Cryptography Security Designs and Enhancements of DNP3-SA Protocol Based on Trusted Computing

Ye Lu[1] and Tao Feng[2]
(Corresponding author: Ye Lu)
College of Electrical and Information Engineering, Lanzhou University of Technology[1]
Lanzhou 730050, China
School of Computer and Communication, Lanzhou University of Technology[2]
(Email: luye528@126.com)
(Received Sept. 1, 2017; revised and accepted Dec. 28, 2017)

Abstract

Although there are several solutions utilized to prevent security threats in DNP3 networks, existing DNP3-SA networks still have severe shortcomings. To solve this security problem, the attack vector and security requirements of DNP3-SA protocol are analyzed, then, a cryptography security designs and enhancements of DNP3-SA protocol is proposed based on the Trusted Computing, which authenticate the identity and security status of the client and server to prevent node sensitive information from being compromised. The new protocol overcomes man-in-the-middle and replay attacks without increasing communication overhead. The protocol is verified by the SPAN tool, and no intrusion path is found, which ensures the integrity, authenticity, freshness and confidentiality of the nodes participating in the communication.

Keywords: DNP3-SA Protocol; Industrial Control System; SPAN; Trusted Computing

1 Introduction

Although the industrial Ethernet protocol based on TCP/IP technology is widely used in SCADA system, the original industrial Ethernet protocol is facing more and more threat of network attacks. The widespread use of the DNP3 protocol in the field of SCADA systems has proven unsafe [5–7]. Therefore, the DNP3 protocol must be researched and improved from the perspective of the communication side to ensure the communication security of the ICS system.

The latest security improvement version DNP3-SA [2] proposes a certification strategy to ensure the integrity of the message. However, literature 5 indicates that the protocol still cannot resist replay attacks. Literature [3] proposes two improvement schemes, the first scheme is verified by SPAN, there is still a replay attack, and the second scheme cannot meet the specification of the original DNP3-SA protocol. Literature [10] proposed ECC-based public key authentication scheme based on the third party trusted institutions to ensure the legitimacy of the client, but cannot guarantee the authenticity of the server identity. It is necessary to use the trusted platform to ensure the authenticity of both sides of the communication, against the attacker posing and tamper with DNP3 server and client. Literature [4] 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 [8] into ICS and proposes a remote secure communication based on the trusted platform module (TPM) built-in key [9]. Literature [1] proposed the use of trusted platform to protect and evaluate the terminal equipment data security and trusted state, but the above studies are not given industrial Ethernet protocol security reinforcement of specific programs. There are no other public research to introduce trusted components into the DNP3 protocol to ensure the safety of field devices.

The main contributions of the paper are as follows: Firstly, the four kinds of attack vectors under the dnp3 protocol are given. Secondly, the DNP3 protocol is introduced into the trusted platform for the first time, and the authentication of the device identity is realized. Finally, the key update and communication sub-protocol was redesigned to resist replay attacks.

The rest of our paper is organized as follows. In Section 2, The attack vector and security requirements of DNP3-SA protocol are analyzed. Subsequently, we propose our enhancements of DNP3-SA protocol based on the Trusted Computing in Section 3 and analyze the security with the span tool in Section 4. In Section 5, The performance of our protocol is analyzed. At last, Section 6

presents the overall conclusion.

2 The Attack Vector and Security Requirements

In this section, we takes the SCADA system as an example to study the security threats faced by the DNP3-SA communication protocol. SCADA system consists of monitoring stations (MS), Human machine interface(HMI) and other equipment such as PLC, IED. DNP3-SA protocol using C / S mode of communication, MS, HMI as DNP3 client communicate with DNP3 field server PLC through the configuration software (CS). PLC program collect the scene data back to the MS and HMI. The DNP3 protocol communication threat model is illustrated in Figure 1.

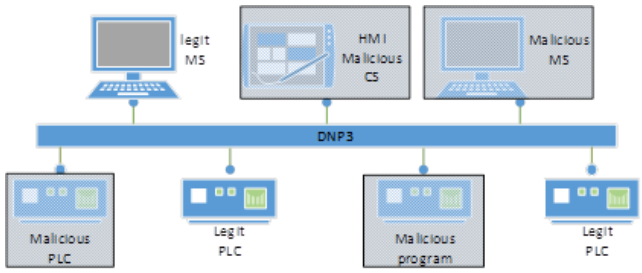

Figure 1: DNP3 Communication threat model

MS as a client to communicate with multiple PLC servers. The gray 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 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 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 [2, 3, 5] based on the authentication of both parties can prevent such attacks.

