Secure Message Delivery Games for Device-to-Device Communications

Emmanouil Panaousis[1], Tansu Alpcan[2],
Hossein Fereidooni[3], and Mauro Conti[3]

[1] Queen Mary University of London
e.panaousis@qmul.ac.uk
[2] The University of Melbourne
tansu.alpcan@unimelb.edu.au
[3] University of Padua
{hossein,conti}@math.unipd.it

Abstract. Device-to-Device (D2D) communication is expected to be a key feature supported by next generation cellular networks. D2D can extend the cellular coverage allowing users to communicate when telecommunications infrastructure are highly congested or absent. In D2D networks, any *message delivery* from a *source* to a *destination* relies exclusively on intermediate devices. Each device can run different kinds of *mobile security software*, which offer protection against viruses and other harmful programs by using real-time scanning in every file entering the device. In this paper, we investigate the best D2D network path to deliver a potentially malicious message from a source to a destination. Although our primary objective is to increase security, we also investigate the contribution of energy costs and quality-of-service to the path selection. To this end, we propose the *Secure Message Delivery* (SMD) protocol, whose main functionality is determined by the solution of the *Secure Message Delivery Game* (SMDG). This game is played between the *defender* (i.e., the D2D network) which abstracts all legitimate network devices and the *attacker* which abstracts any adversary that can inject different malicious messages into the D2D network in order, for instance, to infect a device with malware. Simulation results demonstrate the degree of improvement that SMD introduces as opposed to a shortest path routing protocol. This improvement has been measured in terms of the defender's expected cost as defined in SMDGs. This cost includes security expected damages, energy consumption incurred due to messages inspection, and the quality-of-service of the D2D message communications.

Keywords: game theory, security, device-to-device communications.[4]

1 Introduction

Nowadays, the vast demand for anytime-anywhere wireless broadband connectivity has posed new research challenges. As mobile devices are capable of communicating in both cellular (e.g., LTE) and unlicensed (e.g., IEEE 802.11) spectrum,

[4] The original publication is available at www.link.springer.com.

the Device-to-Device (D2D) networking paradigm has the potential to bring several immediate gains. Networking based on D2D communication [1–4] not only facilitates wireless and mobile peer-to-peer services but also provides energy efficient communications, locally offloading computation, offloading connectivity and high throughput.

Another emerging feature of D2D is the establishment and use of multi-hop paths to enable communications among non-neighboring devices. In multi-hop D2D communications, messages are delivered from a source to a destination via intermediate devices, independently of operators' networks. Relay by device has been proposed by the Telecommunication Standardization Advisory Group (TSAG) in the International Telecommunication Union Telecommunication Sector (ITU-T).

A key question in *multi-hop D2D networks* is, which route should the originator of a message choose to send it to an intended destination? To motivate the application of our model, we emphasize in the need for *localized applications.* In particular, these applications run in a collaborative manner by groups of devices at a location where telecommunications infrastructures:

– are not presented at all, e.g., underground stations, airplanes, cruise ships, parts of a motorway, and mountains;
– have collapsed due to physical damage to the base stations or insufficient available power, e.g., areas affected by a disaster such as earthquake;
– are over congested due to an extremely crowded network, e.g., for events in stadiums, and public celebrations.

Furthermore, relay by device can be leveraged for commercial purposes such as advertisements and voucher distributions for instance in large shopping centers. This is considered a more efficient way of promoting businesses than other traditional methods such as email broadcasting and SMS messaging due to the immediate identification of the clients in a surrounding area. Home automation and building security are another two areas that multi-hop message delivery using D2D communications is likely to overtake our daily life in the near future. Lastly, multi-hop D2D could be leveraged towards the provision of anonymity against cellular operators as proposed in [12].

Due to the large number of areas D2D communications are applicable to, devices are likely to be an ideal target for attackers. Malware for mobile devices evolves in the same trend as malware for PCs. It can spread for instance through a Multimedia Messaging System (MMS) with infected attachments, or an infected message received via Bluetooth aiming at stealing users' personal data or credit stored in the device. An example of a well-known worm that propagates through Bluetooth was Cabir [7], which consists of a message containing an application file called `caribe.sis`. Mabir, a variant of Cabir, was spread also via MMS by sending out copies of itself as a .sis file. Van Ruitenbeek et al. [8] investigated the effects of MMS viruses that spread by sending infected messages to other devices. In addition, Bose and Shin [9] examined the propagation of malware that spread via SMS or MMS messages and short-range radio interfaces while Polla et al. [10] have made a thorough survey on mobile malware.

1.1 Contributions

In this paper, we assume that each device has some host-based intrusion detection capabilities (e.g., antivirus). Therefore, a device would be able to detect malicious application-level events as in [11]. We assume that each device has its own detection rate which contributes towards the overall detection rate of the routes that this device is on. To increase the level of security of a message delivery, the route with the highest detection capabilities must be selected to relay the message to the destination. Apart from security, energy consumption is of crucial importance because devices (e.g., smartphones) usually impose strict energy constraints. This becomes more important due to the limited CPU and memory capabilities that devices have, which entail higher energy cost as opposed to cases where no message inspection takes place.

In this paper, we propose the *Secure Message Delivery* (SMD) protocol. The primary objective of this protocol is to choose the most secure path to deliver a message from a sender to a destination in a multi-hop D2D network. SMD can work on top of underlying physical and MAC layer protocols [5, 6]. Apart from security, SMD respects the energy costs and Quality-of-Service (QoS) of each route. This happens by giving certain weights to each of the involved parameters (security, energy, QoS) with more emphasis to be put on security.

We formulate *Secure Message Delivery Games* (SMDGs) in order to derive an optimal behavior for the SMD. In these games, one or more adversaries, abstracted by the *attacker*, aim at increasing the security damage, incurred to the defender (i.e., network), by injecting malicious messages into the D2D network. On the other hand, the defender chooses the "best route" for message delivery. In SMDGs, the utility of the defender is influenced by: (i) the probability of the delivered message to be correctly classified as malicious or benign before it is delivered to the intended destination; (ii) the *energy cost* associated with *message forwarding*, and *message inspection* on relay devices during message delivery; and (iii) the QoS of the message communications on the chosen D2D path.

The remainder of this paper is organized as follows. Section 2 summarizes the most relevant related work within the intersection of game theory, security and mobile distributed networking. In Section 3 we present the system model whilst Section 4 formulates the SMDGs and it provides their solutions. In Section 5 the SMD routing protocol for D2D networks is described. We present some preliminary simulation results in Section 6 for different number and types of malicious messages distributions, and different D2D network profiles. Section 7 concludes this paper by summarizing its main contributions, limitations and highlighting our plans for future work.

2 Related Work

The papers we discuss in this section have used game theory in favor of security in mobile distributed networks. These address different challenges including *secure routing* and *packet forwarding* [13, 27, 29–31], *trust establishment* [15, 27], *intrusion detection* [15,21,23,24,26], and *optimization of energy costs* [17–19,22].

