SOS: NDN Based Service-Oriented Game-Theoretic Efficient Security Scheme for IoT Networks

Pushpendu Kar, *Senior Member, IEEE,* Sudip Misra, *Senior Member, IEEE,* Ankush Kumar Mandal, Hao Wang, *Senior Member, IEEE,*

***Abstract*—Internet of Things (IoT) is a network of heterogeneous physical devices connected over the Internet. Each of the devices is capable of collecting and processing data. Due to the connection with the Internet, the IoT devices become more susceptible to attacks by malicious nodes, which may result in privacy loss and security breaches. Thus, network security is necessary for the privacy of transmitted messages. In this context, we propose a scheme, *Service-Oriented game-theoretic Security* (SOS), which provides a simple yet robust security solution for IoT networks. Here, we have amalgamated our scheme with Named Data Networking (NDN), which is more of a data content-specific approach, unlike the traditional IP address search. In this scheme, at first, the hop count between the sender and the receiver is used to generate the public key to encrypt the messages by the sender. When the receiver receives this message, it decrypts the message with the help of the decryption function generated by the sender using the hop count between them as the private key. A non-cooperative Stackelberg game-theoretic model is used to model defenders and attackers, which helps to decide strategies to maximize the payoff (profit) of the defenders to protect the network from malicious attacks. The results are further extended for a modified public key encryption technique, which results in the robustness of the security scheme to be used for all real-life network scenarios. Simulation results show that the proposed scheme, SOS, has a better performance compared to the existing state-of-the-art security schemes, UAKMP and CLS, in terms of time complexity, message overhead, throughput, and attack probability.**

***Index Terms*—Internet of Things, Stackelberg game, payoff, Named Data Networking, strategy, hop count, SOA, malicious nodes, privacy, security.**

I. Introduction

Internet of Things (IoT) is a network of various heterogeneous physical components embedded with the software, operating systems, sensors, and actuators, which communicate among themselves over the Internet. The IoT has molded our world into a cyber-physical system, which is equipped with various smart devices of everyday use. Smart appliances, watches, vehicles [1], and medical equipment are embedded with computing systems and are inter-operable over the network, which store important data, pass it through the gateways, transfer the data to the cloud storage, process over it, and lastly, send it again to the users with the help of implanted software and hardware. Connected over the Internet makes security and privacy of the user's data and action a very important issue to be addressed [2], [3], [4]. Moreover, many devices are manufactured with old and obsolete software systems, which make the network even more vulnerable to attacks [5], [6] by cyber-criminals. Including this, efficient and faster communication between components is always desirable in any kind of network. Therefore, we have proposed a simple yet robust security scheme for the IoT network using hop count between the nodes which protects the privacy of the users as well as makes the communication in the network more efficient and faster. In this paper, we have introduced our security scheme with a network, which provides a basic level of intrinsic security within the network. It is the evolution of the architecture that was earlier based on the host but now on data. This new way of the network not only adds another layer of security but also makes the data transmission faster with the help of cache memory by storing all the earlier requests of other nodes in the network.

A. Motivation

IoT devices are embedded with software, sensors, and actuators, which enable the devices to collect and exchange data between themselves. As the devices are all connected to the Internet, the risk of data getting exposed has increased. There is always a risk for using IoT networks for sharing information, as the private data is vulnerable to be compromised. To mitigate this problem, in this paper, we have tried to establish a simple, but robust, security scheme for an IoT network, which can be installed in a Service-Oriented Architecture (SOA). A service-oriented architecture is shown as a software design where various applications can make use of several services that are offered over a network. The concept of services has been utilized in the naming of the paper only to project its real-time usages outside theoretical applications and simulations. Our security scheme stands valid for various networks such as Wireless Sensor Network (WSN), Local Area Network (LAN), Personal Area Network (PAN), Wireless Local Area Network (WLAN), and Internet of Things. It can also be found in internetwork (where two or more computer networks are connected over some devices), which include extranet (Metropolitan Area Network) and intranet as well.

P. Kar is with the School of Computer Science, University of Nottingham Ningbo China, 199 Taikang East Road, Ningbo, 315100, China. E-mail: pushpendu.kar@nottingham.edu.cn

S. Misra is with the Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, West Bengal, India, 721302. E-mail: smisra@sit.iitkgp.ernet.in

A. K. Mandal is with the ASU School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, 85281, Arizona, United States, Email: akmanda2@asu.edu

H. Wang is with the Department of Computer Science, Norwegian University of Science and Technology, A213, Ametyst-bygget, Gjøvik, Norway. E-mail: hawa@ntnu.no

SOA is a system design model consisting of a collection of services that communicate among themselves through normal data transfers or co-ordinate for providing joint services to other components over the network. In an SOA, the entire procedure is dependent on the connection between the two services; involving the service request to the provider and the corresponding response to the consumer. The previously designed network security schemes are complex and they require multiple message exchanges between the parties, which in turn, reduces the data exchange rates between them, and makes them less secure. The infusion of SOA into IoT security increases the software model's versatility and prominent use in other networks.

In this work, we have introduced dual-level security using hop count-based cryptanalysis along with the intrinsic layer of security at the data level which is added by the NDN [7]. This network uses content caching, which makes the data transmission faster and much more reliable. Usability and scalability are one of the main reasons for switching to NDN which is very suitable for an IoT network. The main idea of using a Stackelberg game-theoretical model in the proposed solution is that it resembles the attacker-defender situation, where the attackers follow the strategies of the defenders. The Nash equilibrium is achieved for the Stackelberg model when the payoff for each of the parties is maximum and both of them have chosen their optimum strategies. Deviating from any of the already computed strategies incurs a loss for them. This work proposes a simple IoT security scheme, which helps the nodes to increase the message exchange rate. In our scheme, the sender and receiver don't have to exchange the public and private keys, as both of them are formed from the hop count which is familiar to both. It reduces the encryption algorithm to a two-way cryptanalysis and thus saves one-third (33%) of the total time taken. Other data schemes are less secure because more message exchanges pave the way for more number of times the data is exposed and there are more chances of attacks in these situations.

B. Contribution

In this paper, we propose SOS, a scheme, which makes the security of an IoT network robust and efficient. The proposed scheme infuses SOA to increase its versatility and re-usability. The proposed scheme is novel and it helps in providing a secure network with the help of hop count between the nodes. The use of hop count to provide security for the network had not been considered in the existing literature. The proposed scheme prevents any malicious node from executing harmful activities in the network. When a malicious node attempts to send a message to any network node, the proposed security scheme prevents the network node from receiving the packet [8] and drops it.

The proposed scheme improves the rate of message transfer and reduces the overall time complexity. The overall contributions of this work are summarized below:

1) Proposed a novel scheme, named SOS, to provide robust security in an IoT network.
2) Incorporated service-oriented architecture in the proposed scheme to make it useful in a service-based IoT network.
3) Evaluated the proposed scheme through simulations and compared it with state-of-the-art schemes.

