Journal of Computers Vol. 34 No. 4, August 2023, pp. 25-39
doi: 10.53106/199115992023083404003

Formal Analysis and Improvement of Z-Wave Protocol

Jin-Ze Du*, Jun-Wei Liu, Tao Feng, Zhan-Ting Yuan

School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China
fengt@lut.edu.cn

Received 11 July 2022; Revised 15 November 2022; Accepted 4 January 2023

Abstract. In order to verify the security of the Z-Wave communication protocol, the possible attacks in the protocol are analyzed to reduce user privacy security vulnerabilities. For the communication process and key exchange process between the controller and the node, this paper uses CPN tools to model the Z-Wave S2 protocol, and introduces the Dolev-Yao attack model to verify the security behavior of the protocol. The results show that there is a man-in-the-middle attack when using S2 authentication for device inclusion. In response to this vulnerability, we propose a lightweight static authentication scheme based on HKDF function and XOR operation, which performs authentication between Z-Wave controller and slave device. Secondly, we formally verify the security objectives of the improved scheme, and prove that the optimization scheme can effectively prevent man-in-the-middle attacks in the S2 security mode.

Keywords: Z-Wave protocol, colored Petri net, formal analysis, HKDF

1 Introduction

With the rapid development of wireless sensor networks (WSNs) and other IoT technologies, smart home systems are also advancing. Smart home is the application of Internet of Things technology to connect different kinds of sensors with the Internet. It collects data through sensors and then transmits control commands to control intelligent devices [1]. Smart devices are commonly used in many different situations to collect large amounts of personal information. If attacked by hackers, it may lead to serious privacy and security problems for users [2]. Therefore, it is necessary to detect the possible vulnerabilities in the protocol to ensure the security of the smart home system. The combination of formal analysis and protocol specification is an important way to verify the security of IoT protocol [3]. The model can be built according to the protocol specification, and the protocol model is automatically verified in the state space of the model.

So far, the most studied IoT protocols are LoRaWAN, 5G, Bluetooth, Narrowband IoT, 6LoWPAN, and ZigBee, while Z-Wave research mainly involves vulnerability analysis rather than formal analysis [4]. Hence, we analyze the Z-Wave protocol from a formal perspective. Optimize the protocol to enhance its security. Specifically, our protocol analysis focuses on the joining procedure of a S2 device to the Z-Wave network, which involves using S2 security class. In this phase, a new joining device negotiates with the controller which symmetric keys it will use to protect all communications with the other nodes in the network.

In this paper, colored Petri net (CPN) and Dolev-Yao attack model are used to model the protocol key agreement process [5]. The results show that there is a man-in-the-middle attack vulnerability in the protocol. Aiming at this vulnerability, we designed an identity authentication method based on HKDF algorithm and XOR operation. This method provides confidentiality and integrity, and can resist man in the middle attack. Secondly, our authentication mechanism requires only one round trip message to complete the authentication between the Z-Wave S2 controller and the slave device. This allows for authentication with a smaller number of messages. The HKDF algorithm and XOR operation are light-weight algorithms with low computational complexity and low resource consumption. Therefore, this method has little response to the performance of Z-Wave S2 node and is suitable for equipment with limited resources. The main contributions of this article is as follows:

1) We describe in detail the process of the device joining the Z-wave network. Then we establish the CPN model of Z-Wave protocol and use CPN tools to verify the consistency of the model.
2) On the basis of colored Petri nets, Dolev Yao attack model is introduced to evaluate the security of S2 protocol, which verifies that there is man in the middle attack in the protocol.

* Corresponding Author

3) We propose a mutual authentication scheme for resource-constrained devices and establish a CPN attack model to check the security of the improved scheme.

The remainder of the paper is structured as follows. In Section II, Z-Wave protocol related research achievements are introduced. In Section III, the S2 security class of Z-Wave protocol is described in detail. In Section IV, we define the security evaluation model of Z-Wave S2 protocol, and generate the state space report of the model with CPN Tools to verify the existence of MIMT attack in Z-Wave protocol. In Section V, we illustrate proposed lightweight authentication scheme for mutual authentication between Z-Wave controller and slave Node, and analyze the security of this scheme. Section VI provides future directions for this research work and concludes the work.

2 Background

2.1 Z-Wave Protocol

Z-Wave is a wireless communication protocol for automation equipment in home and business environments. It can run all electrical equipment in the house, such as switches, light, air conditioning (HVAC), televisions and home security. “Inclusion” and “exclusion” refer to the process of including and excluding devices from Z-Wave networks, respectively.

The security command class of Z-Wave ensures the security of the network “inclusion” into new devices. Z-wave has two encryption mechanisms: S0 and S2, which are compatible with the original S0 encryption mechanism. Based on the Z-Wave S2 framework, the Z-Wave network can be logically divided into three security classes: S2 Access Control, S2 Authenticated, and S2 Unauthenticated. The devices that join the Z-Wave network must fall into one of these categories. Since each S2 security class has its own network key, there are three distinct AES-128 keys. It is used for devices, such as door locks and garage openers, that require access control because it offers the highest level of security. Authenticated class is next lower grades and applied to normal household devices, such as sensors and light control. Unauthenticated class is lowest level and is used for devices which cannot be completely authenticated due to limited interface. A device can belong to more than one security classes. These devices require appropriate keys to connect to other devices in each class [6].

To distribute network keys securely, the controller must establish a secure connection with the device. However, it is impossible to establish a secure communication without an encryption key. The Z-Wave system uses Elliptic Curve Diffie-Hellman (ECDH) to deal with this problem. For secure communication, ECDH allows the exchange of temporary keys. Once the secure connection has been established using the ECDH temporary keys, the network key can be securely transferred to the device. ECDH temporary link keys can be used to distribute the S0 network key to S2 devices in order to achieve interoperability with S0 devices. Any key of this kind will be viewed as an unauthenticated class key under S2.

The network communication between a controller and slave node in S2 mode is illustrated in Fig. 1, which is the core part of Z-Wave S2 security layer. The connection process can be roughly divided into three steps:

Phase 1. The main objective is to generate temporary shared keys using ECDH key exchange technology. The first three steps confirm the security class by exchanging KEX GET, KEX Report, and KEX SET commands. In Step 4 and Step 5, the controller exchanges the ECDH public key with the slave device to generate a shared key.

If security inclusion is used, the gateway needs to verify the identity of the new device. The S2 protocol uses device specific keys (DSK) for authentication. That is, the new device provides a unique DSK to the controller for authentication during the inclusion process.

Phase 2. It is a challenge-response protocol based on Nonce. First, the slave obtains a random number from the gateway through the Nonce Get command. The gateway sends a valid random number to the slave node through the Nonce Report command. Second, they confirm that no messages have been tampered with by resending the KEX Report and KEX SET messages. Finally, the slave device sends the Network Key Get command to the gateway to obtain the network key. The gateway sends the Network key report command to inform the slave node of the network key.

Phase 3. Use challenge-response based on random numbers to verify the correctness of the network key.

