are signed by the same CA, and they put restrictions on which nodes are authorised to influence routing.

There are of course other non-security properties to consider such as data and processing overhead, battery consumption, delay, etc., which influence the choice of security mechanism. For instance, the extensive use of digital signatures in the ARAN protocol ensures a higher level of security (e.g. secure authentication of intermediate nodes) at the cost of added processing and data overhead for each hop. Thus, the optimal protocol is not necessarily the one providing the optimal security.

As with conventional intrusion detection systems, detecting misbehaving nodes in MANETs may be erroneous, which in turn may have devastating effects on the Network. Since availability is the primary goal of such systems, labeling a benign node as malicious would in effect constitute a denial-of-service attack by the protocol. Similarly if malicious nodes are undetected, the availability of the entire network would be threatened.

The protocols and mechanisms outlined in Section 2.3 all use anomaly-based detection, where deviations from correct protocol behaviour are considered malicious. Additionally, all protocols rely on obtaining information by promiscuously overhearing neighbour transmissions. A problem here is the possibility of a node having two neighbours (that are not themselves neighbours) transmitting simultaneously, causing a collision only for the node operating in promiscuous mode. Such situations and also the unreliability of the wireless medium makes it very difficult to perform accurate detection.

3. REQUIREMENTS

Most existing work on security in ad hoc networks handles security requirements only superficially. The most relevant work that we are aware of is a study of known problems with existing routing

protocols for ad hoc networks, as presented by Dahill et al. (Dahill, Levine, Royer, & Shields, 2001) and Sanzgiri et al. (Sanzgiri, Dahill, Levine, Shields, & Belding-Royer, 2002). This study led to seven security requirements, covering spoofing of route signalling, fabrication and altering of routing messages, malicious formation of routing loops, route redirection from shortest path, which nodes should be part of route computation and discovery, and exposure of network topology. Ad hoc networks are divided into three categories, each requiring a different level of security. Emergency and response in disaster areas is considered part of the managed-hostile environments group, which should meet all the identified requirements.

A less detailed list of security requirements on routing protocols of ad hoc networks is provided by Zapata and Asokan (Zapata & Asokan, 2002). They are concerned with routing updates, and state the importance of import authorisation, source authentication and integrity of routing information. Data authentication is said to be covered by the combination of the above. Compromised nodes are not considered, as they believe this only to be relevant for military scenarios. Availability is also not covered as they find it unfeasible to prevent denial of service (DoS) attacks when using wireless technology.

Wrona (Wrona, 2002) takes a different approach, and states that ad hoc networks in general have the same security requirements as other communication systems. Ad hoc networks are however extreme in the requirements on the sophistication and efficiency of the security mechanisms themselves, mainly because of the lack of infrastructure and the very dynamic and ephemeral character of relationships between network nodes. However, Wrona does not provide more details on the security requirements.

3.1. Elicitation Method

Tøndel et al. (Tøndel, Jaatun, & Meland, 2008) give an overview of existing approaches to security requirements elicitation, and identify the most commonly recommended steps. A four-step approach is then proposed: 1) Identify security objectives, 2) Asset identification, 3) Threat analysis, and 4) Documentation of security requirements. Objectives are defined as "the high-level requirements or goals that are most important to customers, and the requirements that must be met to comply with relevant legislation, policies, and standards" (Tøndel et al., 2008). Assets are important as "security requirements are primarily needed in order to protect our assets, and this will obviously be impossible to do properly unless we know what these assets are" (Jaatun & Tøndel, 2008). During threat analysis likely attacks against the most important assets are studied.

In this work the requirements elicitation process was performed by the authors, who can be said to be network security experts. As we did not have access to customers, objectives were identified based on previous work in OASIS and based on reading material on ad hoc networks for emergency and rescue operations. Assets were identified in a workshop using the approach described by Jaatun an Tøndel (Jaatun & Tøndel, 2008). This approach is based on brainstorming, something that may seem a bit too unstructured at first glance. Available publications on asset identification however show that brainstorming techniques and similar are used in several approaches - with few problems experienced (Caralli, Stevens, Young, & Wilson, 2007, Jaatun & Tøndel, 2008).

In the workshop assets were prioritised by considering the importance of the confidentiality, integrity and availability of each asset from the viewpoint of system users, owners and attackers. By including different viewpoints we were able to handle the fact that different actor's view of an asset are not directly related (Haley, Laney, Moffett, & Nuseibeh, 2008). Hence most focus is given to assets that are important for attackers as well as system owners and/or system users. In order to keep the method as lightweight as possible we only used four classes of priorities for our assets: high, medium, low and irrelevant. The total value of e.g. the confidentiality of an asset is then the sum of its value from the different viewpoints. This is of course a simplification, but still provides an easy and powerful way of finding which assets (or more correctly, which properties of the assets) are important in the system.

Based on the result of asset identification, we studied the threats towards the most important assets. For the threat modelling we used attack trees as defined by Schneier (Schneier, 1999), as his threat modelling method is well recognised and fits our approach well. A selection of the identified attacks is presented in Table 2. Most attack trees were created in a workshop, the rest was created by one expert and checked by the others at a later point in time. At the end one expert identified and documented security requirements by going through the security objectives, assets and attack trees. The requirements were later checked by the other experts.

3.2. Objectives

The identified security objectives are listed in Table 3. As a basis for identifying these objectives we described what will be the typical usage of the OASIS ad hoc network and the main security issues as we see it.

Table 2. Examples of identified attacks

Attack Tree		Main Attacks Identified
A1	Get access to and use an existing node	Access node, either physically or externally, and either get access to valid access credentials or bypass access control.
A3	Get access to sensitive information	Get access to communication through eavesdropping or routing, and break any encryption. Get access to sensitive information on a node.
A4	Get access to access credentials	Get access to communication or nodes that contain access credentials and break any protection. Find credentials. Guess credentials. Perform social engineering attack.
A7	Destroy integrity of information	Flip bits in communication. Destroy integrity of packets during routing. Destroy integrity of information stored on nodes.

Table 3. Security objectives

Nr.	Objective
O1	Confidentiality: For some information confidentiality will be required by law, e.g. medical information. Mechanisms must thus be in place that is able to offer adequate protection of confidentiality.
O2	Availability vs. confidentiality: As the OASIS ad hoc network is intended used in crisis situations, availability is in many, if not most, cases more important than confidentiality.
O3	Integrity: As there are attackers that may want to attack the integrity of information in order to hamper the operation, integrity should be ensured.
O4	Participation and collaboration: Personnel from different organisations and regions must be allowed to participate and collaborate without compromising the security of the network.
O5	Access control: There is no intention of letting “just anyone” connect to the network and start interacting with it. This is a difference between a first responder network and the “academic ideal” ad hoc network.
O6	User hierarchy: Security solutions should support the hierarchical nature of emergency operations.
O7	Dynamics of responsibility: Security solutions should support dynamics in responsibility and authority.
O8	Limited node resources: Devices typically used for the OASIS ad hoc network will have limited computational power and battery available. The security solutions must take this into account.
O9	Limited bandwidth: The bandwidth available will typically be limited, and this must be taken into account when choosing and implementing security solutions.
O10	Usability: Security solutions must not render the system too difficult or troublesome to use.
O11	Not dependent on central nodes: The ad hoc network should function without any central nodes.

The current predominant communication paradigm for first responders is voice communication over radio networks (e.g. TETRA). MANETS will enable distribution of rich content in uni-, multi- or broadcast mode. In addition to user nodes, we envisage a command post that is operated from a specialised vehicle and possess greater computing resources. In situations where external communication infrastructure is available, both the command post and first responders may connect to external resources (health networks, police networks, etc.).

Many of the challenges of securing MANETs in general (Wu, Chen, Wu, & Cardei, 2007) also apply to MANETs for first responders. However, communication patterns, media diversity, organisational structure and legislative issues constitute both challenges and opportunities for first responders MANETs. While MANETs in the general case should allow anyone to participate, the situation is quite the contrary for first responders. First responders require an access control that prevents nodes from wasting their resources (energy, processing power, bandwidth, etc.) on information that is not relevant for the mission. While this normally requires pre-configuration, the mechanism should be flexible enough to allow temporary access to nodes that have not been pre-configured. This will allow first responders to dynamically include volunteers, experts, etc., in the operation as they see fit.

We have identified two main types of attackers posing a threat to first responder MANETs: news media and terrorists. News media is primarily interested in obtaining information on the tactical operation by launching passive attacks. Information is assumed to be most valuable in real-time, but remains interesting for critics in the evaluation process. Terrorists are interested in obstructing the network operations by launching active attacks to disrupt routing, forge communication, thwart legitimate access, etc. It is possible that a physical terrorist attack (e.g., explosion, fire, etc.) is extended by a follow-up attack on the first responder emergency operation network.