II. Literature Review

Many researchers have proposed schemes for authentication and authorization in IoT networks. Gupta*et al.* [9] have addressed several issues concerning the safety of the services and devices of a network. We also discuss and analyze the different security implementations in various layers of the IoT network. Perception layer security deals with the acquisition and processing of data through various independent units such as temperature or pressure sensors and actuators. In this layer, the adversary's prime target [10], [11] is to hijack stacked-up raw data and destroy the perception devices in the IoT network. Message (data) filtering techniques exist to put an end to these attacks [12]. Network layer security has the main focus on the transmission of collected data. Therefore, the main security challenge in this layer is the controlling and allocation of network resources. The layer also mainly emphasizes the security of wireless networks because most of the IoT devices are connected over the Internet. Denial-of-service (DoS) attack [13] is one of the common attacks in this layer, which keeps all the network resources busy by corrupting the network protocol or by engaging the IoT network with massive traffic and leaving the IoT services unavailable. Many defensive schemes were deployed to counter this attacking technique [14]. The main importance of the application layer is to provide the services requested by the users. The security challenges in this layer are that the attackers can attack the software and corrupt it. The phishing attack is one such attack, where the adversary gets hold of confidential data by forging the authentication and authorization codes via corrupted emails and various other phishing sites. Secure authorization access can mitigate this attack. Kothamyr *et al.* [15] successfully implemented the first-ever two-way handshaking authentication scheme, which was a breakthrough in transport layer security. The scheme is based on already existing Internet protocols, especially DTLS handshaking and exchange of X.509 certificates, which contain the RSA keys. Hummen *et al.* [16] explored the importance of certificates for peer authentication in the IoT. DTLS certificate had overheads, which provoked the authors to propose three various ideas to minimize the DTLS handshake overhead through pre-validation, session expiry, and handshake authorization.

The authors of this paper wanted to propose and design a security scheme to secure an IoT network. They have considered a network architecture, which is data-centric and more content-based than just data transfer between two IP addresses of the sender and receiver. This architecture not only provides a basic layer of additional security but also instills the cache memory idea in storing the earlier lookups to improve response time. All the previous searches are already listed so that any duplicate searches can be easily found out and sent responses for. In this architecture, there is no need to acquire the IP address repeatedly for connection if the session is over or the location changes. NDN supports multiple user network interfaces without total connection between source

and destination as their data content is consistent for various levels.

Today, almost all smart devices [17], [18] worldwide are enabled with various software and Internet connectivity through which they can communicate and interact. The tiny heterogeneous devices also co-operate with each other to provide a plethora of web services. The software industry slowly, but steadily, drifts towards the SOA, especially physical devices for our daily use [19]. The devices in an IoT network get access to the gateways and other resources (services) in a service-oriented architecture [20].

The initial references of using mathematical models were found in the literature of network security, where non-cooperative game theory had been used to depict the interconnection between malicious nodes and the network devices (nodes) [21]. Here, different mathematical models are taken into consideration for different security problems and threats. These games have a crucial concept of Nash equilibrium which denotes the strategy to be used by one party knowing the strategies of all the other parties in the game. This is known as the best response to the mentioned device (node) [22]. Webb [23] in his book has explained the Stackelberg duopoly model, where many firms have to compete against each other for the same set of customers with their products. The leader firm makes his decision and imposes on the followers and all of them have to make strategies according to the decided strategy (one leader multiple followers model).

Unlike the Cournot model [24], where the strategic decisions are made simultaneously without the knowledge of each other. Han *et al.* [25] observed that in some cases, players select their strategies without knowing the opponents' preferences (payoffs) i.e. Bayesian games. This game model is often used to analyze wireless transmission in a network and so we find it accurate to use in IoT networks. These games [26] are security games between the attacker (selects the target) and the defender (allocates resources).

Wilczyski *et al.* [27] in his book wrote about the different models of Stackelberg game theory which have been implemented earlier to map the attacker-defender scenarios in different security implementations. The authors explained that the attackers (followers) can know the utility of the defender (leader) but the defender does not know the attackers' utility. Wazid *et al.* [28] showed the use of various separate public keys and proposed an important three-factor remote user authentication scheme, User Authenticated Key Management Protocol (UAKMP), which offers several functionality features such as offline node sensing, password protection, and other bio-metric update facilities. The three factors are user smart-card, password, and biometric. UAKMP builds six phases in it which are applied for scheme implementation. The steps involve registration of nodes, users, a user login, key agreement, password update, and new node sense. The only fallback is that it doesn't have any practical testing of cluster nodes and gateways in a real-time IoT network. Yeh *et al.* proposed [29] a scheme, which involves a certificate-less signature (CLS) crypto-system to provide transaction security and communication robustness to mobile users and make payment hassle-free. This scheme eliminates heavy computation and is thus suitable for mobile communication architecture. The shortcoming of the scheme is that it involves the generation of a lot of random numbers and matching the numbers every time gives more scope for malicious nodes to attack and capture the generated key.

Synthesis: A brief study of the existing literature reveals that researchers have explored various types of security attacks in different protocol layers and proposed security schemes. Indeed, network security is our prime concern along with the time complexity and message overhead for secure and faster communication. However, in the existing works, complex security schemes have been proposed which require many message exchanges and in turn, make the network vulnerable to attacks. In this work, we create a public key to transform traditional cryptography into a two-way cryptanalysis to provide robust security to the network. IoT network can be modeled into Stackelberg Game Theory to understand the security game but, in this paper, we have tried to develop a security scheme for the secure transmission of data content among the nodes. Just the game model let us understands how the two parties are positioned in the network but won't allow us to make it more secure. Adding the cryptanalysis is the extra layer of security, which makes the scheme more robust and ensures a lesser probability of attacking the network. Even if an attacker is successful in a brute force attempt to know the hop count, there is no way the attacker is able to decrypt the ciphertext with our decryption function. Amalgamating a data-centric architecture, Named Data Networking (NDN), into our security scheme, SOS, added an intrinsic layer of security and made it more robust and strong. We have adopted NDN and merged it with the NHC table of the networking architecture so that we can get more security. The modified NHC tables and encryption techniques (decryption function) are the novelties of this work.

III. SOS: THE PROPOSED SCHEME

A. Problem Description

Let us consider an IoT network having N nodes (defenders) and M of which are malicious (attackers). The security game is a game between the attackers and the defenders. The defender chooses a mixed strategy and the attackers choose their strategy according to the defenders' strategy. A pure defense strategy of a player i is the mapping of resources, which consists of a subset of the covered targets. A mixed defense strategy of a player i is the mapping of resources, which consists of a subset of strategies, where each of the strategies is selected for both attacking and defending the targets. In the proposed security scheme, the malicious nodes can attack the network nodes, when the malign nodes can access the public key generated from the hop count, which is stored in their neighboring hop count (NHC) table. Thereafter, the malicious nodes attack the network nodes and take their position in the network. Suppose m malicious nodes, ($m \in M$) are identified to attack the network with mixed strategies S_i $\forall i \in \{0,1\}$ to replace n network nodes ($n \in N$).

B. Security Model

Our security solution scheme is merged with the NDN to apply to an IoT network. Whenever a network node requests

for a data packet, an Interest packet is sent with the same name as the data packet. This architecture is different from end-to-end data delivery because here multiple network interfaces can be simultaneously achieved. This is known as a pull-based data delivery system. Then the interest packet goes to the producer node which answers the interest with the subsequent data packet. When the Interest packet has finally reached the end node and the content store is matched, then the Data packet is released. The Data packet takes the same route but in a reverse direction. Our scheme represents the fact that after the Interest packet reaches the node where it is supposed to reach, the data packet is encrypted with the key and then sent back to the node which had requested the data packet. The first step of our modified two-way handshake begins at this stage.

In a service-oriented IoT network, every node is assumed to know its respective hop length from its neighbor nodes. After the service is requested by a user in service consumer mode (by the destination devices), the request reaches the middleware, which checks the request and verifies if that particular request is a service provided by the service provider. This layer acts as a service broker. It also contains the Service Layer Agreement (SLA) [30]. The SLA consists of the price for using each service and the security measures for the services.

In this paper, we use hop count as the basis for security in data communication. Before the reception of every data packet, a node decrypts the packet with the private key generated from the hop count between the corresponding pair of nodes stored in the NHC table of the destination node. If the correct hop count from the NHC table is used for decryption, the destination node receives the data packet, otherwise, it discards the packet.

