On the Performance of Key Pre-distribution for RPL-Based IoT Networks

Ayman El Hajjar[1(✉)], George Roussos[1], and Maura Paterson[2]

[1] Department of Computer Science and Information Systems, Birkbeck, University of London, London, England
{a.elhajjar,g.roussos}@bbk.ac.uk
[2] Department of Economics, Mathematics and Statistics, Birkbeck, University of London, London, England
m.paterson@bbk.ac.uk

Abstract. A core ingredient of the *Internet of Things (IoT)* is the use of deeply embedded resource constrained devices, often connected to the Internet over Low Power and Lossy Networks. These constraints compounded by the need for unsupervised operation within an untrusted environment create considerable challenges for the secure operation of these systems. In this paper, we propose a novel method to secure an edge IoT network using the concept of key pre-distribution proposed by Eschenauer and Gligor in the context of distributed sensor networks. First, we investigate the performance of the unmodified algorithm in the Internet of Things setting and then analyse the results with a view to determine its performance and thus its suitability in this context. Specifically, we investigate how ring size influences performance in order to determine the required ring size that guarantees full connectivity of the network. We then proceed to propose a novel *RPL objective function* and associated metrics that ensure that any node that joins the network can establish secure communication with Internet destinations.

1 Introduction

In recent years, with the development of wireless sensor networks, the Internet of Things (IoT) became a reality. This presents many challenges that also did not exist before because of the nature of the IoT. Since the IoT is a collection of heterogeneous networks, it involves not only the same security problems with sensor network, but also more particular ones, such as privacy protection problem, heterogeneous network authentication and access control problems, information storage and management [1].

The research into the IoT security is far more complicated then that of the Internet security in general. Conventional security protocols for the Internet as we know are not suitable for the Internet of Things. Devices in the IoT are different in terms of computation capabilities, memory limitation, processing power and physical limitation (i.e., installed in rural area and unattended). Thus factors such as reliability, scalability, modularity, interoperability, interface and QoS can be hard to achieve [2].

N. Mitton et al. (Eds.): InterIoT 2016/SaSeIot 2016, LNICST 190, pp. 67–78, 2017.
DOI: 10.1007/978-3-319-52727-7_9

Security of the Internet of Things is at the centre of research. The impact of security breaches on humans in an IoT device is much greater than in conventional networks. For example, a breach of a device monitoring the CO_2 level in a room can lead to physical harm to a human being if this device is compromised and is sending data that are not accurate. Thus authentication and authorization are key to ensuring that only authenticated devices (those that share a suitable key) can join the network. The main challenge, when it comes to authentication of various IoT devices, is the design of key storage and distribution mechanisms, because of the nature of the IoT devices and their network architecture [3].

Given the limitation that IoT devices (sensors and actuators) are constrained in term of computational power and storage memory, several of the conventional security methods are not suitable for use.

The purpose of this paper is to investigate the performance of Laurent Eschenauer and Virgil D. Gligor's Algorithm [4] for Distributed Sensor Networks (DSN) in the context of IPv6 Low Power and Lossy Networks (6LoWPAN) Devices for the Internet of Things (IoT). We provide an analysis of the performance of the algorithm when applied in the DSN and IoT context. We also show the ring size needed to guarantee full network connectivity. We then propose a modification of the routing protocol for Low power and Lossy Networks (RPL) Objective function (OF) in order for the key pre-distribution algorithm to achieve a full network connectivity in the context of the IoT.

Section 2 provides an introduction to the Internet of Things, the 6LoWPAN network protocol, the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL) and several solutions that attempts to secure the Internet of Things. Section 3 presents the key pre-distribution algorithm by Eschenauer and Gligor in [4]. In Sect. 4, we present the experiment methodology and design that we carried in order to first validate the results of [4] and second to determine whether those results are applicable in the context of the IoT. In Sect. 5 we provide an overview of the future work that will be carried on to enable key pre-distribution algorithm to become a suitable solution for the IoT. Finally, we present our main conclusions in Sect. 5.

2 Understanding the Problem: Literature Review

Distributed Sensor Networks (DSN) include a large array of sensor nodes that are usually battery powered, have limited computational capabilities and memory. Nodes in a DSN network, collect data and make it available for processing to application components of the network and control nodes. The scale of a DSN network is quite large (tens of thousands). The Internet of Things (IoT) network is a collection of sensor networks (Wireless and Distributed) that share the same characteristics as Distributed Sensor Networks.

