International Conference on Control, Engineering & Information Technology (CEIT'14)

Proceedings - Copyright IPCO-2014

ISSN 2356-5608

Broadcast Based Registration Technique for Heterogeneous Nodes in the IoT

Qazi Mamoon Ashraf[1], Mohamed Hadi Habaebi[2],Gopinath Rao Sinniah[3], Jalel Chebil[4]

[1,2]Department of Electrical and Computer Engineering,International Islamic University Malaysia
[3]Internet of Things, Wireless Communication, MIMOS BHD
[4]Department of Technology and Engineering in Transport, Higher Institute of Transport and Logistics of Sousse, Tunisia

Email: [1]mamoonq@gmail.com, [2]habaebi@iium.edu.my, [3]gopinath.rao@mimos.my, [4]chebil8@hotmail.com

Abstract—**Many traditional techniques for device node registration require manual configuration to achieve the same. Some techniques make use of public-key based schemes to achieve authentication and security. Typically, they are designed to work for multi-hop wireless sensor networks and are able to provide seamless support for mobility. However, these techniques are not considered practical especially for requirements arising in Internet of Things such as in a smart home setup. This paper presents a single-hop, single gateway based node registration technique called RIoT (Registration in IoT) for a similar use case scenario. Advantages of such a method include support for scalability, security and user-friendliness. Some additional contention parameters were introduced to cater for the need of scalability. RIoT was simulated on Contiki OS to test the contention parameters.**

***Index Terms*—Internet of Things, Scalability, Registration, Wireless Sensor Networks, Contiki, Single-Hop, Autonomy**

I. Introduction

Wireless sensor networks (WSNs) are widely deployed for the purpose of remote monitoring and comprise of heterogeneous components and services as well as non-standard interfaces [1]. Nodes in WSNs traditionally make use of registration protocols to establish connection and get attached with gateways or routers. These WSN technologies are important to achieve the vision of Internet of Things (IoT); a platform where traditional Internet is being extended to include diverse objects [2] [3] in an attempt to standardize the ways of communication between heterogeneous entities.

IoT consists of cheap, low processing and energy-efficient devices capable of interacting with the environment. Such devices exist in huge numbers and are expected to become part of our daily life. In addition, these devices may be highly mobile; and constantly may fall within the boundaries of different networks. Issues like seamless handoff may not be applicable as in the case of mobile telephony. A large number of IoT devices leads to the problem of scalability, especially when it comes to configuring each of them manually as required by the traditional registration protocols, with the different settings required by the network.

Another problem that also arises is the scenario if many of such IoT nodes enter a network at the same time and request to register simultaneously. A large number of requests, results in packet-loss through collision. As such, a requirement of scalability also exists to be met. Thus, traditional registration techniques may not be suitable for scalability in IoT, partially due to the fact that they were never designed for that.

This paper presents RIoT; a broadcast-based node registration technique for IoT. The remainder of this article is organized as follows. The use-case scenario for home automation is explained in the second section followed by a brief look at the related work. This is followed by the design of the RIoT protocol and the simulation results. The sixth section presents the preliminary analysis of RIoT and finally the findings are concluded and future work is established.

II. Use Case Scenario

For traditional wireless networks that are deployed, a master gateway, router or an access-point (AP) commonly interfaces the wireless devices with the Internet over a single hop configuration. Wireless Local Area Networks (WLANs) constitute a vital part of the IoT as many IoT devices make use of IEEE 802.11 technology for connecting to WLANs. Similarly, IoT devices may also use IEEE 802.15.4 as their communication standard, but they would still require a master gateway for establishing connection to other devices over the Internet. A single user is expected to own a multitude of these IoT devices, which will be used for various sensing and actuating purposes.