In a traditional IoT network, there are major security concerns for every layer. In that scenario, the network cannot determine which layer has been compromised in the whole system. But in our scheme, as soon as the malicious node sends the data packet to any unknown network node without knowing the hop count, the packet will be dropped, breaking off the node from the network. Our scheme is robust against a major attack in sensor networks, which is a SYN flood attack where the attacker initiates many half-open handshakes for data transmission, and then the server goes down. It is a protocol attack that establishes half-open connections resulting in denial-of-service (all the resources are used up) to other legitimate nodes in the network.

The proposed model provides security against data tampering or manhandling. When an adversary attacks by placing a malicious node in the network, the new node in the network does not know its hop count from its neighbors. Therefore, it will be unable to send any data into the network, as the destination nodes do not find a matching public key generated from the stored hop count in the NHC table of the node. This renders a malicious node failing to send erroneous data to the network nodes. This security scheme works in any condition irrespective of the type of the source node. When the source node is malicious, it does not know where to send the data, as it is unaware of the hop distance to the next node. This type of attack proves very fatal in maintaining the security of the network. In [5] the authors have shown that the WSN security is very important in an IoT network [31] as it can monitor and track the status of the devices and send the status data to the control center and sink nodes. The main objective of the perception layer of an IoT network is the collection of data. The prime security concerns are the forging of data and destroying the network nodes. All sets of nodal attacks fall under the subset of the concerns we have mentioned in our paper. The node capture attacks, malicious code injection attacks, false data injection, spoofing attacks, and DOS attacks are some of the major security threats addressed in our paper. The threat in a false data injection attack is when an external adversary node captures a network node and installs false data in place of normal data which is measured by the IoT device. Then this malicious data is transferred in the whole network which affects the effectiveness of an IoT. The authors have projected that to defend the network against such attacks. Data filtering schemes should be installed in the network which detect and drop the false data before being received by an IoT node. Our scheme, SOS, is not only effective for these attacks but also for other such aforementioned attacks of the same domain. In this way, the proposed security scheme works to provide security to the IoT devices.

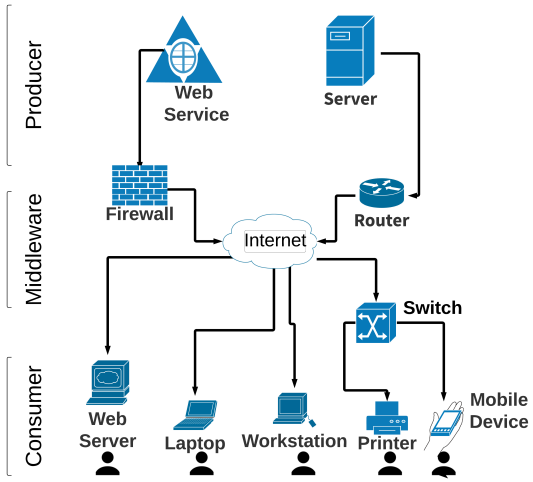


Fig. 1: SOA in IoT

Fig.1, depicts the organization of the proposed scheme in an SOA-based IoT system. The producer nodes offer various web services via servers while the smart devices, such as a personal computer, mobiles, smartwatches, smart appliances are possessed by the users to consume those services. The producers and the consumers negotiate between themselves to avail the offered services. The consumers first contact the middleware and ask to find out a specific service among the various kinds of services provided by the producers. The middleware then checks the number of service requests, costs, associated quality requirements, and matches them with the best possible service. The SLA functions act as a broker between the service producers and the service consumers.

The system model also depicts the cryptanalysis of the proposed security scheme and its dual-level security. The transmitted message has to be encrypted with the public key of

the receiver node, which in this case, is known by the sender node, as both the nodes are connected by the same edge. A general three-way handshake is shown in Fig.2. The first step involves sending the public key to the message sender, the second step is encrypting the message by the sender with the public key, and finally, the message is decrypted by the receiver with the help of the private key. In a synchronous communication, a three-way handshake communication is the most basic protocol of any security-based cryptanalysis scheme. Firstly, the message receiver sends the public key to the sender implying that it is ready for the file transmission. Thereafter, the message sender encrypts the message with the public key and sends it back. At last, the message receiver decrypts the message with the help of a private key. There is no need for any negotiation between the nodes to authenticate each other. Our scheme straightaway reduces these three steps into two by eliminating the first step at the start. If one step out of three is reduced straightway, then it automatically reduces the handshaking time by 33%. As both the sender and receiver nodes know the public key, which is computed from the hop count, it reduces the first step in a synchronous communication, which, in turn, reduces communication time. The time reduction is even more in asynchronous communication, where the sender and receiver communicate in different timelines.

The proposed scheme, *SOS*, allows faster data transmission in a secured manner by simply skipping the first step of the handshake, as both, the nodes already know the common public key. So, simply escaping one out of three stages of message encryption makes it comparatively faster than the general cryptanalysis-based schemes. The source and destination nodes and the NHC table are depicted in Fig.3.

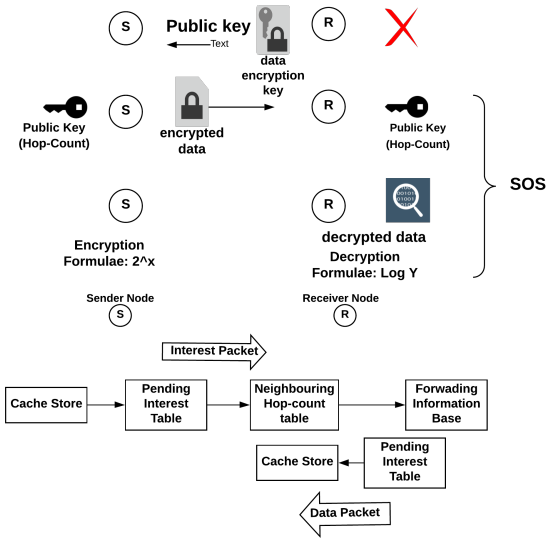


Fig. 2: Cryptanalysis

The authors have tried to utilize a new form of public key cryptography in which there is a public key and a private key but with some restrictions and modifications. One of the most important things to notice is the fact that the public key used in our scheme is not actually known or public to all the nodes in the network. The private key is also only known to the source and destination node where the message is being transmitted. We use the following encryption function for generating a public key from the hop count as shown in Equation 1.

$$P_{ij} = {h_{ij}}^{h_{ij}} \tag{1}$$

where, P_{ij} and h_{ij} are the public key and hop count between nodes i and j. This is done to make the public key more complex for the adversary to predict. This reduces the probability of the attack. The function mentioned above is not a random static function but a dynamic one. The value of this function expression depends on the value of n which changes for every network. We have taken care of the robustness of this scheme so that this scheme can be put to use in other networks without the fear of being compromised. The main reason we have used a dynamic function is so that, after encryption, even if the public key is compromised, the attacker never cracks the actual hop count before encryption. As the function isn't dependent on any constant, at no point in time, the attacker can never get the original hop count even after getting the public key or the decryption function. The message, after being encrypted, is sent to the receiver node and there that node already has a private key to decrypt the message. The receiver uses the following decryption function to get back the hop count from the public key after receiving the data packet.

$$F_{decry} = Log_n P_{ij} \tag{2}$$

where K is the ciphertext (encrypted message). The authors have chosen this modified version of asymmetric key cryptography where the public key for every message transmission is different from each other and no other knows the public key for any other message transmission apart from the one sent by the node itself. There have been previous public key encryption techniques but calculating the product and factoring a large prime number is very computationally expensive and even time-consuming. Here the public key can be formed easily from the hop count and the time complexity is also minimal $O(1)$ and for prime factorization in the traditional methods, the time complexity is $O(Log\ n)$. One of the cons of such asymmetric encryption is that they have to authenticate the received message (digital signatures) from a third party. They would verify the reliability of the public key and provide the encrypted ciphertext with digital signatures. In our modified encryption technique, there is absolutely no inclusion of any third party for authentication. Only one node apart from the receiver knows the hop count and that has to be the sender. The verification is done at the time of the message being sent. This saves an entry point of attack by any malicious nodes. The attacker can compromise the third-party node and hijack the network through that but we terminated that scope. Thus our modified public key encryption process has the pros of both symmetric and asymmetric encryption techniques and at the same time lacks the cons of asymmetric algorithms.