2.1 Internet of Things and 6LoWPAN

Internet of Things is a simple low cost communication network that allows wireless connectivity in applications with limited power and relaxed throughput

requirements [5]. 6LoWPAN concept originated from the idea that "the Internet Protocol could and should be applied even to the smallest devices" and that low-power devices with limited processing capabilities should be able to participate in the Internet of Things [6].

Internet protocols has always been considered too heavy for sensor networks and thus the 6LoWPAN protocol stacks were created. The need for an IP based sensor network made many researchers attempt to adapt existing Internet standards to the creation of interoperable protocols and the development of supporting mechanisms for composable services [7]. Not surprisingly, one of these challenges is security because of the distinct features of sensor networks such as the capabilities of the nodes. In Sect. 2.3, we will review the various attempts to create new security protocols for sensor networks and the IoT or to adapt existing protocols in the context of the IoT.

Given those limitations, another problem arises with IP for the 6LoWPAN network stacks that is relevant to this paper, the topology of the network. Various topologies should be supported by 6LoWPAN networks including mesh and star. Routing for Low Power and Lossy network (RPL) as described in [8], is a routing protocol for 6LoWPAN networks that can solve this problem.

2.2 Routing for Low Power and Lossy Networks RPL

The Routing Protocol for Low-Power and Lossy Networks (RPL) is a distance vector IPv6 routing protocol designed for LLN networks. RPL is designed for networks which comprise of thousands of nodes where the majority of the nodes have very constrained energy and/or channel capacity. To conserve precious resources, a routing protocol must generate control traffic sparingly. However, this is at odds with the need to quickly propagate any new routing information to resolve routing inconsistencies quickly.

RPL organises its topology in a Directed Acyclic Graph (DAG). An RPL DAG must have at least one RPL root and a Destination Oriented DAG (DODAG) is constructed for each root. The root acts as a sink for the topology by storing all routes to all nodes in the DODAG in the routing table. The root may also act as a border router for the DODAG to allow nodes that belong to different DODAGs to communicate [8].

RPL supports three security modes: unsecured, preinstalled and authenticated. Unsecured refers to the security mechanism that is provided in lower layers such as link layer security. Preinstalled and authenticated modes require the use of preinstalled shared keys on all nodes prior to deploying the nodes. Both modes provide security procedures and mechanisms at the conceptual level and are concerned with authentication, access control, data confidentiality, data integrity and non repudiation. This study focuses on the preinstalled mode as a method of securing message transmission between nodes in an RPL DAG instance.

Authentication in the preinstalled mode involves the mutual authentication of the routing peers prior to exchanging route information (i.e. peer authentication) as well as ensuring that the source of the route data is from the peer (i.e. data

origin authentication) [9]. The limitation of the preinstalled mode in its common form, is that it is assumed that a node wishing to join a secured network is pre-configured with a shared key for all neighbours and the RPL root. This means that once this shared key is compromised, all network leaves in the RPL DODAG are compromised.

2.3 Security for the Internet of Things Proposed Solutions

Providing key management for confidentiality and group level authentication in a sensor network is difficult due to the ad hoc nature and limited resources of the distributed sensor network environment. The main challenge in public key algorithms when using in the context of Internet of Things, similarly to sensor networks, is the energy consumption of exchanging public key certificates [10].

Key management protocols can be divided into three categories. Arbitrated keying protocols, Self Enforcing protocols and Pre-Deployed Keying protocols.

Arbitrated keying protocols requires a trusted server such as the use of [11]. They are not suitable for use in the context of the IoT because of the limited energy, communication bandwidth and computational capacities of sensor nodes in an IoT network. The Otway-Rees protocol in [12] is applied in the context of the IoT for one-way authentication; symmetric cryptography with AES is used for encryption. The drawback in one way authentication is that it leaves the network vulnerable to man-in-the-middle attacks.