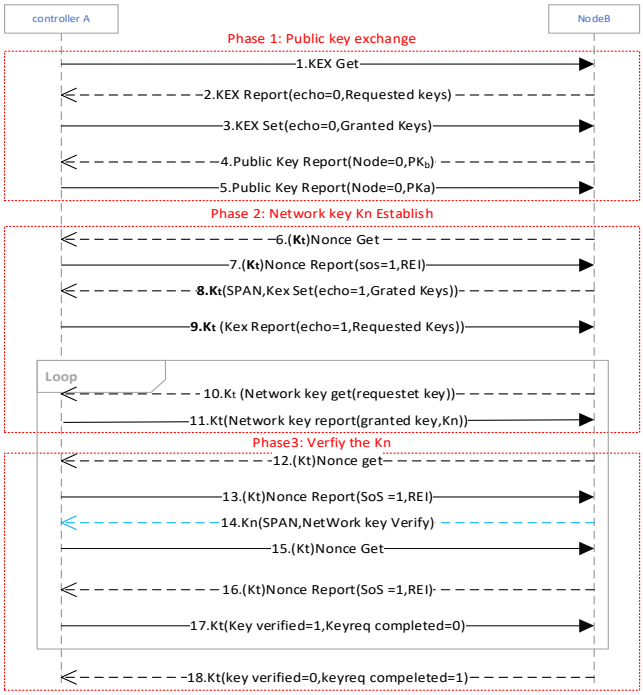

Fig. 1. Z-Wave S2 joining procedure

2.2 Formal Analysis Methods and CPN Tools

Formal analysis is an effective technique for modeling and evaluating protocols and algorithms in the Internet of Things [7]. At present, there are three formal analysis methods: theorem proving, modal logic method and model-based formal analysis. Model based formal analysis method is the most effective protocol security analysis method at present [8].

Petri net theory is a modeling method based on state. A Coloured Petri Nets (CPN) is a graph consisting of two kinds of nodes: places and transitions [9]. A set of arcs link places to transitions to places and transitions. The state of the modelled system is represented by the place. Each place can be marked with one or more tokens, and each token is accompanied by a data value called the token colour. The transitions drawn as rectangles represent the events that can occur in the system. Transitions can be fired. Firing a transition updates the tokens of the places in the net [5].

CNP Tools is a CPN model performance analysis tool, supporting the establishment of colored Petri net model with time and stratification [10]. CPN model uses the state space method to verify system characteristics. CPN Tools can automatically construct the state space of the model, and there is information about the verification of system behavior in the constructed state space, such as whether there is deadlock and the possibility of always reaching a given state [11-13]. This paper uses CPN Tools to study Z-Wave protocol.

2.3 HKDF

A key derivation function (KDF) is a basic and essential component of cryptographic systems. The KDF's main purpose is to deduce one or more strong encryption keys from initial keying materials using a pseudo-random function to prevent brute-force attack or dictionary attack on secret input values [14].

KDF first obtains a pseudo-random key K satisfying the length of the security key from the insecure source key material, and then uses PRF to extend K to one or more pseudo-random keys of the desired length [15]. HKDF is a key deriving function based on HMAC, which contains two steps.

Step 1. Using the original key material IKM and a salt value, a pseudo-random key PRK consistent with crypto-graphic strength is derived. The calculation formula of PRK was shown in Eq. (1).

$$\begin{aligned} PRK &= HKDF - Extract(salt, IKM) \\ &= HMAC - Hash(salt, IKM) \end{aligned} . \tag{1}$$

Step 2: Use PRK, info and L to generate the pseudo-random key OKM with the required output length. The calculation formula of OKM was shown in Eq. (2)

$$\begin{aligned} OKM &= HKDF - Expand(PRK, \inf o, L) \\ &= K(1) \,||\, K(2) ... \,||\, K(t) \end{aligned} . \tag{2}$$

Where:

$$t = [L \,/\, HashLen] . \tag{3}$$

$$K(1) = HMAC(PRK, \inf o \,||\, 0) . \tag{4}$$

$$K(i+1) = HMAC(PRK, K(i) \,||\, \inf o \,||\, i), \quad 1 \le i < t . \tag{5}$$

The derived key OKM replaces the initial secret value as the key of the system. In either case, the "salt" is either a 256-bit random (but not necessarily secret) value, or if not provided, it is set to 0. Its role is to increase the strength of HKDF, ensure independence between the hash function's many uses and support "source-independent" extraction. In the definition of HKDF, the "info" value is optional, which is often of extremely important in applications. Its main function is to link the key material derived with application- and context-specific information. However, there is a special requirement for the choice of "info": it should be independent of the key material IKM value entered.

2.4 Related Work

In recent years, there are few literatures on Z-Wave. Most of the researches focus on the device security and privacy assurance of Z-Wave protocol. So far as we know, formal methods applied to Z-Wave protocol are only discussed in [8]. The authors exploit the ASMETA formal framework to model the protocol to verify the correct behavior of the protocol's security goals. As a result of the verification process, a vulnerability was identified that can be exploited by MITM (Man-In-The-Middle) attacks, which compromise the secrecy of the exchanged symmetric keys.

In [16], authors discovered a new vulnerability in the Z-Wave controller. Attackers use the web server and internet access point to add random rogue Z-Wave devices to a home network, so as to use the controller for continuous attacks. In [17], the authors reverse engineered the Z-Wave routing protocol in terms of frame forwarding and topology management, and found a routing attack vulnerability: adversaries can use the inherent blind trust of routing nodes to modify the topology and routes. In [18], authors propose a misuse-based intrusion detection system (MBIDS), which can detect manipulated packet injection attacks (including Hel attacks) with a detection rate of 99%. In [19], the authors used threat modeling technology to analyze Z-Wave devices. They also examined three Z-Wave attack vectors, including DoS, FOTA and remote add-mode control attacks. they found that if combined with Z-Wave attacks, they can cause serious damage to smart home residents. In [20], the authors present VFUZZ method to evaluate vulnerabilities in Z-Wave devices, found 10 different security vulnerabilities and 7 crashes among the tested devices. The yielded six distinct common vulnerabilities and exposures (CVE) identifiers related to Z-Wave chipset.

To sum up, the research on Z-Wave vulnerabilities has been progressing steadily. However, previous re-search work did not propose vulnerability optimization methods or introduce an attacker model. Thus, on the basis of these studies, we use colored Petri nets and Dolev-Yao attack models to model protocol behavior to verify possible MITM vulnerabilities, and propose a lightweight improvement scheme based on HKDF and XOR operations to prevent man in the middle attacks.

3 Z-Wave Protocol Model

This section simulates the interaction model of Z-Wave network containing devices, which adopts the Hierarchical colored Petri net model. The main simulation contains the first part of the process, which involves DSK authentication. S2 authentication and S2 access control require authentication. During the authentication process, there are two major ways to get the DSK of the device: scanning and decoding the QR code with a smartphone camera or entering the DSK in an input field with a keyboard or keypad.

In the Z-Wave protocol model established in this paper, it is assumed that the gateway uses the second method to obtain the device DSK. The attacker can perform brute force attacks to obtain the device PIN code.