2) Attack vectors based on CS vulnerabilities:
An attacker can exploit a CS to obtain native sensitive information and send a malicious command to the PLC. If the attacker steals the preset key through the controlled CS, the DNP3-SA security improvement protocol [2, 3, 5] 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, causing malfunction. Since the impersonator cannot obtain the session key, the DNP3-SA security improvement protocol [2, 3, 5] based on the authentication of both parties can prevent such attacks.

4) Attack vector based on PLC program:
As the PLC is usually used weak password protection mechanism, an attacker can crack the password and other ways to implant malicious program, in order to obtain sensitive information or cause failure. If the attacker steals the preset key through the controlled PLC, the DNP3-SA security improvement protocol based on the authentication of both parties [2, 3, 5] will not prevent such attacks.

By the above attack vector and the literature [5–7], DNP3-SA protocol mainly exists the following attack types: eavesdropping, tampering, posing, DOS, replay. Therefore, integrity, confidentiality, authenticity and freshness are the security requirements of trusted DNP3 protocol design.

3 The Proposed Scheme

In this section, we propose a cryptography security designs and enhancements of DNP3-SA protocol based on the Trusted Computing which can remedy a range of network attacks. It is composed four sub-protocols: Identity authentication sub-protocol; key agreement sub-protocol; key update sub-protocol and communication sub-protocol. The authentication sub-protocol provides periodic verification and updating of the identity and security status information for the MS and PLC by configuring the trusted platform and increasing the authentication server (AS). 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 data security, and the AS solves the trustworthiness of the device state by periodically querying the PCR of the MS and PLC. The communication sub-protocol improves the NACR mode (non-critical request) and AGM mode (critical request) in the original DNP3-SA protocol and the scheme of literature [3] to protect against replay attacks. It should be noted that, for the first time, this paper introduces the trusted platform into DNP3-SA protocol to complete the device authentication.

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

1) The communication request is initiated by the MS.

2) The base layer of and protocol and AS are reliable.

3) 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.

4) AS knows the expected trusted information of all terminal devices in the SCADA system, that is, a trusted list.

5) MS, PLC known AS's identity authentication public key A_AIK_Pub and Bind-public key KA_Pub.

3.1 Identity Authentication and Key Negotiation Sub-Protocol

Trusted Computing [9] Measure 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 in a non-tamperable manner for user authentication to ensure that the device hardware and software system behaves 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.

The function of the identity authentication sub-protocol is to authenticate each other and prevent the device from being hijacked before the MS and the PLC communicate with each other. Figure 2 is the identity authentication sub-protocol and key agreement sub-protocol message flow, M is the client (MS), O is the server (PLC). O_AIK_Pri, O_AIK_Pub, M_AIK_Pri, M_AIK_Pub are authentication key pair (AIK), KA_Pr, KA_Pub, KM_Pri, KM_Pub, KO_Pri, KO_Pub are Bind-key pairs. PcrO, and PcrM is the trust metric root. K_H is used for HMAC calculations in communication sub-protocols to ensure the integrity of communication data; K_E is used for symmetric encryption of critical data required for high-security communications. Random numbers Na, Nb, Nc, Nd, Ne, Nf, Ng ensure the freshness of the message.

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 agreement process of the message authentication key K_H and the message encryption key K_E after confirming the identity information between M and O with the assistance of A.

AS through the periodic question of MS and PLC PCR, by comparing the white list information to find whether there is unexpected changes in equipment status, so as to ensure that the SCADA system terminal equipment in the course of the operation has not been tampered with. 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 white list 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