Self Enforcing protocols such as Pairwise Asymmetric Keying are based on the Diffie-Hellman key agreement protocol. A proposed solution to use a light-weight DTLS based keying mechanism to secure IoT was suggested in [13]. Although this solution proved to provide a lighter and robust security protocol using pairwise key establishment between nodes, the number of message transfers to establish the secure connection in [13] still introduced a large communication overhead. Pre-deployed keys into nodes prior to deployment in a network offers energy efficient solution to providing confidentiality and group level authentication keys [10].

In the next section we investigate the use of the key management scheme for Distributed Sensor Networks proposed by Eschenauer and Gligor in [4] in the context of the Internet of Things.

3 Key Pre Distribution as a Solution for Securing IoT

Offline key pre-distribution algorithm for DSN by Eschenauer and Gligor [4] describes the method by which keys are distributed to nodes in the network.

This key pre-distribution mechanism ensures that for each direct link between any two nodes in the network, the probability of those two sharing at least a key is 0.5. The authors of [4] concluded that the size of key rings and identifier rings $RING$ does not need to be large in order for a network to guarantee full connectivity and only 50% of those pair of nodes need to have a shared key.

At first, a large pool P of keys K and identifiers I is generated. Each key K in the pool is randomly represented by one of the identifiers I. A certain number of identifiers K and their respective keys K are picked from the pool P randomly and loaded into the memory of the node. This will form the key ring and the identifier ring. This step will be repeated for each node that wishes to join the network.

Now that each node in the network has an identifier ring and a key ring loaded into its memory, nodes can begin the phase of selecting a secure route to any other nodes. Each node broadcast its identifier ring to all neighbouring nodes (neighbouring nodes are the nodes that are within it is transmission range). Each neighbouring node compares the identifier ring it received with its own identifier ring. If the node find a shared identifier between the two identifier rings, it sends a message to the origin node with the shared identifier. Nodes that have a shared identifier can establish a secure direct link by using the key that corresponds to the shared identifier. Nodes that do not share an identifier with the origin node will attempt to create a link with it through other nodes (indirect links by hops).

An example in [4] showed that when a pool contained 100,000 keys, full network connectivity was achieved with only 75 keys in the rings. This is due to the fact that routing in Distributed Sensor Networks (DSN) allows multi hops and indirect hop communication between nodes, thus nodes that do no share an identifier can use another node that it shares an identifier with as an indirect link to reach it.

This paper is attempting to evaluate the performance of this algorithm in the context of the IoT environment when using RPL.

4 Experiment Design and Setup

The experiment was simulated on the Contiki Operating System [14] using Cooja nodes simulator [15]. A C program was coded to generate keys pool, IDs pool, Key rings, ID rings[1]. The simulation file was composed of N nodes and one border router[2]. A script was written in order for the simulation to stop running only when all possible routes were computed and no more routes exist. This was essential to ensure that the routing table we obtain at the end of each simulation is the optimum one for our setting. Finally, a Perl program was coded to analyse logs generated by individual nodes after simulation in order to determine if nodes were able to establish a secure link.

4.1 Experiment Parameters

The parameters selected for the simulation experiments aim to approximately match the characteristics of a recent innovative deployment of IoT technology at

[1] Keys & identifiers were generated randomly using Blum Blum Schub generator. Each node will then choose a set of Keys & identifiers for its key ring and identifier ring randomly using Knuth Shuffle algorithm.

[2] A border router is also the root of the RPL DODAG and it will store the routing table of the simulation (acting as a sink).

the campus of the University of Liverpool in the UK, where 650 students were able to employ a smartphone app to access discounts or coupons in stores or cafeterias, as well as for wayfinding and alerting. Specifically, the overall area of 250 × 250 meters which is a typical area size of a medium sized university campus. Number of users (Network size) is based on an average number of wifi usage at Birbkeck campus during a day which is 2394 users [17]. The main difference between out simulations and the use case we use for motivation is the wireless technology used which was Bluetooth Low Energy (BLE) while in the simulations we use Zigbee.

Parameters related to the environment (control parameters) of the simulation were defined in the experiment configuration. We assumed that the transmitting range for each node is 50 meters (this is the common transmitting range for 6LoWPAN low power devices). We also used the key length *klength* of 64 bits and the ID length *ilength* of 32 bits. Those two sizes were chosen as they are enough, given the number of nodes we simulated in the experiment. The number of bits in ID was chosen to be smaller because of memory constraints in the Internet of things devices. The other reason is that exchanging IDs is not revealing anything as there is no connection between keys and IDs is exchanged. Anyone trying to intercept the messages will not be able to make the connection between the identifier exchanged and the key it represents.

