A Secure Scheme for Group Communication of Wireless IoT Devices

Bashar A. Alohali*, Vassilios G. Vassilakis†, Ioannis D. Moscholios‡, Michael D. Logothetis§
*School of Computing and Mathematical Sciences, Liverpool John Moores University, Liverpool, United Kingdom
†Dept. of Computer Science, University of York, York, United Kingdom
‡Dept. of Informatics & Telecommunications, University of Peloponnese, Tripolis, Greece
§Dept. of Electrical & Computer Engineering, University of Patras, Patras, Greece

***Abstract*—The emergence of the Internet of Things (IoT) is expected to significantly advance the technology development in many application domains such as agriculture, home automation, and healthcare. However, in the IoT era, this development faces serious research challenges in terms of handling large amounts of data, designing efficient system architectures, and implementing appropriate mechanisms for privacy and security assurance. Especially the network security aspect of the IoT is of major importance due to huge amounts of data that the IoT is expected to generate and handle, and considering the limited resources of typical IoT devices. One of the serious security threats are the physical attacks on the IoT devices that operate in remote locations. These are known in the literature as the *node capture attacks*. Motivated by the aforementioned issues, this paper first introduces the background of IoT security and discusses the related challenges. Next, a secure group communication scheme that enables IoT using low energy wireless IP network is described. The proposed approach is based on Shamir's Secret Sharing scheme, which has been enhanced to enable secure group-to-group communication of resource-constrained IoT devices. In particular, we consider the low energy wireless IP networking technology as one of the IoT enablers and the problem of mitigating the negative effects of node capture attacks on IoT devices. Simulation results show significant improvements of the proposed scheme over the traditional public-key based approach.**

***Keywords*—Internet of Things; group communication; secret sharing.**

I. Introduction

Recently, the Internet of Things (IoT) has attracted attention due to its impact in a wide range of application areas, including agriculture, smart grids, industrial control systems, remote healthcare, smart mobility, and road traffic management [1]–[3]. IoT is expected to grow both in terms of its deployment size as well as its expansion to new application areas. The term IoT was coined by Kevin Ashton in 1999 and refers to the connectivity of any entity (also known as "thing"), that has an embedded microprocessor chip, is globally addressable, using, e.g., an IP address, and is connected to a wired or wireless network [4]. This leads to a smart world with ubiquitous computing and provides services that enable remote access and intelligent functionality [5].

IoT enables real-time analysis of data flows that improves the efficiency and reliability of communication systems. For example, connecting all appliances in a smart home can save electricity by efficient monitoring. IoT provides convenience in day-to-day living and makes an intelligent use of resources in a home [6]. Connected devices ranging from sensors to automated transport, will generate huge volumes of data that should be effectively managed and processed.

We recognize the fact that all references to IoT typically involve sensors with different levels of integration to smart devices and the use of heterogeneous networking technologies. Hence, from this perspective, a security scheme concerns the operational and functional aspects of these sensor motes. We use the term IoT to signify such networked devices with sensing elements. Hence, in this paper, we use the term IoT rather than the term Wireless Sensor Network (WSN).

There are large-scale deployments of IoT in infrastructure networks such as water management systems, smart grids, and logistics management. The security risk for an infrastructure that provides such services, termed as *critical infrastructure*, is important. Recent cyber-attacks on critical infrastructure have highlighted the drastic effects on people's lifes and sometimes even in a nationwide scale [7]. The need for a secure and resilient infrastructure and services is evident. In this paper, we identify the most important security threats and requirements for the IoT. We then describe the specific scenario in which we intend to propose a solution. Our proposed scheme is based on Shamir's Secret Sharing scheme [8]. The latter has been enhanced to enable secure group-to-group communication of wireless IoT devices and to mitigate the negative effects of node capture attacks.

The rest of the paper is organized as follows. In Section II, we present the basic IoT architectural elements. In Section III, we discuss a number of representative IoT applications. In Section IV, we identify the most important IoT security challenges. In Section V, we discuss the IoT security risks and secure design requirements. In Section VI, we propose a new secure scheme for the group communication of wireless IoT devices. In Section VII, we present a high-level security analysis, focusing on node capture attacks and replay attacks. In Section VIII, we evaluate our proposed scheme by means of computer simulations, by comparing it with the traditional public-key based approach. We conclude and discuss our future work in Section IX.