Organisations involved in emergency operations are typically hierarchically structured, where information flows upwards and decisions downwards. However, the operational hierarchy is affected by the type of personnel available at any given time, such that dynamics in responsibility and authority must be anticipated. As an example, police commanders are normally in charge of the overall operation, but if none with sufficient authority is present, a fire-fighter officer will assume this role. In addition, personnel from different organisations and regions must be allowed to participate and collaborate without compromising the security of the network. This makes key management for authentication and access control in particular, a troublesome task.

In a crisis situation, it is likely that some medical data will be exchanged. Confidentiality of medical data is required by law to protect the privacy of citizens. However, in the event of an emergency, preserving lives is considered more important than preserving privacy. If confidentiality requirements hamper operations, medical staff will plead just cause in order to ensure availability of data. For the same reason usability is also important, as security mechanisms significantly hampering the performance of first responders are not likely to be used.

For any tactical operation it is vital that commanding nodes (e.g. squad leader) have access to a situation map with the current layout of the network (with optionally geographical position). This coupled with the need for low latency in route discovery makes proactive protocols seem as the better choice.

The limited available resources of devices in MANETs are a prime concern when designing effective security mechanisms. This constraint also applies to the first responder case, but not to the same extent. Devices for first responders are not assumed to be COTS (Commercial Off-The-Shelf), but rather specifically designed to meet communication requirements and to withstand environmental stress. It is thus conceivable that devices for first responders will have far more resources than hand-held devices designed for the common public.

3.3. Requirements Summary

We devised in total 30 security requirements (Tøndel, Jaatun, & Nyre, 2009) relevant for ad hoc networks as used in OASIS. The requirements relevant for the work presented in this paper is these requirements is presented in Table 4. In addition we identified requirements concerning e.g. physical access to nodes, input control and credential quality. The requirements differ from the requirements suggested by Dahill et al. (Dahill et al., 2001) and Sanzgiri et al. (Sanzgiri et al., 2002) in that they cover more than just routing. In our requirements elicitation process we have also focused on objectives, assets and threats, while they mainly focused on problems with existing approaches. Our requirements are also more detailed than those presented by Zapata and Asokan (Zapata & Asokan, 2002) and Wrona (Wrona, 2002). The entries in the final column of Table 4 refer back to the identified objectives or attacks as exemplified in Table3 and 2(see Tøndel et al. (Tøndel et al., 2009) for more details).

Table 4. Selected security requirements

Nr.	Requirement	Source
R8	Network access: Access to the OASIS ad hoc network should require authentication.	A2 A3
R9	Strength network access: The mechanism for access to the OASIS ad hoc network should be able to withstand extensive security testing by security testing professionals.	A2 A5
R10	Link confidentiality: The confidentiality of sensitive information must be protected while sent on the communication link.	A3
R11	End-to-end confidentiality: The confidentiality of sensitive information should be protected end-to-end during communication.	A3
R12	Encryption algorithms: All encryption mechanisms should be implemented with well recognised algorithms.	A3 A4
R13	Encryption keys: All keys used related to encryption should have a key length that is recognised to provide high protection.	A3 A4
R14	Key management: All key management mechanisms should be well known and recognised.	A3 A4
R16	Credential communication: The confidentiality of access credentials must be protected end-to-end during communication.	A4
R20	Transmission errors: For all communication it should be possible to detect transmission errors.	A5-A9
R21	Integrity transmission: Integrity of communication related to access control (or possibly all communication) should be protected while sent on the link in order to detect deliberate changes by attackers.	A5-A9
R23	Detection of misbehaving nodes: The OASIS ad hoc network should include mechanisms for detecting misbehaving nodes.	A8
R26	Identities vs. access rights: Mechanisms must be in place that ensures node users cannot edit their identities and by that increase their access rights.	A6
R27	Identities and spoofing: Mechanisms should be in place that ensures users cannot edit their entities and by that spoof as another user.	A6
R28	Participation: The access control mechanism to the ad hoc network should support participation and collaboration from police, fire and medical professionals from the same or neighbouring districts.	O4
R29	Decentralisation: Access control to ad hoc network should work without any centralised nodes.	O11

4. PROTOCOL OVERVIEW

In this section we outline the main features of our proposed protocol. We first providea basic overview of the OLSR protocol for MANETs, which we base our specification on. Next we describe how a certificate hierarchy is assumed to be organised and the authentication and access control procedure is accomplished. Finally, we give a brief description of our link encryption scheme.

4.1. Optimised Link State Routing Protocol

The Optimised Link State Routing (OLSR) protocol (Jacquet et al., 2001, Clausen & Jacquet, 2003) is a proactive protocol designed for MANETs. The protocol introduces the concept of Multi-Point Relay (MPR) flooding, where only designated nodes rebroadcast messages. Each node selects a subset of its neighbours, called the MPR set, such that every two-hop neighbour can be reached through at least one MPR. By restricting forwarding to only the nodes that have been selected as MPR by the originator, the MPR scheme allows for an optimised packet flooding that greatly reduces the number of broadcasts compared to the general-purpose flooding.

The protocol defines HELLO messages for local link sensing and Topology Change (TC) messages for network wide topology diffusion. Nodes advertise their link set and MPR selection through periodic broadcasts of HELLO messages containing all direct links with corresponding status (e.g. symmetric, MPR, etc.). At the receiving end, the messages are used for link sensing, determine forwarding actions (whether the node is MPR or not) and to build two-hop neighbour topology that forms the basis for MPR selection. The node also maintains an MPR Selector Set containing all neighbours that have selected the node as MPR. HELLO messages are intended for neighbours only and are never forwarded.

Topology Change (TC) messages are periodically flooded in the network to allow nodes to build a complete routing table. The protocol requires that every node having been selected MPR must broadcast TC messages containing at least all neighbours in the MPR Selector Set. This being a minimum, additional links may be advertised for redundancy.

4.2. PKI

The authentication mechanism of our protocol is based on X.509 certificates (Cooper et al., 2008) and requires the establishment of a certification authority (CA) for each organisation participating in the network. The CA operates off-line, i.e. does not participate in the MANET, and is responsible for issuing certificates to all its nodes. The number of hierarchical levels and their structure (geographical, organisational, etc.) is configurable by the user. However, if two nodes that do not share a CA (at some level) are to authenticate each other, at least one of the certificates in the certificate chain must be cross signed, so that they may verify the authenticity of each other's certificate. For first responder organisations that are likely to cooperate, such cross-certification is recommended. The certificates must include an X.509 extension containing a description of the node and the certificate.

Distribution of Certificate Revocation Lists (CRLs) is not trivial, especially when allowing cross signed certificate authorities. In order to limit the size of CRLs and also the impact of failing to distribute CRLs, we propose to limit the validity time of certificates to typically a few months. The process may be automated as part of docking/re-charging procedure at the node's home location (e.g. at the hospital). CAs could have considerably longer validity time (e.g. years) since these are not exposed in the same way as mobile nodes.

In order to provide network access to nodes that do not possess regular first responder certificates, we propose a special short-term certificate. This type of certificate is issued on scene by regular authorised nodes. Whether all regular nodes, or only a subset of such (e.g. high-ranking officers) are authorised to issue short-term certificates is configurable. With validity time set to 24 hours, the need for CRLs is diminished.

4.3. Authentication, Key Establishment and Access Control

In order to verify the authenticity of certificates (i.e. prove ownership) a challenge-response protocol is proposed. The process (depicted in Figure 1) is initiated whenever a new link is discovered (through the reception of a HELLO message) and consists of four main steps:

1. Node B generates a challenge (CKeyID) for node A;
2. Node A signs the challenge (CKeyID) and generates a new one (RKeyID) for node B;
3. Node B verifies the response from A and generates a key;
4. Node A verifies the response from B and stores the received key.

This process serves three main functions as it 1) provides mutual authentication, 2) distribute the authorised node description (contained in the certificate), and 3) establishes a shared secret key.