We carried out the experiment simulations with three different parameters (independent parameters) changing. The Pool size P of keys is the first parameter. Two pools are being generated in each simulation, one for keys, the other for IDs. Both have the same size. The pool size is an important factor that will have a huge impact on the probability of shared keys between motes. The pools size we run simulations for are: 100, 250, 500,750,1,000 and 2,500 motes. The second parameter is the network size N. The third parameter is the ring size RS It was computed using Stirling equation as per [4]. Those independent variables are shown below and in Table 1. For each pool size (P), keys and identifers are generated once for all networks size. To ensure the accuracy of experiment simulations, each simulation will run 5 times. The largest and smallest results were discarded and the average of the remaining three runs is used. The outputs of

Table 1. Independent variables

Pool size (P)	Ring size (RS)	Network size (N)					
100	8	100					
250	13	100	250				
500	18	100	250	500			
750	22	100	250	500	750		
1,000	25	100	250	500	750	1,000	
2,500	41	100	250	500	750	1,000	2,500

each of those experiments are the Number of DAGs $DAGs$ in the routing table and the Number of Shared Keys NSK between nodes that formed a DAG.

4.2 Experiment Results and Analysis

Figure 1 shows the percentage of shared keys for various pools size when changing the density of nodes in the network in a small environment of 250 by 250. As we can see from Fig. 1, the result of percentage of shared keys in the $DAGs$ becomes consistent around 50%. If the network simulated is a Distributed Sensor Network, a 50% of links between various nodes in the DSN network sharing a key is enough to guarantee full connectivity of the network. In a DSN network, nodes that do not share a key can use a neighbouring node as an indirect link as long as the link is secure. This will mean that it will take the connection between two nodes two hops rather than a direct link but both of them will be secure. However this network is an IoT network, therefore nodes that do not share a key in the routing table will be discarded. Point to Point links in RPL routing is not allowed therefore an alternative multihop secure link can not exist.

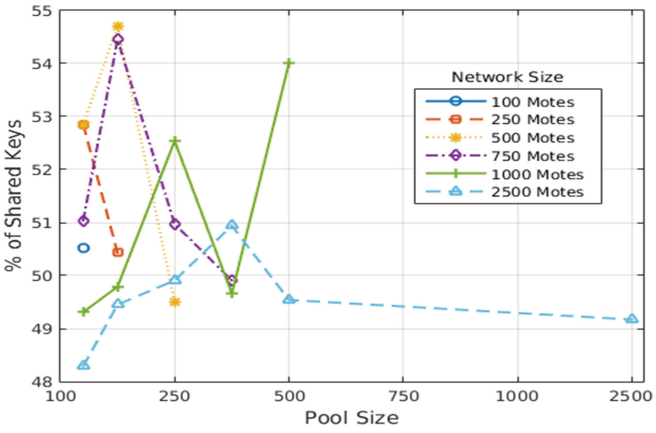


Fig. 1. Number of nodes Vs percentage of shared keys for various pools size

Figure 2 represents the ring size vs the percentage of shared keys in the DAG for various Network size. In this graph, it is very clear that the percentage of shared key $\%NSK$ is hovering around the 50%. We can also validate from Fig. 2 that the size of the ring calculation used in [4] generated a 50% shared keys between nodes in the DSN network. The percentage of $DAGs$ that contains a shared key can also be validated for IoT as 50% of the RPL routing table leaves had a common key ($\%NSK$) in the ring.

