Lu and Feng *EURASIP Journal on Wireless Communications and Networking* (2018) 2018:108
https://doi.org/10.1186/s13638-018-1129-y

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Research on trusted DNP3-BAE protocol based on hash chain

Ye Lu[1,2,3*] and Tao Feng[1,4]

Abstract

To solve the security problem of industrial Ethernet DNP3 protocol broadcast authentication, the attack vector and security requirements of trusted DNP3 protocol are analysed. First, the paper adopts a trusted platform into the control network and authenticates the identity and security status of the DNP3 client and server to prevent node sensitive information from being compromised. Second, a trusted DNP3-BAE broadcast authentication encryption protocol is proposed based on the hash chain method to solve the problem of missing message security authentication mechanism in broadcast mode, which only needs a key to complete the broadcast message authentication for multiple slaves. The new scheme can use the DNP3-SA encryption primitive, without a major upgrade to the existing platform. The protocol is verified by the SPAN tool; the results show that there is no intrusion path, which ensures the integrity, authenticity, freshness, and confidentiality of the communication nodes. At present, there is no public document to introduce a trusted platform into the DNP3 protocol to solve security problems. Performance analysis shows that our solution reduces the overhead of large-scale broadcast authentication at the expense of increased less processing and storage overhead.

Keywords: Industrial control system, DNP3 protocol, Trusted Computing, Span

1 Introduction

With the advance of Industry 4.0, information security research for industrial control network has become a hot topic. More and more cyber-attacks indicate that SCADA is unsafe [1–3]. The industrial Ethernet protocol based on TCP/IP technology is widely used in SCADA system. Although the requirement of remote monitoring of ICS field devices is realized, the original industrial Ethernet protocol is facing more threat of network attacks. The widespread use of the DNP3 protocol in the field of SCADA systems has proven unsafe [4–6]. Once the attacker made through the Ethernet into the internal DNP3 network, they can obtain and control the ICS data and sensitive operations by impersonating the DNP3 client and server. Therefore, the DNP3 protocol must be researched and improved from the perspective of the communication side to ensure the communication security.

The latest security improvement version DNP3-SA only proposed unicast communication under the authentication strategy and did not give the multicast communication authentication and encryption program. An attacker could exploit this vulnerability to modify commands on multiple external sites or execute an unauthorized command to interrupt the service. Although [7] proposed a public key authentication scheme based on ECC for multicast security problems, the computational complexity is high, and special hardware is needed, and the adaptability of high real-time power SCADA system is poor. Therefore, it is urgent to propose a lightweight broadcast authentication encryption scheme. μTESLA protocol [8] is based on a hash chain technology, with low cost and high efficiency, and can be in the symmetric encryption system to complete multi-message authentication, in line with the network real-time and security needs. Literature [9] studied the multi-node authentication scheme of μTESLA protocol in wireless sensor networks, which enabled many sensor nodes to verify the identity of the communication participants.

In addition, the above security improvements to the protocol cannot resist the attacker posing and tamper

* Correspondence: luye528@126.com
[1]College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou, China
[2]Key Laboratory of Gansu Advanced Control for Industrial Processes, Lanzhou University of Technology, Lanzhou, China
Full list of author information is available at the end of the article

with DNP3 server and client attack threats. Literature [10] proposed the introduction of trusted anchor technology into ICS embedded devices to prevent equipment from being impersonated but lacks security for servers and protocols. The Trusted Computing Group (TCG) introduces the Trusted Computing Concept [11] into ICS and proposes a remote security communication based on the trusted platform module (TPM) built-in key [12] but lacks strengthening the security of communications between field devices. Literature [13] proposed the use of trusted platform to protect and evaluate the terminal equipment data security and trusted state, but the security reinforcement of industrial Ethernet protocol is not given. There is no other public research to introduce trusted components into the DNP3 protocol to ensure the safety of field devices.

This paper is organized as follows. In Section 2, the attack vector and loophole of DNP3 protocol broadcast communication mode are analysed. In Section 3, we authenticate the identity and security status of the client and server based on TPM and propose a DNP3-BAE lightweight broadcast authentication encryption protocol based on hash chain method. The protocol can use DNP3-SA encryption primitives without significant upgrades on existing platforms. In Section 4, the SPAN tool is used to verify the security of the scheme. In Section 5, the performance of our protocol is analysed. At last, Section 6 presents the overall conclusion.

2 Analysis of the attack vector and security loophole

In this section, the model of security threats faced by the DNP3-SA communication protocol in SCADA system has been proposed. A SCADA system consists of MS (Monitoring station), HMI (Human machine interface) and other equipment such as PLC and IED. DNP3-SA protocol using C/S mode of communication and MS (DNP3 client) communicate with PLC (DNP3 field server) through the configuration software (CS). PLC program (PLC program) collects the scene data and sends back to the MS. The DNP3 protocol communication threat model is illustrated in Fig. 1.

MS is a client communicating with multiple PLC servers on the site. The shaded part indicates that there is a threat to the current device. The Dolev-Yao adversary model shows that the attacker has enough ability to eavesdrop, replay, tamper with and fake any arbitrary network packets. The following four types of attack vectors are available:

1) Attack vector based on MS impersonator

As the DNP3-SA protocol lacks the identity authentication mechanism, the impersonator can forge DNP3 request message by eavesdropping the PLC communication address and send the malicious control command to the PLC. Since the impersonator cannot obtain the session key, the DNP3-SA security improvement protocol [7, 14] based on the authentication of both parties can prevent such attacks.

2) Attack vectors based on CS vulnerabilities

An attacker can exploit a CS vulnerability to obtain native sensitive information, send a malicious command to the PLC, and tamper with the response message of the PLC. If the attacker steals the pre-set key through the controlled CS, the DNP3-SA security improvement protocol [7, 14] based on the authentication of both parties will not be able to prevent such attacks.