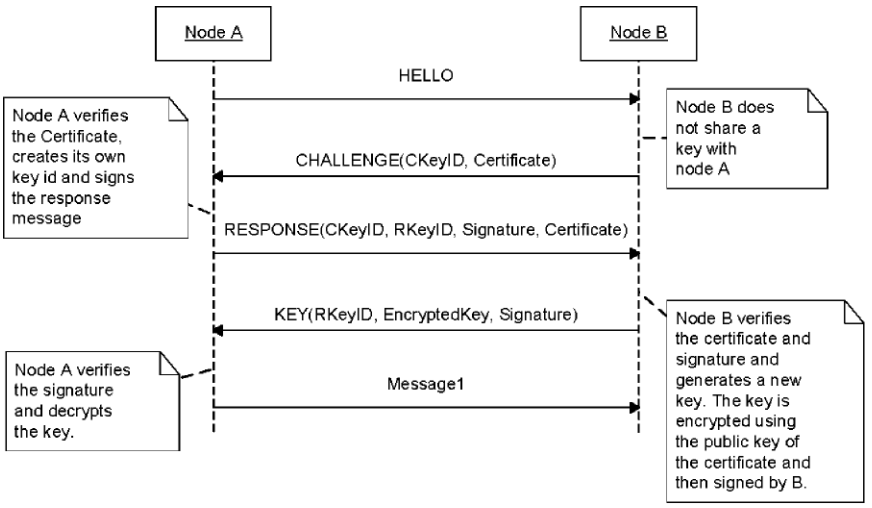
After a successful authentication, the access control mechanism utilises the node description contained in the certificate extension to determine the access level to grant the node. We have defined two levels; where one is granted to all nodes with regular certificates, while the other is granted to nodes with temporary short-term certificates. The latter group is not allowed to be selected MPR and may therefore not interfere in routing protocol updates (except from the ones originating from the node itself).

4.4. Link Encryption

We propose an effective symmetric encryption scheme where messages are encrypted on a per link basis. The scheme relies on the establishment of symmetric keys for each pair of neighbours. These keys are denoted *link keys* and are established during the final step of the authentication and key establishment process described previously.

To reduce the processing overhead for intermediate nodes, the payload is encrypted once using a one-time key, which in turn is encrypted using the link key. Thus, intermediate forwarding nodes need only decrypt and re-encrypt the header field, rather than the entire packet. Additionally, to accommodate broadcast messages, multiple headers are allowed such that all neighbouring nodes may decrypt the one-time key using their link key. This way one need not repeat the entire payload, only the minimal header.

Figure 1. Key establishment process



5. PROTOCOL DESCRIPTION

Our protocol description is based on the OLSR protocol and is aimed at pointing out where the two protocols differ. Hence, we will often refer to the OLSR specification (RFC 3626 (Clausen & Jacquet, 2003)) on matters that are not treated specifically by our security extension.

5.1. Message Formats and Processing

All existing OLSR messages such as TC and HELLO messages are distributed in broadcast mode without explicit addresses of recipients. For our link encryption scheme we therefore define the general encrypted message format (Figure 2) to allow multiple recipients of the per link encrypted message. The summary section contains the number of key blocks (KB_counter) and the type and length of the Message Authentication Code (MAC) (MAC_length). There is one Key Block for each recipient containing a key identifier (Key_id) and the one-time key encrypted with the corresponding key. The MAC and encrypted payload constitutes the rest of the message. By using key identifiers instead of IP addresses, the protocol does not allow adversaries to eavesdrop on the communication in order to get an overview of participating nodes.

The encrypted HELLO message defined for our protocol is identical to the original HELLO message format after decryption. The encrypted TC messages contain a node description in addition to the already specified solution (see Figure 6).

The message formats for our challenge response protocol are given in Figures 3, 4, and 5. The key identifiers (CKeyID/RKeyID) are selected randomly and therefore also serve as nonces.

Figure 2. General encrypted message format encapsulating HELLO and TC messages

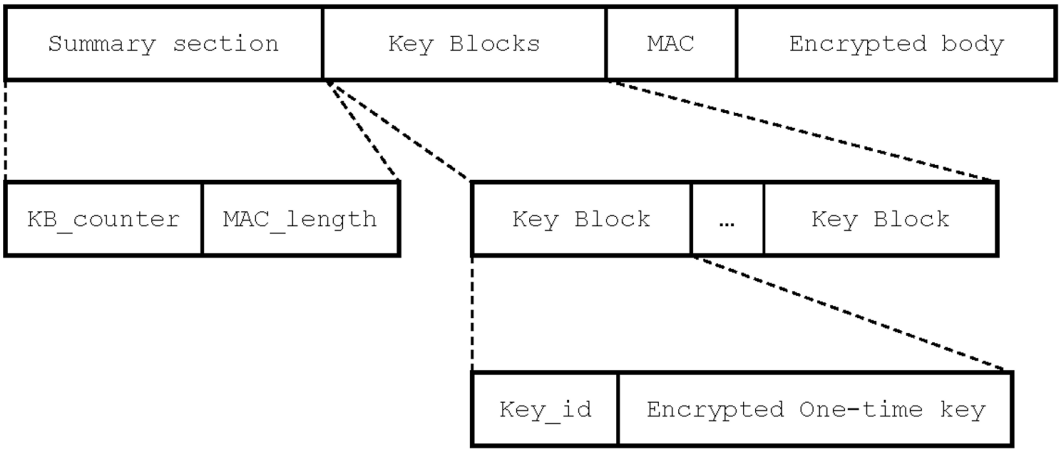


Figure 3. Challenge message format

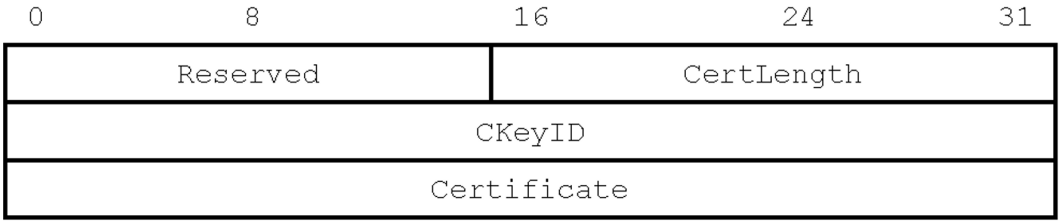


Figure 4. Response message format

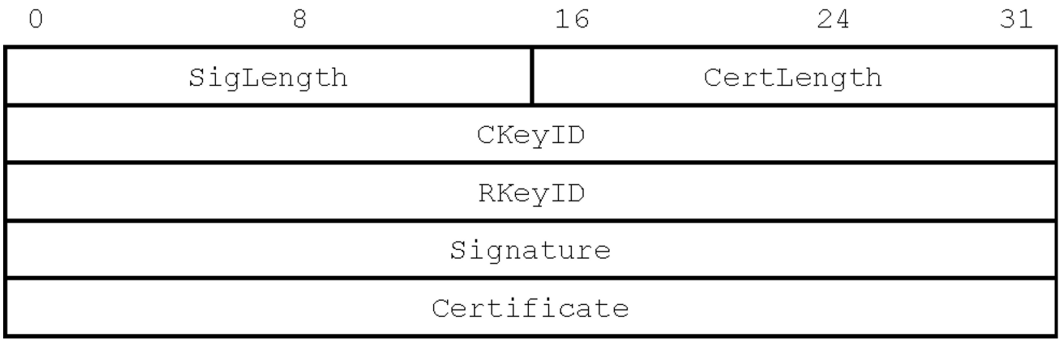


Figure 5. KEY message format

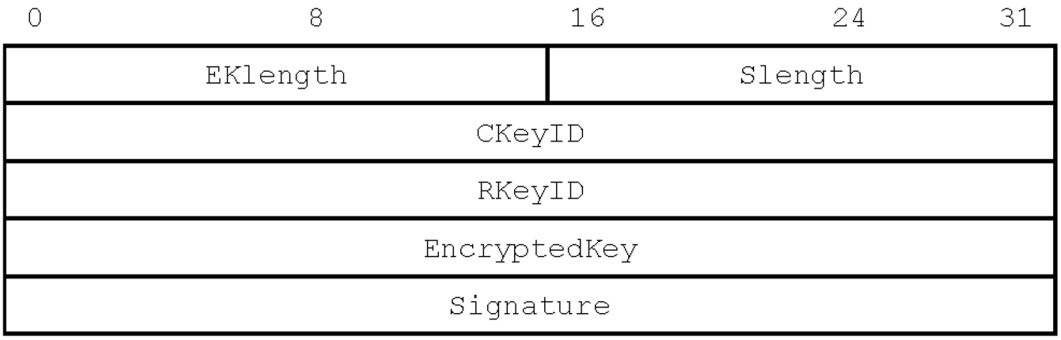
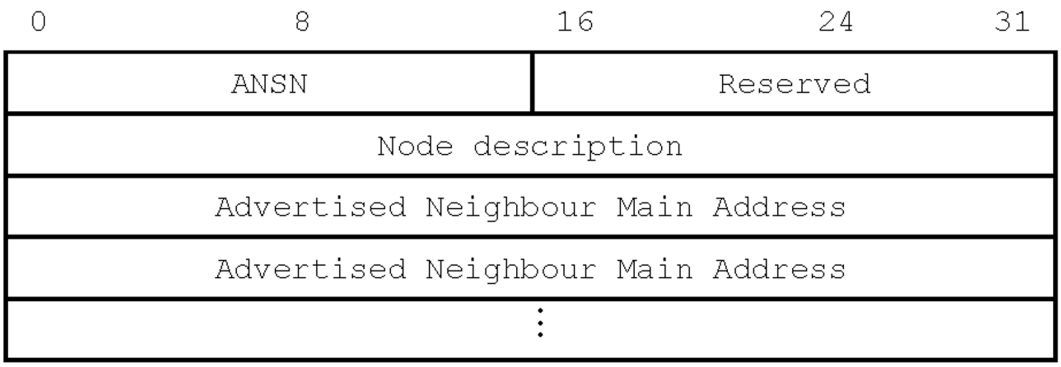