In [27], Sun et al. presented an information theoretic framework to evaluate trustworthiness in ad hoc networks and to assist malicious detection and route selection. According to their mechanism, a source node chooses a route to send a message to a destination by looking up the packet-forwarding nodes' trustworthiness, and selecting the most trustworthy route. Yu et al. examined, in [29], the dynamic interactions between "good" nodes and adversaries in mobile ad hoc networks (MANETs) as secure routing and packet forwarding games. They have derived optimal defense strategies and studied the maximum potential damage, which incurs when attackers find a route with maximum number of hops and they inject malicious traffic into it. Extension of the previous work is presented in [31], where Yu and Liu examined the issues of cooperation stimulation by modeling the interactions among nodes as multi-stage secure routing and packet forwarding games. In [30], the same authors focused on a two-player packet forwarding game stating that nodes must not help their opponents more than their opponents has helped them back. Felegyhazi et al. have studied in [13] the Nash equilibria of packet forwarding strategies with TFT (Tit-For-Tat) punishment strategy in a repeated game.

In [17], the authors presented a Bayesian hybrid detection approach to preserve the energy spent for intrusion detection. In the proposed static game, the defender fixes the prior probabilities about the types of his opponent. The dynamic game allows the defender to update his belief about his opponent's type based on new observed actions and the game history. The authors formulated the attacker/defender game model in both static and dynamic Bayesian game contexts, and investigated the equilibrium strategies of the two players. Lui et al. in [18] put forwarded a more comprehensive game framework and they used cross-feature analysis on feature vectors constructed from the training data to determine the actions of a potential attacker in each stage game. They proposed to use the equilibrium monitoring strategies to operate between a lightweight IDS and a heavyweight IDS. In [19], Marchang et al. proposed a game-theoretic model of IDS for MANETs. They have used game theory to model the interactions between the IDS and the attacker to determine whether it is essential to always keep the IDS running without impacting its effectiveness in a negative manner.

In [23], Patcha et al. provided a mathematical framework to analyze intrusion detection in MANETs. They model the interaction between an attacker and an individual node as a two player non-cooperative signaling game. The sender could be a *regular* or a *malicious* node. A receiving node equipped with an intrusion detection system (IDS) detects a "message/attack" with a probability depending on his belief, and the IDS updates the beliefs according to this message. However, it is not explained how the IDS updates the beliefs according to this message. The same authors have also reinforced the suitability of using game theory for modeling intrusion detection by giving a theoretically consistent model in [24]. They used the concept of multi-stage dynamic non-cooperative game with incomplete information to model intrusion detection in a network that uses host-based IDSs. A cooperative approach is proposed in [21] by

Otrok et al. to detect and analyze intrusions in MANETs. The authors used the Shapley value to analyze the contribution of each node to the network security and proposed pre-defined security classes to decrease false positives. They also considered cache poisoning and malicious flooding attacks. Santosh et al. in [26], employed game theoretic approaches to detect intrusions and identify anomaly behaviors of nodes in MANETs. The authors aimed at building an IDS based on a cooperative scheme to detect intrusions in MANETs using game theoretic concepts.

In [15], Cho et al. developed a mathematical model to analyze and reveal the optimal rate to perform intrusion detection related tasks. They enhanced the *system reliability of group communication systems* in MANETs given information regarding operational conditions, system failures, and attacker behaviors. They have also discussed to prolong the system lifetime and cope with inside attacks. They proposed that intrusion detection should be executed at an optimal rate to maximize the mean time to failure of the system.

Finally in [22], Panaousis and Politis present a routing protocol that respects the energy spent by intrusion detectors on each route and therefore prolonging network lifetime. However, this protocol does not investigate the effect of different malicious messages. It rather takes a simplistic approach according to which the attacker either attacks or not a route.

As we have seen in this section, a substantial amount of game theoretic models for security in distributed mobile networks (e.g., mobile ad hoc networks) have been proposed in the literature. However, none of them addresses all aspects of security, QoS and energy efficiency at the same time. Motivated by this observation, our work contributes to the existing literature by bringing together these three aspects, under a generic but also customizable model provided by the SMDGs. Furthermore, our work defines the adversary's pure strategies to be a set of different malicious messages. And this is not an aspect of investigation of papers identified by our literature review. It is worth noting that we consider the work undertaken, in this paper here, as the first step towards a more complex and advanced game theoretic secure message delivery protocol for D2D networks.

3 System Model

This section presents our system model and its different components. We assume a multi-hop Device-to-Device (D2D) communication network that extends a cellular network (e.g., LTE Advanced) as illustrated in Fig. 1.

Data transmission takes place in the application layer in the form of data units called *messages*. Any device can be the source (`s`) of a message and each message has a final destination (`d`). When `d` is not within the transmission range of `s`, a route must be established to allow message delivery. Therefore, there is an apparent need for the devices to collaborate to relay messages towards `d`.

We refer to the i-th mobile device by s_i, and define the set of all legitimate mobile devices in a mobile network as $S \triangleq \{s_i\}$. When the l-th type of message, denoted by m_l, has to be delivered to a destination device (`d`), a route must be chosen by `s` to serve that purpose. Formally, we denote route j by r_j. The

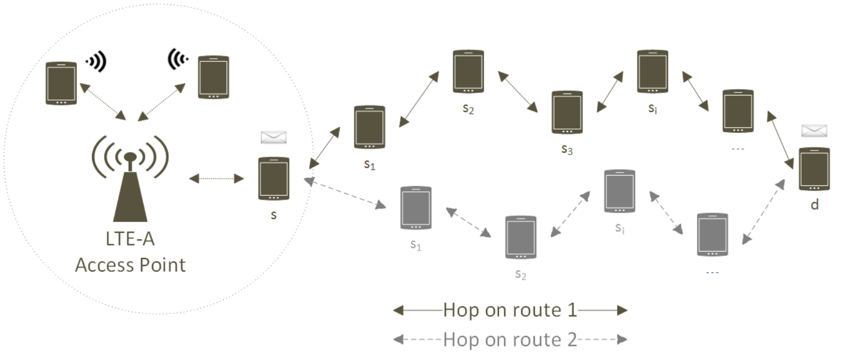

Fig. 1. Example of a D2D network.

devices on r_j must forward m_l towards `d`. We define the set of all routes from `s` to `d` as $R \triangleq \{r_j\}$, and the set of all devices that constitute r_j is expressed by S_j.

We denote the set of all different types of messages[5] by $\mathcal{M}$. This equals the union of the set of all malicious undetected messages ($\mathcal{M}_m$), and the set of all benign messages ($\mathcal{M}_b$). Therefore, $\mathcal{M} \triangleq \mathcal{M}_m \cup \mathcal{M}_b$. An *attack* is defined as the attempt of the attacker to harm `d` through the delivery of a malicious message. When m_l stays undetected prior to be delivered to `d`, we say that it causes harm $\mathcal{H}_l$, which is associated with the damage caused to an asset that the device holds (e.g., data loss). We also assume that any false alarm has loss equivalent to $\mathcal{F}$. The security effectiveness of a device against a malicious message is denoted by $\delta(s_i, m_l)$, and it is equivalent to the detection rate of an attack. The vector $\varDelta(s_i) \triangleq \langle \delta(s_i, m_1), \ldots, \delta(s_i, m_\psi) \rangle$ defines all the different values of security effectiveness of s_i with regard to the different messages. For more convenience, Table 1 summarizes the notation used in this paper.

3.1 Collaborative Detection

In our model, the aim of the devices is to detect malicious messages injected through an *entry point* into the D2D network. We assume that each device that receives a message is responsible for inspecting it by using its detection capabilities to the best level possible. Based on the results of the detection, the device updates the *confusion matrix* of the route. This is a right stochastic matrix, which holds the probability of the different messages being detected correctly, being confused with other messages or being identified as benign. This matrix type was initially proposed in [28] (p. 100).