In home automation, IoT devices will be setup to register with a gateway or AP which will provide services for devices to connect to the Internet. Here, there is also the possibility of the existence of an attacker gateway, which

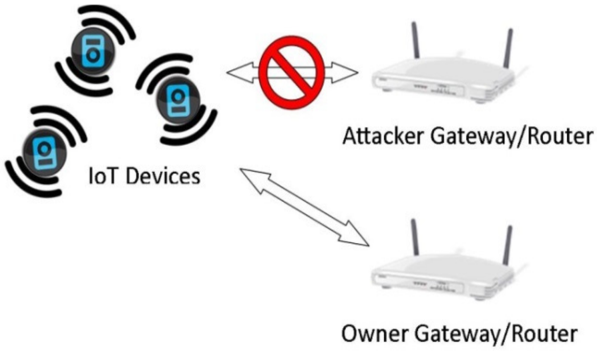

Fig.1.Use Case Scenario for IoT device registration with only the owner's trusted Gateway/Router.

may fall under the radio communication radius. Such a concept is also applicable for WSNs where a large deployment may take place. Scalability issues in terms of delay and collision may arise in each. For simplicity, the scenario requirements are as follows:

- The user buys new IoT devices and wants to immediately set them up for use at home.
- The user (will be referred to as owner from this instance) doesn't have to configure any setting except for a simple grant of permission. This requirement is essential to make the protocol more secure.
- An attacker with a gateway/AP may be within range and access should only be provided to the legitimate device. Fig.1 represents this scenario. This requirement translates directly into a need of some sort of security mechanism.

III. RELATED WORK

Authors of [4] have presented a registration approach called ICatchYou for a multi-sink scenariobased on device discovery. However, such a multi-sink system is not suitable for our use-case as the IoT device should be only attached to the owner's gateway/router and supporting multiple attaching points would lead to complications in security. If the legitimate owner has two gateway devices, it will still not be optimal for a simple IoT device to have the capabilities to be attached to both simultaneously or handle such complicated scenarios.

Table I: Summary of Related Works

Work	Use Case
[4]	Multi-Sink Approach For Registration
[5]	Non- User Friendly, Not suitable for IoT case
[6]	User-Friendly but Complicated Authentication Mechanism, No Manual Entry of Passcodes
[7]	Secure but High Processing Requirements, Owner's Device Broadcasts
[8]	Trust Based Approach for Ad Hoc Networks.

The goals of the protocol proposed by [5] suitably match our use case; however this protocol requires the owner of the device to manually enter access keys at the gateway. Our proposed protocol removes the need to manually enter any information, and proposes to make the whole process as user- friendly as possible.

The protocol proposed in[6] attempts to make the process as user friendly as possible, whereby the owner identifies the devices audio-visually and makes sure that only one registration instance is activated at a time. This approach uses public key authentication system to assure security. For IoT devices, employing public key based security solves a wide variety of security problems and a similar approach is being used in RIoT as well.

Authors of [7] go ahead to describe accelerating a public-key based, multi-user authentication scheme. In this setup, broadcast from an owner's trusted device continues until all reachable nodes receive the broadcast packet. Upon receiving the broadcast packet, the node attempt to verify the attached signature. This work attempts to solve the inefficiency arising in such a scenario as each node has to separately verify the signature of the broadcast packet. Again, the approach assumes the IoT devices to be capable of possessing some processing ability. Scalability might also become an issue in such an approach as nodes are expected to communicate with one another.

RIoT itself is based on some of the concepts derived from [8] which discusses on creation of ad hoc networks using self-configuration procedures. The main concept derived is the use of trust to authenticate the limited resource devices. But due to the intrinsic nature of this protocol to be used for ad hoc networks, it is not possible to adapt it as such for the case of IoT. Also, security mechanisms in this paper make use of both symmetric and asymmetric techniques for exchanging information which might complicate the working of IoT devices, given their small memory size and energy constraints.

The related works are summarized in Table I.