Figure 6. TC message format after decryption



5.2. Information Bases

We extend the information bases for OLSR to include link keys, node descriptions and access level. The link set tuple is extended to include local and neighbour key identifiers (L_local_KID, L_neighbour_KID) and key value (L_key_value). The local key identifier is used whenever a message is sent to a node, while the neighbour key identifier is used whenever a message is received. Local key identifiers must be unique for each node, while neighbour key identifiers need not. The neighbourhood information base is extended to include the authenticated node description extracted from the certificate during key establishment.

The topology information base is extended with a new Node Description Set, where each tuple contains a node address (ND_main_address) and the corresponding node description (ND_node_description).

5.3. Link Sensing

Due to our link encryption scheme, the process of link sensing and neighbour discovery is slightly different from the OLSR protocol. In OLSR, link sensing and neighbour discovery is performed through periodical HELLO message transmissions, containing known links to one-hop neighbours. However, our link encryption scheme requires the establishment of a shared secret key prior to any regular message processing. The process is initiated whenever a HELLO message is received and the sender and receiver do not share a key. After decryption, HELLO messages are processed in the same way as the original OLSR protocol, with some minor changes.

The interpretation of link codes is slightly changed from the original OLSR protocol. We regard a link to be symmetric (SYM_LINK) only if the nodes share a symmetric link and a key has been established. If the link has been detected but no key has been established, the link is considered asymmetric (ASYM_LINK). We also define a new neighbour type for the situation where the link is symmetric and the the node has only restricted access to the network (RES_SYM_NEIGH). Only nodes that have no access restrictions are eligible to be selected MPR and hence to take part in routing control message forwarding.

5.4. Topology Discovery

To discover nodes and links outside the 2-hop neighbourhood all nodes distribute Topology Control (TC) messages containing their advertised neighbour set and node description (see Figure 6). Similar to the HELLO message, TC messages are also encapsulated in the general encrypted message format (Figure 2) and broadcast to all neighbouring nodes with which the nodes share a symmetric link.

The node description is only authenticated to immediate neighbours (during key establishment) and not within the TC message. While such authentication is desirable, it would severely increase the control data overhead and possibly exhaust bandwidth resources. Therefore, whenever a TC message is received from a neighbour, the node verifies that the node description contained in the TC message is identical to the authenticated description received during key establishment. In the event of a mismatch, the TC message is silently discarded. After decrypting the TC message, it is processed according to the original OLSR protocol. If the message is considered valid (i.e. not processed before) the node description set is updated with the new description found in the TC message.

5.5. Routing Table Calculation

The routing table calculation is performed in the same manner as the OLSR protocol, with one slight difference resulting from the split network architecture described in section 4.3, where only nodes with no access restrictions are allowed to forward packets. The routing table calculation must take this into account in order to avoid paths containing limited access nodes.

Thus, in order to compute the routing table for node *X*, a shortest path algorithm is run on the directed graph containing:

1. The neighbour arcs, where *Y* is a symmetric neighbour of *X*;
2. The 2-hop neighbour arcs, where *Y* is a neighbour node with willingness different than WILL_NEVER and *Y*'s node description specifies no access restrictions and *Y*, *Z* belongs to the 2-hop neighbour set;
3. The topology arcs, where there exist an entry in the topology set with *V* as T_dest_addr and *U* as T_last_addr and *U*'s node description specifies no access restrictions.

6. GROUP COMMUNICATION

There exist numerous multicast routing protocols both for regular wired networks and mobile ad hoc networks. Common to all these protocols is that they require nodes to explicitly join a multicast group before data packets are routed to them. Thus multicast group addresses must be established and known to all participants before deploying the network, and all multicast groups must be maintained regardless of the frequency of use. Although this approach is desirable in many situations; greater flexibility is desired for first responders. Small tactical units may need to set up their own group communication in a dynamic fashion, not knowing beforehand who will participate. Similarly, command leader may need to distribute information to nodes with certain characteristics (e.g. medical personnel) occasionally. In both situations it is unpractical to require each participant to sign up for a multicast group before messages are sent.

Our multicast solution utilizes ideas from context-addressable messaging (CAM), by allowing nodes to select their multicast recipients based on node descriptions. While CAM allows nodes to define their own context descriptions, the node descriptions we propose are predefined and must be authenticated by a trusted entity. The node descriptions are assumed to be included in the node certificates and exchanged when nodes connect to each other. The descriptions are then further distributed throughout the network, through control messages of the underlying link state unicast routing protocol. We have defined three major classifications (i.e. descriptions) based on the organisation to which the nodes belong; fire, police and medic.

CAM relies on limited scope flooding for message propagation, which introduces serious transmission overhead and also makes it impossible to provide any confidentiality of messages. We therefore base our solution on explicit multicast routing, where each destination address is explicitly stated in the message header. In traditional IP networks, it is considered impractical (or even impossible) for the sender to individually address each recipient of a multicast transmission - with tens of thousands recipients of an IP-TV pay-per-view multicast of a major sports event the address overhead per frame would be prohibitive. However, in our case of a mobile ad hoc network for first responders in a crisis situation, the number of participating nodes is much lower – typically around 100 nodes or less.

Our proposed protocol is based on the Differential Destination Multicast (DDM) protocol, which allows consecutive messages to only include changes to the destination list and thereby greatly reducing the overhead of each transmitted package. The protocol is completely controlled by the multicast source and hence allows it to select destinations at its own discretion. The list of destination addresses is then partitioned according to their next-hop address gained from the routing table of the underlying unicast protocol and forwarded to the next-hop neighbours. Intermediate nodes receiving the message will similarly partition its destination list according to the next-hop address from its routing table and forward the message to these nodes. The process is then repeated until all destinations have been reached.

Multicast group selection and route calculation may be performed either reactively or proactively. The proactive approach allows nodes to adapt their group selection and calculated next-hop addresses whenever the underlying routing table changes. Thus, when a node needs to send a message to a proactively maintained group, the response time is greatly reduced. Although this approach claims more processing power for maintenance, it is particularly beneficial for nodes or groups where delay is of utter importance. For nodes or groups that do not require extreme response times, the reactive approach ensures that computational overhead is left at a minimum by only performing selection and route calculation whenever the node has data to send.

In Figure 7 we have illustrated how group communication could work in our scenario. Police officer p-1 wished to address all other police officers, and can conclude from the OLSR tables that she can reach p-10, p-11, p-12, p-21, p-22, and p-23. The message from p-1 would appear as shown in Table 5.