Each device that receives a message, follows exactly the same procedure until the message arrives at `d`. At this point, the confusion matrix should have taken the most accurate detection values (ideally is the identity matrix) due to all inspections undertaken by the devices on this route. Collaborative detection of a malicious message along a path requires forwarding state information, which

[5] Very often, we use the terms *types of messages*, and *messages* interchangeably according to the context.

Table 1. Notation.

S	Set of devices	s_i	device i
m_l	message l	$h^\star$	Maximum possible route length in hops
$\mathtt{s}$	Message source	$\mathtt{d}$	Message destination
P^A	Attacker	P^D	Defender
R	Set of routes from $\mathtt{s}$ to $\mathtt{d}$	r_j	j-th route from $\mathtt{s}$ to $\mathtt{d}$
S_j	Set of devices on r_j	$\mathcal{M}$	Set of messages
$\mathcal{M}_m$	Set of malicious messages	$\mathcal{M}_b$	Set of benign messages
$\delta(s_i, m_l)$	Security effectiveness of s_i against m_l	$\Delta(s_i)$	Security effectiveness vector of s_i
σ_i	Security energy cost of s_i	f_i	Forwarding energy cost of s_i
ϵ_i	Total message delivery energy cost of s_i	$\mathtt{e_j}$	Total energy cost on r_j
T	Lifetime of a Nash message delivery plan	$\mathtt{E}$	Vector of energy costs, $\forall\ r_j$ from $\mathtt{s}$ to $\mathtt{d}$
$\mathtt{h_j}$	Number of hops on r_j	$\mathtt{H}$	Vector of hops, $\forall\ r_j$ from $\mathtt{s}$ to $\mathtt{d}$
$C^{(s_i)}$	Confusion matrix of s_i	$C^{(r_j)}$	Confusion matrix of r_j
$\mathcal{F}$	False alarm loss	$\mathcal{H}_l$	Security damage if m_l undetected
w_s	Security cost weight	w_{fa}	False alarm cost weight
w_e	Energy cost weight	w_q	QoS cost weight
D	Payoff matrix of P^D	A	Payoff matrix of P^A
d_{jl}	Utility of P^D for (r_j, m_l)	a_{jl}	Utility of P^A for (r_j, m_l)
$\mathbf{D}^*$	Nash message delivery plan	r^*	Nash route

includes results of the inspections previously conducted on the message. This prevents unnecessary duplication of inspections, thus saving energy.

3.2 Device Confusion Matrix

Given the set of messages $\mathcal{M}$, the linear mapping $C^{(s_i)}\colon \mathcal{M} \rightarrow \mathcal{M}$ describes the *detection capability* of s_i for a message received. This capability is modeled using a stochastic *device confusion matrix* as follows:

$$C^{(s_i)} \triangleq [C^{(s_i)}_{uv}]_{\psi\times\psi},\ \text{where } 0 \leq C^{(s_i)}_{uv} \leq 1,\ \forall u, v \in \{1, \ldots, \psi\}. \tag{1}$$

A confusion matrix value $C^{(s_i)}_{uv}$ denotes the probability of a message u being reported as message v. If $m_u \neq m_v$, then the device confuses one message for another. Such misinterpretation is beneficial for the attacker because the attack associated with the message is not mitigated. If $m_u \in M_m$, and $m_v \in \mathcal{M}_b$, $C^{(s_i)}_{uv}$ is the probability of the D2D network failing to report an attack. If $m_u \in \mathcal{M}_b$, and $m_v \in \mathcal{M}_m$, then $C^{(s_i)}_{uv}$ is the probability of a *false alarm.* One of the objectives of the D2D network must be the confusion matrix to become the identify matrix (*no confusion*) by the time a message is delivered to $\mathtt{d}$. In another sense, if the confusion matrix is the identity matrix, every single malicious message can be detected before it infects $\mathtt{d}$. However this case is not likely to be achieved in practice due to, for instance, 0-day vulnerabilities, and other misclassification errors. To motivate the computation of confusion matrices we present the following example.

Example 1. Assume $S = \{s_1, s_2\}$, and $\mathcal{M} = \{m_1, m_2, m_3\}$. Also, $m_1, m_2 \in \mathcal{M}_m$, and $m_3 \in \mathcal{M}_b$. We also set the false alarm rate equal to 0.05 for both devices. The security effectiveness vectors are $\Delta(s_1) = \langle 0.5, 0.8\rangle$ and $\Delta(s_2) = \langle 0.75, 0.6\rangle$. We also assume that none device confuses a malicious message for another malicious message and therefore $C^{(r_j)}_{uv} = 0, \forall\, u \neq v,\, m_u, m_v \in \mathcal{M}_m$. Then the devices confusion matrices are the following:

$$C^{(s_1)} = \begin{pmatrix} 0.5 & 0 & 0.5 \\ 0 & 0.8 & 0.2 \\ 0.05 & 0.05 & 0.9 \end{pmatrix},\; C^{(s_2)} = \begin{pmatrix} 0.75 & 0 & 0.25 \\ 0 & 0.6 & 0.4 \\ 0.05 & 0.05 & 0.9 \end{pmatrix}. \tag{2}$$

3.3 Route Confusion Matrix

Similarly, given the set of messages $\mathcal{M}$, the linear mapping $C^{(r_j)}\colon \mathcal{M} \to \mathcal{M}$ describes the final detection capability of the D2D network on r_j. This is the *route confusion matrix* for r_j derived from the confusion matrices of the devices that constitute this route. In the problem we examine, the order of detectors does not matter. Therefore, the confusion matrix for each combination can be computed prior to the message delivery.

An advanced way of deriving the route confusion matrix values is to use a boosting meta-algorithm such as *Adaboost* [16]. If we consider that each device detector is a weak classifier then boosting makes classifiers focusing on data that was previously misclassified. The underlying concept of Adaboost is that several weak classifiers can yield a strong classifier. The confusion matrix of a route is a representation of the weighted classifiers on the devices. It is worth mentioning here that boosting is effective only when all devices trust each other. For the boosting scheme to work there is a need for a broadcasting system which updates the classifiers and pre-sets confusion matrices for the combination of detectors. Nevertheless, such a system has to be implemented anyway for updating virus signatures and anomaly detector parameters. Thus, the update of the classifiers can be piggybacked on top of them.

A "naive" alternative to boosting can be a *linear combination algorithm* where each device contributes linearly to the final route detection capability by some weight determined by characteristics of the route (e.g., #hops).

3.4 Energy Costs and QoS

Each time a device receives a message it spends energy: (*i*) to detect any sign of malice (security energy cost, σ_i) and (*ii*) to forward a message towards $\mathtt{d}$ (forwarding energy cost, f_i). The former is determined by all required intrusion detection tasks undertaken during message inspection. The second is related to the energy spent for relaying the message towards the next-hop on the route from $\mathtt{s}$ to $\mathtt{d}$. We denote by ϵ_i the *secure message delivery cost* incurred to a device during message delivery. Formally, we have that $\forall s_i \in S:\ \epsilon_i \triangleq \sigma_i + f_i$.