3.1 HCPN Model of Z-Wave Protocol

In this paper, the bottom-up hierarchical modeling method is used to model Z-Wave protocol. The top-level model of Z-Wave protocol is shown in Fig. 2. Substitution transition Controller and S2-Node represent the master controller and S2 slave devices in Z-Wave network respectively. Substitution transition Net represents Z-Wave wireless network. A-KEX-GET and B-KEX-GET represent the network channel interface in each step.

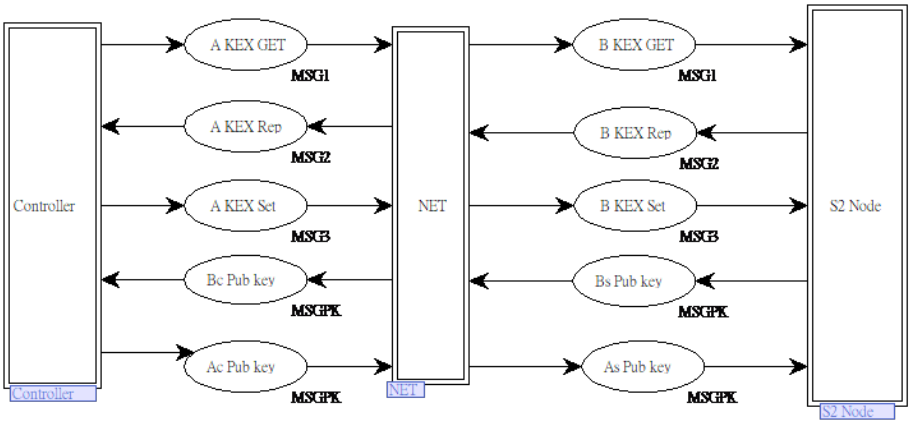

Fig. 2. Top-level model of the Z-Wave protocol

Fig. 3 shows the internal model of Controller, which has a total of five messages. A-KEX-GET is used by the Controller to query the security level supported by the slave node. The data includes Controller node ID, S2 device node ID and Get command. A-KEX-REP receives the response message MSG2. The Stores transition extracts the command classes supported by the device from the MSG2 message and writes them into the Req-Keys Place. The input transition selects one of the security classes and writes the selection results to the A-Gran-key repository. The Join transition encapsulates the token in place A-Gran-key into MSG3 and sends it to the A-KEX-Set interface. The MSG3 message contains the KEX SET command for S2 Node to change settings according to the security class selected by the user.

After the Bc-Pub-key interface receives the MSGPK message, the transition Check combines the PIN code entered by the user with the public key information in the MSGPK to generate a complete public key PKb. At this time, a new token is added to the SKa place, triggering the Kt MUL transition to generate the ECDH shared key.

The green part of the diagram is the timer TA2, which simulates the response time of the controller from sending the KEX SET command to the user input PIN code. The duration of the timer TA2 has a positive effect on preventing an intruder from violently breaking the PIN code.

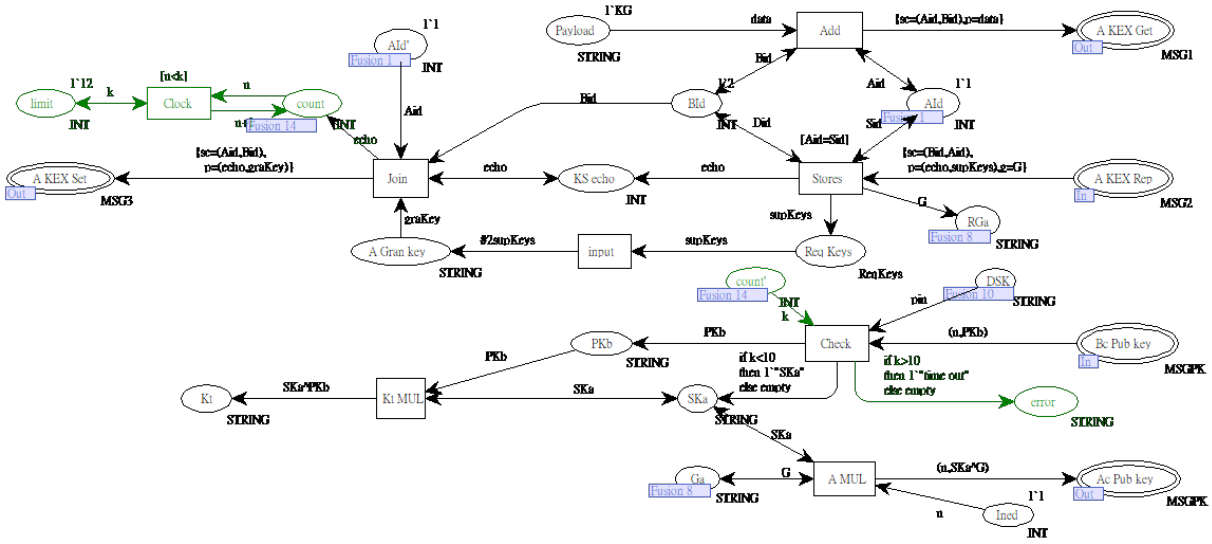

Fig. 3. The internal model of Controller

Fig. 4 shows the internal model of S2-Node. After the B-KEX-GET interface accepts the device request command KEX-GET, the KEX-RePS transition combines the security classes it supports with other information to generate a message MSG2. The message MSG2 is sent to the Controller through the interface B-KEX-Rep. The Curve place represents the elliptic curve information negotiated by the Controller and S2-Node entities. The KEX-SET transition sets the S2-Node parameters according to the received KEX SET command. The B-Mul transition generates the public key PKb according to the existing data. Interface Bs Pub key sends the public key PKb to the controller. The PIN place is the first two bytes of the public key PKb, which represents the pin code used for authentication on the device. the Kt-MUL transition generates the ECDH shared key BKt according to SKb and PKa.