3) Attack vector based on PLC impersonator

As the DNP3 protocol lacks the identity authentication mechanism, impersonators can obtain DNP3 response messages by eavesdropping to obtain PLC and CS communication addresses, causing malfunction. Since the impersonator cannot obtain the session key, the DNP3-SA security improvement protocol [7, 14] based on the authentication of both parties can prevent such attacks.

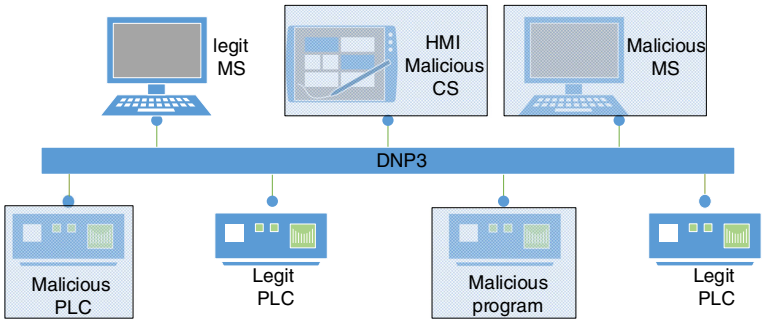

Fig. 1 DNP3 communication threat model

4) Attack vector based on PLC program

As the PLC usually used weak password protection mechanism, an attacker can crack the password and other ways to implant malicious program, to obtain sensitive information or cause failure. If the attacker steals the pre-set key through the controlled PLC, the DNP3-SA security improvement protocol based on the authentication of both parties [7, 14] will not prevent such attacks.

According to the literature [4, 6] and what we know, DNP3 protocol mainly exists in the following attack types: eavesdropping, tampering, posing, DOS, and replay. More specifically, there are mainly three loopholes:

1) The first type of loophole: In the NACR mode, randomly changing the value of the message sequence number "Ksn" and the random number will make the protocol lose synchronization, resulting in unexpected authentication failure of the protocol. The attacker "I" observed that in the continuous NACR mode, the message sequence number "Ksn" will increase by one when the slave sends a new NACR request. The attacker "I" can tamper with the sequence number "Ksn" of the challenge message sent from the station "O", posing as a station "O" and sending it to the master station "M" to induce the master station to generate an incorrect message authentication code, which results in failure of the message authentication. Figure 2 gives the MSC model of this attack behaviour.
2) The second type of loophole: By observing a large amount of information in the cleartext challenge, the attacker "I" can replay the message authentication code "tag_old" intercepted in the previous rounds of the session to the station "O" to trigger the execution of false orders "Prented_RSP" when finding the challenge information with the same message sequence number "Ksn" and random number. Figure 3 shows the MSC model of this attack behaviour.
3) The third type of loophole: By observing a large amount of information in the plaintext challenge, the attacker "I" replays the AGM active request message "Old_AGMRQ" intercepted in the previous rounds of sessions to the station "O" when it finds the challenge information with the same message sequence number "Ksn". This loophole caused station "O" to perform unauthorized critical operations. Figure 4 shows the MSC model of this attack behaviour.

3 Design of the new credible DNP3-BAE protocol

The credible DNP3-BAE protocol proposed in this paper includes two parts: identity authentication and key agreement sub-protocol and key update and broadcast message authentication sub-protocol. The identity authentication sub-protocol provides periodic verification and updating of the identity and security status information for the MS(M) and PLC(O) by configuring the trusted platform and increasing the authentication server (A). There is no research yet using the trusted platform for the identity of the certification in DNP3 protocol device. The key agreement sub-protocol completes the negotiation of the secret key after the authentication succeeds to facilitate the symmetry encryption of the operation data required for high security level communication. The key update sub-protocol periodically updates the key to ensure