In this work, we have used the encipherment mechanism to provide data confidentiality to the packets sent from one node to another node. This process encrypts the data to a non-readable form for unintended persons (malicious attackers). The authors have modeled the problem of network security

into a Stackelberg game between the attackers and the defenders. Adding the cryptanalysis is the extra layer of security, which makes the scheme more robust and ensures a lesser probability of attacking. Even if an attacker is successful by using the brute-force technique to know the hop count, there is no way for him to encrypt the message with our encryption factor $({h_{ij}}^{h_{ij}})$ where h_{ij} is the hop count from node i to j.

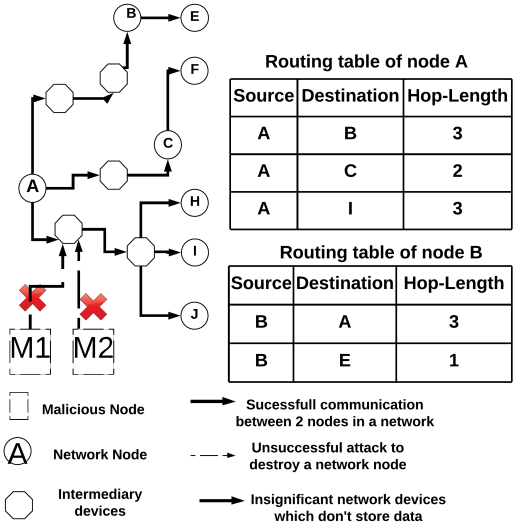
The minimum height of a tree with n nodes is 2n and for security analysis, we always need to consider the worst-case for a defender. The less height of the leaf nodes paves more scope for the attackers to predict the hop count for a successful attack. For a graph with n nodes, the hop count varies from 1 to $2n$. The probability of guessing the public key correctly becomes:

$$P_{hc} = \frac{1}{2n^{2n}} \tag{3}$$

In equation 3, we can clearly see how guessing the probability of any hop count has become very computationally expensive even using the brute force method.

C. Preparation of Neighbor Hop count Table

The scheme, SOS, requires to prepare an NHC table for a node so that only the neighboring nodes can share the hop count among themselves and generate a public key to encrypt the message to be transferred.



Routing table of node A

Source	Destination	Hop-Length
A	B	3
A	C	2
A	I	3

Routing table of node B

Source	Destination	Hop-Length
B	A	3
B	E	1

Fig. 3: IoT Network and NHC Table

In Fig.3, the network backbone shows that any node is connected with another node by only one edge and a node can be connected to more than one node. The dotted lines represent the service requests by the service consumer and the bold lines depict the services provided by the providers. The solid circles represent the end IoT devices and the dotted squares represent the malicious nodes, which try to disrupt the network by sending malicious data to the regular nodes. The crossed lines depict the malicious nodes which fail an attempt, due to lack of next-hop information. The number of networking components on an edge is referred to as the hop count. As an example, the distance between node A and B is 3 hop units. The NHC table in the figure shows a portion of a general routing table, which stores hop count from a node to its neighbor nodes. Each of the network nodes knows the hop length from its adjacent nodes. The other nodes in the network or any malicious node are unaware of the hop count between the neighbor nodes. So, without knowing the hop count, the adversary cannot pass any malicious data in the network, which results in the data being discarded from the network. The problem in question is not a simple modeling of attacker and defender nodes in a network, but we also wanted to develop a security scheme for data transmission having the minimum time complexity and message overhead. There have been many schemes previously designed which have more message exchanges in the scheme, which opened up gates for more attacks by malicious nodes. We have considered the Named Data Networking so that we can use the concept of data caching for faster transmission. Our only goal is to design a scheme that is not just a theoretical model but also something which considers all the cases in a real-life scenario and has tried to abate the chances of a vulnerability attack. Stackelberg's Theory alone doesn't show the method to protect our network from any such attacks. We are dealing with a very complex problem here which concerns the safety of the network and if any node in the network even gets hijacked, the network doesn't leak any data.

D. Proposed Solution

Let the security function $A(s)$ have a maximum value of S_0 when the network is fully secured and reduce to 0 when the network is sabotaged. The network cost is expressed as:

$$C(s) = C_i s_i \tag{4}$$

for the i^{th} node in the network. Let C_1 be the cost for data transmission in the network. The cost of a defender is:

$$C(s_1) = C_1 s_1 \tag{5}$$

If the attacker attacks a node knowing the exact encrypted hop count value from the NHC table of a regular node, then the cost for network transmission is:

$$C(s_2) = C_2 s_2 \tag{6}$$

The total security of the network is:

$$A = S_0(1 + s) \tag{7}$$

where, s is the net security in the IoT network, such that

$$s = s_1 - s_2 \tag{8}$$

where s_1 is the security factor for the defender and s_2 is the attacking chance of the attacker. The main basis of the equation came from the fact that NDN provides an intrinsic level of security which in our case is S_0. The authors of this paper have also researched the fact that this intrinsic security, S_0, can be combined with the net security of the network, s, to form an additional factor in the total security of the network. Apart from the intrinsic network security, the combined net security and the intrinsic security makes up the total security of the network We solve this game by backward induction so that

we can find the subgame perfect Nash Equilibrium. However, the attacker knows the security function of the defender and also its cost function. But the defender knows neither the cost function nor the security function of the attacker.

We consider the function $A(x)$ as the encryption formula of the hop count, where x is the hop count to jump to the next node. So the attacker's chances of an attack are $A(s)$, which reduces from the original total security of the network. It is intended to find the best response of the attackers for every choice of the strategies of the defenders. As attackers already know the choice of the defenders, they provide the best response $s_2(s_1)$ possible for certain choices of the defenders. To attain Nash Equilibrium, we need the maximum payoff of the defenders. The attacker's profit function is given by $P(q_2, q_1)$, and the best response to any choice q_1 is found by partially differentiating the profit function with respect to s_2.

$$\frac{\partial}{\partial s_2}\{(A - C) * s_2\} = 0 \tag{9}$$

which gives

$$\begin{aligned} \frac{\partial}{\partial s_2}[\{S_0(1 + s) - C\}s_2] = 0 \\ s_2(s_1) = \frac{1}{2}(1 + s_1 - \frac{C}{S_0}) \end{aligned} \tag{10}$$

After partial differentiation, we have done a partial double differentiation to check if the critical point is a maximum or a minimum point. If the $\frac{\partial^2 A}{\partial s_2^2}$ is negative it means that the slope of the curve is negative. The critical point is a minimum point, which indicates that this strategy has the minimum payoff for the attacker and the network remains secure. Similarly, if the $\frac{\partial^2 A}{\partial s_1^2}$ is positive, it means that the slope is positive of a convex curve. The critical point is a maximum point, which indicates that the strategy has the maximum payoff for the defender and the network also remains secure.

