Hindawi
Mobile Information Systems
Volume 2021, Article ID 6676946, 9 pages
https://doi.org/10.1155/2021/6676946

Research Article

Network Security Technology of Intelligent Information Terminal Based on Mobile Internet of Things

Ning Sun,[1,2] **Tao Li**,[2] **Gongfei Song**,[2,3] **and Haoran Xia**[4]

[1] *Binjiang College, Nanjing University of Information Science & Technology, Nanjing 214105, Jiangsu, China*
[2] *CICAEET, School of Automation, Nanjing University of Information Science & Technology, Nanjing 210044, Jiangsu, China*
[3] *Key Laboratory of Advanced Control and Optimization for Chemical Processes, Shanghai 200237, China*
[4] *College of Mechanical & Electrical Engineering, San Jiang University, Nanjing 210012, Jiangsu, China*

Correspondence should be addressed to Ning Sun; 001764@nuist.edu.cn

Received 14 October 2020; Revised 21 January 2021; Accepted 12 March 2021; Published 25 March 2021

Academic Editor: Chi-Hua Chen

In the process of implementing the Internet of Things, the object itself has identity information and identification equipment and encounters difficulties in communication security during the process of entering the network communication. Just like the Internet and wireless sensor networks, there are security issues in information transmission. Therefore, it is of great significance to study the mobile Internet of Things network security technology depression to protect the communication information in the mobile Internet of Things. This paper mainly studies the network security technology of intelligent information terminal based on mobile Internet of Things. This article will analyze and compare the mainstream encryption algorithms of the current mobile Internet and choose a safer and more secure HASH algorithm. We study the flow of key management, key generation, key distribution, verification key distribution, key update, key storage, key backup, and key validity time setting for mobile Internet of Things, using an existing identity cryptosystem. Based on encryption, the design technology of key management and authentication in this paper is improved. Compared with other methods, the storage consumption of this method on GWN is relatively medium. In the initial stage, the storage is 32 bytes, then in registration stage 1, it reaches 84 bytes, in registration stage 2, it is 82 bytes, and then, in the login authentication phase, the number of bytes rose and reached 356 bytes in authentication phase 3. Experimental results show that this protocol has certain advantages in ensuring safety performance.

1. Introduction

To allow participants in the Internet of Things to avoid the security and privacy issues brought about by the universal network basic platform as much as possible, the Internet of Things must achieve simple and safe completion of various user control behaviors. Networking technology research must fully consider security and privacy. And with the continuous improvement of "object" automation capabilities and autonomous intelligence, the problems of object recognition, identity, stealth, and the role of objects in the role they play will become the focus of academic scholars.

As the Internet of Things is becoming more and more widely used, the number and differences of devices are becoming greater, and the early public key encryption technology has been unable to meet the information security needs of the Internet of Things [1, 2]. As an important means of protecting nodes in the Internet of Things, key management and authentication methods have become a research hotspot at this stage. How to propose a set of correct keys under the conditions of low-power consumption, low computing, and high security of the nodes Management and authentication methods is of great significance to study the network security technology of intelligent information terminals based on mobile Internet of Things [3, 4].

Raban Y conducts a relatively balanced long-term predictive study to identify the major threat drivers and identify emerging technologies that may have a significant impact on defense and attack capabilities in cybersecurity. The main tools he uses are online scanning and online surveys conducted by subject matter experts to assess the potential impact of emerging threats and several emerging technologies on network defense capabilities and network attack capabilities. His investigation revealed that network resilience, homomorphic encryption, and blockchain may be considered technologies that primarily contribute to defense capabilities. On the other hand, the Internet of Things, biological hacking, human-machine interface (HMI), and autonomous technologies mainly increase attack capabilities. In the middle, he found that autonomous technology, quantum computing, and artificial intelligence all contribute to defense and attack capabilities, and the impacts on both are roughly similar [5]. *Cavelty MD* used network security research as an empirical research location, which can prove that there are two different ways of understanding network technology in society. The first category of people believe that network technology is nonpolitical, flawed, and important and needs to be repaired to create more security. The other party understands them as political tools in the hands of social participants, without considering technical (possible) possibilities. He suggested focusing on a third understanding to bridge the gap between each other: technology is defined as the embodiment of social knowledge. He believes that, corresponding to this, research on cyber politics will benefit from two innovations: cyber security as the focus of social practice (making and stabilizing by spreading knowledge about vulnerabilities) and the use of practical attention and elimination of these loopholes [6]. *Ahlawat P* studied the problem of node capture from an adversarial perspective. In this view, the enemy intelligently uses various vulnerabilities in the network to establish a cost-effective attack matrix. To resist such attacks, defenders or network designers can construct a similar attack matrix. The defender will identify a set of key nodes and use the key trade-off relationship to assign a key dominance level to each node of the network. The key dominance quantifies the possibility of attacking specific nodes. It is used to determine the length of the hash chain. It can also be used to improve the security of path key establishment and key update of the proposed scheme. The performance of the scheme is analyzed with other existing schemes. His results show that the performance of the scheme is better than the recovery ability of the node capture, the number of hash calculations is reduced, the probability of key leakage of the proxy node is reduced, and the key is updated. The number of links that were revoked during the process decreased [7]. His research provides technical measures and reference programs for this article.