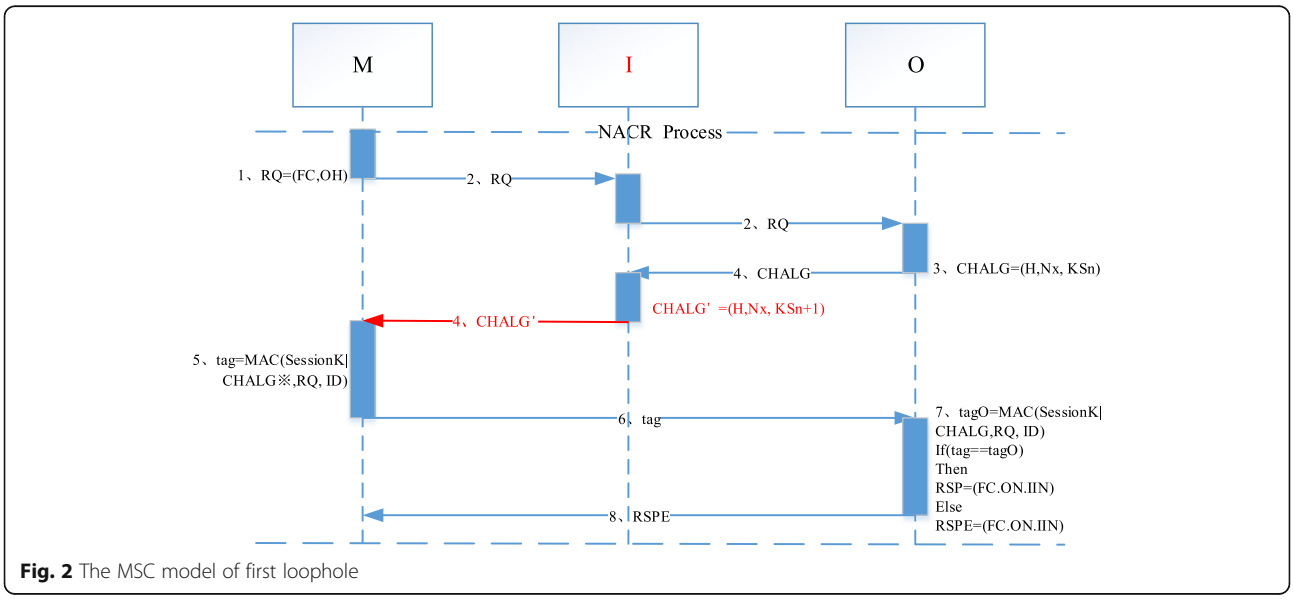

Fig. 2 The MSC model of first loophole

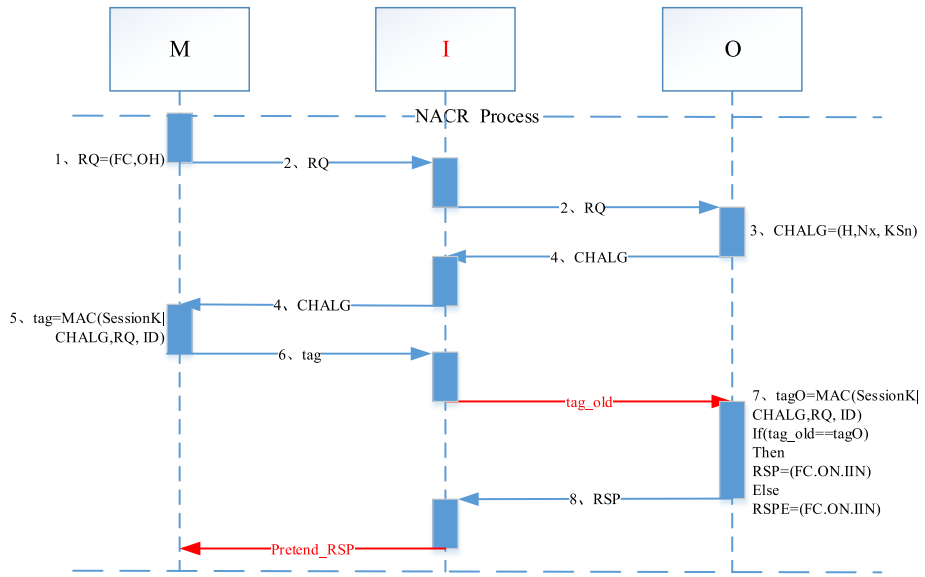

Fig. 3 The MSC model of second loophole

data security, and the AS solves the trustworthiness of the device state by periodically querying the PCR of the MS and PLC. The broadcast authentication sub-protocol completes the sending and authentication of the broadcast message based on the hash chain. The biggest advantage of this method is that only one key can be used to complete the broadcast message authentication of multiple slaves, which can greatly reduce the large-scale SCADA broadcast communication overhead.

Before the protocol is designed, assume that the communication participant has the following knowledge:

1) The MS communicates with multiple slave PLCs at the same time.
2) The communication request is initiated by the MS.
3) The base layer of the protocol and AS are reliable.
4) MS, AS, and PLC are based on TPM hardware to achieve a trusted function. All commands beginning with TPM_ are done in TPM hardware and software.
5) A trusted list is assigned to the SCADA system, which is the expected trusted information of all terminal devices.
6) MS, PLC known AS's identity certificate public key A_AIK_Pub and bound public key KA_Pub.

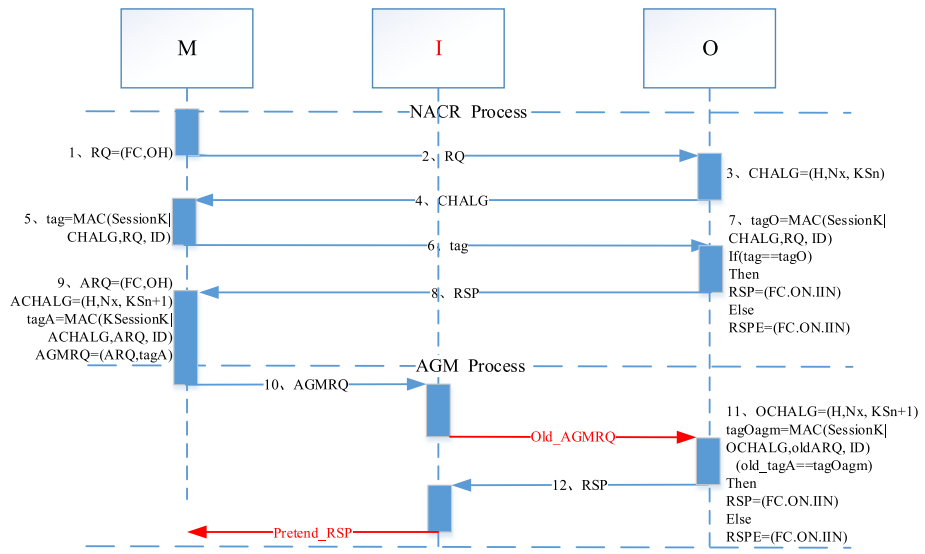

Fig. 4 The MSC model of third loophole

3.1 Authentication and key agreement protocol

Trusted Computing [9, 12] measures the hardware and software reliability of the device through the trusted metric root in the BIOS of TPM device. The measurement results are stored in the platform configuration register (PCR) inside the TPM and cannot be tampered. PCR is used for user authentication and protects the hardware and software systems of the equipment in line with expectations. When the terminal device is initialized, the authentication key pair (AIK) is created by the TPM. The private key of the AIK is stored in the device TPM. Verifying the AIK private key signature can guarantee the authenticity of the device identity. Bind-Key is a pair of public and private key pairs that the TPM uses to decrypt small-scale data (such as a key). The encrypted data must be decrypted on a device with a Bind- private key.