Double differentiating the security function A with respect to s_2, we see that:

$$\frac{\partial^2 A}{\partial S_2^2} = -2S_0 < 0 \tag{11}$$

Therefore, the equilibrium point is known to be a minimum point, which protects the motive of taking a suitable security function. However, the defender knows that the attacker chooses the strategy $s_2(C)$ based on the network cost C. If the defender chooses s_1 and the attacker chooses $s_2(s_1)$ as the best response, then the defender's profit is:

$$\begin{aligned} \frac{\partial}{\partial s_2}\{(A - C) * s_1\} = 0 \\ \frac{1}{4}(s_1 - 1 + \frac{C}{S_0}) = 0 \end{aligned} \tag{12}$$

Hence, partially differentiating defender's profit function with respect to attacking strategy (s_1) we maximize defender's security payoff to find out the Nash equilibrium of defender's strategy, s_1^*:

$$\begin{aligned} \frac{\partial A(s_1, s_2)}{\partial s_1} = 0 \\ s_1^* = \frac{1}{3}\left(\frac{C}{S_0} - 1\right) \end{aligned} \tag{13}$$

Further double differentiating the security function A with respect to s_1 we find:

$$\begin{aligned} \frac{\partial^2 A}{\partial S_1^2} \\ = \frac{\partial}{\partial s_1}(S_0 + 2S_0 s_1 - s_0 s_2 - C) \\ = 2S_0 > 0 \end{aligned} \tag{14}$$

Therefore, the equilibrium point is known to be the maximum point, which protects the objective of selecting a suitable security function. The network will be completely secured at the equilibrium point as the defenders have the maximum payoff strategy to defend the network and the attackers have the minimum payoff strategy to attack.

E. Nash Equilibrium

The proposed security scheme, SOS, reaches an equilibrium point where both the parties (attackers and defenders) have the maximum payoff. That point in a game-theoretic model is referred to as a Nash Equilibrium point. The most optimal strategies for both the defender and the attacker are:

$$s_1^* = \frac{1}{3}\left(\frac{C}{S_0} - 1\right) \tag{15}$$

$$s_2^* = \frac{1}{3}\left(1 - \frac{C}{S_0}\right) \tag{16}$$

The above solutions constitute the mixed strategy subgame with perfect Nash equilibrium for the security Stackelberg game between the attacker and the defender. The attacker chooses an attacking strategy after knowing the defender's strategy, while the defender chooses any strategy of its choice. The physical significance of the Nash equilibrium is that the defender does not get more benefit by choosing any other strategy other than the strategy in Nash equilibrium. The strategy at Nash equilibrium has the maximum payoff, which means that the network is most secured at the equilibrium point. On the other hand, the Nash equilibrium for the attacker implies that the attacker is successful in destroying the security of the network and its payoff is also maximum for the strategy at Nash equilibrium. The physical significance of the Nash equilibrium is that knowing the defender's strategy, the attacker won't change its strategy to gain maximum from the attack which again, in turn, is beneficial for the defender. In this equilibrium, both the parties are at their best strategies and so can't deviate from that. Nash equilibrium is attained when both the attackers and defenders are in the game with their known strategies and the game takes place.

F. The proposed algorithm

In SOS, a node can communicate with another node in the network if the destination node verifies that the hop count

received by the packet matches the same between the source and the destination node. The process of message verification and security of the proposed scheme is depicted in Algorithm 1. First, the algorithm creates an NHC table for every node in the network. Then every node, on receiving the message, checks the NHC table to compute the next shortest path to reach a neighbor node and then transmits the message to that node.

Any external malicious node does not have the information regarding the NHC table and the hop counts to jump to the next network node. If a malicious node tries to communicate with a legitimate network node using an arbitrary hop count, the malign node fails as the hop count does not match the hop count in the NHC table, which results in the packet being dropped.

As soon as a node generates an Interest Packet, it is first checked in the Cache Store to see if the Data Packet needed by the node is already present as a result of earlier requests. Then it goes on to the Pending Interest Table to check if an earlier request has already been made, which has not been fulfilled yet. Then, if the requested data packet isn't present in any of them, the packet checks for the hop count from the NHC table and then passes on to the Forwarding Information Base. After the request has been responded with a Data Packet, initially, the Pending Interest Table is checked to see the status of any pending requests. Then data content is stored in the Cache Store so that for any future requests, the responses are stored and can be easily retrieved from this store. Every NDN device maintains these data structures. The path for Interest Packet is from the source node to destination and the reverse for a Data Packet which brings the data content back to the source node.

The proposed scheme prevents a malicious node from attacking a network node by encrypting the messages using the public key, which is computed from the hop count between the source and the destination nodes. A destination node receives a packet from a source node only if the decrypted hop count in the packet matches the hop count between the source and the destination nodes stored in the NHC table of the respective nodes. By this algorithm, a malicious node is neither able to inject erroneous packets into the network, nor it can receive a packet from the network nodes, as the malicious node does not know the hop count to the network nodes.

***Theorem** 1:* The best-case and the worst-case time complexities of the proposed scheme *SOS* are $O(1)$ and $O(n)$, respectively, where n is the number of nodes in the network.

Proof: In the presented scheme, SOS, let the cryptanalysis and transition from one node to another take constant time. In the best case, there exists two regular nodes and one attacking node in the network. Here, only one node has to be traversed, so that the distance can be covered in unit time. So, the best case time complexity is O(1). The worst-case scenario arises when all the network nodes are attacked by malicious nodes. The regular nodes have to perform the two-way cryptanalysis in unit time for a single node. The time computation for two-step cryptanalysis for one node takes a unit time. So, for n nodes take '$a \times n$' units of time. If b is the constant time required for checking the NHC table, the time complexity for the scheme, SOS, for n nodes is:

$$T(n) = an + b = O(n) \tag{17}$$

Algorithm 1 SOS

Inputs:
N : Number of Nodes
$rou\_table[][]$: NHC Table
Hop_{ij} : hop count between two nodes i and j
Output:
prn: Boolean variable, which indicates whether the packet is received or not received; if 0 pkt is not received else received

Begin
for i=1 to N **do**
  for j=1 to N **do**
    $rou\_table[i][j] \leftarrow Hop_{ij}$
  end for
end for
if $packet \neq 0$ **then**
  $P_{hop} \leftarrow$ hop count of the received packet
  $S_{id} \leftarrow$ source id of the received packet
  $D_{id} \leftarrow$ destination id of the received packet
end if
if $P_{hop} == rou\_table[S_{id}][D_{id}]$ **then**
  $prn \leftarrow 1$
else
  $prn \leftarrow 0$
end if
return prn
End

***Theorem** 2:* If an adversary is successful in attacking any single node in the network, then the probability of another successful attack on its immediate neighbor is also increased.

Proof: In the proposed scheme, the information of the hop count of the immediate neighbors is stored in the NHC table and only the two connecting nodes know about the hop count between them. If a malicious node gets the hop count value from the NHC table, then it can successfully exploit its neighboring nodes in the network. In a network of n nodes, the maximum number of neighbor nodes any particular node can be connected to is $(n-1)$. So, from the $(n-1)$ hop counts in the table, the malicious node has to choose anyone and the probability becomes:

$$p_{new} = \frac{1}{n-1} \tag{18}$$

IV. SIMULATION RESULTS

A. Simulation Configuration

We have used MATLAB simulator to simulate the proposed scheme, SOS. In the simulation, we have deployed 50-400 nodes randomly over a terrain of 500×500 m^2. The following parameters have been considered in the simulation for performance evaluation:

1) *Attack Probability*: The probability that the malicious node can successfully attack a regular node in a network

by decrypting the public key (hop count) from the message. Let s_i be the value of the i^{th} attacking attempt of a malicious node to a legitimate node of a network having total N nodes. If there are n number of successfully attacking attempts then the attacking probability, ap, is computed as:

$$ap = \frac{\sum_{i=1}^{N} s_i}{n} \tag{19}$$

2) *Message Overhead*: The total number of control messages sent for implementing the proposed scheme. The message overhead of the proposed scheme includes the number of messages sent during the preparation of the NHC table.
3) *Throughput*: Throughput is computed as the number of messages sent per unit time for implementing our proposed scheme, i.e., the message count per round of computation.
4) *Computation Time*: The total time taken to find out the minimum length path between two nodes and the time taken to check if a malicious node can successfully decrypt the public key (hop count), is considered as the computation time.