II. IoT Architectural Elements

This section presents the main elements of a typical IoT architecture. They are mentioned using a high level taxonomy to help in identifying and defining the components required by the IoT. The three IoT components that can enable intelligent and ubiquitous computing are [9]:

1) *Hardware*: Includes the embedded processing

and communication hardware, sensors and actuators.
2) *Middleware*: Is responsible for on-demand storage as well as the required computing tasks to support data analytics.
3) *Visualization and analysis tools*: Are user-friendly and available on different platforms and for different applications.

In the following, we briefly discuss the most important enabling technological developments that implement the three components indicated above. First of all, the Radio Frequency IDentification (RFID) technology enables the microchips for wireless data communication [10]. RFID has the capability to automatically identify anything within a given range and acts as an electronic barcode. RFID inactive tags are not powered with the use of battery; instead, they utilize the reader's interrogation signal power to communicate with the RFID reader. This has resulted in a wide range of interesting applications, especially in the retail and supply-chain management sector. An example that can be given to explain this concept is its applicability in Intelligent Transportation Systems (ITS), such as the registration stickers or the replacement of tickets. On the other hand, the RFID active tags are powered from their own battery and are able to instantiate the communication. A few examples from applications using RFID active tags are those in port containers that are responsible to monitor cargo [11].

Another important IoT enabler is the WSN technology [12]–[14]. This refers to cost-effective, efficient, and low power miniature electronic apparatuses for usage in remote sensing applications. WSNs have significantly increased the capability of utilizing a sensor network which currently consists of a large number of intelligent sensors and can collect, process and analyze the distribution of valuable data and information that are gathered in a variety of environments. However, the technical challenges that must be addressed in order to exploit better the huge potential of WSNs, are multidisciplinary and substantial in nature. To be more specific, the data of the sensor are shared between sensor nodes and are thereafter sent to a centralized (or distributed) system for analysis.

III. IoT Applications

IoT applications can be classified based on the network availability type, heterogeneity, repeatability coverage, scale, user involvement, and impact. Thus, various IoT applications can be broadly categorized into four application domains [15]: i) personal and home, ii) enterprise, iii) utilities, and iv) mobile.

An IoT application of the first category (personal and home) utilizes the information gathered by the sensor, that is only used by those who own the network. For example, nowadays it is possible for a smartphone using an iOS, Android or Windows Phone operating system to communicate via multiple interfaces (e.g., Bluetooth, WiFi) for interconnecting sensors that measure physiological parameters. Also, these types of IoT applications allow the control of home equipment such as refrigerators and air conditioners, enabling a comfortable home and an efficient energy management [16], [17].

The second category refers to an enterprise-based application. It mainly relates with the businesses and deals with the Network of Things (NoT) within a business environment. The data and the information gathered by these networks is selectively released and is only used by their owners. For instance, a common application that can describe this condition is the environmental monitoring which was developed and implemented to keep track of the number of residents within the building. This can be achieved, for example, through a light sensor. As a result, the sensors in this case represent a fundamental component and used for automation, security, and climate control reasons.

Another important category refers to the utility applications. Such applications are already used by several utility organizations mainly for managing their resources and optimizing, e.g., the electricity cost vs. profit. However, such application requires the deployment and management of expensive network infrastructure in order to be capable enough to monitor critical utilities, hence achieving an efficient and effective resource management. Usually, such networks are used by large businesses (on a national scale) that have the potential to afford costly satellite or cellular communications [18]. Such an IoT application can continuously monitor every electricity consumption and generation point within a house and modify the way electricity is consumed. Hence, it can be said that this application is environmentally friendly as it achieves efficient energy consumption [15].