This paper mainly studies the network security technology of mobile Internet of things. According to the seven steps of key generation, key distribution, key validation, key storage, key update, and key validity, the key management of Internet of Things is studied and analyzed. Security analysis, basic theorem and proof, key analysis, authentication analysis, and identity information security analysis of the technical scheme are proposed in this paper. Experimental results show that the proposed protocol has some advantages in security performance.

2. Intelligent Terminal Network Security Technology

2.1. Network Security Key Technology

2.1.1. Establish the Key. The APSZM primitive provides services for establishing a secure link and key management between devices. Each device has an original key, which is set when the device is initially installed [8, 9]. The APSzM primitive performs key exchange and update on the basis of the original key. When a certain device wants to establish a secure communication with another device, it must first establish a secure communication key [10]. Use the security primitive APSZM. MAKELINE. KEY can establish a security key. As shown in Figure 1, it is a timing diagram for key establishment. When two devices want to establish a security key, one acts as the initiator, and the other acts as the responder. The initiator's ZDO will generate an APSZM. MAKELINE. The KEYreq request is sent to the APS layer with the responder's address and the key establishment start method and protocol. The initiator can choose to use the responder's parent node as a contact, just set puse Parent to true in the corresponding parameter, and set the physical address of the parent node in the parameter presParent MAC Addr [11, 12]. After receiving the request, APS will generate a SKKE frame and send it to the network. If a responder receives a request to establish a key, and there is a master key stored in it that is related to the initiator, the APS layer will send an indication primitive to the ZDO layer, which will send a response to the ZDO layer after processing. At the APS layer, this response contains the parameter ACCEPT, which ultimately determines whether to agree to establish a key with the initiator. If agreed, the ASP layer establishes a security key with the initiator through the SKKE protocol [13, 14]. As shown in Figure 1, set the time for the key.

2.1.2. Key Transmission. APSzM's key transmission service plays a key role when the keys need to be transmitted between devices. The key transmission service can transmit the link key, master key, or network key of the program and the master key of the trust center. The key transmission is also composed of an initiator and a responder. When the initiator ZDO layer transmits a key to a responder, it directly generates a primitive for key transmission and sends the primitive to the APS layer. The APS layer is receiving. After the request, the corresponding frame is generated and sent to the network [15, 16]. When the ASP layer of the responder receives the key transmission command frame from the network, it decompresses, decrypts, and authenticates the frame. When the command is confirmed to be legal, it checks whether the address of the command responder is the same, and whether the command has a chain. If road key, master key, and network key are the same, they inform the ZDO layer that a key transmission command has been received [17, 18].

2.1.3. Key Request Service. APSZM provides the key request service to the upper layer. When a device wants to obtain the key of other devices such as the trust center of the current

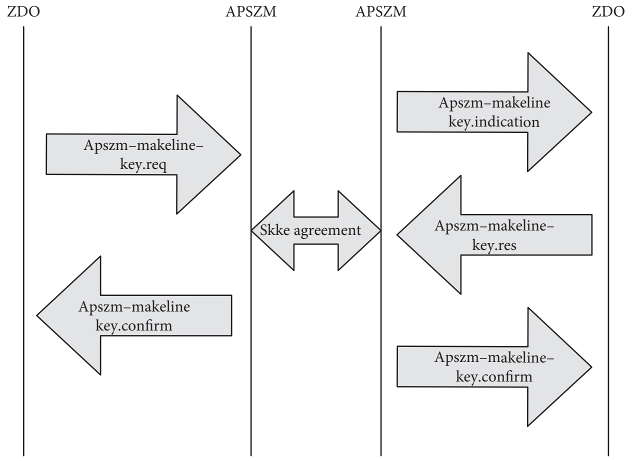

Figure 1: Timing of key establishment.

wireless network or the end-to-end master key, it can use the APSZM in the key request service. GET-KEY. req primitives to achieve. The ZDO layer sends this primitive to the APS layer. After receiving the primitive, the APS layer constructs a command frame according to the parameter value. The command in the command frame is APSC GET-KEY, and the data is 0x08 [19, 20]. If the key type is 2, it means that the link key is requested, and if the key type is 1, it means that the network key is requested. The partner address may be 0 or 8 bytes in length. For the application key request, the partner address is the 64-bit physical address of the corresponding device. For network key requests, the partner address is not used.

After the request key frame is constructed, it must go through security processing and then pass NLDE. DATA. req sends the data to the network. When the ASP layer receives a request key command frame, it first decompresses and authenticates the frame data and then sends an instruction to the ZDO layer. After receiving the instruction, the ZDO layer can determine whether to send the command to the sender. Transfer the key, or the key to its partner.

2.1.4. Key Conversion Service. APSZM within APSZM. CHANGE. KEY primitive is used to provide key conversion services to the ZDO layer. When a certain node device (usually a trust center) wants to notify other devices to convert a new network key, some primitives are used. The ZD0 layer sends APSZM. CHANGE. KEY primitive, and the APS layer receives the primitive to construct the command frame according to the parameter value and network conditions, through security processing, and then call NLDE-DAl to divide. req sends the data out. When the APS layer of the target device receives the data, it decompresses and authenticates the data. After successful pass, it sends an instruction to the ZDO layer. After receiving the instruction, the ZD0 layer replaces the old network key with the key related to KeySeqNumber [21].