Figure 5 is the identity authentication sub-protocol and key-key negotiation sub-protocol message flow, M is DNP3 communication in the client (MS), **O** for the server (PLC). O_AIK_Pri, O_AIK_Pub, M_AIK_Pri, and M_AIK_Pub are the identity certificate key pairs (AIK); KA_Pr, KA_Pub, KM_Pri, KM_Pub, KO_Pri, and KO_Pub are the binding key pairs (Bind-Key); PcrO and PcrM are the trust metric root. KH is used for HMAC calculations in communication sub-protocols to ensure the integrity of communication data; KE is used for symmetric encryption of critical data required for high-security communications. Random numbers Na, Nb, Nc, Nd, Ne, Nf, and Ng ensure the freshness of the message. Note that O as a DNP3 server cannot initiate a request, all communication initiated by M.

Steps 1 to 14 describe the M and O request AIK signatures of the device status information (PCR value) to the opposite party to complete the two-way authenticate process with the assistance of AS. Among them, the TPM_E and TPM_D commands encrypt and decrypt the PCR and protocol data respectively.

Steps 15 to 20 describe the negotiation process of the message authentication key KH and the message encryption key KE after confirming the identity information between M and O with the assistance of A.

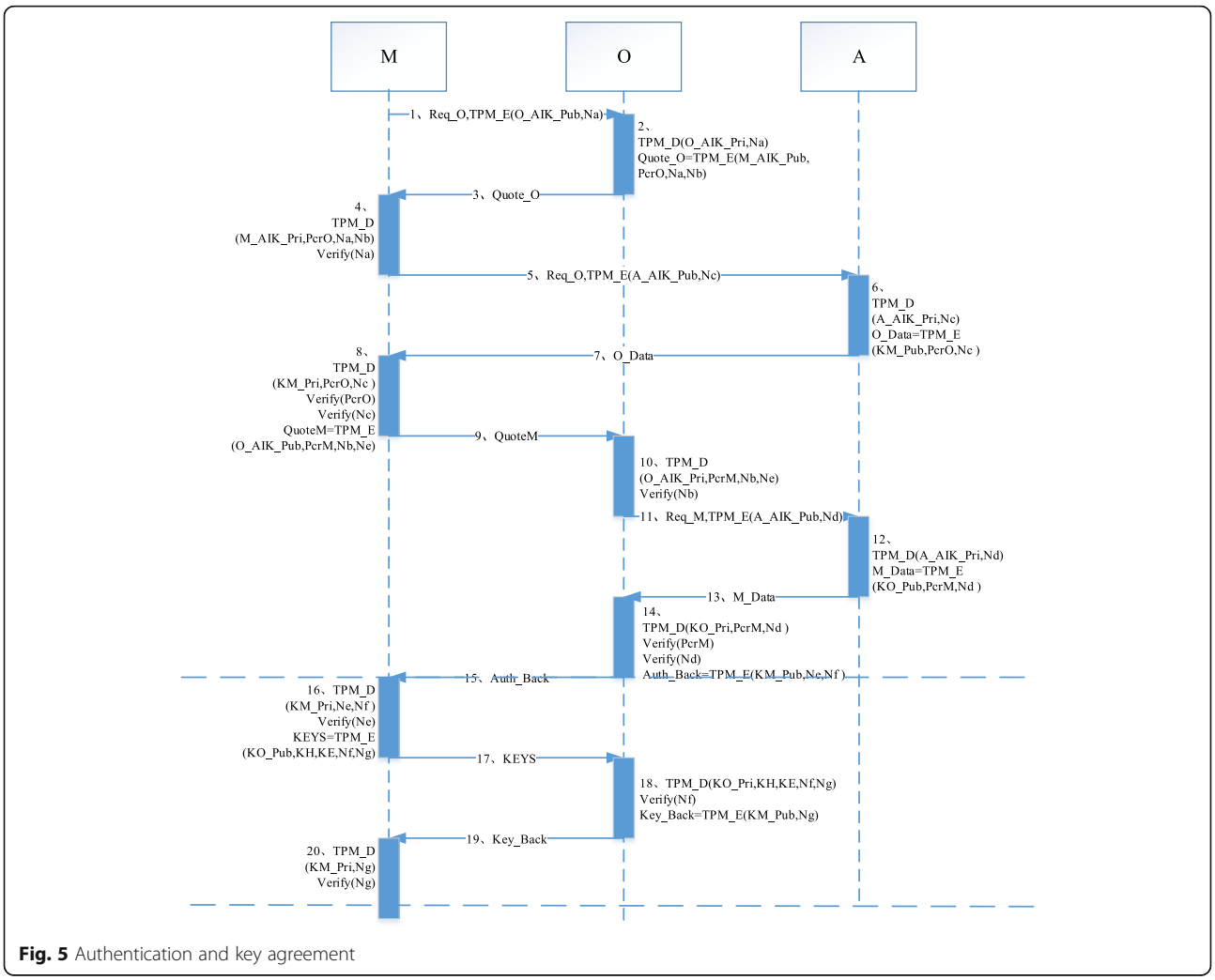

Fig. 5 Authentication and key agreement

AS periodically polls MS and PLC for PCR and compares it with whitelist information to find out whether there is any unexpected change in equipment status and ultimately ensure that the SCADA system terminal equipment has not been tampered with during operation. Depending on whether the verification result is successful or failed, the AS will decide whether to update the ICS device status: (1) If the verification is successful, AS does not do anything; (2) if the authentication fails, or if the administrator updates the whitelist information on the AS, the update process is initiated by the AS. Upon receipt of an AS-initiated update notification, the MS or PLC sets the symmetric keys KH and KE negotiated in the pre-Key Negotiation sub-protocol to be invalid and re-initiates the identity authentication sub-protocol, this process is not repeated by the length limit.