IV. IoT Security Challenges

Providing appropriate levels of IoT security is an extremely important yet a very challenging task. The IoT infrastructure and devices are sensitive to a number of potential vulnerabilities, attacks, and design challenges. One important implication is that the data generated and used in IoT is subject to user privacy and data integrity attacks. Other issues include the IoT failure due to the physical faults of devices or malicious intrusions. This may be especially complicated in case the data managed by IoT relates to a critical information impacting people's lives, such as energy, transportation, business or health.

Connected things are typically resource-limited devices with small storage capacity and energy. This makes them vulnerable to a number of potential attacks and risks. As a result, sensitive IoT data may be blocked and manipulated, with severe financial and security implications. Security of connected devices may be enhanced by the *security by design* approach, in which all security and privacy risks are addressed in a process of things' design and implementation. In order to improve things against attacks, new security protocols, encryption methods and algorithms must be developed taking into consideration memory and computing limitations of connected devices [19]. Security architecture of IoT should also address the issue of fault tolerance, since device failures may be quite common in an IoT system. Filtering bogus and manipulated data and ensuring data identity are critical tasks for a robust IoT security system.

Connected devices will generate great volumes of data that should be transmitted, processed, managed and analysed. In a centralized IoT approach, data management is primarily realized by cloud computing systems. Therefore, the security of this data to a large extent depends on security measures undertaken by cloud service providers. Data security in cloud systems depends on the protection

of the virtualization process and relies on safe allocation and reallocation of resources [20]. Interactions between the Hypervisor and the Virtual Machines (VMs) must be properly organized to prevent data exposure when resources are reallocated from one VM to another. Such a need arises due to a shared and distributed nature of computer resources in a cloud architecture. Data security may be also compromised by the malicious traffic going from one VM to another. In order to mitigate this risk, traffic monitoring and firewalls between VMs may be used as effective counter-measures. Another useful technique is the segregation and isolation of different VMs classes from each other [21].

IoT security faces a number of challenges that originate from the features of embedded computer devices, RFID, networking technologies, and machine-to-machine (M2M) communication. IoT is susceptible to replay attacks via the compromised communication or attacks that directly target IoT devices. The latter refers to an attacker's ability of observing the network traffic and resending the captured packets at a later time to obtain access to unauthorized resources. According to a typical method of realizing a replay attack, a malicious user can eavesdrop on communications and resend old packets again multiple times in order to waste system or device resources [22].

V. IoT Security Risks and Secure Design Requirements

We consider a scenario that involves a set of wireless devices/nodes, which are equipped with sensing elements and term them as *things*. These things are organized into groups. Each group has a node that provides an interface to connect to the rest of the network. This node, termed as the gateway node (GW), is connected to other similar gateway nodes in the network to form a path to an upstream server. All things in the network communicate with the server in order to deposit the data they generate.

Given this scenario, there are two basic security risks to be considered. Firstly, the wireless communication opens the risk of attackers eavesdropping on the traffic in the network and using the traffic characteristics and the traffic data to break into the network. Being able to gather packet data, enables several typical attacks, such as the replay attacks, the Sybil attacks, and the impersonation attacks. This necessitates that all the communication between the end points is completely private. The packet in transit should not contain any data in a form that is directly readable or easily decipherable. Secondly, there is always a risk of physical capture or damage of the *thing*. In the event of a physical capture, the attempt would be to either manipulate the node to behave in a manner that the attacker decides or to be able to access any secret information that is stored in the *thing* and then to utilize that information to launch attacks.

Based on the two aforementioned risks, we lay down our security requirements:

1) All *things* in the network should be authenticated when they join the network.
2) The gateways are allowed to forward data only from *things* that have already been authenticated.
3) All communication between *things* and with the server must be completely private.

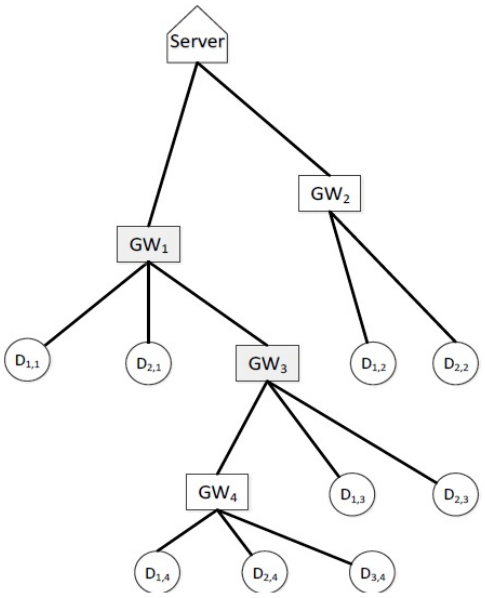

Fig. 1. A hierarchical model for IoT security (simple example).