IV. THE PROTOCOL

Our protocol RIoT, standing for "Registration in IoT", aims to provide analmost-autonomous registration technique between IoT devices and gateways/APs with security mechanisms. The advantage of such a method lies in the requirements of user-friendliness and scalability that it aims to cater to. The protocol results in mutual authentication of both parties involved. Five messages are described:

A. *Registration Beacon Message*

The Registration Beacon Message (RBM) contains three parts. First is the Node ID which may also be printed on the body of the IoT device, or any other similar identifier. The second part will be the product brand identifier or trademark code. Third part will contain the MAC address of the IoT device. Upon powering on, this message will be continuously broadcasted by the IoT device to any interested listeners without any encryption. The message will be broadcasted after a pre-defined contention interval t_{RBM}+ a random time t_{r1} where t_{r1}= [0, t_{RBM}]. Thus for a given value of t_{RBM}, the actual broadcast will happen after t_{RBM}+ $_{tr1}$ time intervals. Thus, the maximum contention time between two successive broadcasts would be $2t_{RBM}$. The total time in this stage is given by:

$$t = N_1\ (t_{RBM}) + \sum t_{r1}$$

Where N_1 is the number of broadcasts required until the acknowledge message is received.

B. Registration Acknowledge Message (ACK)

Once the gateway receives the RBM, it will reply with a Registration Acknowledgement Message which will contain the public key information of the AP. The ACK will be sent after a minimum of t_{ACK} seconds of receiving the beacon message. To send the ACK, the gateway will use MAC layer based communication protocol.

The ACK won't be sent immediately but after a contention period, so as to try to allow other nodes to finish their broadcasting within the total broadcast interval of 0-$2t_{RBM}$ seconds. The ACK will be sent after the contention interval t_{ACK} + a random time t_{r2} where is defined similarly as $t_{r1.}$ Henceforward, t_{r2} = [0, t_{ACK}]. Clearly, a good combination of the two values can prove to increase efficiency whereas a poorly chosen combination can severely result in collisions, and unnecessary delay. Our preliminary tests investigated this behavior. The total time so far is calculated as:

$$t = N_1\ (t_{RBM}) + \sum t_{r1\ +}\ t_{ACK} + t_{r2}$$

Information Displayed on a User Interface:

Furthermore upon receiving the RBM; the Node details, time of request, and the MAC address will be displayed on a suitable display such as a web interface belong to the gateway. The owner will then have to allow or disallow the device or group of devices via a simple click. This way the gateway authenticates the device and adds an entry in its list of trusted devices.

C. Restricted Public Key Transfer Message

This message allows the IoT device to be accessed and actuated solely by the gateway, which is an important security requirement. This is done by selectively passing the public key of the IoT device to only the gateway. The public key in such a scenario can be termed as a restricted public key (RPK). Restriction is thus manifest that the public key is not exactly public, because it will be only sent to the gateway. The gateway will then use the RPK to encrypt all actuator messages that it wants to send to the IoT device. Such a measure is helpful as no sniffer will be able to impersonate the gateway by possessing the RPK. To send the RPK to only the gateway, low range communication methods can be used. For user-friendliness, the RPK can be sent using near field communication technology or by RFID. Thus, the owner just touches the new IoT device to the NFC device in the gateway. Clearly, using this method no malicious device can get hold of the public key, unless and until the IoT device is brought really close. This is where the concept of trust is being utilized.

Information Displayed on a User Interface:

To achieve this restricted communication of the RPK, the web interface will ask the owner to bring the specific new device closer to the gateway. The gateway will then associate the freshly acquired RPK with the devices Node ID, and MAC address. Thus the gateway will know how to encrypt any actuation message destined towards a particular node. This way the gateway authenticates itself to the IoT device for future communication.

D. Sensor Value Message

The IoT device may be a sensor in whole or in part, and may be required to send sensor values to any destination. Any data originating from the IoT device would be encrypted using the Public Key of the gateway. It is recalled that the IoT device will receive the Public Key of the gateway in the ACK message. There will be an option to enable or disable encryption for sensor-only IoT devices.