3.2 Key update and broadcast message authentication sub-protocol

Figure 6 shows the flow of key update and broadcast message authentication sub-protocol. **R** is a random bit string whose length is 128 bits, and H represents the hash function used to calculate the hash chain label: S1 = H (R), S2 = H (S1), ..., Sn = H (Sn-1), which represents a hash chain label generated from a single random value (R). Sn is the last tag of the hash chain, and n is the hash chain index. Req(FC) is the key update request, where FC is the Function Code. Chalg (H, Nx, KSn) is the key update response, where H is the hash algorithm used, KSn is the message serial number, and Nx is the random number. MAC (K_H, KS, KSR) and E (K_E, K_E_new, KH_new, Sn, n) respectively authenticate and encrypt the message. BM represents the broadcast information; Si is the current hash tag; *i* is the index of the current hash tag; tagA is the MAC tag; Sstored, Nstored, Sreceived, Nreceived, and Scalc store the values of Sn, *n*, Si, *i* (where $1 < i < n$), and Scalc; and Scalc represents the variable used to find the next hash tag for the hash chain.

Based on the hash chaining method, the master M uses the redistributed credentials, and the external site O can only verify the current hash of the current broadcast message, preventing forward attacks, effectively protecting the DNP3 broadcast from tampering, replay and inject attack. The scheme uses only one symmetric key hash chain mechanism in multiple broadcast messages

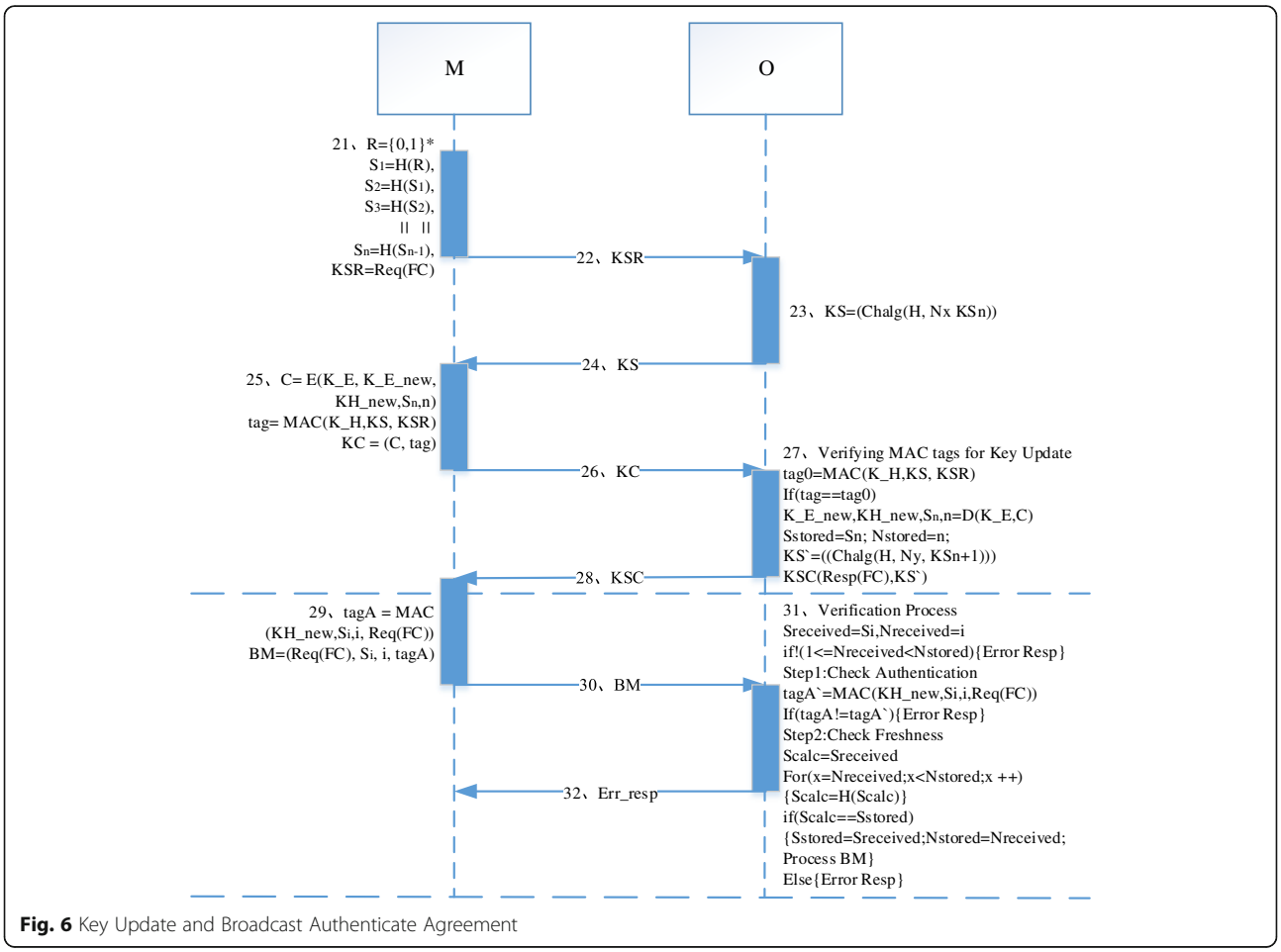

Fig. 6 Key Update and Broadcast Authenticate Agreement

instead of multiple broadcast messages using multiple keys of the HMAC mechanism, reducing the communication overhead, especially for large-scale broadcast communications. In addition, this method does not require a public key infrastructure (PKI), using only the original encryption primitives (SHA-1, SHA-256, AES-GMAC, and AES) of DNP3-SA, reducing the cost of updating the device.

4 Performance analysis

Since the authentication and key negotiation sub-protocol is initiated only when the device status information is changed (as before the first communication, the authentication fails) and the message is processed using the dedicated TPM hardware and software, the part of the protocol Performance costs has less impact on the time overhead of the entire broadcast communication. The key update and broadcast message authentication sub-protocol is used frequently in broadcast communications, which uses the encryption primitives in the DNP3-SA specification without the TPM-related time overhead, but adds some communication, computation and storage overhead.

Table 1 shows the comparison of calculation and storage overhead for key update and broadcast message authentication sub-protocol with DNP3-SA unicast protocol (NACR and AGM mode) in broadcast communication environment and gives the performance overhead of the three protocols in the case of a man-in-the-middle attack at probability p. **N** is the number of PLCs, n is the number of broadcast messages, and the number of hash tags is generated. MIRB, MIRN and MIRA represent tampering, injection and replay attacks for BAE, NACR and AGM respectively.