The total *route energy cost* on r_j, when a message is delivered over r_j, is denoted by $\mathtt{e_j}$, and it is derived by $\mathtt{e_j} = \sum_{s_i \in S_j} \epsilon_i$. The energy costs of all routes between $\mathtt{s}$ and $\mathtt{d}$ are given by the vector $\mathtt{E} \triangleq \langle \mathtt{e_1}, \ldots, \mathtt{e}_\xi\rangle$.

Apart from security and energy efficiency, QoS is an important consideration when deciding upon message delivery. We denote by $\mathtt{h_j}$ the number of hops on r_j. In this paper, we measure the QoS of a route as $\mathtt{h_j}/\mathtt{h}^\star$, where $h^\star \triangleq N_S - 1$, and N_S is the total number of devices in the D2D network. The number of hops of all routes $r_1, \ldots, r_\xi$ from $\mathtt{s}$ to $\mathtt{d}$ are given by $\mathtt{H} \triangleq \langle \mathtt{h}_1, \ldots, \mathtt{h}_\xi \rangle$.

In this paper, we assume a best effort message delivery service without acknowledgments. Along with having higher end-to-end delay due to this assumption, as the number of hops increases the probability of a message to be lost is higher. This is due to mobility, which is meant to be common in D2D networks. It is worth noting here that our model does not consider real-time multimedia communications because they require higher bandwidth than what a typical multi-hop D2D network provides.

3.5 Network Profiles

To allow the expression of different *network profiles*, we have defined an importance costs vector $[w_s, w_{fa}, w_e, w_q]$. By w_s, we denote the security importance weight which accounts for the level of importance the defender gives to some expected security damage (e.g., data theft); w_{fa} is the importance of the false alarm cost (i.e., cost for dropping an innocent message); w_e is the importance that the defender places into the energy cost which can influence the network lifetime and speed up network fragmentation; and w_q is the importance of the QoS for the defender which accounts for the message success delivery rate and end-to-end delay. This vector allows the network designer to define their *network profile* based on their requirements, measured in terms of security, energy preservation, and QoS.

4 Secure Message Delivery Games

In this section, we use game theory to model the interactions between a D2D network (the *defender*) and any adversarial entity (the *attacker*). The latter aims at launching an attack against a device by sending a malicious message to it through the network's entry point as depicted in Fig. 1. Formally, we define the set of players as $\mathcal{P} \triangleq \{P^D, P^A\}$.

The objective of P^D is to securely deliver a message to the intended destination $\mathtt{d}$. By secure delivery we refer to the message being relayed through the network and collaboratively inspected by the devices on its way to $\mathtt{d}$, in order to mitigate any security risk inflicted by P^A. Therefore the security objective of P^D is to correctly detect and filter out malicious messages before they reach their destination. Every request for message delivery to $\mathtt{d}$ defines a *Secure Message Delivery Game* (SMDG).

4.1 Game Characterization

The SMDG is a non-cooperative two-person zero-sum game. The explanation to the zero-sum nature of SMDG is that we have assumed that the attacker aims at inflicting the highest possible damage to the defender. We could model a game where the benefit of the attacker is smaller than the loss of the defender. However, we have left this for future work along with the investigation of different attacker profiles that are associated with different payoffs.

The defender primarily aims at delivering the message securely to $\mathtt{d}$ while the attacker aims at infecting $\mathtt{d}$ with some malware attached to a malicious message as we mentioned previously. The SMDG is a repeated game since players make their decisions once for a pair of $\langle \mathtt{d}, T\rangle$, where T is a predefined timeout, and $\mathtt{d}$ is the destination device for which the game is played. Afterwards, they repeat the game for either every other destination or when T expires. The value of T may depend on the devices' mobility. For instance, high mobility dictates small T in order valid routes to be discovered.

In SMDG, the players make their decisions concurrently without any *order of play*. However, an order of play can be imposed as an alternative where the attacker becomes the *leader* and the defender the *follower* of a Stackelberg game. Nevertheless, this consideration is out of the scope of this paper.

4.2 Strategies and Payoffs

The pure strategies of P^D consists of all routes from $\mathtt{s}$ to $\mathtt{d}$. Therefore, the action set of P^D is defined as $\mathcal{A}^D \triangleq R = \{r_1, r_2, \dots, r_\xi\}$. On the other hand, the pure strategies of P^A are the different messages that P^A can choose to send to $\mathtt{d}$. A message can be one of the following:

$$\{\mathtt{malicious_1}, \dots,\ \mathtt{malicious_n}, \mathtt{harmless},\ \mathtt{surveillance}\} \tag{3}$$

Then, the finite action set of the attacker is defined as:

$$\mathcal{A}^A \triangleq \mathcal{M} = \{m_1, \dots, m_\psi\} = \{m_1, \dots, m_n\} \cup \{\mathtt{harmless}, \mathtt{surveillance}\}.$$

We denote by $G_{\mathtt{d}} \triangleq \langle D, A\rangle$ an $\xi \times \psi$ bi-matrix game where the P^D (i.e., row player) has a payoff matrix $D \in \mathbb{R}^{\xi\times\psi}$ and the payoff matrix of P^A (i.e. the column player) is denoted by $A \in \mathbb{R}^{\xi\times\psi}$.

P^D chooses as one of their pure strategies one of the rows of the payoff bi-matrix $(D, A) \triangleq (d_{j,l}, a_{j,l})_{(r_j, m_l)\in[\xi]\times[\psi]}$. For any pair of strategies, $(r_j, m_l) \in [\xi]\times[\psi]$, P^D, P^A have payoff values equivalent to $d_{j,l}$ and $a_{j,l}$, respectively. The payoff of the defender for a given pair of players' pure strategies (r_j, m_l) follows:

$$U_D(r_j, m_l) \triangleq d_{j,l} \triangleq -w_s(1 - C_{ll}^{(r_j)})\mathcal{H}_l \ - \ w_{f_a}(1 - C_{ll}^{(r_j)})\mathcal{F} - w_e e_j - w_q h_j. \tag{4}$$

Generally, the first term is the expected security damage (e.g., data theft) inflicted by the attacker due to malicious messages being undetected while the second term expresses the expected cost of the defender due to false alarms. This accounts for benign messages that are dropped due to being detected as malicious. The next to last term is the energy cost of the defender when message delivery takes place over r_j while the last term expresses the expected QoS experienced on this route. Since players act independently, we can enlarge the strategy spaces, so as to allow the players to base their decisions on the outcome of random events. Therefore we consider the mixed strategies of both P^D and P^A. The mixed strategy $\mathbf{D} \triangleq [q_1, \dots, q_\xi]$ of the defender is a probability distribution over the different routes from $\mathtt{s}$ to $\mathtt{d}$, where q_j is the probability of delivering a message via r_j. We refer to a mixed strategy of P^D as the *message delivery plan*. On the other hand, the attacker's mixed strategy $\mathbf{A} \triangleq [p_1, \dots, p_\psi]$ is a probability distribution over the different messages, where p_l is the probability of choosing m_l.