However, in a Distributed sensor network as in [4], if two nodes do not share a key they can still communicate using an indirect link (multi-hop). In an IoT network using RPL routing, multi hop alternative route is not possible. A node

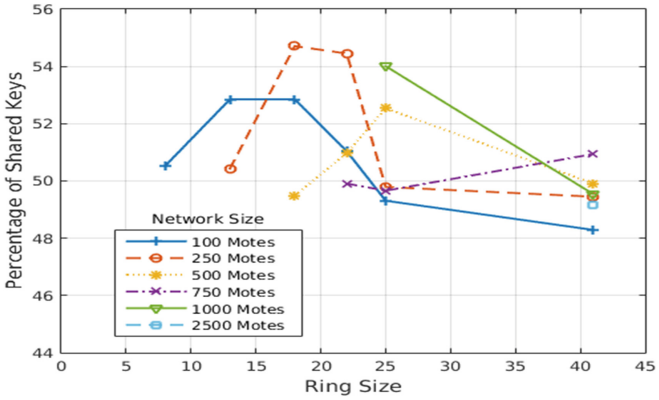
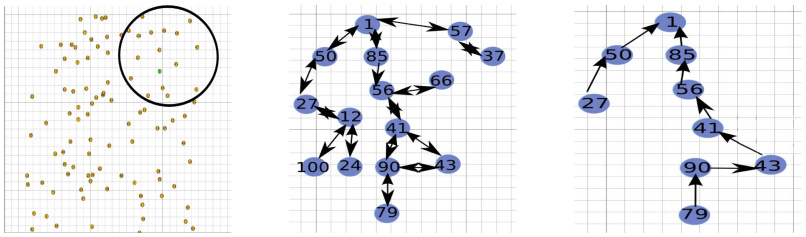


Fig. 2. Ring size Vs percentage of shared keys for various networks size

is only able to communicate with its preferred parent as per the routing table. In our experiment, if this node does not share a key with its preferred parent, then the link between those two nodes does not exist. Therefore the node will not be in the routing table and any sub leaves will also be discarded. Figure 3 show a simulation example of a 100 nodes network and how the routing table for a small subset of this network appear when simulated in the context of the Distributed Sensor Networks versus in the context of the Internet of Things. From this figure we can conclude that many nodes will be discarded if we use the key pre-distribution algorithm in its current form. This will result in an IoT network a lot smaller than the one we started with. The remaining nodes that were discarded, if the algorithm left as it is, will have to start the process of randomly selecting a new key ring and identifier ring. Nodes in the routing table will then check again whether all leaves in the routing table share a key.



(a) Network subset (b) Routing table for DSN (c) Routing table for IoT

Fig. 3. Comparison of routing table for a snippets from a simulation of 100 nodes in the context of DSN Vs. IoT

4.3 Larger Key Rings

Having a small ring size for a considerably large network is a characteristic of the key pre-distribution algorithm in [4]. However and as shown in Table 2, the rings size used for previous experiment did not achieve full connectivity of the network. One alternative that we thought is essential to investigate is the size of the ring. Table 2 below show how we experimented with the ring size, modifying it until we reached 100% connectivity of the network.

Table 2. Simulation experiments over various rings size

Original values			Experiment											
			1		2		3		4		5		6	
N	RS	SK %	RS	SK %	RS	SK %	RS	SK %	RS	SK %	RS	SK %	RS	SK %
100	8	50.52	18	84.16	22	100								
250	13	50.43	30	98.18	36	100								
500	18	57.14	30	83.17	45	99.07	48	100						
750	22	49.47	30	71.95	45	92.87	60	99.40	63	100				
1,000	25	57.14	30	63.44	45	89.28	60	97.32	75	99.53	77	100		
2,500	41	48.19			45	59.37	60	92.46	75	97.11	100	99.64	104	100

Figure 4 show a comparison of rings size when the key pre-distribution algorithm is used in distributed Sensors network and in RPL over IoT network for various network sizes. It is very clear that the size of the ring that achieves a full network connectivity in [4] does not apply to the Internet of Things network when using RPL. To achieve full connectivity of the network, a ring size of 77 key/identifier is needed for a pool size of 1000 in comparison of a ring size of 25 key/identifier for the same pool. This is a big difference that will have a large impact on the network performance. Figure 5 show the rings size needed for various network sizes to achieve a guaranteed full connectivity between all nodes within the RPL routing table.

As we can see from Table 2 above, 104 keys were needed in the key ring to achieve a 100% guaranteed connectivity in the RPL routing table in comparison with only 41 keys in a ring needed for DSN networks . We have used 64 bits key and 32 bits identifier. This will mean that key ring and identifier ring will take up around 1.38 kb of memory storage in each node. In this experiment, we have also used Zolertia node Z1 which features a powerful a 16-bit RISC CPU, 16 MHz clock speed, 8 KB RAM and a 92 KB Flash memory. This means that at least 90 kB of Flash memory is still free to use for operating system and other applications.