TABLE I: SIMULATION PARAMETERS

Parameter	**Value**
Number of nodes	50-400
Communication Range	150
Increase in number of nodes for each simulations	50
Number of node levels	8
Range of generated random hop count	10
Range of coordinates	250-2000
Range of maximum edge lengths	56-86

B. Results and Discussion

In this section, we discuss and analyze the simulation results of the proposed scheme, SOS. The probability of attacks by malicious nodes with a varying number of network nodes is shown in Fig.4. The plot shows that the attacking probability of the malicious nodes increases with the increase in the number of regular nodes in the network and decreases in the maximum value of the hop count between network nodes. The attacking probability of a network increases with the decrease in the maximum hop count between the network nodes because the probability of matching the correct hop count by the malicious nodes increases, as the value of the maximum hop count decreases.

The message overhead of the SOS with the increasing number of network nodes is presented in Fig.5. The sending and receiving of "HELLO" messages are required to form the NHC table. The increase in the number of nodes in a network corresponds to the computation of more hop counts, which results in more messages being sent and received. Therefore, the message overhead of the network with the proposed security scheme increases with the increase in the number of nodes in the network.

The throughput of the network with the varying number of network nodes is presented in Fig.6. In the plot, we see that, due to the increase in the number of nodes in the network, the message overhead gradually increases. The more the number of network nodes, the more the number of messages transmitted through the network, which in turn, increases the throughput of the network. The plot also presents the throughput for different maximum hop lengths between the network nodes with a varying number of nodes in the network.

In the plot, we can also see that the throughput of the network does not change with the increase in the maximum hop count between the network nodes. The possible reason is that the increase in the maximum hop length between nodes does not change the number of message transfers in the network. The change of computation time of the proposed scheme with a varying number of nodes in the network and varying maximum hop length between network nodes is plotted in Fig.7. From the plot, we can infer that, with the increase in the number of nodes in the network, the computation time also increases gradually and there is not much change in computation time with the change in maximum hop length between the network nodes. The possible reason behind this is that, with the increase in the number of nodes in the network, the number of neighbor nodes of any node in it also increases, which in turn, increases the preparation time of the NHC table of every individual node in the network.

However, with the increase in the hop length between the nodes, the number of neighbor nodes of any node in the network does not change much, which keeps the computation time nearly unchanged.

The attacker's and defender's optimum strategies are represented in Fig.8. The Nash equilibrium points for the proposed scheme lies on the straight-line, as found from our simulations. The coordinates of the points in this 2-D plane are positive for the defender and negative for an attacker, as per the Nash equilibrium of strategies. In our paper, the graph for the Nash equilibrium is shown in Fig. 8. The figure represents the attacking strategies between the defenders (X-axis) and the attackers (Y-axis). The straight line with a negative slope shows that the defenders have to have a positive strategy (to defend the network) and the attackers have a negative strategy against the security of the network. The straight-line aligns well with our assumption that both of their strategies are linearly dependent on each other and the change of any of the strategies causes them to deviate.

Through this security scheme, we want to show that, there should be a positive strategy for the defender to defend the network and protect against the malicious nodes from attacking it and that the attacker should have a negative value of optimal strategy, so that the malicious nodes can have the minimum payoff, which is again, highly desirable for network security.

s_1^* and s_2^* are the Nash equilibrium strategies of defenders and attackers, respectively. s_0 is the highest attainable security in the network, the value of which is 1.

From the simulation, we can see that, for all the network nodes, there is at least one case in which the attacker cannot successfully attack. Here, C is the cost measured in terms of

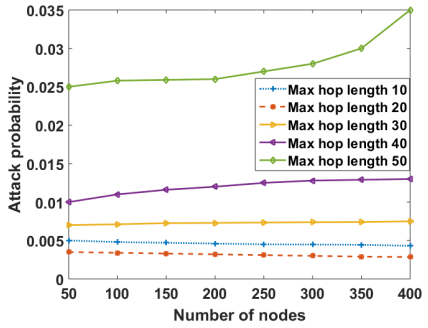


Fig. 4: Attack probability

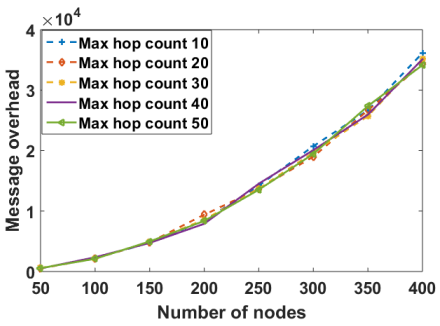


Fig. 5: Messages overhead

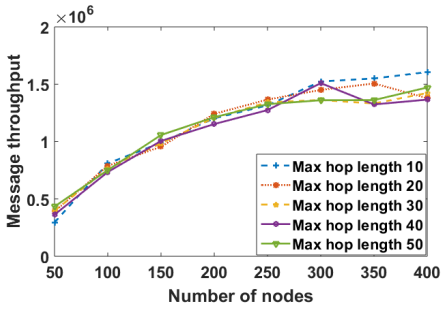


Fig. 6: Throughput

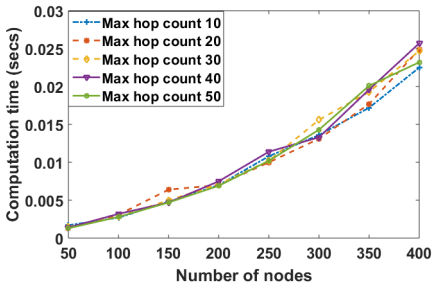


Fig. 7: Computation time

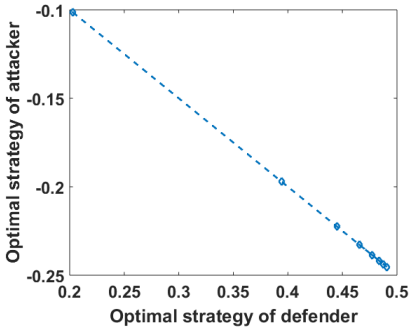


Fig. 8: Nash equilibrium strategies

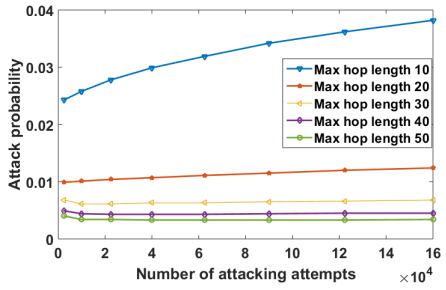


Fig. 9: Successful attack

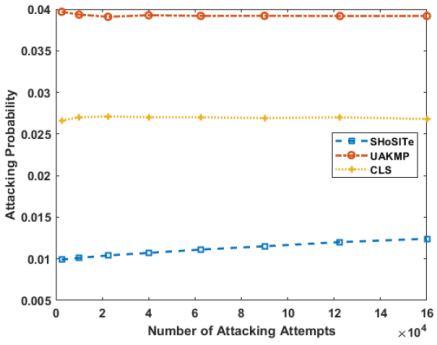


Fig. 10: Attack probability versus number of attack attempts

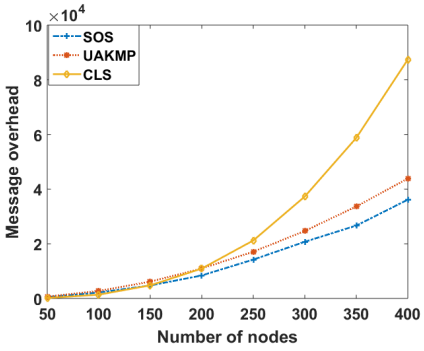


Fig. 11: Message overhead versus varying number of nodes

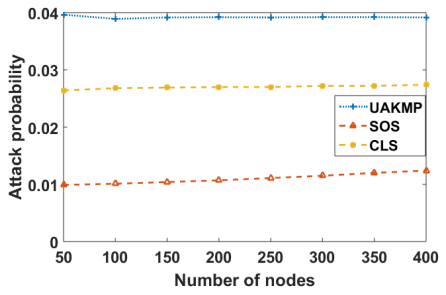


Fig. 12: Attack probability versus varying number of nodes

throughput of the regular network nodes to find out the Nash equilibrium point of the proposed strategy, as shown in the following equations.