4) Secret information being stored in individual *things* should be avoided when possible. If there is need to have some secret information to stored in a *thing*, this information must not be sufficient to spoof the identity of the compromised *thing* and authenticate it with the network using that identity. This requirement, therefore, excludes the use of schemes that use a shared secret that is stored in the *thing*.

A simple example of our considered hierarchical model for IoT security is shown in Fig. 1. Note that the hierarchy in the nodes is topological and not necessarily functional from the point of view of the applications. Also, *things* may join or leave the network at any point in time. Hence, the hierarchical dependencies are highly dynamic and may change over time.

Our security design is driven by the requirements mentioned above. The basic design parameter, as is common with all applications for *things*, is the resource efficiency. Computing, storage, and energy are the typical resources whose utilization has to be efficient. Shared secrets, both static and dynamic, that are used by secure schemes require being stored. These secrets, therefore, should require minimum storage resources. Also, the computing requirements of the security scheme should be at a level that the energy consumption is below a predefined limit.

A centralized security scheme often requires shared secrets to be stored on the end devices. Communication within the group also requires shared secrets, causing the number of shared secrets to increase. Such solutions are not scalable since the amount of storage for secrets increases with the number of *things* that are communicated with. In addition to this, the stored secrets should be such that by capturing the secrets of one member of a group, the attacker is not able to take on the identity of the captured member. Therefore, distributed secret sharing, where a key is derived from multiple secrets from within the group, should be considered. The objective is to ensure that the capture of a member will minimally impact the group, the rest of the network, and the provided services.

Finally, the secure scheme should ensure that the communication requirements to implement security are as minimal as possible, since communication utilizes the on-board energy, which is limited. The volume of communications for implementing security, that is, the security communications overheads must be minimal.

VI. The Proposed Secure Scheme

We observe that the network topology of Fig. 1 is a tree with the server as a root. The server could play a limited role in the secure scheme to ensure that the design requirements are conformed to. The *things* (leaf nodes) communicate within the group and with the gateway, when there is a need to send data to the server. If the security scheme is designed in a distributed way, it can be generic enough to be used in a wide range of topologies, such as a *tree topology*, a *mesh topology* and a *partial mesh (cluster-tree) topology*.

In the following sub-sections, we illustrate the process of key generation and group membership verification. We also briefly discuss the processing and storage overhead.

A. Key generation

The server creates a random master key, KS. This key will be used to generate new keys for each child (which must be a gateway when the parent is a server) with ID GW_g (i.e., GW_1 and GW_2 in Fig. 1):

$$KGW_g = F(KS||GW_g) \tag{1}$$

where $F(\cdot)$ is a secure one-way hash function and $||$ is the concatenation operator. The key KGW_g is stored at the corresponding GW_g. The server does not need to store a copy of this key, since it can be generated from the KS when needed. Continuing with the example of Fig 1., a similar procedure is followed by GW_1 and GW_2. We observe that GW_1 has two child IoT devices $D_{1,1}$ and $D_{2,1}$ (the generic notation for IoT devices is: $D_{n,g}$ where n is the device number and g is the group number) and one child gateway, GW_3. Hence, GW_1 will use its master key KGW_1 to generate keys for its child nodes:

$$KD_{n,g} = F(KGW_g||D_{n,g}) \tag{2}$$

$$KGW_{g'} = F(KGW_g||GW_g) \tag{3}$$

where g' is the group of the child gateway. Similarly, GW_3 will generate keys for its child nodes and so on. The keys are stored at the corresponding child nodes. As mentioned before, the parent nodes do not need to store the keys since they can generate keys when required.