However, the original plan was to use as in [4] a pool of 100,000. A simple calculation can give us an estimation of 4,600 keys and identifiers in each ring in order to guarantee connectivity in the network using RPL protocol. Ring size of 4,600 keys and identifiers will take up around 54 kb of memory storage in

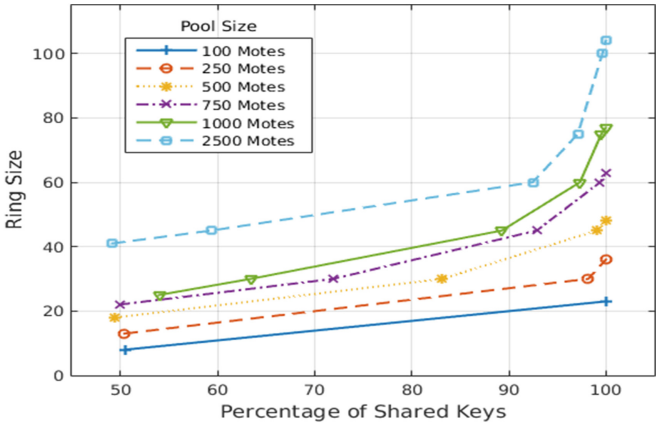


Fig. 4. Various rings size to achieve 100% of shared keys for different Pool size

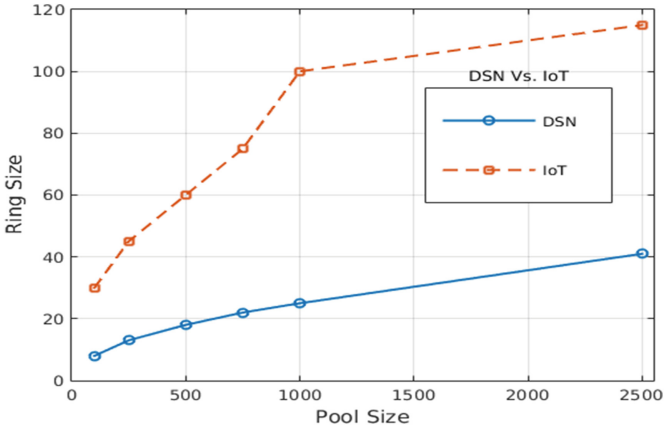


Fig. 5. Rings size in DSN Vs. rings size in IoT for various Pool size for 100% connectivity

each node. That is more than half of the memory present for the Zolertia node (Zolertia [18] has the largest amount of memory in Contiki. TMote sky node [19] is widely used and it has only 48 kb of memory which is not enough if using 4,600 keys and identifiers in each ring).

Computation overhead is another aspect that needs to be looked at. Comparing two identifiers rings will require a processing power that is very scarce. When running the same experiment using 4,600 and 104 keys in a ring, we note that during comparison of the key ring between two nodes, nodes processing power were around 87% used for 23 s. We can conclude that for a larger key ring size, nodes will not be able to cope with the computation power required and this will add a huge overhead on the network performance and the routing table establishment.

5 Conclusion and Future Work

In this paper, we investigated the performance of the key pre-distribution algorithm for distributed sensor networks on the IoT devices. We experimented with the variables and simulated small scale networks of 100 nodes to large scales network of 2500 nodes. Up until this point, we believe we have proved that the key pre-distribution algorithm achieve the 50% probability of the nodes to have a shared key, however it does not guarantee a full connectivity of the network when used in the context of the IoT. The use of RPL protocol in IoT gives a 0.45 probability of leaves in the RPL table with a shared key, which means that not all the network is able to communicate as the RPL only uses leaves that are in the routing table.

The next step in this research will be to explore alternatives solutions to secure leaves in the RPL routing table that do not share a key. In the coming few months, we will be developing a new Objective function metric.

The Objective Function uses several routing metrics to form the DODAG based on some algorithm or calculation formulas. Metrics are carried in DAG metric containers embedded in the DIO messages. The DAG metric containers at the moment are divided into two categories, node metrics and link metrics. In node metrics, nodes exchange information metrics about node state, node energy and hop count. in Link metric, nodes exchange link related information such as throughput, latency and link reliability.