When considering mixed strategies, the defender's objective is quantified by the utility function:

$$\begin{aligned} U_D(\mathbf{D},\mathbf{A}) = \sum_{j=1}^{\xi}\sum_{l=1}^{\psi} q_j d_{j,l}\, p_l = -w_s[\sum_{m_l\in\mathcal{M}_m}\ \sum_{r_j\in R} q_j\,(1-C_{ll}^{(r_j)})\, p_l\, \mathcal{H}_l] \ - \\ w_{f_a}[\sum_{m_l\in\mathcal{M}_b}\ \sum_{r_j\in R} q_j\,(1-C_{ll}^{(r_j)})\, p_l\, \mathcal{F}] \ - w_e\mathbf{D}\mathtt{E}^T \ - w_q\mathbf{D}\mathtt{H}^T, \qquad (5) \\ \text{where } j\in\{1,\ldots,\xi\},\ l\in\{1,\ldots,\psi\}. \end{aligned}$$

Because SMDG is a zero-sum game, the attacker's utility is given by $U_A(\mathbf{D},\mathbf{A}) = -U_D(\mathbf{D},\mathbf{A})$. This can be interpreted as, the attacker can cause the maximum damage to the defender.

4.3 Nash Equilibrium

SMDG is a two-person zero-sum game with finite number of actions for both players, and according to Nash [20] it admits at least a Nash Equilibrium (NE) in mixed strategies. Saddle-points correspond to Nash equilibria as discussed in [28] (p. 42).

The following result, from [14], establishes the existence of a saddle (equilibrium) solution in the games we examine and summarizes their properties.

Theorem 1 (Saddle point of the SMDG). *The Secure Message Delivery Game defined admits a saddle point in mixed strategies,* $(\mathbf{D}^*,\mathbf{A}^*)$*, with the property that*

$$\mathbf{D}^* = \arg\max_{\mathbf{D}}\min_{\mathbf{A}} U_D(\mathbf{D},\mathbf{A}),\ \forall\mathbf{A}\ \ \textit{and}\ \ \mathbf{A}^* = \arg\max_{\mathbf{A}}\min_{\mathbf{D}} U_A(\mathbf{D},\mathbf{A}),\ \forall D.$$

Then, due to the zero-sum nature of the game the following holds:

$$\max_{\mathbf{D}}\min_{\mathbf{A}} U_D(\mathbf{D},\mathbf{A}) = \min_{\mathbf{A}}\max_{\mathbf{D}} U_D(\mathbf{D},\mathbf{A}).$$

The pair of saddle point strategies $(\mathbf{D}^*,\mathbf{A}^*)$ *are at the same time security strategies for the players, i.e., they ensure a minimum performance regardless of the actions of the other. Furthermore, if the game admits multiple saddle points (and strategies), they have the ordered interchangeability property, i.e., the player achieves the same performance level independent from the other player's choice of saddle point strategy.*

Our results can be extended to non-zero sum, bi-matrix games. In the latter case, the existence of a NE is also guaranteed, but the additional properties hold only in the case where the attacker's utility is a positive affine transformation (PAT) of the defender's utility.

Definition 1. *The Nash message delivery plan, denoted by* $\mathbf{D}^*$*, is the probability distribution over the different routes, as determined by the NE of the SMDG.*

The minimax theorem states that for zero sum games NE and minimax solutions coincide. Therefore, $\mathbf{D}^* = \arg\min_{\mathbf{D}}\max_{\mathbf{A}} U_A(\mathbf{D},\mathbf{A})$. This means that regardless of the strategy the attacker chooses, the *Nash message delivery plan* is the defender's security strategy that guarantees a minimum performance.

We can convert the original matrix game into a linear programming (LP) problem and make use of some of the powerful algorithms available for LP to derive the equilibrium. For a given mixed strategy $\mathbf{D}$ of P^D, P^A can cause a maximum damage to P^D by injecting a message $\widehat{m}$ into the D2D network. In that case, the utility of P^D is minimized and it is denoted by $U_D(\mathbf{D}, \widehat{m})$ (i.e., $U_D^{\min} = U_D(\mathbf{D}, \widehat{m})$). Formally, P^D seeks to solve the following LP:

$$\max_{\mathbf{D}} U_D(\mathbf{D}, \widehat{m}\,)$$

$$\text{subject to} \begin{cases} U_D(\mathbf{D}, m_1) - U_D(\mathbf{D}, \widehat{m})e \geq 0 \\ \qquad\qquad\vdots \\ U_D(\mathbf{D}, m_\psi) - U_{\mathcal{D}}(\mathbf{D}, \widehat{m})e \geq 0 \\ \mathbf{D}e = 1 \\ \mathbf{D} \geq 0 \end{cases} \Rightarrow \begin{cases} \sum_{j=1}^{\xi} q_j d_{j,1} - U_D(\mathbf{D}, \widehat{m})e \geq 0 \\ \qquad\qquad\vdots \\ \sum_{j=1}^{\xi} q_j d_{j,\psi} - U_D(\mathbf{D}, \widehat{m})e \geq 0 \\ \mathbf{D}e = 1 \\ \mathbf{D} \geq 0 \end{cases}$$

In this problem, e is a vector of ones of size ξ.

5 The Secure Message Delivery Protocol

In this section, we present the *Secure Message Delivery* (SMD) routing protocol whose routing decisions are taken according to the *Nash message delivery plan.* SMD increases security in a D2D network by mitigating the risk of adversaries harming legitimate devices via, for instance, malware attached to messages. SMD has been designed based on the mathematical findings of the SMDG and its main goal is to maximize $U_D(\mathbf{D}, \mathbf{A})$.

According to SMD, each time a request for message delivery to d is issued, s has to compute the *Nash message delivery plan* by solving an SMDG for this destination. To this end, the device uses its latest information about confusion matrices, QoS and energy costs. Then, the message is relayed and collaboratively inspected by the devices on its way to d. The objective of the network (i.e., P^D) is to correctly detect and filter out malicious messages before they infect d.

5.1 SMD Considerations

The SMD protocol takes routing decisions that increase the probability of detecting malicious messages. Apart from security, SMD utilizes standard approaches to take into account (i) the energy costs resulting from message forwarding and inspection, and (ii) the QoS of the chosen route. According to SMD, the devices maintain routing tables with at least three metrics per route:

- the route confusion matrix,
- the total expected energy cost on this route and,
- the shortest path in terms of number of hops (i.e., QoS).

If the only factor affecting the routing decision was security, then the route with the highest detection capability would be always chosen. This would result to a faster depletion of this route's energy as opposed to when a combination of different routes is chosen. Consequently, the D2D network would suffer fragmentation across the entire topology and consequently security would be reduced.

This is the motivation behind considering energy costs upon path selection. Nevertheless, while the shortage of a device's battery can be solved by, for example, by using mobile solar cells as discussed in [4], and QoS might not be so much of a concern for message communications, secure message delivery remains a critical issue.

The formulation of the defender's utility function allows a device to decide how important the expected QoS and energy costs are compared to the expected security damage. For instance, the defender can decide to set the energy costs equal to 0 when a constant source of energy supply is available or to give a higher importance to security losses than QoS.

Due to the best effort nature of the communications (as a result of the multi-hop environment) the higher the number of hops (i.e., QoS) of a route the more likely a message is to be lost during its delivery via that route. QoS accounts for a successful message delivery rate and therefore the defender might never really want to ignore it. In general, SMD allows network designers to customize the protocol based on the *network profile* of the D2D network. In any case, all defender's preferences are reflected to the *Nash message delivery plan.*

5.2 Routing

Getting inspired by the functionalities of the well-known *Dynamic Source Routing* (DSR) [25] routing protocol, SMD consists of two main stages.