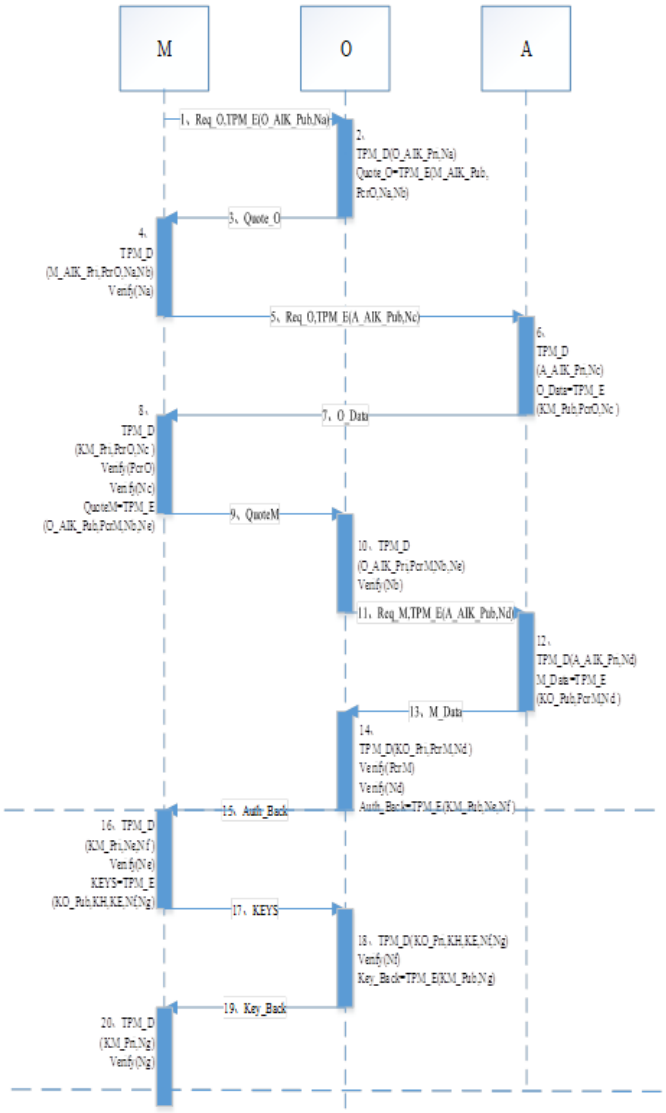

Figure 2: Authentication and key agreement

the symmetric keys K_H and K_E negotiated in the pre-agreement 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 Communication Sub-Protocol

Figure 3 depicts our key update and communication sub-protocol. Steps 21 to 29 refers to the key update sub-protocol, Steps 30 to 41 describe the communication sub-protocol. Our Scheme encrypts the Ksn to ensure the confidentiality of the serial number, preventing replay attacks, use the random number Nx, Ny to ensure the freshness of the message, and use the message authentication code MAC to verify the integrity of the message.

Literature [3] indicates that there is a replay attack on the DNP3-SA protocol, and the attacker can replay the response message of the server (O) to cheat the client (M) to generate a valid MAC tag (old message), and then execute the command on the server. This attack is fatal to critical infrastructures and suggests two solutions to improve this flaw. Solution 1 calculate MAC on the challenge message and solution 2 implement the Ksn as the sole component of the AGM operation. But, the two solutions are verified by the span tool, the result show that there are still replay attacks in both approaches. This is because the Ksn is plain text, the attacker is still able to guess the next Ksn, and then launch replay attacks. In addition, the server can not obtain the current Ksn serial number in solution 2, causing the protocol to fail.

4 Security Verification

Our scheme is described using the role-based formalized protocol language HLPSL, and the SPAN tool is used to verify the security of the protocol. The SPAN tool simulates the protocol functions and intruder behavior 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. 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 article uses "master" to represent MS, the entity M in Figure 4, use "out" to represent PLC, the entity O in Figure 4, use "server" to represent AS, the entity A in Figure 4. Limited to space, Figure 4 depicts the communication process for the master role.