2.2. Data Fusion Method. Wireless sensor network, as the main component of the sensing layer of the Internet of Things, is limited by many factors such as energy, storage capacity, transmission rate, and robustness. Among them, energy limitation is arguably the biggest challenge faced by wireless sensor networks. In wireless sensor networks, energy is mainly consumed in two aspects: transmission consumption and computing consumption. Among them, the energy consumed in the process of data transmission is the most [22]. Therefore, how to carry out complex environmental monitoring and reporting on sensor nodes with limited energy is an important problem to be solved urgently in wireless sensor networks. Since wireless sensor network is a data-centered network, data processing technology in the network can be used to reduce this excessive energy consumption; that is, data fusion technology can be used to solve these problems. Figure 2 shows the location of the data fusion technology in the perception layer.

Through the theoretical analysis and simulation testing, the role of data fusion technology is discussed in detail. The results of the study prove that the ratio of the energy cost of the network when using data fusion technology and not using data fusion technology is as follows:

$$\lim_{d\longrightarrow\infty}\frac{N_D}{N_A}=\frac{1}{k}, \tag{1}$$

where d is the distance from the sensor node to the fusion node, k is the number of data collection source nodes, ND is the number of data transmissions in the network using data fusion, and $N\,\hat{}$ is the number of data transmissions in the network without data fusion. Formula (1) shows that the larger the overall size of the network (the larger the d), the more energy saved by data fusion technology: the more data source nodes (the larger k), the more energy can be saved by data fusion technology [23, 24]. This shows that the use of data fusion is extremely important to save the energy consumption of wireless sensor network transmission [25].

2.3. Flow Prediction Method. An effective intrusion detection method based on Markov traffic prediction model is introduced. This method performs anomaly detection independently by predicting the traffic of each node, without special hardware support and cooperation between nodes. Suppose that the current state of the sensor node is i, the next state is j, and the transition probability of the state $p_{i,j}$ is expressed by

$$p_{i,j}=p\left\{X_{m+1}=j\middle|X_m=i\right|\right\}. \tag{2}$$

The probability that the sensor node transitions from state i to state j once is recorded as $p_{i,j}$, which is called the matrix:

$$p=\left\{\begin{matrix} p_{11},p_{12},\dots,p_{1N}\\ p_{21},p_{22},\dots,p_{2N}\\ \cdot\quad\cdot\quad\cdot\\ p_{N1},p_{N2},\dots,p_{NN}\end{matrix}\right\}. \tag{3}$$

Transfer matrix is for one time (or one step). Suppose that the sensor node in the initial state

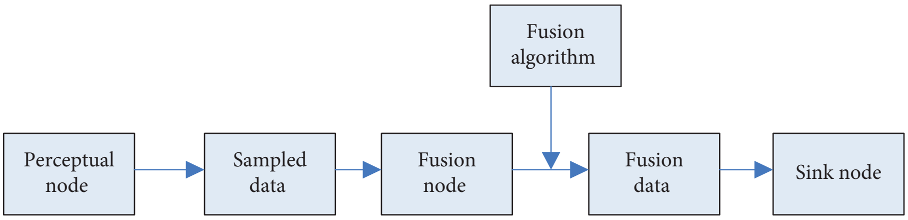

Figure 2: Process diagram of data fusion at the perception layer.

$S^{(0)} = (S_1^{(0)}, S_2^{(0)}, \ldots, S_N^{(0)})$ of $k = 0$ is known, and the state after seven transfers is $S^{(0)} = (S_1^{(k)}, S_3^{(k)}, \ldots, S_N^{(k)})$, $k = l$, 2,..., then,

$$S^{(k)} = S^{(0)} \begin{bmatrix} p_{11}, p_{12}, \ldots, p_{1N} \\ p_{11}, p_{12}, \ldots, p_{1N} \\ . \quad . \quad . \\ p_{N1}, p_{N2}, \ldots, p_{NN} \end{bmatrix}. \tag{4}$$

Equation (4) is a Markov prediction model. The state $S^{(k)}$ of the system after k transitions depends only on the initial state $S^{(0)}$ and the transition matrix P. Let B_s represent the number of data packets transmitted by the sensor node in the initial state of S and the interval of Δt, then the number of transmission packets of the node in any state $S^{(k)}$ can be predicted by equation (3), and then by monitoring the sensor node or the deviation of the predicted flow and the actual flow of the cluster head node to detect whether the sensor network has been intruded. The disadvantage of this method is that the node overhead is relatively large.

By using the Genetic Algorithm to optimize the traffic matrix, a method to enhance DDOS attack detection is proposed, and the traffic matrix is improved through the following operations: (1) reconstruction of the hash function to reduce hash conflicts, and (2) the use of packet-based window size Instead of based on the time window size to reduce the cost of calculation. Then, calculate the variance (Variance, V) through the flow matrix. If $V < T$, an alarm is generated, where T represents the threshold, and V is calculated according to

$$V = \frac{1}{K} \sum_{j=0}^{n} \sum_{i=0}^{n} \left(M_{(i,j)} - u \right)^2, \quad if\, M_{(i,j)} \neq 0, \tag{5}$$

$$u = \frac{1}{K} \sum_{j=0}^{n} \sum_{i=0}^{n} \left(M_{(i,j)} \right)^2, \quad if\, M_{(i,j)} \neq 0. \tag{6}$$

Among them, $M(i, j)$ represents the elements in the traffic matrix, and k represents the number of nonzero elements in M.