B. Group verification

In this subsection, we describe the process of verification of an IoT device in a group controlled by a gateway. This scheme is based on Shamir's Secret Sharing scheme and its extensions [8], [23], and has been adapted to enable secure group communication of resource-constrained wireless IoT devices. Verification is based on a distributed key generation scheme, according to which, the secret is split and distributed among the group members. In particular, assume that the number of nodes in a group is N. Then the secret is split into N parts, and each part is distributed to each one of the N nodes. At least, K parts out of N can be put together to recover the secret. Any less than K parts cannot recover the secret.

Key generation and distribution

Consider a group g controlled by a gateway GW_g. The group members are denoted by $D_{n,g}, n = 1, \ldots, N$. GW_g is responsible for group key generation and distribution. It performs the following steps:

1) It selects two large prime numbers, p and q, so that q divides $p-1$ and the finite field $GF(q)$ is a unique subgroup of $GF(p)$.
2) It selects two random polynomials, $f_1(x)$ and $f_2(x)$, with coefficients in $GF(p)$ and with degrees $K-1$, where K is the minimum number of secret shares required to construct a key.
3) It generates two tokens, $f_1(x_n)$ and $f_2(x_n)$, for each group member $D_{n,g}$.
4) It selects four random integers, $w_{n,1}$, $w_{n,2}$, $d_{n,1}$, and $d_{n,2}$ in $GF(q)$, where $w_{n,1} \neq w_{n,2}$, for each $D_{n,g}$.
5) For each $D_{n,g}$, it selects g_n that is a generator of $GF(q)$.
6) It generates the key $KD_{n,g}$ for each $D_{n,g}$ according to (4), below.
7) It sends $w_{n,1}, w_{n,2}, d_{n,1}, d_{n,2}$, and each g_n, $H(KD_{n,g})$ to every $D_{n,g}$, where $H(x)$ is a one-way hash function.

$$KD_{n,g} = g_n^{(d_{n,1}f_1(w_{n,1})+d_{n,2}f_2(w_{n,2}))} \bmod p \tag{4}$$

Key reconstruction

When receiving a message from an IoT device, $D_{n,g}$, another IoT device, say $D_{i,g} (i \neq n)$, of the same group g will try to validate the sender by reconstructing the sender's key, $KD_{n,g}$. In particular, each $D_{i,g}$ performs the following steps:

1) It computes its corresponding Lagrange component, c_i, based on (5), below.
2) It computes an auxiliary parameter $e_i = g_n^{c_i} \bmod p$.
3) It sends e_i to every other IoT device.
4) It receives $e_n (n \neq i)$ from each $D_{n,g} (n \neq i)$.
5) It computes the key $KD_{n,g}$ based on (6), below.

$$c_i = \sum_{l=1}^{N} d_{n,l} f_{n,l} \prod_{n=1, n\neq i}^{N} \frac{w_{n,l} - x_n}{x_i - x_n} \bmod p \tag{5}$$

where N, as mentioned before, is the number of $D_{n,g}$ in group g.

$$KD_{n,g} = \prod_{n=1}^{N} e_n \bmod p \tag{6}$$

C. Processing and storage overhead

The key generation and storage process, described in the previous subsection, introduces very limited additional processing and storage overhead. Assume that the generation of a single key requires x CPU cycles and the storage of a single key requires y storage units. For example, in the network of Fig. 1 the processing overhead for the server and the gateways GW_1, GW_2 is: $O_S^{proc} = 3x$, $O_{GW_1}^{proc} = 3x$, $O_{GW_2}^{proc} = 2x$, respectively. Note that by convention $O_{D_{n,g}}^{proc} = 0$, since the end devices do not generate any keys. Finally, the storage overhead for every node is the space required to store a single key, since each node needs to store only its own key. That is, $O_S^{stor} = O_{GW_i}^{stor} = O_{D_{n,g}}^{stor} = y$.

VII. Security Analysis

In this section, we present a high-level security analysis for the proposed scheme. Note that the scheme primarily mitigates the node capture attack. Mitigation of the replay attack is not intrinsic to the key generation and reconstruction, but depends on how the elements of the key are communicated to the involved parties.