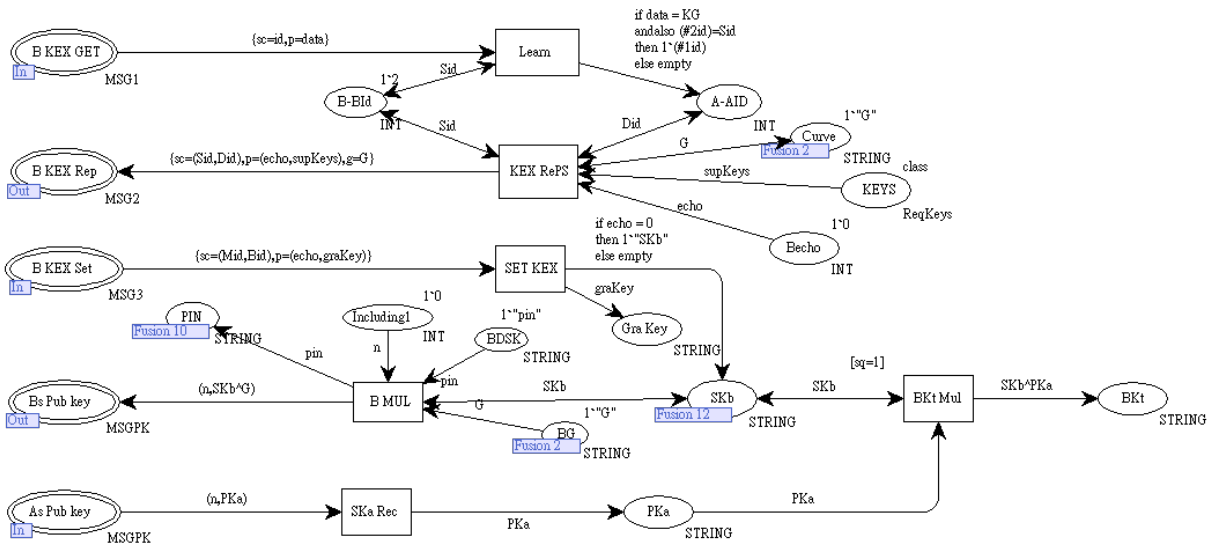

Fig. 4. The internal model of S2-Node

3.2 Dolev-Yao Threat Model

Assuming that the prevailing entity transmits messages over insecure channels, an attacker may communicate with the controller by disguising as a legitimate S2 device. The notation δ symbolizes a polynomial-time (t) bounded adversary. Dolev-Yao attacker model is adopted to evaluate the security of Z-Wave protocol [21]. In this model, the capabilities of adversary δ are as follows.

1) Adversary δ can detect any message transmitted over an insecure channel.
2) Adversary δ can intercept, modify, decompose, insert and redirect the transmitted message.
3) Adversary δ can decrypt the encrypted message and view the message content only if δ possesses the decryption key.
4) The mobile terminal cannot be captured by adversary δ, so δ cannot extract the private key of the gateway.
5) Adversary δ can establish the connection between controller and adversary δ as well as the other be-tween δ and the S2 device. The attacker can launch a MITM attack through these two connections.

3.3 Attacker Based Evaluation Model

According to the capability of the attacker in the threat model, the Z-Wave protocol security evaluation model is established. The attack process is divided into three attack models: MKG, MKS and MPK.

The Security assessment model MKG is shown in Fig. 5. Place Ch-m is a fusion place, indicating whether the place is attacked or not. There are places with the same fusion mark in MKS and MPK models.

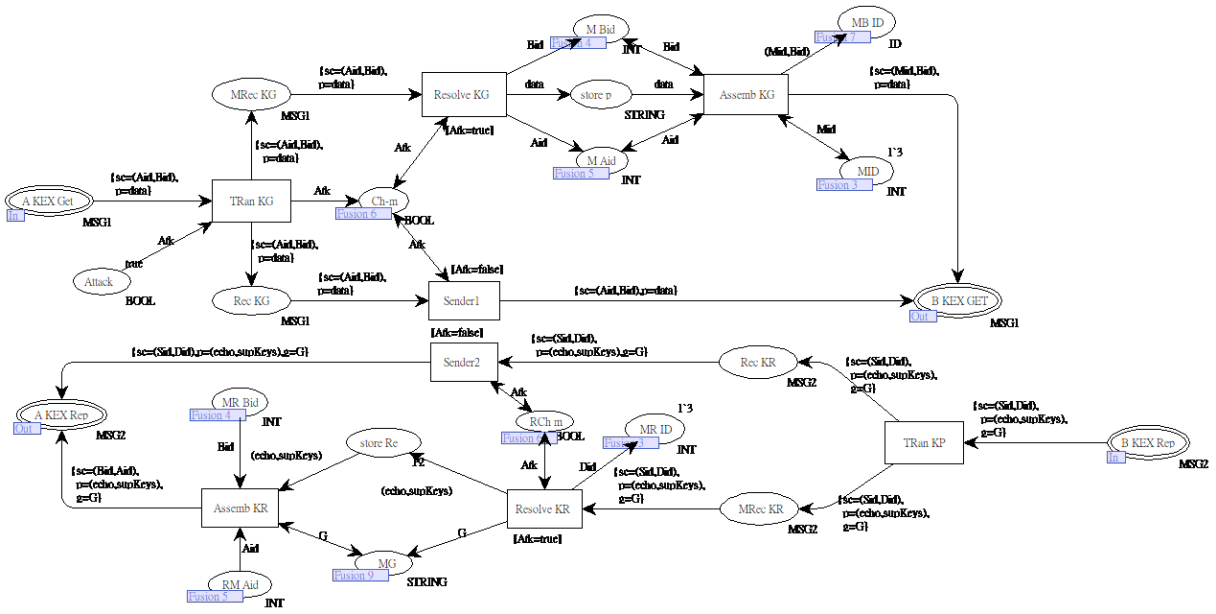

Fig. 5. Attacker-based Z-Wave protocol security evaluation model MKG

After A-KEX-Get accepts the message MSG1, the transition Tran-KG selects the attacker from the At-tack place and writes one taken in Ch-m place. At the same time, the received messages are stored in MRec-KG and Rec-KG. If the value of Ch-m is false, the transition Resolve-KG is fired. The attacker uses Resolve KG to split the MSG1 message into three parts: controller ID, device ID, and command class, which are respectively stored in the place M-Bid, M-Aid, and store-p. Transition Assemble-KG modifies controller ID to attacker ID, regenerates message MSG1 and sends it to network interface B-KEX-GET. At this point, a connection is established between the attacker and S2-Node. If the value of Ch-m is true, the transition Sender1 is fired, indicating that there is no MITM attack.

Transition Tran-KP is fired when a message can be sent. The transition Resolve-KR splits MSG2 into two parts. Transition Assemb-KR tampered with MSG2 and redirected it to interface A-KEX-Rep. At this point, the attacker and Controller established a connection.

The main functions of this model are as follows: (1) Attacker sends KEX GET messages to S2 Node by imitating the controller to establish a session between attacker and S2 Node; (2) The attacker imitates S2 Node to send KEX SET message to the controller to establish a session between the controller and attacker.

The security evaluation model MKS is shown in Fig. 6. After A-KEX-Set receives the message MSG3, the transitions Assemble-KS and Resolve-KG split and reassemble MSG3. The attacker simulates the controller to send a KEX SET message to the S2-Node node, but the attacker does not collect information of interest.

The security evaluation model MPK is shown in Fig. 7. In the Controller page, there are places with the same fusion tag as MDSK, which are used for DSK identity authentication of Controller.