Figure 5 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

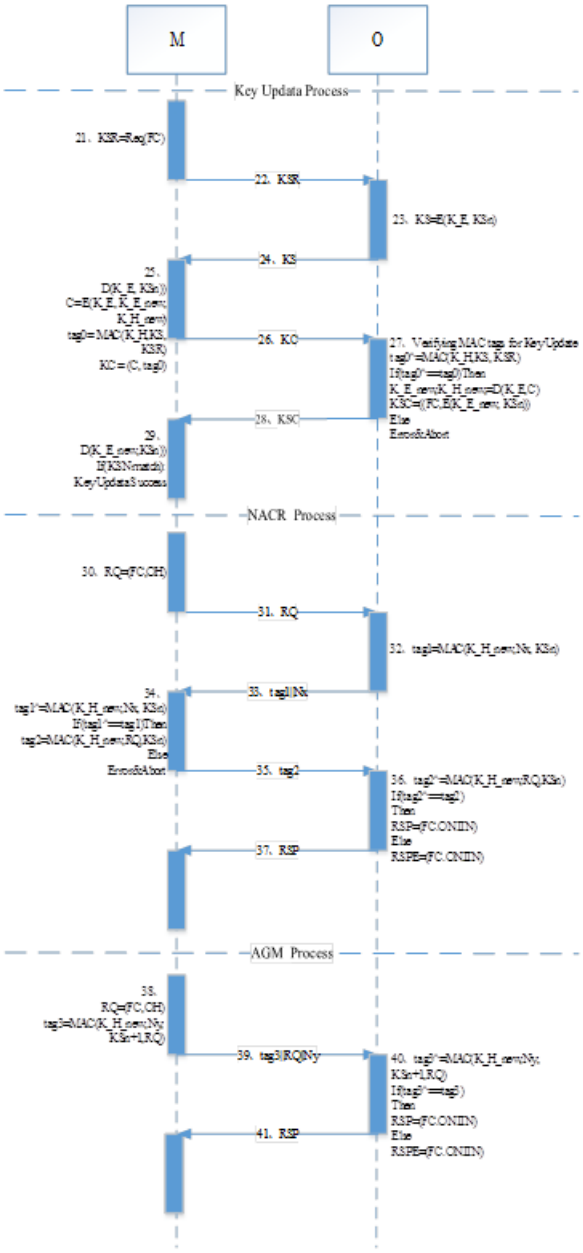

Figure 3: Key update and communication

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
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
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 4: Communication process of entity M

authentication sub-protocol and the key-agreement 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.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
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
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 5: Attacker knowledge and security goals

As shown in Figure 6, the SPAN authentication result of the authentication and key agreement sub-protocol is security (SAFE). the message sequence of the protocol given by SPAN analyzes 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.

The security target and authentication process of the key update sub-protocol and communication sub-protocol

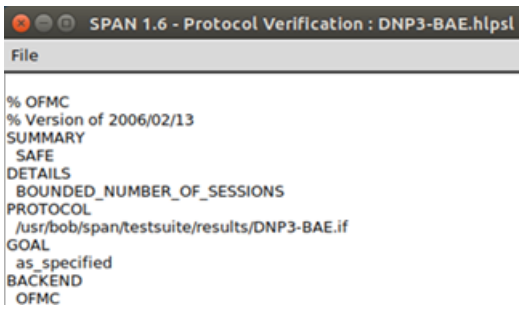

Figure 6: SPAN verification results

are similar to the authentication sub-protocol. The SPAN verification result is also safe and will not be repeated. In summary, our scheme can meet the safety requirements of the protocol proposed in Section 2, which guarantee the identity and status of the communication entity, the integrity of the protocol data, the freshness of the random number, the confidentiality of the protocol data under the high security level, and can withstand the replay attacks that still exist in literature [3].

5 Performance Analysis

In this section, we provide the overhead analysis of the fixed protocol to show that our approaches indeed maintain communication, processing and storage overheads at the cost of a minor increased cost in calculate. Since the authentication and key agreement sub-protocol is initiated only when the device status information is changed (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 have less impact on the time overhead of the communication. The Key update and communication sub-protocol is used frequently, which uses the encryption primitives in the DNP3-SA specification without the TPM-related time overhead, but adds some communication, computation and storage overhead.

This section presents a comparison of communication, Calculate and storage overheads between the standard DNP3-SA, solutions of [3] and our scheme. It is to be noted that the comparison is based on the total counts of messages(n) within the DNP3-SA protocol but not in byte size. This is because most of the messages have similar byte sizes. Here n is used to denote the approximate number of commands to be exchanged between a master station and an outstation per user and during a time interval between two key updates. Sol1 and Sol2 denote the proposed solution 1 and solution 2 in literature [3]. M represent the modification attack, R represent the replay attack, S represent the spoofing attack and H refers to hijacking attack.

To understand Table 1, one must take three things into consideration for the communication and storage over-

Table 1: Performance analysis and comparison(AGM)

Scheme	Communication	Calculate overhead		Storage		M	R	S	H
		MS	PLC	MS	PLC				
Ours	2(n+4)≈0(n)	n+2≈0(n)	n+2≈0(n)	5	5	√	√	√	√
DNP3-SA	2(n+4)≈0(n)	n+1≈0(n)	n+1≈0(n)	4	4	√	×	×	×
[3]-Sol1	2(n+4)≈0(n)	n+2≈0(n)	n+2≈0(n)	4	4	√	×	×	×
[3]-Sol2	2(n+4)≈0(n)	n+1≈0(n)	n+1≈0(n)	4	4	√	×	×	×

heads. First, there is a key update process that occurs before the NACR or AGM operates. The key update process has 4 headcounts of messages per round and per user (refer to Figure 3). Second, before AGM can successfully operate, there must be at least a run of the NACR operation, which implies the number of communicated messages in NACR will also be considered in AGM. Third, we consider the performance overhead of critical information transmission mode (AGM).