A. Node capture attack

The proposed scheme is based on splitting the secret key into parts in a manner that a minimum number of parts is required to reconstruct the key. The split parts are distributed among the group members. In particular, for a group of N members, a secret key is split into N parts so that at least K parts are required to reconstruct the secret key. Anything less will not be able to reconstruct the key. An attacker will therefore find it extremely hard to obtain the secret key; this would require him/her to know the number K and also to obtain at least K parts. This means that the attacker must capture K out of the N members. Hence, in our scheme, we can avoid storing the whole secret key in the memory of the IoT device.

B. Replay attack

The replay attack refers to the strategy of capturing packets from a specific node and analyzing them to guess the secret key. Under our scheme, this type of attack is also made harder, since only key fragments are exchanged. In general, when talking about replay attacks, two approaches can be considered. The first is the replay of packets that exchange the key parts. Replaying these packets does not serve any purpose other than a weak attempt of a denial-of-service attack, since the packet has to be received, processed and then discarded by the receiver. The second approach refers to the replay of data packets that are destined upstream. These packets are encrypted and a replay will cause the packet to be received processed and then discarded. Furthermore, exchanging time-stamped messages can act as an additional security layer by ensuring that the messages are recent and genuine. Finally, the use of *nonces* can enhance the mitigation of replay attacks even more [24].

VIII. Evaluation

A. Experimental setup

In this section, we evaluate our proposed scheme by means of computer simulations. We use the Riverbed Modeller 18 simulation tool [25]. Our considered network consists of 10 groups. Each group has $N = 230$ group members. In particular, the members of each group are 10 ZigBee Coordinators, 50 ZigBee Routers, and 170 ZigBee End Devices. Configurations for the aforementioned node types are available in the simulation tool. We generate 10 different mesh topologies and randomly assign nodes into groups with a random key distribution hierarchy. In the simulations, the hierarchy in each topology is static. However, due to the generation of multiple topologies, this approach represents closely enough a realistic dynamic IoT scenario. In Fig. 2, we show an example of modelling the key distribution within two groups using the Riverbed Modeller.

Our aim is to study the impact of the *node capture attack* on the security of the group communication of

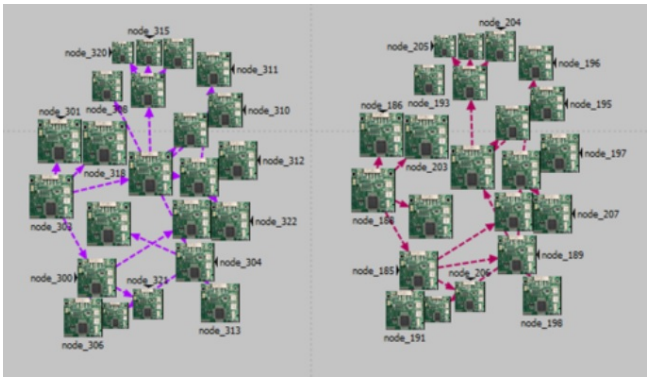

Fig. 2. Modelling the key distribution hierarchy with the Riverbed Modeller.

IoT devices. In particular, we determine the number of compromised nodes when a subset of the nodes has been captured by an attacker. That is, when the attacker has captured one or more nodes, he/she can attack other nodes of the group by exploiting the existing vulnerabilities of the group communication scheme in use. In this study, we compare our proposed secure group communications scheme with the traditional public-key (PK) based authentication.

We simulate different numbers of captured nodes as follows: i) from 1 to 10 captured nodes to launch a *low-intensity* node capture attack, and ii) from 10 to 100 captured nodes to launch a *high-intensity* node capture attack. Afterwards, for each of the two approaches (the PK-based and the proposed), we determine how many nodes the attacker is able to compromise as a result of the node capture attack. Our depicted results are mean values across 10 different random network topologies and group configurations.