Taking the performance overhead of the BAE scheme as an example, $4N + n$ on the communication overhead means that four messages are exchanged to N external sites and n broadcast authentication messages (see Fig. 5) during the key update, which corresponds to O $(N + n)$. For the calculation overhead, 2n + 2 indicates that the MS needs to generate n hash values and $n + 1$ message authentication codes during the broadcast message authentication phase and needs to complete one encryption operation during the key update phase. Similarly, each PLC needs 4Nn cryptography; for storage overhead, $2n + 2$ means that the MS stores n indexes (i) and n hash values (Si), and two keys (KE and KH). Similarly, the $8N$ at the PLC side means that each PLC stores eight parameters: KH, KE, chalg, Sn, i, Scalc, Sstored, and Nstored. The results of the comparison in Table 1 show that BAE has significantly reduced communication overhead at the minor cost of increasing the computational and processing overhead compared to its corresponding NACR and AGM; especially in the case of intermediate attacks such as modification, injection and reproduction, BAE communication overhead is also significantly reduced, applied to a wide range of broadcast communications; this advantage will be more significant. In addition, compared with NACR and AGM, the computational overhead of the master station in the BAE scheme is also significantly reduced.

Table 2 shows the scheme using a symmetric (Sym) and asymmetric (Asym) cryptography method to compare with the BAE scheme. The results show that the BAE scheme requires only one broadcast authentication key, and the Sym scheme needs to establish n broadcasts for n broadcast messages, and the demand participants must be honest. The Asym program also needs basic public facilities to distribute $N + 1$ public and private key pairs.

5 Simulation results

The trusted DNP3-BAE protocol is described using the role-based formalized protocol language HLPSL, and then, the SPAN tool is used to verify the security of the protocol. The SPAN tool simulates the protocol functions and intruder behaviour described in the HLPSL language and gives the corresponding attack path if the protocol is insecure. Taking the identity authentication sub-protocol as an example, the HLPSL language is used to describe the three roles (MS, PLC and AS) processes and hybrid role participating in the protocol, wherein the role process defines a communication process and an entity variable for the role receive and response message. The hybrid role process defines protocol variables, attacker knowledge, and protocol validation targets. This

Table 1 Performance analysis and comparison

Scheme	Communication overhead	Calculate overhead		Storage overhead	
		MS	PLC	MS	PLC
Ours	$4N + n \approx$ O $(N + n)$	$2n + 2 \approx$ O (n)	$4Nn \approx$ O (Nn)	$2n + 2 \approx$ O (n)	$8N \approx$ O (N)
NACR	$4N(n + 1) \approx$ O (Nn)	$Nn \approx$ O (Nn)	Nn ≈ O (Nn)	4	$4N \approx$ O (N)
AGM	$8N + 2Nn \approx$ O (Nn)	$N(n + 1) \approx$ O (Nn)	$Nn \approx$ O (Nn)	4	$4N \approx$ O (N)
MIR_B	$nP \approx$ O (n)	$nP \approx$ O (n)	$nP \approx$ O (n)	/	
MIR_N	$2NnP \approx$ O (Nn)	$NnP \approx$ O (Nn)	$NnP \approx$ O (Nn)	/	
MIR_A	$2NnP \approx$ O (Nn)	$NnP \approx$ O (Nn)	$NnP \approx$ O (Nn)	/	

Table 2 Scheme analysis and comparison

Scheme	Method	Infrastructure	Keys	Computational complexity
Ours	Hash chain	/	1	n*HMAC, n*MAC,1*AES
BAE-Sym	Session key	/	N	n*HMAC, n*MAC,n*AES
BAE-Asym	Digital signature	PKI	2 N + 2	n*PKI

article uses master to represent MS, the entity M in Fig. 7; uses out to represent PLC, the entity O in Fig. 5; and uses server to represent AS, the entity A in Fig. 7. Limited to space, Fig. 7 depicts the communication process for the master role.

Figure 8 depicts the attacker's knowledge and security objectives, including entities (m, o and a) and plaintext information in the protocol process. The plaintext information refers to the cryptographic algorithm and the public key used. The security objective of the identity authentication sub-protocol and the key-negotiation sub-protocol is to ensure the confidentiality of the authentication key KH and the encryption key KE used in the communication sub-protocol, PCR (PcrO and PcrM), and all random numbers such as Na Etc., with strong authentication (authenticity and freshness).

The SPAN authentication result of the authentication and key agreement sub-protocol is security (SAFE). As shown in Fig. 9, the message sequence of the protocol given by SPAN analyses the protocol security from the perspective of the intruder and fails to form an intrusion path. This result indicates that the sub-protocol can securely authenticate the identity and status information of M and O and can safely exchange the keys KH and KE used by subsequent communications because the attacker cannot obtain the AIK key and bind-private key, cannot tamper with PCR and random number signature, and also cannot decrypt the key KH and KE.