SMD - Stage I. In the first stage, $\mathtt{s}$ broadcasts a Route REQuest ($\mathtt{RREQ_d}$) to discover routes towards $\mathtt{d}$. Each device that receives a $\mathtt{RREQ_d}$ acts similarly by broadcasting it towards $\mathtt{d}$ and caches relevant information (i.e., originator of the request, ID of the $\mathtt{RREQ_d}$). When $\mathtt{d}$ receives a $\mathtt{RREQ_d}$, it prepares the $\mathtt{RREP_d}$ and sends it back towards $\mathtt{s}$ by using the reverse route which is built during the delivery of $\mathtt{RREQ_d}$ to $\mathtt{d}$. Each $\mathtt{RREP_d}$ carries information about the route. This information includes the route confusion matrix (E_1), the total energy costs due to inspection and forwarding on this route (E_2), and the total number of hops (E_3). All three fields are updated while the $\mathtt{RREP_d}$ is traveling back to $\mathtt{s}$.

Each device, involved in route discovery, that receives $\mathtt{RREP_d}$, it updates E_1 by using boosting (e.g., Adaboost) or simply a linear combination algorithm without learning features. The same device (e.g., s_i) updates E_2 by adding its total energy cost ϵ_i to the route energy cost. Lastly, E_3 is increased by 1 in every hop from $\mathtt{s}$ to $\mathtt{d}$.

According to SMD, after $\mathtt{s}$ sends a $\mathtt{RREQ_d}$ it has to await for some timeout T_{req}. Within this period $\mathtt{s}$ aggregates $\mathtt{RREP_d}$ messages and updates its routing table with information from those messages.

SMD - Stage II. In the second stage, $\mathtt{s}$ uses its routing table to solve the SMDG by computing the *Nash message delivery plan* $\mathbf{D}^*$. The latter has a lifetime equivalent to T, as defined earlier. Then, $\mathtt{s}$ probabilistically selects a route according to $\mathbf{D}^*$ to deliver the message to $\mathtt{d}$. The chosen route is called the *Nash route* and it is denoted by r^*. Note that for the same $\mathtt{d}$ and before T expires, $\mathtt{s}$ uses the same $\mathbf{D}^*$ to derive r^*, upon a message delivery request. Algorithm 1 summarizes the main SMD functionalities.

It is worth noting here that the complexity of the SMD protocol measured in terms of the number of messages exchanged in performing route discovery is $\mathcal{O}(2N_S)$, where N_S is the total number of devices in the D2D network.

```
Data: $\mathtt{s}, \mathtt{d}, m_l$
Result: $m_l$ delivered
STAGE 1:
$\mathtt{s}$ seeks for a route to $\mathtt{d}$ by broadcasting $\mathtt{RREQ_d}$
if device $s_i$ receives $\mathtt{RREQ_d}$ then
    if $s_i \neq \mathtt{d}$ then
        $\mathtt{s} \leftarrow s_i$
        Execute Algorithm 1
    else
        Send an $\mathtt{RREP_d}$ back towards $\mathtt{s}$ using the reverse route $r_j$
    end
end
STAGE 2:
if device $s_i$ receives $\mathtt{RREP_d}$ then
    if $s_i \neq \mathtt{s}$ then
        Update $C^{(r_j)}$, $\mathtt{e_j}$, $\mathtt{h_j}$
        Attach $\langle C^{(r_j)}, \mathtt{e_j}, \mathtt{h_j} \rangle$ to the $\mathtt{RREP_d}$
        Relay $\mathtt{RREP_d}$ back towards $\mathtt{s}$
    else
        Cache $\langle C^{(r_j)}, \mathtt{e_j}, \mathtt{h_j} \rangle$ to the routing table
        break;
    end
end
$\mathtt{s}$: Derive the Nash message delivery plan $\mathbf{D}^*$
$\mathtt{s}$: Choose $r^*$ probabilistically as dictated by $\mathbf{D}^*$
$\mathtt{s}$: Deliver $m_l$ to $\mathtt{d}$ over $r^\star$
```

Algorithm 1: SMD Stages.

6 Performance Evaluation

6.1 Simulation Parameters

In this section, we evaluate the performance of SMD by simulating 30 devices and 6 routes between `s` and `d`. The number of devices per route is selected randomly and the maximum number of devices per route has been set to 10. The number of malicious messages vary from 2 to 20 with an incremental step of 2.

We consider different network profiles to assess the performance of the SMD protocol. Note here that the network profile refers to the preference of the D2D network in terms of security (i.e., risk appetite), QoS (i.e., delay in message delivery), energy cost (i.e., spent for message inspection and message forwarding), and false alarm (probability of dropping benign messages) as determined by the *cost importance vector*.

We have used a uniform random generator to create the security effectiveness values for all devices. From these values the simulator creates all devices' confusion matrices. Then, we derive the route confusion matrices by using the Algorithm 2. Note that Algorithm 2 is executed by each device at the step of Algorithm 1 where $C^{(r_j)}$ is updated. This is a linear algorithm (less efficient than boosting due to lack of learning features) which allows us to get some preliminary results about the performance of SMD. This algorithm implements a weighted method according to which each device contributes to the route security effectiveness by

(Device security effectiveness) × *(1/Maximum number of hops in the network).*

The final route detection capability not only depends on the detection capability of each device on the route but also on the number of devices. As a result of this, the longer a route is the better its final security effectiveness.

After the route confusion matrices have been derived, the simulator computes the *Nash message delivery plan* for each of the network profiles presented in Table 2. We evaluate the performance of SMD by measuring the defender's expected cost when `s` uses SMD instead of a shortest path routing protocol. According to the latter, `s` chooses the path with the minimum number of hops to `d`. For each message delivery and protocol used we compute the defender's total expected cost which includes security, false alarm, energy and QoS costs.

Table 2. The importance cost vectors used in our simulations.

Network Profile	w_s	w_{fa}	w_e	w_q	**Network Profile**	w_s	w_{fa}	w_e	w_q
`Security`	10	0.5	0	0	`Security & Energy Efficiency`	5	0.5	5	0
`Security & QoS`	5	0.5	0	5	`Security & QoS & Energy Efficiency`	4	0.5	3	2.5

We have considered 10 Cases each representing a different attacker's action set akin to different number of available malicious messages namely; $2, 4, \ldots, 20$. For each Case we have simulated 1,000 message deliveries for a fixed network topology and we refer to the run of the code for the pair ⟨Case,#message deliveries⟩ by the term Experiment. We have repeated each Experiment for 25 independent network topologies to compute the standard deviation. We do that for all 10 Cases and each type of *attacker profile*.