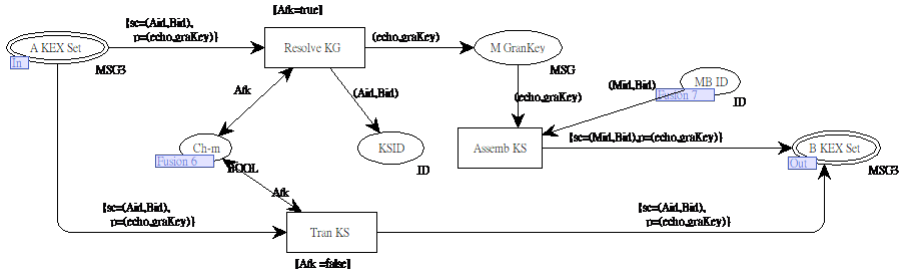

Fig. 6. Attacker-based Z-Wave protocol security evaluation model MKS

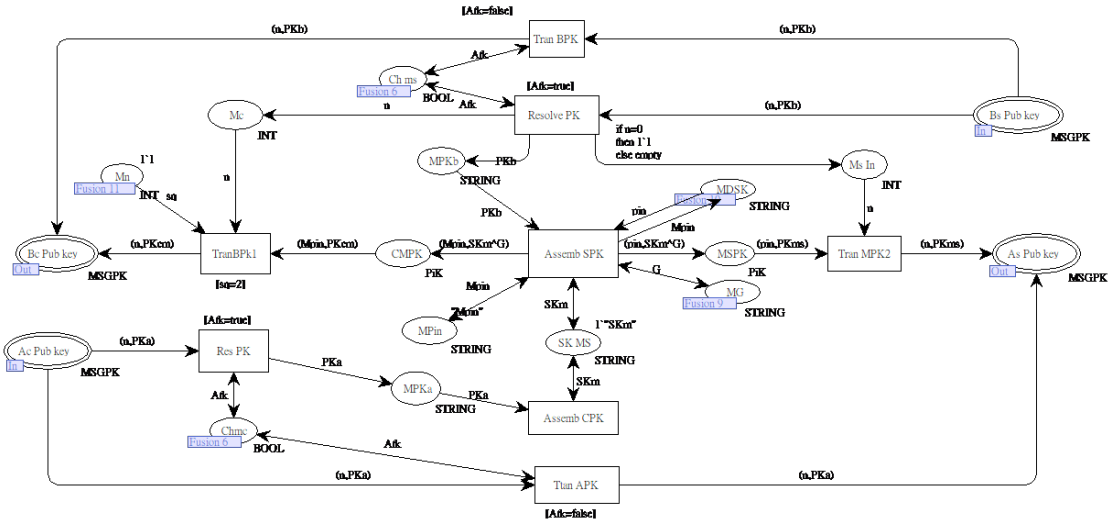

Fig. 7. Attacker-based Z-Wave protocol security evaluation model MPK

The Bs-Pub-key receives the MSGPK message, which contains the ECDH public key PKb sent by S2 Node. Attackers use transition Resolve PK to split received messages to extract information of interest. The transition Assemb-SPK generates ECDH public key PKcm and PKms according to existing information. The attacker uses PKcm and PKms to conduct Key Negotiation with the controller and S2 device respectively.

After S2-Node receives the message, it combines the private key SKb to generate the ECDH shared key. After receiving the attacker's public key, the controller generates a temporary key Kt and sends its public key PKa to the interface Ac-Pub-key. After receiving the MSGPK message, the attacker generates a tempo-rary key shared with the controller and stores it in the Ktcm place. If the shared key exists in the Ktcm place, the attack is considered successful.

3.4 Security Assessment Model Analysis

Table 1 is an excerpt from the state space report of Z-Wave protocol attacker model. This report presents many behavior attributes of the model, including dead nodes, live transitions, dead transitions, etc. It can be seen from Table 1 that the number of state space nodes and State space arcs of the model is consistent with the number of SCC graph nodes and SCC graph arcs. Therefore, there is no wireless loop and iteration in the model. Dead markings indicate that there are 3 terminal states in the generated graph.

After introducing the Dolev-Yao attacker model, we found that the number of dead nodes did not increase, but the number of dead transitions increased by 5. Through analysis, we can know that the new dead transitions are caused by attackers. First, the attacker hijacks the session between the Controller and S2 Node to establish the session between the Controller and the attacker. Secondly, the attacker establishes a session with S2 Node, making the session between the Controller and S2 Node impossible.

Table 1. State space reports - Z-Wave S2 CPN model

Type	Initial	MITM attack
State space nodes	58	165
State space arcs	146	297
SCC graph nodes	85	165
SCC graph arcs	146	297
Dead markings	3	3
Living transitions	0	0
Dead transitions	0	5

Through the above analysis, it is can be seen that the introduced attacker effectively attacked the key exchange process of the original model. This verified that there is a MITM vulnerability in the Z-Wave protocol. Attackers can brute force the PIN code to pass the authentication of the authorized controller. To further attack, it generates the key shared with the legal controller to obtain the grant key associated with the security class.

4 Improvement and Evaluation of Z-Wave Protocol

4.1 Improvements to the Z-Wave Protocol

Based on the analysis results of the security assessment model, we propose a lightweight two-way authentication mechanism between Z-Wave controller and S2 device. In order to avoid additional communication overhead, minimal modification of the protocol is required to meet resource-constrained IoT devices [22-26]. The scheme uses only two handshake messages to complete the mutual authentication between devices. HMAC-based key derivation function (HKDF) and XOR operation are used during the handshake. Fig. 8 shows the messages transmitted between the controller and S2 device during the authentication process. The HKDF function can prevent the attacker from learning the information related to the original key value and prevent the original protocol from being attacked violently in the process of key exchange. The XOR operations is lightweight, which meets the equipment requirement [27]. The data are encrypted and decrypted by AES algorithm on Z-Wave controllers and slave devices. Table 2 shows the list of notations used in our scheme.

Table 2. The list of notations used in our scheme

Symbol	Description
(SK_a, PK_a)	The Controller's private and public keys pair
(SK_b, PK_b)	The S2-Node's private and public keys pair
N_1, N_2	Temporary secret parameters picked by Controller
K	Pseudo random key
Salt	A 256-bit random number
Info	Optional context and application specific information
$h(\cdot)$	hash function
N_3	A random challenge code generated by S2 Node
$\oplus$	Bitwise XOR operation

The authentication process for the improved scheme is described as follows:

Step 1: Controller $\rightarrow$ S2Node:(E_1,N_0)

1) After receiving the public key PK_b of the slave device, the controller generates $DH_k = SK_a \cdot PK_b$ and two random challenge code $N_0 \leftarrow \{0,1\}^{128}$, $N_1 \leftarrow \{0,1\}^{128}$.
2) The controller uses HKDF function to generate pseudo random key $K = HKDF(slat, DH_k, \inf o)$, where $salt = N_0$, info is the information related to the protocol context.
3) The controller chooses a random number N_0 to computes the identifier as $V_1 = h(N_0 \otimes PK_b)$ then computers a masked nonce m_1 as $m_1 = V_1 \otimes N_1 \otimes DH_k$. It uses the pseudo-random key K to encrypt m_1 and V_1 as $E_1 = E_K(V_1, m_1)$ and sends E_1, N_0 to the S2 Node via an unsecure channel.