E. Device Actuation Message

To actuate a device, the owner will trigger the gateway via a secure channel over the Internet, through a suitable interface such as a mobile App, or a web browser. The owner may also send a particular value straight to the IP address of the device. In both cases, the gateway is responsible to encrypt the incoming message using the RPK of the destination IoT device. But since the encrypted value of a particular message can be sniffed and replayed by an attacker, the IoT device has no method to differentiate and may thus be controlled by the attacker. However, to solve this problem, the gateway will generate a random number each time it wants to transmit to the device. The random number will be encrypted with RPK and the message will be encrypted with the newly formed random number. Both will be attached to the message and sent to the IoT device.

The IoT device upon reception will extract the random number and then use it to recover the control parameters. This way the same actuator message will result in different encrypted messages, and as long as these messages are understood by the device only once then no malicious actuating will take place.

This protocol essentially takes care of the worst case scenario when all the nodes join the network at once. If the mobile nodes join slowly, then the packets won't undergo any collisions and the nodes will be registered very efficiently- as soon as they start broadcasting. The protocol handles lost and/or corrupted packets in the following manner. When a Broadcast Packet is lost, the node will retry again after t_{RBM}seconds. When an ACK packet is lost, then no attempt is made to retry, as it is known that the node will retry to send another broadcast soon. This message flow is summarized in Fig 2.

V. Simulation and Results

The simulation was done in COOJA simulator that is provided with Instant Contiki. The simulation was carried out for optimizing the duration of registration using RIoT and finding out the optimal value of contention parameters of $t_{RBM}$and t_{ACK}. Table II presents the simulation mote specification.

Values of t_{RBM} and t_{ACK} were varied for the initial set of simulations. The observations received will lead to select optimized values for the two variables for different population sets of nodes and subsequently help in selecting the right combination for scalability, such as when many IoT devices are switched on together. The simulation was done for 10 motes arranged randomly, but within range of the transmission from the gateway/router node. Three set of figures have been identified as preliminary results, and are presented next. The x-axis in each graph reflects the value of t_{RPM}in seconds, and has been labeled as 'Mobile Node Broadcast Delay'. Five sets of values have been plotted for each figure, and represent the results returned when t_{ACK} was varied from 1s to 5s.

ınMote Specification

WSN mote

ı17 16-bit RISC CPU@16MHz

GHz @250kbps

ain: 9dB

28(TX) byte data buffering[4]

5B Flash Memory, 8KB RAM

IE Address

VI. Analysis

A. *PDR for ACK Packets*

Fig. 3 shows PDR % for Gateway Acknowledgements Packets. Here,

PDR % = ∑ Packets Received by nodes/∑Packets Sent by Gateway x 100.

The conventional use of PDR is for data packets, but here it is being used as an indicator for the control messages to achieve the registration. As can be seen, PDR % is generally highest for 5s ACK delay as compared to other ACK delay values. PDR doesn't seem to be affected significantly by mobile node broadcast delay, however higher broadcast delay durations lead to a higher PDR %. For values from 1s gateway ACK delay, no result achieved 100% PDR ever. On the other hand, 5 s gateway ACK delay achieves 100% ACK for 7 cases. Fig. 3 basically highlights that gateway ACK

delay has a higher effect on the overall PDR, and that to increase PDR; the gateway ACK may be increased. Once the values reach saturation, then the Mobile Node Broadcast Delay can be increased.

For 10 nodes, PDR is affected significantly by gateway ACK delays and majorly by node broadcast delays. To achieve a good PDR, it is recommended to keep ACK delay high as well as the delay for node broadcast. Lower values may result in a lower PDR. For scalability, in the worst case scenario, we might need higher ACK delay values. Future work will be to further understand this will be after increasing the number of nodes and then comparing the average PDRs.

B. *Packet Reception Traffic*

Fig. 4 shows the total number of packets generated by nodes that made it without collision to the gateway to achieve registration. All graphs tend to converge towards a single line for higher values of node broadcast delays. The value lies between 20 to 30 packets. Thus for 10 nodes, an average of 2 to 3 packets have to be registered per node for all to get registered by the node. This value doesn't depend much on either ACK delay or broadcast delay for values after the value for ACK delay equals the value of the broadcast delay. When broadcast delay is smaller than ACK delay, more packets are generated. This could happen because a node takes longer to receive the ACK, and it keeps trying again and again in that duration. This also may signify that a gateway might receive 2-3 packets after the first packet, during the time it delays the acknowledgment.