Figure 7. Group communication scenario

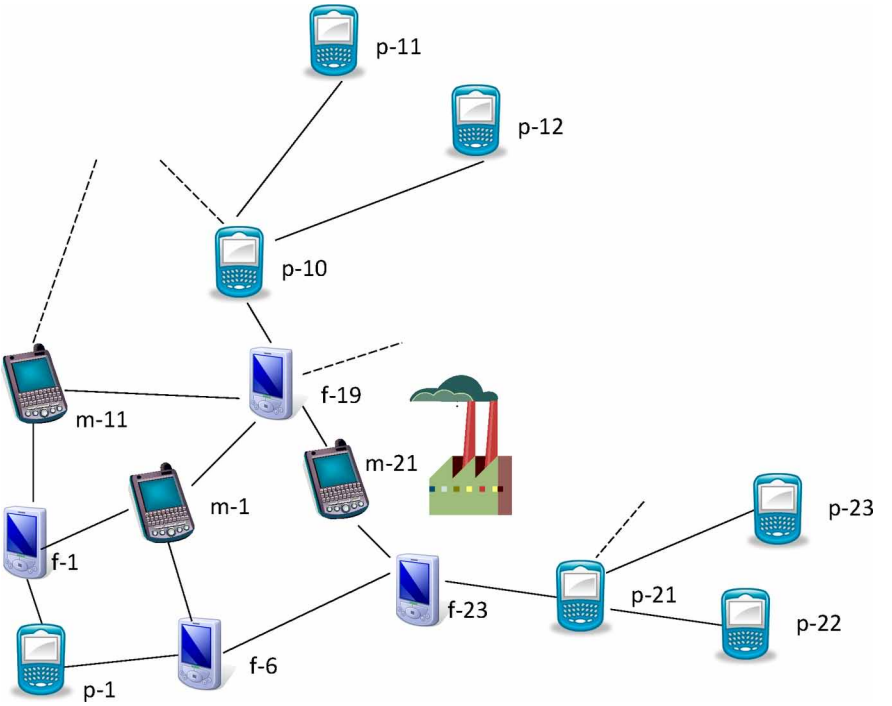


Table 5. The message from p-1

Next Hop	f-1
Destination	p-10, p-11, p-12
Keys	<key>
Next Hop	f-6
Destination	p-21, p-22, p-23
Keys	<key>
	Body

The node f-1 is given the responsibility to forward the message to p-10, p-11 and p-12, while f-6 must forward it to p-21, p-22 and p-23. For the sake of brevity, we will only follow the last next-hop. Node f-6 is not in the destination list and hence forwards the packet without changing the received list. The message from f-6 would appear as shown in Table 6.

Table 6. The message from f-6

Next Hop	f-23
Destination	p-21, p-22, p-23
Keys	<key>
	Body

Node f-23 is not in the destination list either, so it to forwards the packet without changing the destination list. The message from f-23 is shown in Table 7.

Node p-21 is in the destination list, causing it to remove itself from the list and partition the remaining nodes according to its routing table. The node processes the body and forwards the message with the content shown in Table 8.

Both node p-22 and p-23 are in the respective destination lists and hence remove themselves before processing the message body. Since the destination list is empty, no messages are forwarded. A similar chain will follow from the node f-1.

7. MULTICAST DATA TRANSMISSION

The multicast protocol builds on the Differential Destination Multicast (DDM) protocol. However, our approach makes use of the link keys of the routing protocol in order to provide confidentiality on a per-link basis. We call our version the Encrypted DDM (EDDM).

In our approach the protocol makes use of the node descriptions contained in the certificates for group management and hence there is no sign-up process required to join or leave multicast groups. Note that it is not mandatory to use the node descriptions to form groups. Destinations may also be selected independently or on the basis of other conditions by the source.

7.1. Packet Formats

The protocol uses two type of packets; data packets and synchronization packets. Data packets carry the actual content distributed between multicast group members, while synchronisation packets are used between neighbours to refresh the destination list if updates are missing.

7.2. Information Repository

EDDM may, as in the original DDM protocol, be run in a stateless mode requiring no additional storage by intermediate nodes. However, to reduce processing and transmission overhead we specify to use the soft-state mode where forwarding sets are stored by each intermediate node. The forwarding set

Table 7. The message from f-23

Next Hop	p-21
Destination	p-21, p-22, p-23
Keys	<key>
	Body

Table 8. Results after node p-21 processes the body and forwards the message

Next Hop	p-22
Destination	p-22
Keys	<key>
Next Hop	p-23
Destination	p-23
Keys	<key>
	Body

(FS) contains a per source subset indicated by where *k* is the address of a node for which this node is forwarding messages (i.e. an upstream node). The contains the following fields:

- **Grp address:** The multicast group address;
- **Upstream address k:** The address of the closest upstream node. This is the node EDDM messages are received from;
- **Sequence number:** A number used to identify missing packets with message updates;
- **DST addresses:** A list of addresses that the node is responsible for forwarding;
- **Time:** The time until the entry is removed.

The FS contains all the addresses that the node is responsible for forwarding to. However, since these nodes may have different next-hop destinations, the set is partitioned into Direction Sets (DS) based on the next-hop address (i.e. closest downstream node) retrieved from the routing table of the underlying unicast protocol. The sets are labelled, corresponding to the subset of destination addresses in FS that is forwarded to node *u*. The contains the following fields:

- **Downstream address u:** The address of the next-hop neighbour
- **Sequence number:** A number used to identify changes in the direction set.
- **Force refresh:** A flag indicating whether destination list should be refreshed for the downstream neighbour;
- **Forwarding DST addresses:** A list of addresses that use the downstream neighbour as next hop;
- **Time:** The time until the entry is removed.

7.3. Packet Generation

This section describes the process of generating data and synchronisation packets for the EDDM protocol.

Whenever a source has data to send to a multicast group, it performs the following steps to create and distribute the message:

1. Recipients are retrieved from the node descriptions provided by the unicast routing protocol and placed in the Forwarding Set (FS);
2. A one-time key is generated and used for encrypting the payload of the message;
3. Each destination address is added to the DS corresponding to the next-hop address retrieved from the routing table and initial sequence number is set to 1;
4. For each node u in the DS, an EDDM block is created with:
    a. One-time key set to the one previously generated;
    b. Type set to R-block;
    c. ESN is set to DSu sequence number;
    d. Number of addresses is set to reflect the number of addresses in DSu;
    e. Each DST address in DSu is appended;
    f. A CRC is computed on the basis of the content of the block;
    g. The content is encrypted using the link key corresponding to the neighbour address u, and placed behind its key identifier in the EDDM block;
5. The fields in the summary section are set as:
    a. Ver is set to 1;
    b. DSN is stepped once;
    c. Len is set to the number of EDDM blocks created;
    d. TOL is set to a node specific validity time;

   e. Group address is set to the selected group description, or set to zero if destinations are selected outside predefined groups;
   f. SRC address is set to this nodes address;
   g. MAC is computed based on the preceding header fields, the encrypted payload and the one-time key;
6. The packet is passed to the IP interface with destination address set to broadcast and sender address set to this node's address.

A synchronisation packet is generated by setting the multicast group description and source address of the multicast session to synchronise. The force refreshing flag is set to "force once" if the node is operating in soft state mode and "force always" if it is operating in stateless mode. The MAC is computed on the basis of header contents and the link key with the receiving upstream neighbour.

7.4. Packet Processing

Upon receiving an EDDM packet from a node k, the following procedure is done:

1. The packet is dropped if one of the following conditions are met:
   a. There is an entry in the duplicate set containing the multicast group description, source address and sequence number (DSN);
   b. There is no EDDM block containing the node's key identifier;
2. The EDDM block is decrypted and the CRC and MAC computed:
   a. If the computed CRC or MAC does not match the ones in the packet, it is dropped;
3. If the difference between the ESN and the sequence number in the stored is greater then 1, the packet is dropped and a synchronisation message is sent to k. (i.e. an update to the forwarding set is missing, full synchronisation is required);
4. The is updated with the received destination list according to the EDDM block type. That is; the DST addresses are added, removed or used to replace the if type is -, - or R-block, respectively. E-blocks require no changes;
5. The Sequence number of the is set to ESN;
6. A New Direction Set (NDS) is created based on the next-hop address of each entry in the FS;
7. For each, computed in the previous step:
   a. Create an empty EDDM block;
   b. Retrieve the link key and key identifier for u;
   c. If exists: Compute a D- or E type EDDM block depending on the difference between the and the:
      i. Otherwise: Create a new R type EDDM block containing the entire set;
8. If !=:
   a. Update sequence number; and
   b. Replace the with the:
      i. ESN is set to sequence number;
      ii. Number of addresses is set to reflect the number of addresses in;
      iii. Each DST address in is appended;
      iv. A CRC is computed on the basis of the content of the block;
      v. The content is encrypted using the link key corresponding to the neighbour address u, and placed behind its key identifier in the EDDM block;
      vi. The block is added to the outgoing EDDM packet;
9. The summary section of the EDDM packet is copied from the received packet and the Len field is updated to reflect the number of EDDM blocks in the packet;
10. The packet is passed to the IP interface with destination address set to broadcast and sender address set to this node's address.

Whenever a node receives a synchronisation packet, it verifies the MAC using the link key and updates the stored Force Refresh flag to value contained in the packet. The packet is never forwarded.

8. SIMULATION SPECIFICATION

For simplification, we do not actually perform cryptographic operations in the simulation software, but instead represent them by adding delays to the processing. Content of data packets are consequently not actually encrypted or signed. Instead, flags are used to indicate whether the signature is valid or not. This will allow us to consider the effect the solution has on throughput, delay and energy consumption. Furthermore, it allows us to see what kind of performance capacity one would require for the network to be of practical use.