In Table 1, Ours row, the total messages involved the operation is shown as $(2(n+4) \approx O(n))$. This value $(2(n+4) \approx O(n))$ is derived because there are 4 messages from the key update process, 4 messages from the NACR operation and two $2n$ messages from the AGM operation (i.e. NACR must run before AGM) for n commands. The calculate overhead in ours is $n+2$, similar to [3] - Sol1, meaning that 2 MAC computations in NACR operation and one MAC computation in each AGM operation (n). Asymptotically, this value corresponds to $O(n)$. For storage overhead in ours, 5 values are expected to be stored on both stations. This is because both store value like the keys (KH and KE), 2 MACs, content of the challenge message.

In comparing our scheme to proposed solutions 1 and 2 ([3] - Sol1 and [3] - Sol2), our new protocol overcomes the shortcomings of the two proposed solutions of [3], which have man-in-the-middle attacks and replay attacks at a minor increase in calculate overhead and storage overhead, without increasing communication overhead.

6 Conclusion

In this article, we proposed a cryptography security designs and enhancements of DNP3-SA protocol based on the Trusted Computing without increasing communication overhead, which authenticate the identity and security status of the DNP3-SA client and server to prevent node sensitive information from being compromised. The protocol is verified by the SPAN tool, and no intrusion path is found, which ensures the integrity, authenticity, freshness and confidentiality of the nodes.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61462060, No.61762060), The authors would like to thank the anonymous reviewers for their helpful comments and suggestions.

References

[1] R. Amin, S. H. Islam, A. Karati, *et al.*, "Design of an enhanced authentication protocol and its verification using AVISPA," in *IEEE International Conference on Recent Advances in Information Technology*, 2016.

[2] R. Amoah, S. Camtepe, E. Foo, "Securing DNP3 broadcast communications in SCADA systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 4, pp. 1474–1485, 2016.

[3] R. Amoah, S. Camtepe, E. Foo, "Formal modelling and analysis of DNP3 secure authentication," *Journal of Network and Computer Applications*, vol. 59, pp. 345–360, 2016.

[4] F. Brasser, B. E. Mahjoub, A. R. Sadeghi, *et al.*, "TyTAN: Tiny trust anchor for tiny devices," in *IEEE Design Automation Conference*, pp. 34, 2015.

[5] J. A. Crain, S. Bratus, "Bolt-on security extensions for industrial control system protocols: A case study of DNP3 SAv5," *IEEE Security & Privacy*, vol. 13, no. 3, pp. 74–79, 2015.

[6] C. Cremers, M. Dehnel-Wild, K. Milner, "Secure authentication in the grid: A formal analysis of DNP3: SAv5," in *European Symposium on Research in Computer Security*, pp. 389–407, 2017.

[7] D. Lee, H. Kim, K. Kim, *et al.*, "Simulated attack on DNP3 protocol in SCADA system," in *The 31th Symposium on Cryptography and Information Security*, 2014.

[8] P. Maene, J. Gotzfried, R. D. Clercq, *et al.*, "Hardware-based trusted computing architectures for isolation and attestation," *IEEE Transactions on Computers*, PP(99): 1-1, 2017.

[9] Trusted Computing Group, *TCG Trusted Network Communications: IF-MAP Metadata for ICS Security, Specification Version 1.0 Revision 46*, 15 Sept. 2014. (`https://trustedcomputinggroup.org/wp-content/uploads/IFMAP\_Metadata\_For\_ICS\_Security\_v1\_0r46.pdf`)

[10] B. Vaidya, D. Makrakis, H. T. Mouftah, "Authentication and authorization mechanisms for substation automation in smart grid network," *IEEE Network*, vol. 27, no. 1, pp. 5–11, 2013.

Biography

Lu Ye was born in 1986, CCF member. He is a doctoral student at LanZhou University of Technology, His research interests include security of industrial control system.

FENG Tao was born in 1970, researcher/PhD supervisor, CCF senior member, IEEE and ACM member. He graduated from Xidian University, and then worked at school of computer and communication in Lanzhou University of Technology. His research interests include Cryptography and information security.