Intrusion detection system based on neighbor node traffic is introduced. This method considers that nodes that are close to each other in space have similar behavior. If the behavior of a node is significantly different from that of the neighbors, the node is considered a malicious node. This detection technique is regional and unsupervised and adapts to the dynamic changes of the network. Let sensor node $a_i (i = 1, 2, \ldots, n)$ monitor its direct neighbor node $N(a_i) = \{b_{i1}, b_{i2}, \ldots, b_{im}\}$, where n represents the number of nodes in the sensor network, and m_i represents the number of neighbor nodes of the a_i node. Let node a_i monitor node $\{b_{i1}, b_{i2}, \ldots, b_{im}\}$, and the corresponding attribute vector set is $F(a_i) = \left\{ f(b_{ij}) j = 1, 2, \ldots, m_i \right\}$. If the Euler distance from $f_k(b_{i,j})$ to the set $\{f_k(b_{i1}), f_k(b_{i2}), \ldots, f_k(b_{im}),\}$ is greater than δ_k, then the node $b_{i,j}$ is considered as the malicious node by the node a_i. If the attribute value of the node $f_m(b_{i,j})$ is less than γ_m, the node is also regarded as a suspicious node, where γ_m is a threshold set to reduce threats, and the formal definition of the rules is as follows:

$$\begin{aligned} & f_k\left(b_{i,j}\right) - \mathrm{AVG}\left(f_k\left(b_{i,1}\right), f_k\left(b_{i,2}\right), \ldots, f_k\left(b_{i,m_i}\right)\right), \\ & > \delta_k \text{ and } f_m\left(b_{i,j}\right) > \gamma_m. \end{aligned} \tag{7}$$

3. Experimental Detection of Terminal Network Security Technology

3.1. Experimental Setup. Hardware selection for this experiment: the user node used is MICAz. The hardware device of the device is the AA size component model that can be modified by the user: MPR2400 and MPR2600. The processor is 8 MHz AtMegal28 L, using IEEE 802.15.4, and the wireless model uses the 2.4 G frequency band. The memory and energy resources are limited, the computing power is underground, the data and storage space is 4K bytes, and the flash memory space is only 128K bytes. Among the IoT application nodes, the overall performance of the node is at a low to medium level, which can relatively represent the capabilities of IoT user terminals.

3.2. Data Set Selection. The experimental data set is a public comment data set. Taking an urban area as an example in the experiment, the map space is limited to a square area of 8 km × 8 km. The entire map space is divided into 80 × 80 grids, and the area of each grid is 100 × 100 m^2. The a priori query probability in each grid is obtained by counting the number of comments of all points of interest in the grid. As shown in Table 1, according to the number of points of interest, this article divides the points of interest into the following categories: high-density points of interest, medium-density points of interest, and low-density points of interest. In the experiment, ATM, gas station, and Starbucks were selected as representatives of the above three types of points of interest.

In the experiment, the performance of the proposed algorithm under different density of interest points is

Table 1: Types of points of interest.

Types of interest	Quantity	Points of interest	Quantity
High-density points of interest	200	ATM	251
Point of interest	50–200	Gas station	112
Low-density points of interest	50	Starbucks	35

investigated from four different angles: computational cost and communication cost. Since the same interest points are examined in the experiment, the meta-information such as score and average price is almost the same, so this paper only considers the distance factor top-k sorting, but the algorithm in this paper supports the use of multiple factors for top-k sorting. The details of the experimental parameters are shown in Table 2. Examples include appearance of experimental parameters, significance of experimental parameters, and numerical values of experimental parameters.

3.3. Technical Realization. In this paper, this protocol is analyzed experimentally and compared with other related protocols in terms of computing cost, security performance, communication cost, and storage consumption, to prove its effectiveness. In the previous chapter, a key management scheme based on identity is proposed. In this paper, these methods are simulated, quantitative results are given, and the effectiveness of these methods is verified.

For the implemented software environment, MICAz is TinyOS, an operating system based on open source wireless sensors. TinyOS is an embedded open source operating system designed for wireless sensor network design and development. The component-based operating system (component) architecture allows it to be quickly updated, thereby reducing the limitations of code size sensor network storage. TinyOS is a higher professional operating system for low-power wireless devices, mainly used in the field of sensor networks, pervasive computing, personal area networks, smart homes, and smart meters.

TinyOS and its applications are implemented through a language (nesC) for developing component-structured programs. It is a C-based programming language and a mechanism for organizing, naming, and connecting components to become a robust embedded network system. Support bidirectional concurrency.

4. Influence Analysis of Self-Similar Parameters

4.1. Traffic Aggregation without Self-Similarity. Brownian motion is a random process with independent increments that is normally distributed. The mathematical model of Brownian motion can be described by Wiener process. Mandelbrot extended the one-dimensional Brownian motion model B (t) to fractional Brownian motion Bh (t). H represents the Hurst parameter. If the Hurst parameter value of fractal Gaussian noise is greater than 0.5, it will cause the decay of the queue part to become slower. When $H = 0.5$, it corresponds to a typical short correlation model. Fractal Gaussian noise is an incremental process of fractal Brownian motion and has a strict first-order self-similar process. Fractal Gaussian noise can be used to generate data with precise white similar parameters, mean and variance. In this paper, the random number midpoint setting method is used to generate the flow of fractal Gaussian noise. Here, we first generate two types of flows with the same mean, variance and self-similar parameters. Set the mean value $m = 321$, variance $v = 333$, and Hurst = 0.4 to study the characteristics of the aggregated flow after the two types of traffic with the same parameters and without self-similar characteristics are aggregated.