The unpredictable behaviour of nodes, both in terms of traffic generation and movement is difficult to capture accurately. We assume that the different types of nodes (fire, police, and medic) will move in somewhat different patterns:

- Police nodes will be responsible for perimeter protection and hence will be close to stationary after being deployed;
- Fire nodes will be mainly located at the centre of the operation area and will move short distances relatively often;
- Medic nodes will locate and treat patients in the centre, before moving them to the perimeter medical base. Thus, they will exhibit a "back and forth" movement from the perimeter to the centre of the operation with pauses in between.

Although there are mobility models that capture some of the identified behaviours, none of them do so for all. Also, combining different mobility models for the same scenario is troublesome, especially when location restrictions apply (e.g. keep police nodes at the perimeter). Our decision was therefore to use the random waypoint model (Bettstetter, Resta, & Santi, 2003) with different parameter settings for all types of nodes. In the random waypoint model a node selects a random destination and random node speed (within a preconfigured interval). Once the node reaches its destination, it waits for a predetermined period of time known as the pause time. In order to allow for the simulation to vary speed and pause time, we define scaling factors for each node type (see Table 9). The table specifies that if node speed interval is set to, and pause time set to t=c, then applying the scaling factors yields speed interval (2a,4b) and pause time (4c) for medic nodes.

Traffic sources are assumed to be both burst type (on-off) and constant bit rate (CBR), in order to model both the distribution of short messages and streaming of video. In order to allow for variation of the number of sources, we define a ratio of burst type/CBR sources to be 4:1. That is, 80% of sources will generate burst type traffic, while the remaining 20% will generate CBR traffic.

8.1. NS-2 Simulation Environment

The Network Simulator II (NS-2) (*ns-2 Simulator*,)) is a popular discrete event simulator targeted for network research. It provides a comprehensive collection of libraries for modelling and simulating

Table 9. Scaling factors for node types

Node Type	Speed Interval		Pause Time (t)
Police	1	1	10
Fire	1	2	1
Medic	2	4	4

MANETs, including mobility models, propagation models, routing protocols and data sources. The class hierarchy is implemented using C++ while simulation scripts are written using the Object Tcl (OTcl) scripting language. The rationale for using two different languages is that protocol implementation requires a powerful language generating fast executable code. Defining simulations on the other hand, requires fast turn-around time for making adjustments (i.e. no re-compilation required). Hence, C++ and OTcl are used for different purposes, exploiting their strengths.

Although NS-2 provides a great number of protocols and models, it does not provide all. We will base our implemented protocol on the OLSR implementation contributed by the University of Murcia (UM-OLSR) (*UM-OLSR*,)).

8.2. Simulation Parameters

The scenario described earlier is placed in a virtual grid of 1000x1000 meters with nodes' transmission range set to 100 meters. The number of nodes is fixed to 20 throughout the simulation.

Node movement is modelled through the random waypoint model with node speed varied in the interval 1m/s to 10m/s, and the pause time varied from 30 to 300 seconds. The scaling factor of the different node types (see Table 9) is applied to these figures to compute the speed and pause time for each type of node.

The encryption delay is specified using benchmark results from the Crypto++ cryptographic library. For symmetric key encryption the delay is computed based on the number of bytes to encrypt, while asymmetric encryption the delay is specified per operation. The chosen symmetric key algorithm is the Advanced Encryption Standard (AES) using 128-bit keys, while the hash algorithm is the Secure Hash Algorithm (SHA-256) with 32-byte hash. For asymmetric encryption and signature generation we have chosen Elliptic Curve Cryptography (ECC) using 256-bit keys. The estimated delay and encryption speed is given in Table 10.

8.3. Metrics

The performance evaluation of our proposed protocol is done using the following metrics:

- **Packet delivery ratio:** The fraction of data packets sent that are actually received at the destination node. Packets are counted by the source and final destination such that intermediate forwarding is not included;
- **End to end delay:** The average delay of data packets measured from the time the packet is sent to the time it is received by the destination node;
- **Control message ratio:** The fraction of control messages sent per data packet. Packets are counted on a per link basis, such that a single packet traveling two hops will count as two packets.

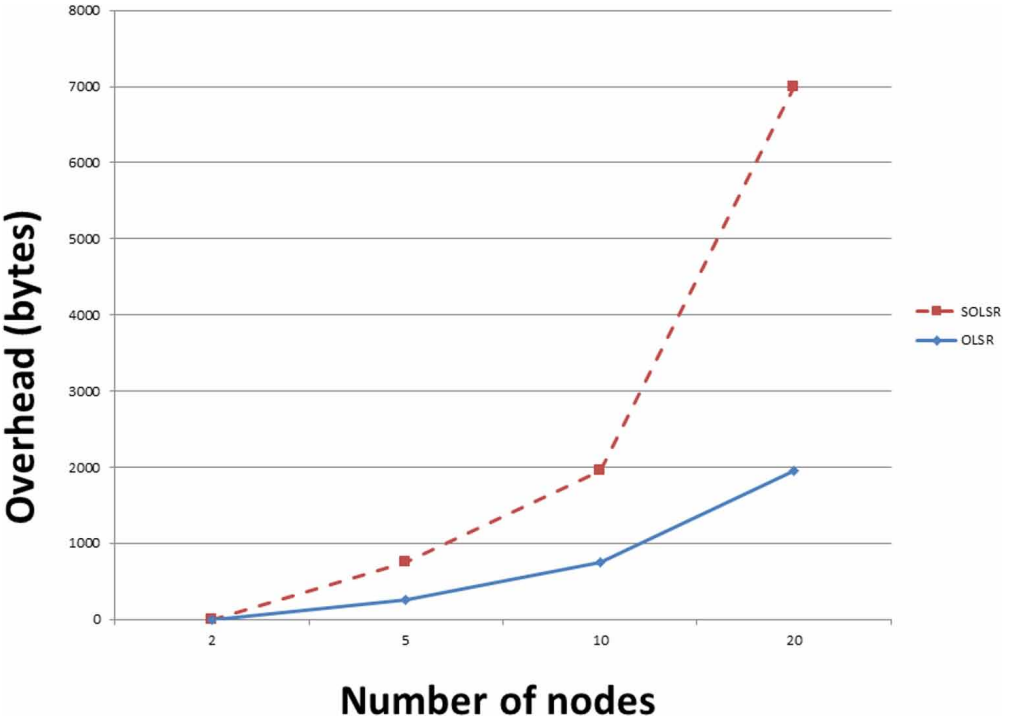
8.4. Simulation Results

Our simple simulations show that while encryption introduces a significant overhead in the routing traffic (see the SOLSR graph in Figure 8), it does not appreciably influence the dropping of packets

Table 10. Throughput and delay due to encryption

Algorithm	Throughput MB/s	Delay per Operation ms
Symmetric key (AES)	84	-
Hash function (SHA-256)	81	-
Asymmetric key encryption (ECC)	-	7.15
Asymmetric key signature (ECC)	-	7.45

Figure 8. Routing overhead (bytes) as function of number of nodes



as traffic increases compared to regular OLSR or AODV (see Figure 9). For more details regarding the simulation results, see (Karim, 2009).

9. DISCUSSION

In Table 4 we listed the relevant security requirements. We will now explain to what level our proposed solution fulfils these requirements, and the trade-offs that have been made in the design of the solution. An overview of requirement fulfillment can be found in Table 11.

Requirements R12 and R13 relate to the choice of encryption algorithm and key length, and this is strictly speaking dependent on how the prototype eventually will be implemented, but we have suggested using AES with 128-bit keys, which would fulfill these requirements.

Requirements R20 and R21 are partly handled by the underlying network protocol, in that transmission errors are detected by the IP CRC, and in addition we provide a MAC which protects the integrity of transmitted information.

Our communication scheme which relies on possession of a valid certificate and knowledge of corresponding private key fulfills requirement R8, and since all data on the link between nodes is encrypted, requirement R10 is also fulfilled. Requirement R14 is partly fulfilled, as we have provided mechanism for key exchange, and have outlined a mechanism for certificate revocation, but complete fulfillment of this requirement and specification of how it will be implemented on the handheld units is considered future work.