Step 2: S2Node $\rightarrow$ Controller:(E_2, N_3)

1) Upon receiving E_1, N_0 from the Controller, S2 Node computers $K = HKDF(slat, DH_k, \inf o)$, where $salt = N_0$.
2) V_1' and m_1 is the decryption of E_1 using the pseudo random K. S2 Node checks whether $N_1' = h(N_0 \otimes PK_b)$ or not, where PK_b is the stored public key. If same, it calculates $N_1 = m_1 \otimes V_1 \otimes DH_k$.
3) S2 Node generates a random challenge code $N_3 \leftarrow \{0,1\}^{128}$ and computes $E_2 = E_k[N_3 \otimes N_1]$ using the pseudo-random secret key K for user verification and sends E_2, N_3 to the controller.

Step 3: Verify

1) Upon receiving E_2, N_3 from the S2 Node, Controller computes $P = Dec_k[N_3 \otimes N_1]$ and checks whether $P \otimes N_1$ is equal to N_3 or not. If they are equal, the authentication is completed between the controller and the new device, and the network key exchange can continue.

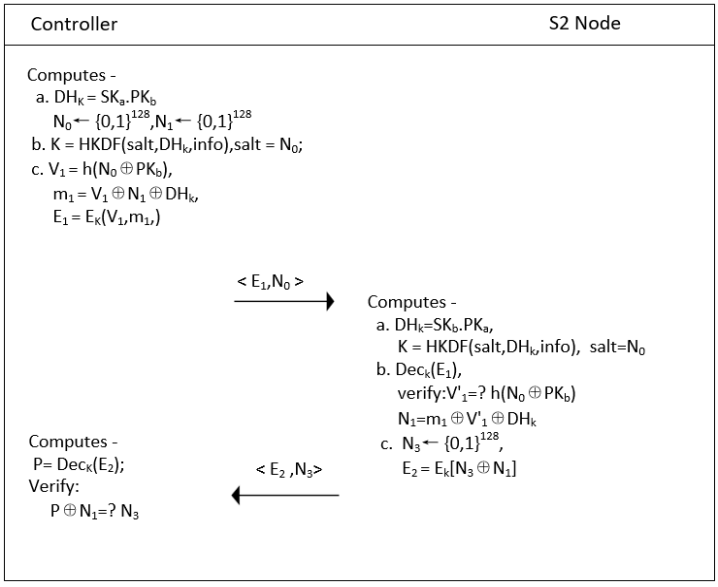

Fig. 8. Proposed scheme: authentication phase

4.2 Evaluation Model for the New Protocol

According to the improved scheme, we add an improved model to the original protocol model and introduce the same attack model as the original protocol. We set up a new evaluation model to analyze the security of the improved protocol.

The improved internal model of the Controller is shown in Fig. 9. The blue part in the figure is the improved part, which is described as follows. The transition HKDF is used to simulate the HKDF derivation function to generate a pseudo-random key okm. The HXOR transition uses random challenge codes (N0, N1), Kt, and pin to generate a verification message m1. The token of the V1 place is updated to m1. Transition AEnc-Kh is fired, which generates authentication message AU1 and sends it to interface A-Auten.

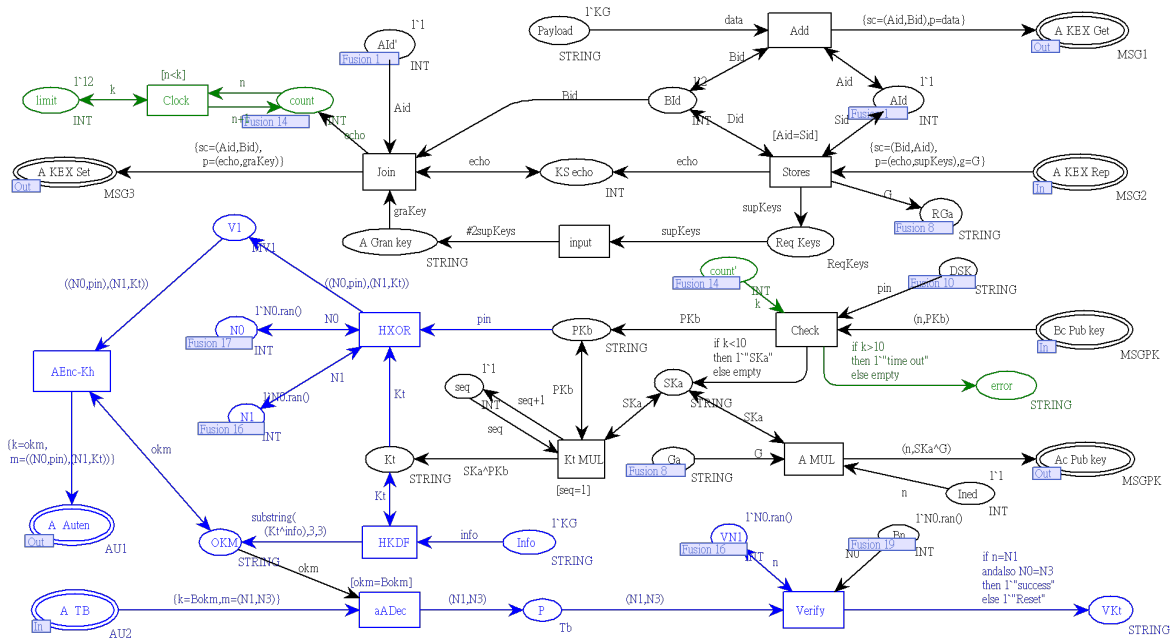

Fig. 9. Internal model of the new scheme to alternative transition Controller

After the interface A-TB receives the authentication message AU2, the transition ADec decrypts the received cipher text, and stores the decrypted plain text in the place P. Transition Verify verifies whether N1 and N3 have been tampered with by the attacker and writes the verification result to the VKt place.

The improved S2-Node internal model is shown in Fig. 10. The blue part in the figure is used to simulate authentication process. Its operation process is as follows. BDec transition is fired when B-Auten receives the message. It uses the token in BOKM' to decrypt and split the AU1 message. The token information of AV1 and Bm1is updated. Once Bverify has been fired, the tokens of BN01 and PKb will be consumed. This process indicates that S2 Node verifies the identity of the Controller and prevents attackers from tampering with the public key PKb and N0. If the verification result is False, a new token is written into Reset, which represents that the Controller is a malicious controller. If the verification result is True, the identity information will be stored in the BV1. Transition BXOR uses BKt, Kt and N0 to calculate N1. BN place will obtain the token value. N0.ran generates a token in the BNonce place, indicating that the S2 device generates a random code N3. Transition Enc-Tb encrypts N1 and N3 with the pseudo-random key Bokm and writes a new token in the interface B-TB.