As shown in Figures 3 and 4, after the two types of data traffic with the same mean, variance, and Hurst coefficients and without self-similar characteristics are aggregated, the aggregated traffic still has no self-similar characteristics, and the aggregated traffic Hurst parameter value and composition aggregate traffic Hurst value of the flow source are almost the same.

After data traffic without self-similarity is aggregated, the value of the self-similarity parameter of the aggregated traffic is approximately the same as the value of the traffic source that constitutes the aggregated traffic. Therefore, when the traffic with the same Hurst parameters and no gate similar characteristics is aggregated with the data traffic with the same properties, there is still no self-similarity in aggregated traffic. This is because the suddenness of traffic without self-similarity is extremely small. After traffic aggregation. Although the traffic density increases, it is still relatively stable on the whole and will not produce a sudden burst of traffic in a short time. Therefore, traffic that does not have self-similar characteristics still does not have self-similar characteristics after aggregation.

4.2. Traffic Aggregation with Self-Similar Characteristics. The network traffic has a self-similar characteristic, indicating that the data is not particularly stable and bursty. After the aggregation of bursty traffic, the burstiness of the traffic will be more obvious. It is still due to the mutual influence of each service flow that the burstiness is weakened, and the self-similarity of the traffic is reduced. This is a problem worthy of study. First, there are self-similar characteristics of the aggregated traffic after two types of traffic with the same mean, variance, and Hurst parameters are aggregated. First, generate two kinds of flow with mean value $m = 318$, variance $v= 323$, and self-similarity coefficient $H = 0.8$, and analyze the self-similarity characteristics of the aggregated flow after the two types of flow are aggregated.

As shown in Figure 5, after two types of traffic with the same mean, variance, and Hurst parameters, and with self-citrus-like characteristics are aggregated, the aggregated traffic still has self-similar characteristics.

The self-similarity parameter of the polymerization flow rate is 0.82021, which is much greater than 0.5, slightly less than 0.85, and also greater than the average of the two flow rates of 0.7. The self-similar value of the aggregated flow is equal to the largest of the flow sources, which is different from the conclusion of this article. This is because, after the traffic aggregation, it is no longer a strict second-order self-

Table 2: Parameter settings.

Parameter name	Appearance	Meaning	Numerical value
θ	Algorithm a	Similarity threshold	When $k = 1$, $\theta = 0$; when $k \geq 21, = 1/2$
η	Algorithm b	Number of candidate regions	$5s_i$
λ	Algorithm c	Information entropy threshold	High-density points of interest:1.5; medium-density points of interest:2.5, low-density points of interest:3.5

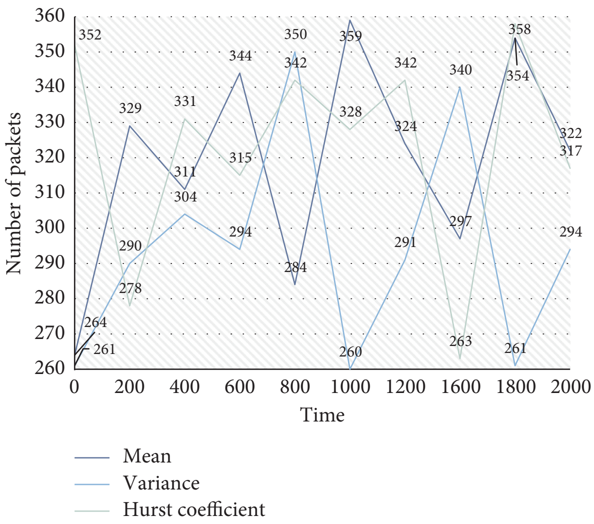

Figure 3: Aggregated traffic source 1 data.

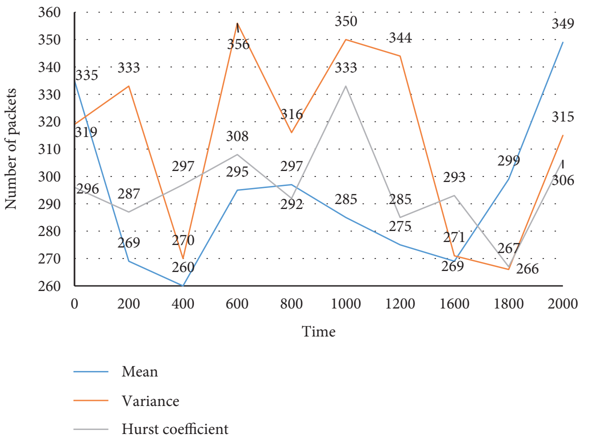

Figure 5: Aggregated traffic source 2 data and Hurst values.