B. Results

In Figs. 3 and 4, we present the number of compromised nodes versus the number of captured nodes for the two approaches and for low and high intensity attacks, respectively. For example, according to Fig. 3, if the attacker is able to capture 4 nodes, then he/she can compromise (on average) 30 nodes if our proposed approach is used, and 80 nodes if the traditional PK-based

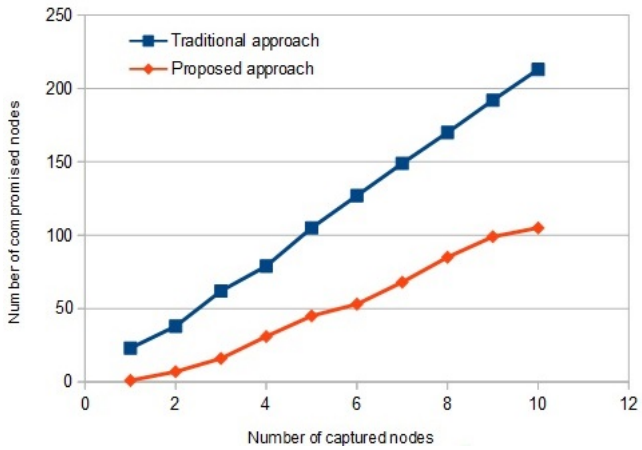

Fig. 3. The impact of a *low-intensity* node capture attack. The number of compromised nodes in the traditional and in the proposed approaches.

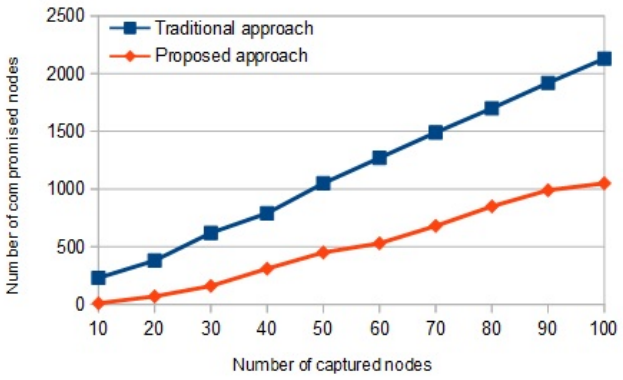

Fig. 4. The impact of a *high-intensity* node capture attack. The number of compromised nodes in the traditional and in the proposed approaches.

approach is used. The superiority of the proposed scheme is even greater when the attack intensity increases. At the same time, as shown in Fig. 4, the PK-based approach has completely failed and a large number of network nodes have been compromised when the number of captured nodes is 100.

IX. Conclusion and Future Work

The IoT is the interconnection of networked and intelligent objects to the Internet or Internet-like infrastructure. The *things* refer to physical objects, such as home appliances, medical devices, and intelligent devices. They are enhanced with computing and communication technology and have the ability to interconnect and communicate through embedded RFID chips, barcodes, or sensors. IoT has begun to form the basis of many critical infrastructures, which are often nation-wide. Often these infrastructures are deployed in remote and unattended locations. Malicious users can physically access them and can cause damage. The risks range from making the devices non-functional to capturing the information within them and re-programming them to behave maliciously.

In this paper we proposed a secure scheme for group communication based on Shamir's Secret Sharing scheme. The enhanced scheme can be used to enable secure group-to-group communication of low-capability IoT devices and to mitigate the negative effects of physical attacks, such as node capture. We study the performance of our scheme against node capture attacks. Our results show that significant security improvements over the traditional PK-based approache can be achieved.

In our future work, we plan to perform comparisons with other proposed approaches for secure group communication, focusing on resource-constrained IoT devices. We are also going to quantitatively evaluate the required storage overhead and take into consideration practical storage and computational capabilities of modern IoT devices. Finally, we plan to enhance our proposed scheme so that protection against replay attacks is also supported.

References

[1] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," IEEE Transactions on Industrial Informatics, vol. 10, no. 4, pp. 2233-2243, Nov. 2014.

[2] H. Shariatmadari, R. Ratasuk, S. Iraji, A. Laya, T. Taleb, R. Jäntti, and A. Ghosh, "Machine-type communications: Current status and future perspectives toward 5G systems," IEEE Communications Magazine, vol. 53, no. 9, pp. 10-17, Sept. 2015.