The security objective of the key update and broadcast message sub-protocol is to ensure that the data exchanged between M and O is confidential (high security level) and has integrity, using random numbers with strong authentication. The security target and authentication process of the key update sub-protocol and the broadcast authentication sub-protocol are like the authentication sub-protocol. The SPAN verification result is also safe and will not be repeated. In summary, the credible DNP3-BAE protocol proposed in this paper can guarantee the identity and status of the communication entity, the integrity of the protocol data, the freshness of

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
role master(M, A, O: agent,
            M_AIK_Pub, O_AIK_Pub, A_AIK_Pub,
            KM_Pub, KA_Pub, KO_Pub : public_key,
            SND_OM, RCV_OM, SND_AM, RCV_AM: channel (dy))
played_by M
def=
  local  State : nat,
         Na, Nb, Nc, Ne, Nf, Ng : text,
         KH, KE : symmetric_key
         %%REQ_O, PcrO, PcrM: text
  init   State := 0
  transition
  1. State = 0   /\ RCV_OM(start) =|>
     State' := 1 /\ Na'=new()
                 /\ SND_OM({Na'}_O_AIK_Pub)
  2. State = 1   /\ RCV_OM({PcrO.Na.Nb}_M_AIK_Pub) =|>
     State' := 2 /\ Nc'=new()
                 /\ SND_AM({Nc'}_A_AIK_Pub)
                 /\ request(M,O,master_out_na,Na)
  3. State = 2   /\ RCV_AM({PcrO.Nc}_KM_Pub) =|>
     State' := 3 /\ Ne'=new()
                 /\ SND_OM({PcrM.Nb,Ne}_O_AIK_Pub)
                 /\ request(M,O,master_out_nc,Nc)
  4. State = 3   /\ RCV_OM({Ne.Nf}_KM_Pub) =|>
     State' := 4 /\ Ng'=new()
                 /\ SND_OM({KH,KE,Nf,Ng}_KO_Pub)
                 /\ request(M,O,master_out_ne,Ne)
  5. State = 4   /\ RCV_OM({Ng}_KM_Pub) =|>
     State' := 5 /\ request(M,O,master_out_ng,Ng)
end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Fig. 7 The communication process of entity M

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
role environment()
def=const m, o, s : agent,
       o_AIK_Pub, m_AIK_Pub, a_AIK_Pub, i_AIK_Pub
       kO_Pub, kM_Pub, kA_Pub, kI_Pub: public_key,
       master_out_na, master_out_nb, master_out_nc,
       master_out_nd, master_out_ne, master_out_nf,
       master_out_ng, kh, ke, pcro, pcrm: protocol_id
intruder_knowledge =
       {m, o, s, o_AIK_Pub, m_AIK_Pub, a_AIK_Pub,
        i_AIK_Pub, kO_Pub, kM_Pub, kA_Pub, kI_Pub}
composition
      session(m, a, o, o_AIK_Pub, m_AIK_Pub, a_AIK_Pub,
              i_AIK_Pub, kO_Pub, kM_Pub, kA_Pub, kI_Pub)
   /\ session(m, a, i, o_AIK_Pub, m_AIK_Pub, a_AIK_Pub,
              i_AIK_Pub, kO_Pub, kM_Pub, kA_Pub, kI_Pub)
   /\ session(i, a, o, o_AIK_Pub, m_AIK_Pub, a_AIK_Pub,
              i_AIK_Pub, kO_Pub, kM_Pub, kA_Pub, kI_Pub)
end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
goal
  secrecy_of kh, ke
  authentication_on master_out_na
  authentication_on master_out_nb
  authentication_on master_out_nc
  authentication_on master_out_nd
  authentication_on master_out_ne
  authentication_on master_out_nf
  authentication_on master_out_ng
  authentication_on pcro
  authentication_on pcrm
end goal
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Fig. 8 Attacker knowledge and security goals

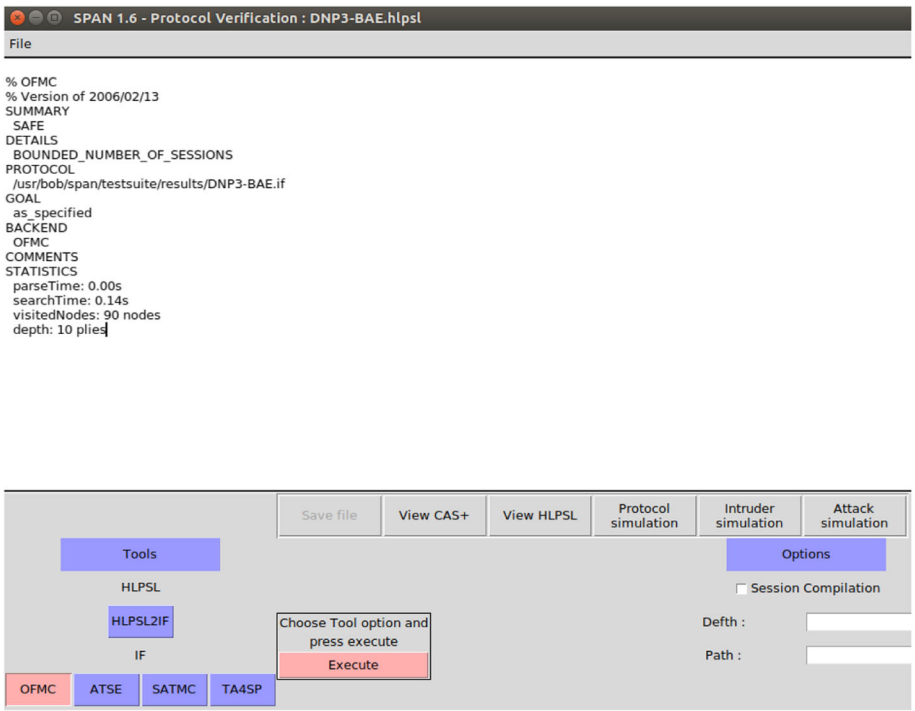

Fig. 9 Verification result by SPAN

the random number, and the confidentiality of the protocol data under the high security level and meet the safety requirements of the protocol proposed in Section 2.

6 Conclusions

This paper analysis the security requirements of trusted DNP3-SA protocol and designs a trusted DNP3-BAE broadcast authentication and encryption protocol based on hash chain method for the lack of security authentication encryption mechanism of DNP3-SA protocol broadcast communication mode. The SPAN verification results show that our solution not only solves the problem that the existing DNP3 network sensitive operation data is easy to be stolen and tampered by the illegal communication entity, but also solves the legitimate communication entities which are controlled to lead to security certification information disclosure problems. The protocol can use DNP3-SA encryption primitives without the need for major upgrades to existing platforms, and performance analysis shows that our solution reduces the overhead of large-scale broadcast authentication at the expense of minor increased processing and storage overhead.