$$s_1^* = \frac{1}{3}\left(\frac{C}{S_0} - 1\right)$$
$$s_2^* = \frac{1}{3}\left(1 - \frac{C}{S_0}\right) \quad (20)$$

The variation of attacking probability with the variation in the number of attacking attempts and maximum hop length between nodes in the network is shown in Fig.9. The plot shows that the attacking probability increases with the increase in the number of attack attempts and is least for the case with the maximum hop count. The possible reason is that the increase in the number of attacking attempts increases the number of successful attacks by the malicious nodes and larger hop lengths increase the chances of matching hop count randomly generated by the malicious nodes to successfully attack the network nodes.

The plot in Fig.10 presents the comparison of the attacking probability of the proposed scheme with the existing schemes for varying attacking attempts. In the plot, we can see that the attacking probability of SOS increases with the increase in the number of attack attempts. The possible reason is that, in SOS, the increase in the number of attack attempts increases the probability of matching hop count, which is randomly generated by the malicious nodes.

Fig.11, shows that, with the increase in the number of network nodes, the message overhead of all the schemes is increasing. But the increase in message overhead is the lowest in SOS. On the other hand, the UAKMP scheme has the highest increase in message overhead with the increase in the number of nodes in the network. The possible reason is that, in the

CLS and UAKMP schemes, apart from initial registration, the authentication of delivered messages requires a key exchange, which increases the number of message exchanges between the network nodes. So the existing schemes have a higher message overhead compared to the proposed scheme.

Fig.12 shows that the attacking probability using SOS is the least as compared to that in CLS and UAKMP. Apart from SOS, the existing schemes have less relation of proportionality with the increasing number of network nodes. They are directly associated with the number of message exchanges between a user and the service provider in the service-oriented IoT architecture. In CLS and UAKMP, for every node, the provider has to provide the same set of registration ids and public keys to access the service. Both of the existing security schemes have higher attacking probability compared to the proposed scheme, however, the attacking probability of the existing schemes does not change with the increase in the number of nodes in the network.

As the authentication of the proposed schemes is performed using key exchange, the increase in the attacking attempts keeps the attacking probability of the existing schemes near-constant. Authentication in the proposed scheme does not depend on key exchange, so the attacking probability of SOS is lesser than that of the CLS and UAKMP schemes.

V. CONCLUSION

In this work, we studied the problem of higher time complexities and message overheads of existing IoT security schemes having also high attacking probability. The existing schemes proposed two-step cryptanalysis, which results in more data exchanges which again leads to higher chances of attacks by the adversaries. To solve this problem and improve the security of an IoT network, we proposed a Stackelberg game-based security scheme, SOS, that provides a simple but robust security solution with reduced attacking probability and message overheads. The proposed scheme uses hop count between the source and destination nodes to generate the public key by using an exponential encryption function. The proposed scheme reduced the three-step cryptanalysis to a single step by maintaining a hop count in the NHC table of both the source and destination nodes.

Simulation results showed that SOS outperforms the state-of-the-art schemes in terms of the attacking probability and message overhead. The limitation of SOS is that it fails to secure the insignificant networking devices, which only forward data packets and do not form or store the NHC table (e.g. switches and hubs). Our scheme is very relevant and functions with total support for intermediary network devices such as routers, wireless access points (WAP). For a successful execution of the scheme, the end devices are connected to the communication links, switches, and routers. Middleware denotes the software layer which connects the operating system and the applications which use those OS.

The authors have tried to study the domain to get the background knowledge and understand the complexities which are present to innovate a new security scheme. Our security scheme can be considered for cyber-physical systems, WSN, IoT networks. We have added a layer of security so that when implemented on real-time networks, the scheme doesn't come off as very trivial. We have added the data-centric NDN to utilize content caching for faster transmission. We have modified the encryption methods from public key cryptography so that our scheme is more efficient and faster and has fewer scopes for attacks from malicious nodes. The authors have configured the security scheme to be fit for any wireless sensor network and IoT network. To make our scheme more robust and sturdy, we had even integrated it with Service-oriented architecture. This increases its utility and makes it applicable for any real-time network. Some simple use cases can be your phone, office laptop, personal computer, office WiFi and office server together form a network that is susceptible to attacks by any malicious attackers. In this scenario, our scheme comes into play for providing security to this network.

In the future, we plan to implement our scheme on a real testbed and secure all the networking devices in an IoT network. As the proposed scheme only works on an IoT network with stationary nodes, in the future, we have a plan to extend the scheme to work on a mobile IoT network.

VI. ACKNOWLEDGMENT

The authors would like to extend sincere thanks to the University of Nottingham Ningbo China for supporting this research project under the Faculty Inspiration Grant (I01190900047).

REFERENCES

[1] E. Municio, G. Daneels, M. De Brouwer, F. Ongenae, F. De Turck, B. Braem, J. Famaey, and S. Latré, "Continuous athlete monitoring in challenging cycling environments using iot technologies," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 875–10 887, 2019.

[2] S. Kulkarni, S. Durg, and N. Iyer, "Internet of Things (IoT) security," in *Proceedings of the* 3^{rd} *IEEE International Conference on Computing for Sustainable Global Development (INDIACom)*, Banff, Alberta, Canada, 2016, pp. 821–824.

[3] M. Ammar, G. Russello, and B. Crispo, "Internet of Things: A survey on the security of IoT frameworks," *Journal of Information Security and Applications*, vol. 38, pp. 8–27, 2018.

[4] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in internet-of-things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1250–1258, 2017.

[5] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125–1142, 2017.

[6] M. Radovan and B. Golub, "Trends in IoT security," in *Proceedings of the* 40^{th} *International Convention on IEEE Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, May 2017, pp. 1302–1308.

[7] D. Saxena, V. Raychoudhury, N. Suri, C. Becker, and J. Cao, "Named data networking: a survey," *Computer Science Review*, vol. 19, pp. 15–55, 2016.

[8] R. Li, K. Makhijani, H. Yousefi, C. Westphal, L. Dong, T. Wauters, and F. De Turck, "A framework for qualitative communications using big packet protocol," in *Proceedings of the ACM SIGCOMM 2019 Workshop on Networking for Emerging Applications and Technologies*, 2019, pp. 22–28.

[9] K. Gupta and S. Shukla, "Internet of Things: Security challenges for next generation networks," in *Proceedings of the IEEE International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH)*, Noida, India, Aug. 2016, pp. 315–318.

[10] X. Yang, J. Lin, W. Yu, P.-M. Moulema, X. Fu, and W. Zhao, "A novel en-route filtering scheme against false data injection attacks in cyber-physical networked systems," *IEEE Transactions on Computers*, vol. 64, no. 1, pp. 4–18, 2015.