In this paper we consider 2 different attacker profiles; *Uniform* and *Nash*. A *Uniform* attacker chooses any of the available messages with the same probability whilst a *Nash* attacker plays the attack mixed strategy given by the NE of the SMDG. Therefore, we have totally simulated

10 (Cases) × *1,000 (Message deliveries)* × *25 (Runs of each experiment)* × *2 (Attacker profiles) = 500,000 Message deliveries.*

Per message delivery, the simulator chooses an attack sample from the attack probability distribution which is determined by the attacker profile. The simulator aggregates the cost values of each Experiment for both SMD and the shortest path routing protocol.

```
Data: $C^{(s_i)}, C^{(r_j)}$
Result: Updated $C_{uv}^{(r_j)}$
for $u \in \mathcal{M}$ do
    for $v \in \mathcal{M}$ do
        if $u \in \mathcal{M}_m$ then
            if $v == u$ then
                $C_{uv}^{(r_j)} \leftarrow C_{uv}^{(s_i)}/h^\star + C_{uv}^{(r_j)}$
            end
            if $v \in \mathcal{M}_b$ then
                $C_{uv}^{(r_j)} \leftarrow 1 - C_{uu}^{(r_j)}$
            else
                // probability a malicious message $u$ to be
                   confused with another malicious message
                $C_{uv}^{(r_j)} \leftarrow 0$
            end
        end
        if $u \in \mathcal{M}_b$ then
            if $v \notin \mathcal{M}_b$ then
                // $f_a$: device false alarm rate
                $C_{uv}^{(r_j)} \leftarrow f_a/h^\star + C_{uv}^{(r_j)}$
                $f_a^{route} \leftarrow C_{uv}^{(r_j)}$
            else
                // $f_a^{route}$: route false alarm rate
                $C_{uv}^{(r_j)} \leftarrow 1 - f_a^{route}$
            end
        end
    end
end
```

Algorithm 2: How a device s_i updates the route confusion matrix.

6.2 Simulation Results

We have plotted the improvement on the total expected defender's cost when SMD is chosen as opposed to the shortest path routing protocol. The plots illustrate different number of available malicious messages, attacker profiles and importance cost vectors, in Figures 2 and 3.

From both figures we notice that SMD outperforms the shortest path routing protocol with the highest improvement to be achieved under the "`Security`" network profile. From Fig. 2 we notice that the average values of this improvement fluctuate approximately within the range $[30\%, 43\%]$. The second best performance is achieved under the "`Security & QoS`" network profile and it is only slightly better than the improvement we get under the "`Security and Energy Efficiency`" profile. The lowest improvement is noticed under the "`Security & QoS & Energy Efficiency`" network profile with the mean values to be within

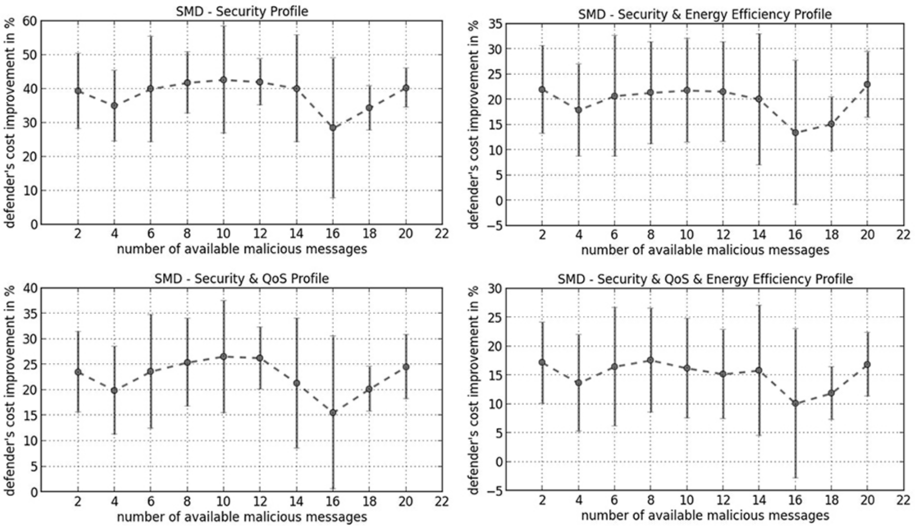

Fig. 2. Simulation results in presence of a uniform attacker.

the range [10%, 18%]. We notice the same trends for a Nash attacker as illustrated in Fig. 3. One difference in the results is that under the network profile `Security & QoS` the difference in improvement compared to the `Security & Energy Efficiency` is more pronounced as opposed to the scenarios with a Nash attacker. We also notice that for all network profiles SMD improves the defender's expected cost in a greater degree in the presence of a Uniform Attacker rather than a Nash attacker although the defender chooses the Nash routing plan in either cases (since it minimizes the maximum potential cost inflicted by the attacker). This is due to the attacker maximizing the minimum defender's expected cost at the NE as stated in Theorem 1. On the other hand, the uniform attacker follows a naive distribution to inject different messages into the D2D network and therefore achieving a worse performance than the Nash attacker.

As a generic comment, the more focused objectives SMD has the higher the improvement of the defender's expected cost is, compared to a shortest path protocol. We also notice that the standard deviation is large in all Experiments. This can be explained by looking at the results from the different Experiments in more detail. By doing so, we noticed that occasionally the same routes are chosen by both SMD and the shortest path routing protocol. This can be explained by the number of available routes being only 6 in our simulations here. The generic trends demonstrate the improvement that SMD introduces even without the use of a boosting algorithm. These preliminary results are promising and we have plans for further investigations when a boosting algorithm (e.g., Adaboost) is used and a larger number of devices and routes are given. In addition, we are planning to examine different mobility levels and see how these affect the expected defender's cost under different network profiles with SMD.

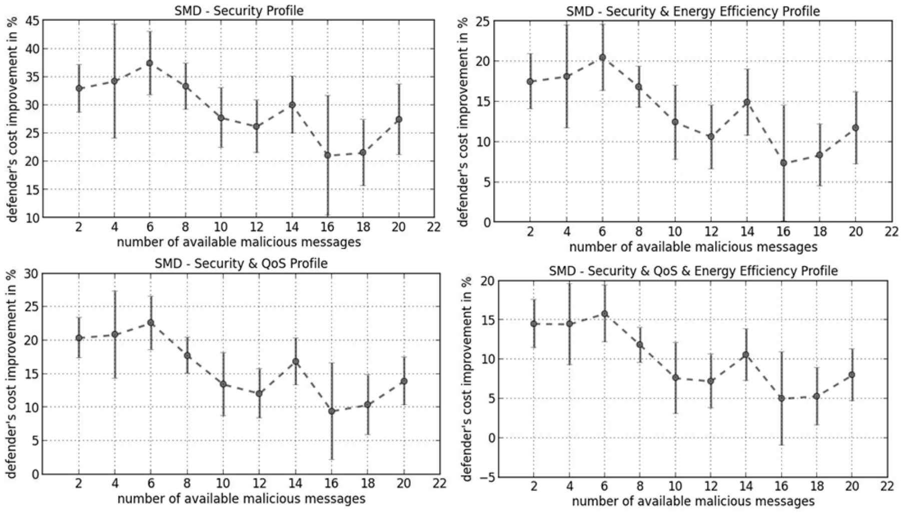

Fig. 3. Simulation results in presence of a Nash attacker.

7 Conclusions

In this paper we have investigated secure message delivery for device-to-device networks in a hostile environment with possible malicious behavior. We have formulated *Secure Message Delivery Games* (SMDGs) to study the interactions between the *defender* (i.e., device-to-device network), and different adversaries, which are abstracted by the player called *attacker*. The defender seeks the "best route" to deliver a message from a source device to a destination device whilst the latter aims to harm the destination with mobile malware attached to a message. The defender solves an SMDG to derive the *Nash message delivery plan* (i.e., Nash mixed strategy). Then, the defender probabilistically chooses a route according to this plan and delivers the message to the destination. Due to the multi-hop nature of the network, intermediate devices relay the message towards the destination. Apart from forwarding, the relaying devices are responsible for the inspection of the message to identify malicious signs and therefore providing security for the D2D message communications.