[3] V. G. Vassilakis, I. D. Moscholios, J. S. Vardakas, and M. D. Logothetis, "On the digital certificate management in advanced metering infrastructure networks," Proc. IEICE Information and Communication Technology Forum (ICTF), Poznan, Poland, July 2017.

[4] D. A. Gratton, The Handbook of Personal Area Networking Technologies and Protocols, Cambridge University Press, 2013.

[5] H. Chaouchi, The Internet of Things: Connecting Objects, John Wiley & Sons, 2013.

[6] P. Parwekar, "From Internet of things towards cloud of things," Proc. 2nd International Conference on Computer and Communication Technology (ICCCT), Allahabad, India, Sept. 2011, pp. 329-333.

[7] C.-W. Ten, G. Manimaran, and C.-C. Liu. "Cybersecurity for critical infrastructures: Attack and defense modeling," IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 40, no. 4, pp. 853-865, June 2010.

[8] A. Shamir, "How to share a secret," Communications of the ACM, vol. 22, pp. 612-613, Nov. 1979.

[9] S. Distefano, G. Merlino, and A. Puliafito, "Enabling the cloud of things," Proc. 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Palermo, Italy, July 2012, pp. 858-863.

[10] X. Zhu, S. K. Mukhopadhyay, and H. Kurata, "A review of RFID technology and its managerial applications in different industries," Journal of Engineering and Technology Management, vol. 29, no. 1, pp. 152-167, March 2012.

[11] K. Hwang, J. Dongarra, and G. C. Fox, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Elsevier Science, 2013.

[12] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: A survey on recent developments and potential synergies," The Journal of Supercomputing, vol. 68, no. 1, pp. 1-48, April 2014.

[13] B. A. Alohali and V. G. Vassilakis, "Security of wireless sensor network (WSN) in smart grid," Proc. 2nd International Conference on Internet of Things and Cloud Computing, Cambridge, UK, March 2017.

[14] I. M. M. E. Emary and S. Ramakrishnan, Wireless Sensor Networks: From Theory to Applications, Taylor & Francis, 2013.

[15] D. Bandyopadhyay and J. Sen, "Internet of things: Applications and challenges in technology and standardization," Wireless Personal Communications, vol. 58, no. 1, pp. 49-69, May 2011.

[16] H. Zhou, The Internet of Things in the Cloud: A Middleware Perspective, Taylor & Francis, 2013.

[17] J. S. Vardakas, N. Zorba, and C. V. Verikoukis, "Power demand control scenarios for smart grid applications with finite number of appliances," Applied Energy, vol. 162, pp. 83-98, January 2016.

[18] R. Ratasuk, N. Mangalvedhe, and A. Ghosh, "Overview of LTE enhancements for cellular IoT," Proc. 26th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Hong Kong, April 2015, pp. 2293-2297.

[19] B. A. Alohali and V. G. Vassilakis, "Secure and energy-efficient multicast routing in smart grids," Proc. 10th IEEE International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, April 2015, pp. 1-6.

[20] B. A. Alohali and V. G. Vassilakis, "Protecting data confidentiality in the cloud of things," Int. J. of Hyperconnectivity and the Internet of Things, vol. 1, no. 1, pp. 29-46, Jan. 2017.

[21] J. Cucurull and S. Guasch, "Virtual TPM for a secure cloud: fallacy or reality?," Proc. RECSI, Alicante, Spain, Sept. 2014, pp. 197-202.

[22] A. Miri, Advanced Security and Privacy for RFID Technologies, Information Science Reference, 2013.

[23] L. Harn, "Group authentication," IEEE Transactions on Computers, vol. 62, no. 9., pp. 1893-1898, July 2013.

[24] S. W. Lee, H. S. Kim, and K. Y. Yoo, "Efficient nonce-based remote user authentication scheme using smart cards," Applied Mathematics and Computation, vol. 167, no. 1, pp. 355-361, Aug. 2005.

[25] Riverbed Modeller, http://www.riverbed.com (last accessed May 20, 2018).