[11] J. Granjal, E. Monteiro, and J. S. Silva, "Security for the internet of things: a survey of existing protocols and open research issues," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1294–1312, 2015.

[12] J. Lin, W. Yu, and X. Yang, "Towards multistep electricity prices in smart grid electricity markets," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 1, pp. 286–302, 2016.

[13] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zualkernan, "Internet of things (IoT) security: Current status, challenges and prospective measures," in *Proceedings of the* 10^{th} *International Conference for Internet Technology and Secured Transactions (ICITST)*, London, UK, Dec. 2015, pp. 336–341.

[14] S. U. Maheswari, N. Usha, E. M. Anita, and K. R. Devi, "A novel robust routing protocol RAEED to avoid DoS attacks in WSN," in *Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES)*, Tamil Nadu, India, Feb. 2016, pp. 1–5.

[15] T. Kothmayr, C. Schmitt, W. Hu, M. Brünig, and G. Carle, "DTLS based security and two-way authentication for the Internet of Things," *Ad Hoc Networks*, vol. 11, no. 8, pp. 2710–2723, 2013.

[16] R. Hummen, J. H. Ziegeldorf, H. Shafagh, S. Raza, and K. Wehrle, "Towards viable certificate-based authentication for the internet of things," in *Proceedings of the 2nd ACM workshop on Hot topics on wireless network security and privacy*. ACM, 2013, pp. 37–42.

[17] R. Chen, J. Guo, and F. Bao, "Trust management for service composition in SOA-based IoT systems," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Istanbul, Turkey, Apr. 2014, pp. 3444–3449.

[18] Y. Zhang, L. Duan, and J. L. Chen, "Event-driven SOA for IoT services," in *Proceedings of the IEEE International Conference on Services Computing (SCC)*, Anchorage, Alaska, USA, Feb. 2014, pp. 629–636.

[19] W. Lv, F. Meng, C. Zhang, Y. Lv, N. Cao, and J. Jiang, "Research on Unified Architecture of IoT System," in *Proceedings of the IEEE International Conference on Computational Science and Engineering (CSE) and Embedded and Ubiquitous Computing (EUC)*, vol. 2, Guangzhou, Guangdong, China, Jul. 2017, pp. 345–352.

[20] V. Issarny, G. Bouloukakis, N. Georgantas, and B. Billet, "Revisiting service-oriented architecture for the IoT: a middleware perspective," in *Proceedings of the International Conference on Service-Oriented Computing*, Macau, China, 2016, pp. 3–17.

[21] S. M. Mousavi, M. Moghadasi, and G. Fazekas, "Dynamic resource allocation using combinatorial methods in Cloud: A case study," in *Proceedings of the* 8^{th} *IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, Debrecen, Hungary, Sep. 2017, pp. 73–78.

[22] K. Leyton-Brown and Y. Shoham, "Essentials of game theory: A concise multidisciplinary introduction," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 2, no. 1, pp. 1–88, 2008.

[23] J. N. Webb, *Game theory: decisions, interaction and Evolution*. Springer Science & Business Media, 2007.

[24] V. Radonji, V. Aimovi-Raspopovi, A. Kosti-Ljubisavljevi, and S. Mladenovi, "Competitive pricing using cournot game in next generation networks," in *the* 10^{th} *International Conference on Telecommunication in Modern Satellite Cable and Broadcasting Services (TELSIKS)*, vol. 1, Oct 2011, pp. 297–300.

[25] Z. Han, D. Niyato, W. Saad, T. Başar, and A. Hjørungnes, *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge university press, 2012.

[26] J. B. Clempner and A. S. Poznyak, "Stackelberg security games: Computing the shortest-path equilibrium," *Expert Systems With Applications*, vol. 42, no. 8, pp. 3967–3979, 2015.

[27] A. Wilczyński, A. Jakóbik, and J. Kołodziej, "Stackelberg security games: Models, applications and computational aspects," *Journal of Telecommunications and Information Technology*, 2016.

[28] M. Wazid, A. K. Das, V. Odelu, N. Kumar, M. Conti, and M. Jo, "Design of secure user authenticated key management protocol for generic iot networks," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 269–282, 2018.

[29] K.-H. Yeh, "A secure transaction scheme with certificateless cryptographic primitives for IoT-based mobile payments," *IEEE Systems Journal*, 2017.

[30] B. K. Ray, S. Khatua, and S. Roy, "Negotiation based service brokering using game theory," in *Proceedings of the IEEE Applications and Innovations in Mobile Computing (AIMoC)*, Kolkata, India, Feb. 2014, pp. 1–8.

[31] C. Mahieu, F. Ongenae, F. De Backere, P. Bonte, F. De Turck, and P. Simoens, "Semantics-based platform for context-aware and personalized robot interaction in the internet of robotic things," *Journal of Systems and Software*, vol. 149, pp. 138–157, 2019.

Pushpendu Kar has completed all his bachelor, master, and PhD degrees in Computer Science and Engineering. Currently, he is an Assistant Professor with the School of Computer Science, University of Nottingham Ningbo China. Prior to this, he was a Postdoctoral Research Fellow with the Department of ICT and Natural Sciences, Norwegian University of Science and Technology, Norway, the Department of Electrical and Computer Engineering, National University of Singapore, and the Energy Research Institute, Nanyang Technological University, Singapore. His research interests include Mobile Ad Hoc Networks, Wireless Sensor Networks, the Internet of Things, Content-Centric Networking, Big Data Analytics, and Blockchain. He is a recipient of the Erasmus Mundus Postdoctoral Fellowship of the European Commission, the European Research Consortium for Informatics and Mathematics Alain Bensoussan Postdoctoral Fellowship of the European Union, and the Science and Engineering Research Board Overseas Postdoctoral Fellowship of the Department of Science and Technology, Government of India. He is a senior member of IEEE.

Sudip Misra received a Ph.D. degree in computer science from Carleton University, in Ottawa, Canada. He is a professor in the Department of Computer Science and Engineering at Indian Institute of Technology Kharagpur, India. Before this, he was associated with Cornell University (USA), Yale University (USA), Nortel Networks (Canada), and the Government of Ontario (Canada). He was awarded the IEEE ComSoc Asia Pacific Outstanding Young Researcher Award at IEEE GLOBECOM 2012. He was also awarded the Canadian Government's prestigious NSERC Post-Doctoral Fellowship and the Humboldt Research Fellowship in Germany.

Ankush Kumar Mandal is presently, a computer science master's at the Arizona State University and had received his B.Tech degree from the National Institute of Technology, Durgapur, India. He is currently interning as a Software Developer at CYR3CON, Arizona. Ankush is receiving an Engineering Graduate Fellowship for his entire master's in recognition of his extraordinary academic achievements. He worked as a security intern at Smart Wireless Applications and Networking (SWAN Group), IIT Kharagpur from May to July 2018. His research interests include Wireless Sensor Networks and the Internet of Things.

Hao Wang holds a B.Eng. degree and a Ph.D. degree, both in computer science. He is currently an Associate Professor in the Department of Computer Science, Norwegian University of Science and Technology, Norway. He has authored or co-authored over 160 papers in peer-reviewed conferences and journals. His research interests include big data analytics and industrial Internet of Things, high-performance computing, and safety-critical systems. He served as a TPC Co-Chair for the IEEE CPSCom 2020, the IEEE CIT 2017, and ES2017, and reviewers for prestigious journals, such as IEEE TKDE, TBD, TETC, T-IFS, and ACM TOMM. He is a member of the IEEE IES Technical Committee on Industrial Informatics. He is a senior member of IEEE and a member of ACM.