We have proposed the *Secure Message Delivery* (SMD) routing protocol which takes routing decisions according to the *Nash message delivery plan*. Apart from security, the protocol respects energy costs and end-to-end delay with the ability to be customized to consider each objective at a different degree. We have undertaken simulations to show how much SMD improves the defender's expected utility compared to a shortest path routing protocol. We believe this improvement will be more pronounced when we implement boosting techniques for the computation of the final intrusion detection capabilities (i.e., confusion matrices) of the routes. We have also plans to take into account the remaining energy of each route in the utility function of the defender, and investigate the impact of mobility to the results. Lastly, future work will consider a networkwide extension of the per-message game where the attacker aims to spread a mobile malware while the defender is attempting to stop it.

References

1. Doppler, K., Rinne, M., Wijting, C., Ribeiro, C.B., Hugl, K.: Device-to-device communication as an underlay to LTE-advanced networks. IEEE Communications Magazine 47(12), 42–49 (2009)
2. Feng, D., Lu, L., Yuan-Wu, Y., Ye Li, G., Li, S., Feng, G.: Device-to-device communications in cellular networks. IEEE Communications Magazine 52(4), 49–55 (2014)
3. Fodor, G., Dahlman, E., Mildh, G., Parkvall, S., Reider, N., Miklos, G., Turanyi, Z.: Design aspects of network assisted device-to-device communications. IEEE Communications Magazine 50(3), 170–177 (2012)
4. Nishiyama, H., Ito, M., Kato, N.: Relay-by-Smartphone: Realizing Multihop Device-to-Device Communications. IEEE Communications Magazine 52(4), 56–65 (2014)
5. Jianting Y., Chuan M., Hui Y., Wei Z.: Secrecy-Based Access Control for Device-to-Device Communication Underlaying Cellular Networks. IEEE Communications Magazine 17(11), 2068–2071 (2013)
6. Daohua Z., Swindlehurst, A.L., Fakoorian, S.A.A., Wei X., Chunming Z.: Device-to-device communications: The physical layer security advantage IEEE Communications Magazine, 1606–1610 (2014)
7. F-Secure: Bluetooth-Worm:SymbOS/Cabir. `http://www.f-secure.com/v-descs/cabir.shtml` (accessed June 2004)
8. Van Ruitenbeek, E., Courtney, T., Sanders, W. H., Stevens, F.: Quantifying the effectiveness of mobile phone virus response mechanisms. In: Proc. of the 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 790–800) (2007)
9. Bose, A. Shin, K.G.: On mobile viruses exploiting messaging and bluetooth services. In: Proc. of the Securecomm and Workshops, pp. 1–10 (2006)
10. La Polla, M., Martinelli, F., Sgandurra, D.: A survey on security for mobile devices. IEEE Communications Surveys and Tutorials 15(1), 446–471 (2013)
11. Miettinen, M., Halonen, P., Hatonen, K.: Host-based intrusion detection for advanced mobile devices. In: Proc. of the 20th International Conference on Advanced Information Networking and Applications (AINA), vol. 2, pp. 72–76 (2006)
12. Ardagna, C.A., Conti, M., Leone, M., Stefa, J.: An anonymous end-to-end communication protocol for mobile cloud environments. IEEE Transactions on Services Computing (2014)
13. Felegyhazi, M., Buttyan, L., and Hubaux, J.-P.: Nash equilibria of packet forwarding strategies in wireless ad hoc networks. IEEE Transactions on Mobile Comput. 5(5), 463–476 (2006)
14. Basar, T., Olsder, G. J.: Dynamic noncooperative game theory. London Academic press, 2nd Edition (1995)
15. Cho, J.H., Chen, I.R., Feng, P.G.: Effect of intrusion detection on reliability of mission-oriented mobile group systems in mobile ad hoc networks. IEEE Transactions on Reliability 59(1), 231–241 (2010)
16. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 119–139 (1997)

17. Liu, Y., Comaniciou, C., Man, H.: A Bayesian game approach for intrusion detection in wireless ad hoc networks. In: Proc. of the Workshop on Game theory for communications and networks (GameNets) (2006)
18. Liu, Y., Comaniciou, C., Man, H.: Modelling misbehaviour in ad hoc networks: A game theoretic approach for intrusion detection. International Journal of Security and Networks 1(7), 243–254 (2006)
19. Marchang, N., Tripathi, R. A game theoretical approach for efficient deployment of intrusion detection system in mobile ad hoc networks. In: Proc. of the International Conference on Advanced Computing and Communications (ADCOM), 460–464 (2007)
20. Nash, J.F.: Equilibrium points in n-person games. In: Proc. of the National Academy of Sciences 36(1), pp. 48–49 (1950)
21. Otrok, H., Debbabi, M., Assi, C., Bhattacharya, P.: A cooperative approach for analyzing intrusions in mobile ad hoc networks In: Proc. of the International Conference on Distributed Computing Systems Workshops (ICDCSW) (2007)
22. Panaousis, E.A., Politis, C.: A game theoretic approach for securing AODV in emergency mobile ad hoc networks. In: Proc. of the 34th IEEE Conference on Local Computer Networks (LCN), pp. 985–992, Zurich, Switzerland (2009)
23. Patcha, A., Park, J. M.: A game theoretic approach to modeling intrusion detection in mobile ad hoc networks. In: Proc. of the 5th Annual IEEE SMC Information Assurance Workshop, pp. 280–284 (2004)
24. Patcha, A., Park, J. M.: A game theoretic formulation for intrusion detection in mobile ad hoc networks. International Journal of Network Security 2(2), 131–137 (2006)
25. Johnson, D.B., Maltz, D.A.: Dynamic source routing in ad hoc wireless networks. Mobile computing, Springer US, 153–181 (1996)
26. Santosh, N., Saranyan, R., Senthil, K.P., Vetriselvi, V.: Cluster based co-operative game theory approach for intrusion detection in mobile ad-hoc grid. In: Proc. of the International Conference on Advanced Computing and Communications (ADCOM), pp. 273–278 (2008)
27. Sun, Y.L., Yu, W., Han, Z., Liu, K.J.R.: Information theoretic framework of trust modeling and evaluation for ad hoc networks. IEEE Journal on Selected Areas of Communication 24(2), 305–317 (2006)
28. Alpcan, T., Basar, T.: Network Security: A Decision and Game-Theoretic Approach. Cambridge University Press (2010)
29. Yu, W., Ji, Z., Liu, K.J.R.: Securing cooperative ad-hoc networks under noise and imperfect monitoring: strategies and game theoretic analysis. IEEE Transactions on Information Forensics and Security 2(2), 240–253 (2007)
30. Yu, W., Liu, K.J.R.: Game theoretic analysis of cooperation stimulation and security in autonomous mobile ad hoc networks. IEEE Transactions on Mobile Computing 6(5), 507–521 (2007)
31. Yu, W., Liu, K.J.R.: Secure cooperation in autonomous mobile ad-hoc networks under noise and imperfect monitoring: A game-theoretic approach. IEEE Transactions on Information Forensics and Security 3(2), 317–330 (2008).