7 Method

In this work, we aim to solve the security problem of industrial Ethernet DNP3 protocol in broadcast mode. We adopt a trusted platform into the control network to authenticate the identity and security status of the DNP3 client and server. We proposed a trusted DNP3-BAE broadcast authentication encryption protocol based on the hash chain method to solve the problem of missing message security authentication mechanism in broadcast mode. The simulation results are generated using Span software.

Abbreviations

A: Authentication server of DNP3 communication network; BM: The broadcast information; Chalg(H, Nx, KSn): The key update response; E(K_E, K_E_new, KH_new, Sn, n): Encrypt the message; H: The hash function used to calculate the hash chain label; I: The index of the current hash tag; KA_Pr, KA_Pub: Binding key pairs of A with TPM (Bind-Key); KE: The key used for symmetric encryption of critical data; KH: The key used for HMAC calculations in communication sub-protocols; KM_Pri, KM_Pub: Binding key pairs of M with TPM (Bind-Key); KO_Pri, KO_Pub: Binding key pairs of O with TPM (Bind-Key); KSn: The message serial number; M: Master station and Client of DNP3 communication network; M_AIK_Pri, M_AIK_Pub: Identity certificate key pair for the client of DNP3 communication network; MAC (K_H, KS, KSR): Authenticate the message; Na, Nb, Nc, Nd, Ne, Nf, Ng, Nx: Random numbers; O: Out station and Server of DNP3 communication network (PLC); O_AIK_Pri, O_AIK_Pub: Identity certificate key pair for the server of DNP3 communication network; PcrO, PcrM: The trust metric root for TPM; R: A random bit string (128 bit); Req(FC): The key update request and FC is the Function Code; S1 = H (R), S2 = H (S1), ..., Sn=H(Sn-1): A hash chain label generated from a single random value (R); Scalc: The variable used to find the next hash tag for the hash chain; Si: The current hash tag; Sstored, Nstored, Sreceived, Nreceived, Scalc: Store the values of Sn, *n*, Si, *i* (where $1 < i < n$) and Scalc; tagA: The MAC tag

Acknowledgements

The authors would like to thank the reviewers for their thorough reviews and helpful suggestions.

Funding

The authors acknowledge the National Natural Science Foundation of China (Grant: 61462060), the National Natural Science Foundation of China (Grant: 61762060), and Gansu Science and Technology Plan Youth Science and Technology Fund Project (Grant 1610RJYA008).

Authors' contributions

LY is the main writer of this paper. He proposed the main idea, proposed the secure protocol of DNP3-SA for broadcast mode, completed the simulation, and analysed the performance. FT introduced the Hash chain algorithm in Key update and Broadcast Authenticate Sub protocol. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Author details

[1]College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou, China. [2]Key Laboratory of Gansu Advanced Control for Industrial Processes, Lanzhou University of Technology, Lanzhou, China. [3]National Demonstration Center for Experimental Electrical and Control Engineering Education, Lanzhou University of Technology, Lanzhou, China. [4]School of Computer and Communication, Lanzhou University of Technology, Lanzhou, China.

Received: 30 January 2018 Accepted: 25 April 2018
Published online: 08 May 2018

References

1. AA Mohsen, F Adil, T Zahir, et al., An efficient data-driven clustering technique to detect attacks in SCADA systems. IEEE Trans. Inf. Forensics Secur. **11**(5), 893–906 (2016)
2. Y Yang, H-Q Xu, L Gao, et al., Multidimensional intrusion detection system for IEC 61850-based SCADA networks. IEEE Trans. Power Delivery **32**(2), 1068–1078 (2017)
3. K Kazukuni, Cyber physical security for industrial control systems and IoT. IEICE Trans. Inf. Syst. **E99D**(4), 787–795 (2016)
4. A Raphael, C Seyit, F Ernest, Formal modelling and analysis of DNP3 secure authentication. J. Netw. Comput. Appl. **59**, 345–360 (2016)
5. N Farhad, M Todd, H Simon, et al., Critical infrastructure protection security layer for DNP3 devices. Int. J. Manuf. Res. **7**(1), 72–85 (2012)
6. A Raphael, C Seyit, F Ernest, Securing DNP3 broadcast communications in SCADA systems. IEEE Trans. Ind. Inf **12**(4), 1474–1485 (2016)
7. B Vaidya, D Makrakis, M Hussein, Authentication and authorization mechanisms for substation automation in smart grid network. IEEE Netw. **27**(1), 5–11 (2013)
8. M Damiano, M Massimo, A semantic analysis of key management protocols for wireless sensor networks. Sci. Comput. Program. **81**, 53–78 (2014)
9. R Guo, W Qiaoyan, J Zhengping, et al., An efficient and provably-secure broadcast authentication scheme in wireless sensor networks. J. Int. Technol. **16**(6), 977–985 (2015)
10. Y Lin, J Deng, J Wang, et al., A-CACHE: an anchor-based public key caching scheme in large wireless networks. Comput. Netw. **87**, 78–88 (2015)
11. N Surya, J Zic, L Dongxi, et al., A mobile and portable trusted computing platform. EURASIP J. Wirel. Commun. Netw. **2011**(1), 1–19 (2011)
12. H Tan, W Hu, J Sanjay, A remote attestation protocol with Trusted Platform Modules (TPMs) in wireless sensor networks. Sec. Commun. Netw. **8**(13), 2171–2188 (2015)
13. K Gao, Z Wang, A Ningyu, et al., Construction of the immune system of cyber security for electric power supervise and control system based on trusted computing. Gongcheng Kexue Yu Jishu/Adv. Eng. Sci. **49**(2), 28–35 (2017)
14. CJ Adam, B Sergey, Bolt-on security extensions for industrial control system protocols: a case study of DNP3 SAv5. IEEE Secur. Priv. **13**(3), 74–79 (2015)