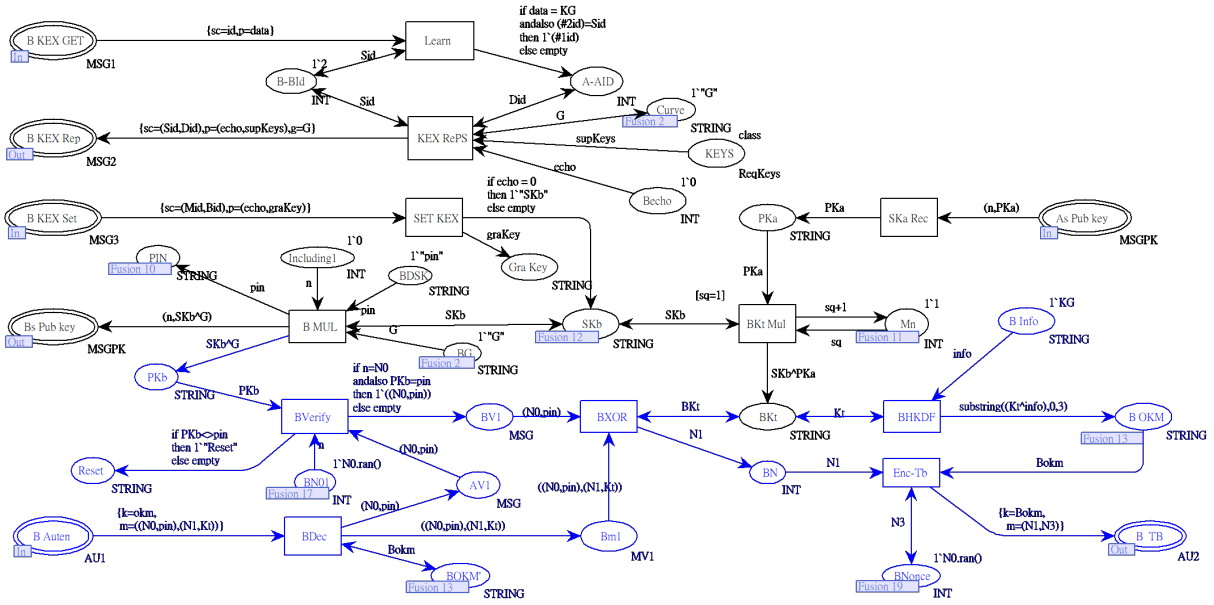

Fig. 10. Internal model of the new scheme to alternative transition S2 Node

The security evaluation model of the improved protocol is shown in Fig. 11. The blue part simulates the attacker's ability to replay messages. Depending on the ability of the supposed attacker, the intercepted messages cannot be viewed and modified because the attacker has not obtained the decryption key.

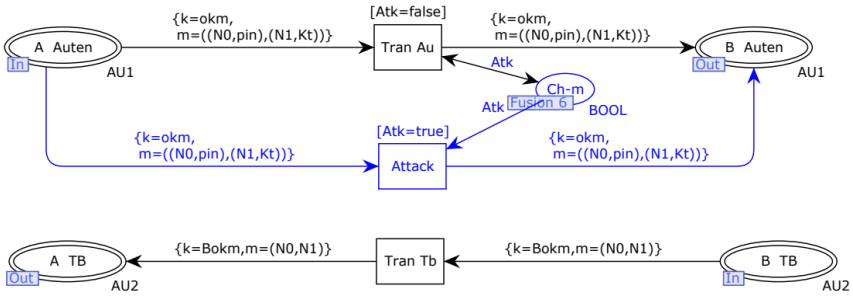

Fig. 11. A Security evaluation model for improved protocols

4.3 Security Evaluation of the New Protocol

Table 3 shows the full state space reports of Z-Wave S2 CPN model with all attacks stated in the specification. The table is composed of two parts: Initial Scheme and New Scheme. AT^{False} implies that attack parameter is set to false. AT^{True} implies that attack parameter is set to true, the attacker attacks the improved Z-Wave protocol according to the attack capability assumed in Section 4.2.

In Table 3, depicts that the State Space Nodes and Arcs have consistent values with the Strongly Connected Components (SCC) Graph Nodes and Arcs. In models, SCC is used to determine the reachability of nodes and iterations. A model with the same state space and SCC values confirms that every node can be accessed from every other node. Furthermore, there are no loops in the model that may result in infinite occurrences of the model.

Table 3. State space reports -New Z-Wave S2 CPN model

Type	Initial Scheme	New Scheme	
		AT^{True}	AT^{False}
State space nodes	58	249	126
State space arcs	146	493	213
SCC graph nodes	85	249	139
SCC graph arcs	146	493	236
Dead markings	3	4	4
Living transitions	0	0	0
Dead transitions	0	11	13

The report also shows that there exist 4 dead marking and 11 dead transitions in the state space of AT^{True}. The new dead marking indicates the termination status of a failed authentication request. Moreover, a closer inspection on the dead transitions revealed that 5 of the 11 dead transitions are marked with attack activities. For example, the attacker intercepts KEX Get and KEX REP requests, making transitions Sender1 and Sender2 unable to be enabled. The remaining dead transitions are caused by authentication failures. Authentication message AU1, AU2 using pseudo-random key (OKM) encryption. The attacker cannot decrypt and view the authentication messages, but can only replay the intercepted messages. As a result, transitions Tran-Au and Tran-Tb cannot be enabled. Secondly, after the interfaces B-Auten and B-Auten receive the message, the attacker fails the authentication, which makes the transition BXOR, Enc, Verify and ADec unable to be enabled. The analysis shows that the number of new dead transitions meets the expected goal, which verifies that the new scheme can prevent MITM attack.

The state space of AT^{False} has 4 dead nodes and 13 dead transitions. This is the verification result when the attacker parameter is set to False. The results are consistent with our expectations, implying that the authentication request from the controller to the S2 device has been successfully executed.

From the above, it can be seen that the behavior of the CPN model of the new Z-Wave protocol has achieved the expected results, verifying the security of the protocol. It shows that the authentication scheme we propose provides confidentiality and resists man-in-the-middle attacks. In the new protocol, unauthenticated malicious nodes cannot enter the communication between the controller and the slave device. The authentication technology we propose is secure and free from intermediate attacks.

4.4 Security Analysis and Comparison of the New Scheme

Our proposed technique for authentication provides confidentiality and resists man-in-the-middle attacks. According to the assumption in Section 4.3, attacker can capture and modify the message between the Controller and S2 Node. When attacker catches the authentication message <E1, N0>, it attempts to brute force attack on received encrypted payload. Since the encryption key for the authentication message is a pseudo-random key generated by the HKDF function, which prevents an attacker from brute force, an attacker cannot modify the message between Controller and S2 Node. It can be concluded that our proposed authentication scheme is secure against man-in-the-middle attacks and provides confidentiality.

The proposed authentication technique also provides for the integrity of messages between the Controller and S2 Node. In our scheme, attackers can capture <E1, N0>, <E2, N3> messages, but cannot modify the payload of these messages. If attacker modifies any of the payload of the message, the receiver can easily detect that attacker has modified the message, or the message is not coming from the intended sender. We assume that the attacker modified the message <E1, N0>. S2 Node accepts the message and decrypts the payload of the message using Key K to obtain the 128-bit verification codes V1 and N0. Then, S2 Node compares N0, PKb with V1 after Hash and XOR operations. If the message is not created by the Controller, the authentication cannot pass. Similarly, the Controller can also verify whether the message comes from the S2 node. In summary, the scheme can authenticate the sender of the message and determine the integrity of these messages.