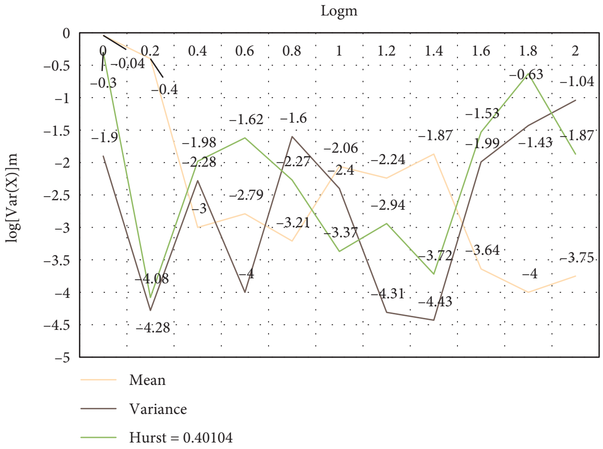

Figure 4: Hurst value.

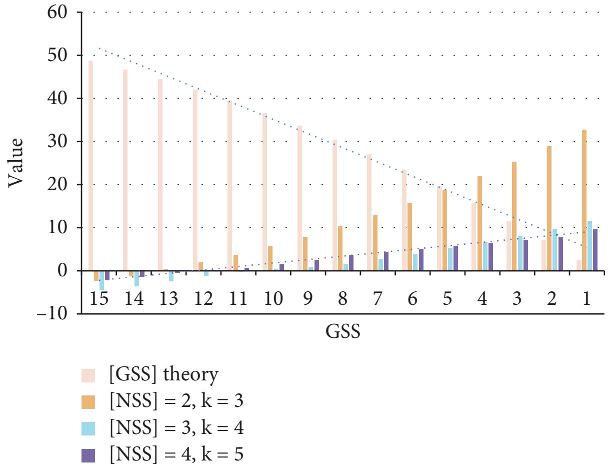

Figure 6: Comparison of the number of nodes to be captured by a collusion attack when different values of |GSS|.

similar process but shows a gradual self-similar characteristic. The H value of the aggregated flow is related to the H value of all flow sources, and the value of the aggregated flow is greater than the largest. It can be seen that when the two types of traffic with self-similar characteristics are aggregated, their mean and variance are the same, but the Hurst parameters are different, and the larger the self-similar parameter, the greater the impact on the aggregated flow.

5. Security Analysis of Network Security Technology

As shown in Figure 6, where the global message set $|I| = 15$, the abscissa is the value of |GSS|, the ordinate is the number of capture nodes required by the attacker, and $\overline{NSS^l}$ is 2, 3, and 4, respectively The number of nodes is captured. As can be seen from the figure, after the global message set |I| is determined, the value of $\overline{NSS^l}$ is not as large as possible. The optimal value should be the value of $\overline{NSS^l}$ determined after the anonymous request of k, rounded down or up at the intersection.

On the other hand, by adding round updates of the global secret information set, the scheme will be able to resist statistical analysis attacks at the same time. However, these are only for the security achieved by passive attackers, and the double verification information security data fusion protocol proposed in this section uses the double secret

Table 3: Distribution of transmission costs at various levels of the end-to-end solution.

Level	Number of nodes node transfer bytes (bytes)	End-to-end solution	Number of bytes transmitted by the node in the double verification scheme (bytes)
1	3	2156	14
2	9	736	14
3	27	236	14
4	81	82	14
5	243	32	14
6	729	8	14
7	2187	2	14

information set method, without adding hash calculation, in accordance with the protocol. Its own security features provide the integrity verification of MAX/MIN nonlinear fusion, which can resist the tampering attacks of active attackers, etc., which improves the security of nonlinear fusion and greatly expands its application scenarios.

6. Practical Performance Analysis of Safety Technology

No matter whether it is homomorphic encryption or traditional encryption algorithms, there is no available solution that can support the operation of taking the most value of the ciphertext directly. It did not appear. Data fusion based on the hop-by-hop encryption method always decrypts and exposes the plaintext at the fusion node, which cannot withstand internal node attacks and fails to meet the requirements of secure data fusion. This section compares the simulation energy consumption of the nonfusion method of nonfusion data of double verification, end-to-end encryption, and hop-by-hop encryption fusion on an embedded multinode platform.

Although the energy consumption of the end-to-end scheme is slightly lower than that of the duplex scheme on average to a single node (because the camouflaged data brings additional energy consumption to the duplex scheme), the ciphertext to be transmitted by the end-to-end scheme during the transmission process continues to increase. The more you go to the upper node, the greater the transmission overhead, and the energy consumption will rise sharply. As shown in Table 3 and Figure 7, the end-to-end solution of the 7-layer network topology is compared with the multiple-layer node transmission cost. As can be seen from the table, the overhead of the top node reaches hundreds of the bottom node thousands of times; this will cause the upper nodes to quickly consume electricity and paralyze the network. In the duplex scheme, the transmission cost of a single node at each level is the length of the global information set due to the execution of the maximum value fusion process, which evenly divides the energy consumption of the network.

When nonlinear MAX/MIN fusion is to be performed at the perception layer of the Internet of Things, the double verification of the disguised data security fusion protocol can provide privacy, confidentiality, and fusion of data between nodes while ensuring low transmission overhead and computational complexity. The integrity verification is a safe and efficient nonlinear data fusion scheme suitable for the perception layer of the Internet of Things.