User (and node) identities are handled by use of x.509 certificates, and since these cannot be changed without invalidating the certificate, users can neither modify their own access rights nor pretend to be someone else without access to the other person's private key; this fulfills requirements R26 and R27. Our proposed PKI scheme also supports decentralised access control (R28), since we

Figure 9. Packet drop rate (packets/sec)

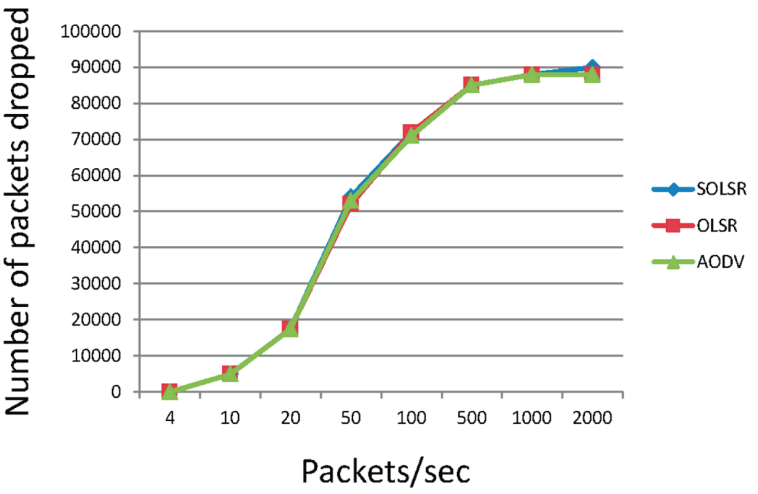


Table 11. Fulfillment of security requirements

ID	Requirement Title	Fulfilment
R8	Network access	OK. Requires valid certificate and knowledge of corresponding private key.
R9	Strength network access mechanism	Relevant, but is too early to test for this.
R10	Link confidentiality	Provides encryption on the link between nodes.
R11	End-to-end confidentiality	Not provided.
R12	Encryption algorithms	Not decided, but has suggested AES which fulfils this requirement.
R13	Encryption keys	Not decided, but has suggested 128-bit keys with AES, which fulfils this requirement.
R14	Key management	Partly fulfilled. Have provided mechanism for key exchange, and have outlined mechanism for revocation, but complete fulfilment of this requirement is considered future work.
R16	Communication of access credentials	Do not send access credentials with the current solutions.
R20	Transmission errors	OK. Handled by underlying protocol. In addition, we provide MAC.
R21	Integrity of transmitted information	OK. Provide MAC.
R23	Detection of misbehaving nodes	This is not solved. Most anomaly detection mechanisms give a lot of false positives, and may therefore not be suitable for emergency and rescue operations.
R26	Identities vs. access rights	OK. Access rights are dependent on valid certificate. If changing the certificate, it becomes invalid.
R27	Identities and spoofing	OK. It is not enough to get access to another user's certificate. An attacker would also need to have access to the corresponding private key.
R28	Support participation and collaboration	OK. Use one certificate authority per district. Personnel from other districts get 24-hour certificates. This means that they will be able to use the network, but not take part in routing.
R29	Decentralised access control	OK. Central authorities only needed prior to system use.

only require a regional authority to issue certificates prior to system use, and not during deployment of the ad hoc network. This also supports participation and collaboration (R28), where personnel from other districts, NGOs, volunteers, etc. can be issued with 24-hour certificates. This means that they will be able to use the network, but not take part in routing.

Requirement R11 on end-to-end confidentiality of messages depends on the applications that would run on the handheld devices. Our link encryption scheme does not provide end-to-end security since intermediate nodes are able to decrypt, and possibly change the content of the message, without the receiver noticing. However, the distribution of routing information is mainly done by broadcasting messages, which makes end to end confidentiality meaningless. To allow for end-to-end message authentication would require either the full distribution of certificates (periodically) or dynamic establishment of symmetric keys between all nodes in the network. In either case, the resource consumption is significant and also scalability issues would arise as the number of nodes in the network increases. Another possibility would be to have all nodes share a single network wide symmetric key. However, this approach makes key management and key agreement a particularly demanding task. Key renewal may then render the network inoperable for a period of time (until the new key is fully distributed), which is considered unacceptable for emergency and rescue operations. As availability in most cases are considered more important than confidentiality (Objective 2 in Table 3), an end-to-end security solution was not considered feasible for our purposes.

Although we had initially intended to explore possibilities for new mechanisms for detection of misbehaving nodes, we chose to abandon this particular goal, leaving requirement R23 unfulfilled. This was decided not only because of time constraints, but also because of input from the literature; such as Zapata and Asokan (Zapata & Asokan, 2002) who argue that compromised nodes are only an issue in military scenarios. Furthermore, the current state-of-the art in misbehaviour detection is exclusively anomaly-based, which carries with it a high rate of false positives. It is clearly unacceptable to cut a fire-fighter off from the OASIS ad hoc network just because her communication pattern is somewhat unusual.

There is considerable risk involved in admitting actors who are not first responders to the network through temporary access. However, by restricting the participation so as to not interfere with the routing protocol operation, the associated risk is greatly reduced. It is also assumed that dynamically granting of access is required in order to take full advantage of the MANET potential.

Certificate revocation is handled through the distribution of Certificate Revocation Lists (CRLs) to authorised nodes. As long as nodes only receive CRLs from the certificate issuer, the task is fairly easy. However, as certificates are cross-signed by other issuers, CRL distribution becomes increasingly hard. By limiting the validity time of the certificates, the size and complexity of the CRL is greatly reduced.

Our simulation shows that although the routing overhead of our SOLSR solution is significant with increasing number of nodes, there is no practical difference in packet drop rate when compared to stabdard OLSR and AODV.

10. CONCLUSION

We have presented a secure ad hoc network scheme for first responders in a crisis situation that provides access control and confidentiality of information, and also offers a group communication mechanism. This scheme balances the need for protection with requirements for availability and efficiency, and takes advantage of the hierarchical structure of such operations.

REFERENCES

Bettstetter, C., Resta, G., & Santi, P. (2003). The node distribution of the random waypoint mobility model for wireless ad hoc networks. *IEEE Transactions on* Mobile Computing, *2*, 257–269.

Bradley, K., Cheung, S., Puketza, N., Mukherjee, B., & Olsson, R. (1998). Detecting disruptive routers: A distributed network monitoring approach. *IEEE Network*, *12*(5), 50–60. doi:10.1109/65.730751

Buchegger, S., & Boudec, J.-Y. L. (2002). Performance analysis of the confidant protocol. In *Proceedings of the 3rd ACM international symposium on mobile ad hoc networking & computing* (p. 226-236). Lausanne, Switzerland: ACM. doi:10.1145/513800.513828

Caralli, R. A., Stevens, J. F., Young, L. R., & Wilson, W. R. (2007). Introducing OCTAVE Allegro: Improving the Information Security Risk Assessment Process. CMU/SEI. Available from http://www.cert.org/archive/pdf/07tr012.pdf

Clausen, T., & Herberg, U. (2010a, September). Router and link admittance control in the optimized link state routing protocol version 2 (olsrv2). In Proceedings of the 2010 international conference on Network and system security (NSS). Academic Press.

Clausen, T., & Herberg, U. (2010b). Vulnerability analysis of the optimized link state routing protocol version 2 (olsrv2). In Proceedings of the 2010 IEEE international conference on Wireless communications, networking and information security (WCNIS) (p. 628-633). IEEE. doi:10.1109/WCINS.2010.5544732

Clausen, T., & Jacquet, P. (Eds.). (2003). Rfc3626: Optimized link state routing protocol (olsr). IETF, The Internet Society. Retrieved from http://www.ietf.org/rfc/rfc3626.txt

Cooper, D. Santesson, Farrell, Boeyen, Housley, R., & Polk, W. (2008). Rfc5280: Internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile. IETF, The Internet Society. Retrieved from http://www.ietf.org/rfc/rfc5280.txt

Dahill, B., Levine, B. N., Royer, E., & Shields, C. (2001, August). A secure routing protocol for ad hoc networks. Electrical Engineering and Computer Science, University of Michigan.