The gateway in this scenario requires an average of 2 to 3 advertisement packets to get all nodes registered in the worst case scenario. These results also have important effects on battery optimization and energy saving, as more packets generated means more energy spent. Future work may be to further expand the x-axis to a wider range of values, to see if the number of packets always touches 10 for 10 nodes. Also, there is a scope of optimization to reduce the number of advertisement packets to make it closer to an average of 1 packet.

C. *Timing for Scalability*

Fig. 5 shows the average time to register for all nodes. In this graph, the total time of registration for 10 nodes was divided by 10 to get the average time. It can be observed that excluding the lower values of the node broadcast delay, roughly the same amount of average time is taken as much as the values of the broadcast delay. Thus it takes roughly, 2 seconds per node to register at a mobile node broadcast delay of 2 s.

Another important observation is that the slope of 5s ACK delay is lower than slope of 1s ACK delay, which means the rate is slowly decreasing as ACK delays are increasing. This means a higher value of ACK delay may be better suited for higher values of node broadcast delay, but

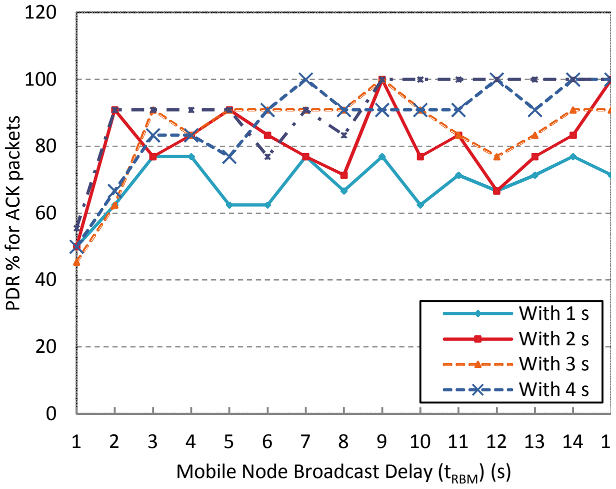

Fig 3. PDR% for ACK packets vs Mobile Node Broadcast Delay (t_{RBM}) plotted at different values of t_{ACK} for 10 nodes initiated together.

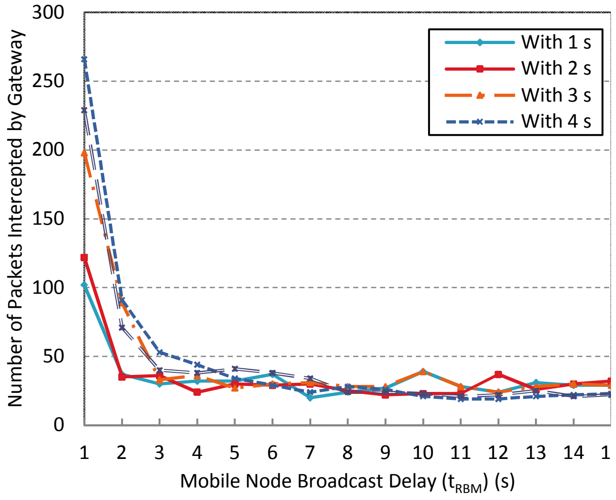

Fig.4. Number of packets intercepted by the Gateway vs Mobile Node Broadcast Delay (t_{RBM}) plotted at different values of t_{ACK}