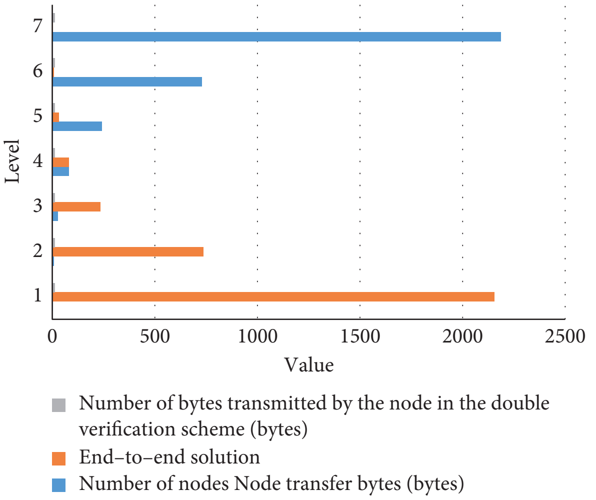

Figure 7: End-to-end solution transmission cost distribution at each level.

7. Conclusions

This paper aggregates traffic with different characteristics and analyzes the influence of Hurst parameters, variance, traffic density, and human participation on aggregated traffic characteristics. The analysis shows that when the variances are the same, after the aggregation of traffic without self-similar characteristics, the aggregated traffic still has no self-similar characteristics. After the aggregation of traffic with self-similar characteristics, the aggregated traffic still has self-similar characteristics. When the traffic without self-similar characteristics is aggregated with the traffic with self-similar characteristics, the characteristics of the aggregated flow are related to the flow sources that make up the aggregated flow. The self-similar parameters of the aggregated flow increase with the variance of the flows without the self-similar characteristics. and decrease with the variance of service traffic with self-similar characteristics.

When the mean and variance of the flow are the same, the larger the Hurst parameter value, the stronger the influence on the self-similar characteristics of the aggregate flow. The greater the variance of the flow source, the greater the impact on the aggregate flow. We analyzed the flow characteristics of real

Internet traffic, aggregated them with the IoT traffic generated by FGN, and found that when the traffic burst is particularly large, the self-similarity of aggregated traffic will be reduced, but it always has self-similarity. This is because the burstiness of the traffic is too strong, and the aggregated traffic cannot effectively absorb the bursty traffic.

The method proposed in this paper has some advantages in computing power, security, and even storage space. It adopts a lightweight authentication model and USES simple xor and hashing functions to ensure the forward and backward performance of key management and authentication. Considering all objects in the Internet of Things as user nodes improves the scalability of the key management and authentication scheme proposed in this paper.

This paper focuses on the nonlinear security data fusion protocol based on disguised data, so as to enable MAX/MIN data fusion in the Internet of Things system. Aiming at the problems of the existing schemes, the article proposes a double verification nonlinear secure data fusion protocol. By using the method of double global secret information group, the advantage of hidden data bits itself is used to complete the integrity verification of the fusion data results to improve the security of the solution that has been expanded, and the scope of application scenarios of the solution has been widened.

Data Availability

All the data sets used in this paper are from the network traffic audit log data of a power enterprise. At the same time, in order to increase the data of all kinds of intrusion behaviors, a part of security audit data set is added to the data set after analyzing and processing Hadoop platform to form the intrusion detection data set in this paper. This data set uses 2 million network traffics as training data, while the other 1 million data sets are test data sets. There are four types of intrusion: port scanning attack, DoS attack, local user's unauthorized access, and remote host's unauthorized access. Among the 39 types of intrusion attacks found, there are 22 kinds of training data sets provided this time. It is worth mentioning that each record has 53 dimensional attributes, and the last attribute is its category. The data is generally composed of the following four aspects: first, fully consider all the basic characteristics of network connection, such as destination IP address, source IP address, source port, destination port, and other attribute fields. Second, consider the content characteristics of network connection: the data part of the data package contains the user's remote access and operating system sensitive file instructions and login system password and other information. Third, consider the time characteristics of traffic: based on the time correlation of network attacks, some connections with the connection within 2 S before the current connection are counted, assuming the percentage of the same host and service type with the current connection within 2 S, etc. Fourth, fully consider the traffic statistical characteristics of the specified host: the actual network attack behavior will be longer than the time span of 2 S. In order to find out the attack, count the relationships between the 100 connections before the current connection and the link; for example, count the percentage of the same host and service type between the first 100 connections and the current connection. The format of a normal and an abnormal network connection data is shown below. 192, 112, 211, 25, 202, 206, 187, 45, 4532, 80, 31, 2, tcp, smtp, 0, 1684, 363, 0, 0, 0, 0, 01, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0.00, 0.00, 0.00, 1.00, 0.00, 0.00, 104, 66, 0.63, 0.03, 0.01, 0.00, 0.00, 0.00, 0.00, 0.00, normal. 192, 119, 131, 65, 202, 206, 225, 130, 7642, 25, 24, 0, tcp, private, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 38, 1, 0.00, 0.00, 1.00, 1.00, 0.03, 0.55, 0.00, 208, 1, 0.00, 0.11, 0.18, 0.00, 0.01, 0.00, 0.42, 1.00, portsweep. Because the data set contains discrete and continuous data, so, we need to standardize and normalize the data in order to fit the input of neurons and avoid the situation of large numbers eating decimals.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61973168, 61973170, and 41675156), and University-Industry Collaborative Education Program of Ministry of Education (201901059005, 201802048012, and 201702177099).