Prior to our work, research in literature [16-19] has revealed protocol implementation vulnerabilities, Z-Wave network key retrieval, rogue controller insertion, Z-Wave threat identification, and DoS on Z-Wave controllers. Compared with this paper, these researchers evaluate the device security and privacy assurance of Z-Wave protocol, but do not propose corresponding hardening methods. Based on the security analysis, this paper improves the security vulnerabilities and adopts an effective verification method for the proposed scheme. Secondly, the formal model detection method based on colored Petri net and Dolev-Yao model theory are adopted in this paper,

which can provide an intuitive and accurate graphical description of protocol security research methods.

5 Conclusion

This paper mainly studies the security of the Z-Wave. Firstly, the article introduces the Z-Wave S2 security class, focusing on the connection process when a new device joins the network. During this process, a new device joins the Z-Wave network and obtains a symmetric key to communicate in the future. Second, we present a formal analysis of the process and propose a lightweight two-way authentication scheme to avoid adding malicious nodes to the Z-Wave network, which uses only two handshake messages for authentication and is suitable for resource Limited equipment. Finally, we conduct a formal analysis of the new scheme, which shows that the new scheme can effectively prevent man-in-the-middle attacks. The results show that the new scheme can effectively prevent man-in-the-middle attacks and has good confidentiality and integrity. However, in the text, we mainly focus on the security of the protocol itself, without considering the performance of the protocol, such as the simulation analysis of the authentication time, power consumption and other parameters of the improved scheme. In the future research work, we will analyze the protocol performance while enhancing the protocol security, and try to find the balance between security and performance.

Acknowledgement

This work is supported by the National Science Foundation of China (No. 62162039, No. 61762060), Key Research and Development Program of Gansu Provincial Department of Science and Technology (No. 20YF3GA0).

References

[1] Z. Krasniqi, B. Vershevci, Smart home: Automatic control of lighting through z-wave IoT technology, in: Proc. UBT International Conference, 2020.
[2] I. Unwala, Z. Taqvi, J. Lu, IoT security: ZWave and Thread, in: Proc. 2018 IEEE Green Technologies Conference, 2018.
[3] S. Zroug, L. Kahloul, S. Benharzallah, K. Djouani, A hierarchical formal method for performance evaluation of WSNs protocol, Computing 103(6)(2021) 1183-1208.
[4] G. Kambourakis, C. Kolias, D. Geneiatakis, G. Karopoulos, G.M. Makrakis, I. Kounelis, A state-of-the-art review on the security of mainstream IoT wireless PAN protocol stacks, Symmetry 12(4)(2020) 579.
[5] K. Jensen, L.M. Kristensen, Coloured Petri nets: modelling and validation of concurrent systems, Int J Softw Tools Technol Transfer 9 9(2009).
[6] B. Fouladi, S. Ghanoun, Security evaluation of the Z-Wave wireless protocol, Black hat USA 24(2013) 1-2.
[7] K. Hofer-Schmitz, B. Stojanović, Towards formal verification of IoT protocols: A Review, Computer Networks 174(2020) 1-6.
[8] M. Lilli, C. Braghin, E. Riccobene, Formal Proof of a Vulnerability in Z-Wave IoT Protocol, in: Proc. SECRYPT, 2021.
[9] X. Gong, T. Feng, J. Du, Formal modeling and security analysis method of security protocol based on CPN, Tongxin Xuebao/Journal on Communications 42(2021) 240-253.
[10] R. Amoah, S. Camtepe, E. Foo, Formal modelling and analysis of DNP3 secure authentication, Journal of Network Computer Applications 59(2016) 345-360.
[11] J. Diaz, D. Arroyo, F.B. Rodriguez, A formal methodology for integral security design and verification of network protocols, JoJournal of Systems and Software 89(2014) 87-98.
[12] E. Coronado, V. Valero, L. Orozco-Barbosa, M.E. Cambronero, F.L. Pelayo, Modeling and simulation of the IEEE 802.11 e wireless protocol with hidden nodes using Colored Petri Nets, Software Systems Modeling 20(2)(2021) 505-538.
[13] F.-Y. Luo, T. Feng, L. Zheng, Formal Security Evaluation and Improvement of Wireless HART Protocol in Industrial Wireless Network, Security Communication Networks 2021(2021) 1-15.
[14] H. Krawczyk, Cryptographic extraction and key derivation: The HKDF scheme, in: Proc. Annual Cryptology Conference, 2010.
[15] M. Fischlin, C. Janson, S. Mazaheri, Backdoored hash functions: immunizing HMAC and HKDF, in: Proc. 2018 IEEE 31st Computer Security Foundations Symposium, 2018.

[16] J.D. Fuller, B.W. Ramsey, Rogue Z-Wave controllers: A persistent attack channel, in: Proc. 2015 IEEE 40th Local Computer Networks Conference Workshops, 2015.
[17] C.W. Badenhop, S. Graham, W.P. Ramsey, B. Mullins, L. Mailloux, The Z-Wave routing protocol and its security implications, Computers & Security 68(2017) 112-129.
[18] J.D. Fuller, E.W. Ramsey, M.J. Rice, J.M. Pecarina, Misuse-based detection of Z-Wave network attacks, Computers & Security 64(2017) 44-58.
[19] K. Kim, K. Cho, J. Lim, Y.H. Jung, M.S. Sung, S.B. Kim, H.K. Kim, What's your protocol: Vulnerabilities and security threats related to Z-Wave protocol, Pervasive Mobile Computing 66(2020) 101211.
[20] C.K. Nkuba, S. Kim, S. Dietrich, H. Lee, Riding the IoT Wave With VFuzz: Discovering Security Flaws in Smart Homes, IEEE Access 10(2022) 1775-1789.
[21] D. Dolev, A. Yao, On the security of public key protocols, IEEE Transactions on Information Theory 29(2)(1983) 198-208.
[22] Z.U. Rehman, S. Altaf, S. Iqbal, An efficient lightweight key agreement and authentication scheme for WBAN, IEEE Access 8(2020) 175385-175397.
[23] C. Patel, N. Doshi, Secure Lightweight Key Exchange Using ECC for User-Gateway Paradigm, IEEE Transactions on Computers 70(11)(2021) 1789-1803.
[24] K. Sowjanya, M. Dasgupta, S. Ray, Elliptic Curve Cryptography based authentication scheme for Internet of Medical Things, Journal of Information Security and Applications 58(2021) 102761.
[25] M. Troncoso, B. Hale, The Bluetooth cyborg: Analysis of the full human-machine passkey entry AKE protocol, Cryptology ePrint Archive 2021(2021) 83.
[26] S.G. Oliver, T. Purusothaman, Lightweight and Secure Mutual Authentication Scheme for IoT Devices Using CoAP Protocol, Computer Systems Science and Engineering 41(2)(2022) 767-780.
[27] A. Gupta, M. Tripathi, T.J. Shaikh, A. Sharma, A lightweight anonymous user authentication and key establishment scheme for wearable devices, Computer Networks 149(2019) 29-42.