We propose to add Shared Identifier Secure Link Objective Function (SISLOF) to RPL objective function metrics. SISLOF objective function will be used to quantify the shared key discovery (node metric) between two nodes that can form a direct link (neighbouring node) using a Boolean value, of 0 or 1, where 0 indicates that the two nodes do not share a common identifier and 1 indicates that the two nodes do share one or more common identifier. Further to this, the SISLOF will compute other link metrics in order to determine the suitability of the link if two links exist both with a shared key, in term of ETX and node rank.

By doing this, we ensure that any node that joins the routing table can communicate securely as only the nodes that fulfil the requirement of the SISLOF will be able to join the RPL DODAG.

References

1. Zhao, K., Ge, L.: A survey on the internet of things security. In: 9th International Conference on Computational Intelligence and Security (CIS), Leshan, pp. 663–667 (2013). doi:10.1109/CIS.2013.145
2. Tan, L., Wang, N.: Future internet: the internet of things. In: 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Chengdu, pp. V5-376–V5-380 (2010). doi:10.1109/ICACTE.2010.5579543
3. Gan, G., Lu, Z., Jiang, J.: Internet of things security analysis. In: International Conference on Internet Technology, Applications (iTAP), Wuhan, pp. 1–4 (2011). doi:10.1109/ITAP.2011.6006307

4. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In: Atluri, V. (ed.) Proceedings of the 9th ACM Conference on Computer, Communications Security (CCS 2002), pp. 41–47. ACM, New York (2011). doi:10.1145/586110.586117
5. Shelby, Z., Bormann, C.: 6LoWPAN: The Wireless Embedded Internet - Part 1: Why 6LoWPAN?, EE Times (2011). http://www.eetimes.com/document.asp?docid=1278794
6. IEEE Computer Society, 802.15.4 - Low Rate Wireless Personal Area Networks (LR-WPANs), IEEE standard for local and metropolitan area networks, IEEE, USA (2011)
7. Internet of Things, Strategic Research Roadmap; European Commission - Information Society and Media DG, European Commission, Brussels, Belgium (2009)
8. Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., Alexander, R.: RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks, In: Winter, T., Thubert, P., (eds.) IETF draft (2012). https://tools.ietf.org/html/rfc6550
9. Taso, T., Alexander, R., Dohler, M., Daza, V., Lozana, A., Richardson, M. (eds.) A Security Threat Analysis for the Routing Protocol for Low-Power and Lossy Networks (RPLs) RFC 7416, IETF trust (2015). https://tools.ietf.org/html/rfc7416
10. Carman, D.W., Kruus, P.S., Matt, B.J.: Constraints and Approaches for Distributed Sensor Network Security, NAI Labs Technical Report, 1 September 2000
11. Neuman, C., Yu, T., Hartman, S., Raeburn, K.: RFC 4129: The Kerberos Network Authentication Service (2005)
12. Noack, M.: Optimization of Two-way Authentication Protocol in Internet of Things, Master Thesis, University of Zurich, Communication Systems Group, Department of Informatics, Zurich, Switzerland (2014)
13. Porambage, P., Kumar, P., Gurtov, A., Ylianttila, M., Harjula, E.: Certificate based keying scheme for DTLS secured IoT draft-pporamba-dtls-certkey-00, IETF, June 2013
14. Contiki Operating system. http://contiki-os.org
15. Ostrelind, F.: A sensor Network Simulator for the Contiki OS, February 2006. http://soda.swedish-ict.se/2296/1/SICS-T-2006-05-SE.pdf
16. Swedberg, C.: University Caters to Students Seeks Efficiencies Through Beacons, IoT Journal, September 2016. http://www.iotjournal.com/articles/view?14936
17. IP Services, Birkbeck University of London, 23 August 2016. http://www.bbk.ac.uk/its/services/kpis/wifi-usage
18. Zolertia Low power wireless module for IoT and WSN. http://zolertia.io/z1
19. Moore, S.: Tmote Sky, August 2013. http://wirelesssensornetworks.weebly.com/1/post/2013/08/tmote-sky.html