References

[1] G. Oluwafunmilayo, "An assessment of cybersecurity technologies in the selected universities in southwestern Nigeria," *International Journal of Computer Applications*, vol. 178, no. 50, pp. 11–18, 2019.

[2] M. Thurber, "CCX Technologies releases cybersecurity hardware," *Aviation International News*, vol. 50, no. 8, p. 6, 2019.

[3] P. Astromskis, "Legal technologies and cyber security in the singularity age," *Law Review*, vol. 16, no. 2, pp. 34–57, 2017.

[4] A. A. M. Al-Dherasi and A. Annor-Antwi, "Dependence on blockchain technology for future cybersecurity advancement: a systematic analysis," *Computer Science and Information Systems*, vol. 1, no. 1, pp. 1–13, 2019.

[5] Y. Raban and A. Hauptman, "Foresight of cyber security threat drivers and affecting technologies," *Foresight*, vol. 20, no. 4, pp. 353–363, 2018.

[6] M. D. Cavelty, "Cybersecurity research meets science and technology studies," *Politics & Governance*, vol. 6, no. 2, p. 22, 2018.

[7] J. Wider, "C-Suite innovators discuss advances in technology, cybersecurity, interoperability, analytics, and more," *Health Management Technology*, vol. 39, no. 3, pp. 6–9, 2018.

[8] T. Kasama, "3 cybersecurity technologies : darknet monitoring and analysis:3-1 long-term darknet analysis in NICTER," *Journal of the National Institute of Information & Communications Technology*, vol. 63, no. 2, pp. 25–31, 2016.

[9] B. Alluhaybi, M. S. Alrahhal, A. Alzhrani et al., "A survey: agent-based software technology under the eyes of cyber security, security controls, attacks, and challenges," *International Journal of Advanced Computer Science & Applications*, vol. 10, no. 8, p. 211, 2019.

[10] W. Hooper, “Cybersecurity for media technology products,” *Smpte Motion Imaging Journal*, vol. 126, no. 1, pp. 1–4, 2017.

[11] C. D. Calhoun, “Incorporating blended format cybersecurity education into a community College information technology program,” *Community College Journal of Research and Practice*, vol. 41, no. 6, pp. 344–347, 2017.

[12] M. Albettar, “Evaluation and assessment of cyber security based on Niagara framework: a review,” *Journal of Cyber Security Technology*, vol. 3, no. 3, pp. 125–136, 2019.

[13] Q. Ye, “A modular approach for implementation of honeypots in cyber security,” *Advances in Computational Ences and Technology*, vol. 11, no. 2, pp. 105–115, 2018.

[14] S. Suzanne, R. Aftin, C. Seth et al., “The evolving state of medical device cybersecurity,” *Biomedical Instrumentation & Technology*, vol. 52, no. 2, pp. 103–111, 2018.

[15] M. Orcutt, “Venture capitalists chase rising cybersecurity spending,” *Technology Review*, vol. 119, no. 2, pp. 72-73, 2016.

[16] T. Takahashi, “6 security architecture techniques 6-1 cybersecurity information discovery technique and knowledge base,” *Journal of the National Institute of Information & Communications Technology*, vol. 63, no. 2, pp. 143–148, 2016.

[17] C. S. Kruse, B. Frederick, T. Jacobson, and D. K. Monticone, “Cybersecurity in healthcare: a systematic review of modern threats and trends,” *Technology and Health Care*, vol. 25, no. 1, pp. 1–10, 2017.

[18] R. Ganesan, S. Jajodia, A. Shah, and H. Cam, “Dynamic scheduling of cybersecurity analysts for minimizing risk using reinforcement learning,” *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 1, pp. 1–21, 2016.

[19] S. T. Hamman, K. M. Hopkinson, R. L. Markham, A. M. Chaplik, and G. E. Metzler, “Teaching game theory to improve adversarial thinking in cybersecurity students,” *IEEE Transactions on Education*, vol. 60, no. 3, pp. 205–211, 2017.

[20] N. Radziwill and M. Benton, “Cybersecurity cost of quality: managing the costs of cybersecurity risk management,” *Software Quality Professional*, vol. 19, no. 4, pp. 25–43, 2017.

[21] T. Butts, “Cybersecurity for broadcasters,” *TV Technology*, vol. 37, no. 2, p. 4, 2019.

[22] W. G. . Wendy, “Measuring information security and cybersecurity on private cloud computing,” *Journal of Theoretical & Applied Information Technology*, vol. 97, no. 1, pp. 156–168, 2019.

[23] D. N. Burrell, “Assessing the value of executive leadership coaches for cybersecurity project managers,” *International Journal of Human Capital and Information Technology Professionals*, vol. 10, no. 2, pp. 20–32, 2019.

[24] Y. Tsuda, N. Kanaya, T. Tomine et al., “4 cyber-security technologies: live network monitoring and analysis technologies 4-1 NIRVANA-kai: a real-time visual siem system Against targeted attacks,” *Journal of the National Institute of Information & Communications Technology*, vol. 63, no. 2, pp. 67–75, 2016.

[25] S. G. Langer, “Cyber-security issues in healthcare information technology,” *Journal of Digital Imaging*, vol. 30, no. 1, pp. 117–125, 2017.