Dearlove, C. (2004). OLSR Simulation, Implementation and Ad Hoc Sensor Network Application. In *Olsr interop & workshop*. Retrieved from http://olsrinterop.free.fr/papers/BAE-OLSR-experience-paper.pdf

Haas, Z. (1997, October). A new routing protocol for the reconfigurable wireless networks. In *Proceedings of 6th IEEE international conference on universal personal communications, IEEE ICUPC'97* (Vol. 2, p. 562-566). IEEE. doi:10.1109/ICUPC.1997.627227

Haley, C., Laney, R., Moffett, J., & Nuseibeh, B. (2008). Security requirements engineering: A framework for representation and analysis. *IEEE Transactions on Software Engineering*, *34*(1), 133–153. doi:10.1109/TSE.2007.70754

Herberg, U., Clausen, T., & Milan, J. (2010). Digital signatures for admittance control in the optimized link state routing protocol version 2. In Proceedings of the 2010 international conference on Internet technology and applications (p. 1 -4). Academic Press. doi:10.1109/ITAPP.2010.5566285

Hu, Y.-C., Perrig, A., & Johnson, D. B. (2005). Ariadne: A secure on-demand routing protocol for ad hoc networks. *Wireless Networks*, *11*(1-2), 21–38. doi:10.1007/s11276-004-4744-y

Hughes, J., Aura, T., & Bishop, M. (2000). Using conservation of flow as a security mechanism in network protocols. In Proceedings of the 2000 IEEE symposium on Security and privacy S&P 2000 (p. 132-141). Academic Press. doi:10.1109/SECPRI.2000.848451

Jaatun, M. G., & Tøndel, I. A. (2008). Covering your assets in software engineering. In *Proceedings of the Third international conference on availability, reliability and security (ARES 08)* (pp. 1172–1179). Academic Press. doi:10.1109/ARES.2008.8

Jacquet, P., Muhlethaler, P., Clausen, T., Laouiti, A., Qayyum, A., & Viennot, L. (2001). Optimized link state routing protocol for ad hoc networks. In Multi topic conference, 2001. IEEE INMIC 2001. technology for the 21st century. proceedings. IEEE international (p. 62-68). IEEE. .2001.995315 doi:10.1109/INMIC.2001.995315

Johnson, D., Hu, Y., & Maltz, D. (2007). Rfc4728: The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. IETF. Retrieved from http://www.ietf.org/rfc/rfc4728.txt

Johnson, D., & Maltz, D. (1996). *Dynamic source routing in ad hoc wireless networks* (pp. 153–181). Kluwer Academic Publishers. doi:10.1007/978-0-585-29603-6_5

Karim, S. M. A. (2009). Simulation of New Security Elements in an Ad Hoc Network. Unpublished master's thesis, NTNU/KTH.

Marchang, N., & Tripathi, R. (2007). A game theoretical approach for efficient deployment of intrusion detection system in mobile ad hoc networks. In Proceedings of the international conference on Advanced computing and communications ADCOM 2007 (pp. 460-464). doi:10.1109/ADCOM.2007.58

Marti, S., Giuli, T. J., Lai, K., & Baker, M. (2000). Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of the 6th annual international conference on mobile computing and networking* (p. 255-265). Boston, MA: ACM. doi:10.1145/345910.345955

Meissner, A., Luckenbach, T., Risse, T., Kirste, T., & Kirchner, H. (2002). Design challenges for an integrated disaster management communication and information system. In *Proceedings of the first IEEE workshop on disaster recovery networks (DIREN 2002)* (Vol. 24).

Michiardi, P., & Molva, R. (2002). Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Advanced communications and multimedia security: IFIP TC6/TC11 sixth joint working conference on communications and multimedia security*, Portoroz, Slovenia, September 26-27 (p. 107). Academic Press.

ns-2 Simulator. (n.d.). Retrieved from http://nsnam.isi.edu/nsnam/index .php/Main_Page

Otrok, H., Mohammed, N., Wang, L., Debbabi, M., & Bhattacharya, P. (2008, March). A game-theoretic intrusion detection model for mobile ad hoc networks. *Computer Communications*, *31*(4), 708–721. doi:10.1016/j.comcom.2007.10.024

Papadimitratos, P., & Haas, Z. (2006). Secure data communication in mobile ad hoc networks. *IEEE Journal on* Selected Areas in Communications, *24*(2), 343–356.

Papadimitratos, P., & Haas, Z. J. (2002). Secure routing for mobile ad hoc networks. In Proceedings of the SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002, 27–31. 10.1.1.12.2420

Papadimitratos, P., & Haas, Z. J. (2003). Secure link state routing for mobile ad hoc networks. In *Proceedings of the 2003 symposium on applications and the internet workshops (Saint'03 workshops)* (p. 379). IEEE Computer Society. doi:10.1109/SAINTW.2003.1210190

Perkins, C. E., Belding-Royer, E. M., & Das, S. (2003). Rfc3561: Ad hoc on-demand distance vector (AODV) routing. IETF. Retrieved from http://www.ietf.org/rfc/rfc3561.txt

Perkins, C. E., & Royer, E. M. (1999). Ad-hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications* (pp. 90–100). IEEE.

Perrig, A., Canetti, R., Tygar, D., & Song, D. (2002). The tesla broadcast authentication protocol. *RSA CryptoBytes*, *5*, 2–13.

Pužar, M., Plagemann, T., & Roudier, Y. (2008). Security and privacy issues in middleware for emergency and rescue applications. In *Proceedings of the Second International Conference on, Pervasive Computing Technologies for Healthcare PervasiveHealth 2008* (pp. 89-92). Academic Press. doi:10.1109/PCTHEALTH.2008.4571037

Ramkumar, M., & Memon, N. (2005). An efficient key predistribution scheme for ad hoc network security. *IEEE Journal on Selected Areas in Communications*, *23*(3), 611–621. doi:10.1109/JSAC.2004.842555

Sanzgiri, K., Dahill, B., Levine, B. N., Shields, C., & Belding-Royer, E. M. (2002). A secure routing protocol for ad hoc networks. In *Proceedings of the 10th IEEE international conference on network protocols* (pp. 78–89). IEEE Computer Society. doi:10.1109/ICNP.2002.1181388

Sanzgiri, K., LaFlamme, D., Dahill, B., Levine, B., Shields, C., & Belding-Royer, E. (2005). Authenticated routing for ad hoc networks. *IEEE Journal on Selected Areas in Communications*, *23*(3), 598-610. doi:10.1109/JSAC.2004.842547

Saxena, N., Tsudik, G., & Yi, J. H. (2007). Threshold cryptography in P2P and MANETs: The case of access control. *Computer Networks*, *51*(12), 3632–3649. doi:10.1016/j.comnet.2007.03.001

Schneier, B. (1999, December). Attack trees: Modeling security threats. *Dr. Dobb's Journal*.

Tøndel, I. A., Jaatun, M. G., & Meland, P. H. (2008). Security requirements for the rest of us: A survey. *IEEE Software*, *25*(1), 20–27. doi:10.1109/MS.2008.19

Tøndel, I. A., Jaatun, M. G., & Nyre, Å. A. (2009). Security requirements for MANETs used in emergency and rescue operations. In *Proceedings of the 1st international workshop on security and communications systems*. NTNU.

UM-OLSR. (n.d.). Retrieved from http://masimum.dif.um.es/?Software:UM-OLSR

Wang, M., Lamont, L., Mason, P., & Gorlatova, M. (2005). An effective intrusion detection approach for OLSR manet protocol. In *Proceedings of the 1st IEEE ICNP workshop on Secure network protocols (NPSEC)* (p. 55-60). IEEE. doi:10.1109/NPSEC.2005.1532054

Wrona, K. (2002, September). Distributed security: Ad hoc networks & beyond. In *Ad hoc network security pampas workshop* (pp. 16–17). London: RHUL.

Wu, B., Chen, J., Wu, J., & Cardei, M. (2007). A survey of attacks and countermeasures in mobile ad hoc networks. In Wireless network security (pp. 103–135). Springer US. doi:10.1007/978-0-387-33112-6_5

Zapata, M. G., & Asokan, N. (2002). Securing ad hoc routing protocols. In *Proceedings of the 1st acm workshop on wireless security* (pp. 1–10). Atlanta, GA: ACM.

Zhou, L., & Haas, Z. (1999). Securing ad hoc networks. *IEEE Network*, *13*(6), 24–30. doi:10.1109/65.806983

Martin Gilje Jaatun is a Senior Scientist at SINTEF Digital in Trondheim, Norway. He graduated from the Norwegian Institute of Technology (NTH) in 1992, and received the Dr. Philos degree in critical information infrastructure security from the University of Stavanger in 2015. He is an adjunct professor at the University of Stavanger, and was Editor-in-Chief of the International Journal of Secure Software Engineering (IJSSE). Previous positions include scientist at the Norwegian Defence Research Establishment (FFI), and Senior Lecturer in information security at the Bodø Graduate School of Business. His research interests include software security, security in cloud computing, and security of critical information infrastructures. He is vice chairman of the Cloud Computing Association (cloudcom.org), vice chair of the IEEE Technical Committee on Cloud Computing (TCCLD), an IEEE Cybersecurity Ambassador, and a Senior Member of the IEEE.