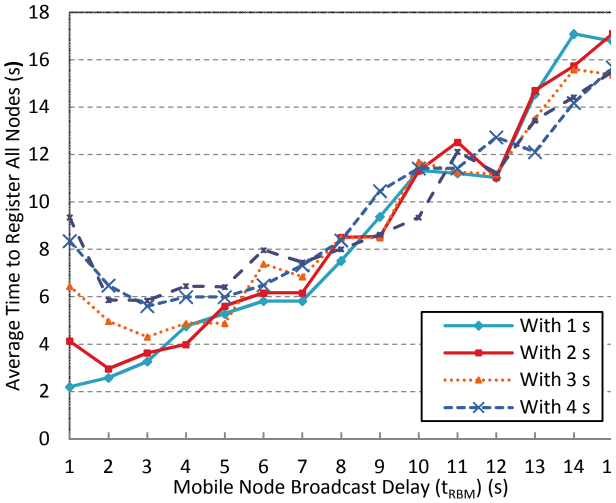

Fig.5. Average time to register for 10 nodes vs t_{RBM} plotted at different values of t_{ACK}

overall sacrificing by increasing the time to register all the nodes.

It was also observed that the average time to register all nodes increases at a decreasing rate for higher values of gateway ACK. This graph gives important insights if we want the network to register faster. However to maintain efficiency, the slope has to be brought in consideration as well.

VII. Summary and Future Work

RIoT was proposed as a protocol specifically aimed for user-friendliness in the IoT as well as addressed towards the scalability requirement of IoT. A suitable implementation for a smart home was also presented. As per the results, packet delivery ratioof 100 percent was achieved for higher values of the contention variables in a worst case scenario. This happened as no ACK packet was lost due to collision. Also, it was observed that the gateway receives an average of 2 to 3 advertisement packets before the ACK is received the IoT device. Support for security was also presented. The future work will look at increasing the number of nodes and then compare effect on the average time to register (per node). Also, the scope of registration will be not restricted to a smart home setup. Instead a general registration protocol will be designed.

Acknowledgement

We are thankful to Zeldi Suryady for his timely suggestions and also to Mohammad Tahir, Teo Kar Hoey, Usman Sarwar and Shariq Haseeb. This research was supported by the Ministry of Higher Education Malaysia (MOHE) under Exploratory Research Grant Scheme (ERGS) number ERGS13-023-0055.

VIII. References

[1] S. Rajasegarar, C. Leckie and M. Palaniswami, "Anomaly Detection in Wireless Sensor Networks," *IEEE Wireless Communications,* vol. 15, no. 4, pp. 34-40, 2008.

[2] A. Iera, C. Floerkemeier, J. Mitsugi and G. Morabito, "Special Issue of the Internet of Things," *IEEE Wireless Communcations,* vol. 17, no. 6, pp. 8-9, 2010.

[3] J. Zheng, D. Simplot-Ryl, C. Bisdikian and H. T. Mouftah, "Ed., The Internet of Things," *IEEE Communications ,* vol. 49, no. 11, pp. 30-31, 2011.

[4] R. Silva, J. Sa Silva, M. Simek and F. Boavida, "A new approach for multi-sink environments in WSNs," in *IFIP/IEEE International Symposium on Integrated Network Management IM'09*, 2009.

[5] N. Taesombut, V. Kumar, R. Dubey and P. V. Rangan, "Secure registration protocol for media appliances in wireless home networks.," in *IEEE International Conference on Multimedia and Expo ICME'03*, 2003.

[6] F. Hohl and E. Kovacs, "Secure and easy-to-use registration of mobile and stationary devices to wireless ad-hoc CE networks.," in *Second IEEE Consumer Communications and Networking Conference*, 2005.

[7] X. Fan and G. Gong, "Accelerating signature-based broadcast authentication for wireless sensor networks.," *Ad Hoc Networks,* vol. 10, no. 4, pp. 723-736, 2012.

[8] R. Lacuesta, J. Lloret, M. Garcia and L. Penalver, "A Secure Protocol for Spontaneous Wireless Ad Hoc Networks Creation," *IEEE Transactions on Parallel and Distribted Systems,* vol. 24, no. 4, pp. 629-641, 2013.

[9] Texas Instruments, "2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver," Texas Instruments, 7 March 2013. [Online]. Available: http://www.ti.com/lit/ds/symlink/cc2420.pdf. [Accessed 15 August 2013].