Efficient Key Management Scheme for SCADA System

Jian Shen, Jin Wang, Yongjun Ren, Jianwei Zhang and Shunfeng Wang

Abstract Currently Supervisory Control And Data Acquisition (SCADA) system intends to be connected to the open operating environment. Thus, protecting SCADA systems from malicious attacks is getting more and more attention. A key management scheme is essential for secure SCADA communications. In this paper, we propose an efficient key management scheme for SCADA systems with good security properties and performance.

Keywords Supervisory control and data acquisition (SCADA) · Key management · Secure SCADA communications

Introduction

In order to deliver critical services, such as water, sewerage and electricity distribution, nations are increasingly depends on Supervisory Control And Data Acquisition (SCADA) systems. As the change of the operating environment in SCADA system from close to open, the risk of SCADA incidents occurring is increasing. Nowadays, SCADA system has been exposed to a wide range of network security problems. If the SCADA system is damaged from the attacks,

J. Shen (✉) · J. Wang · Y. Ren
School of Computer and Software, Jiangsu Engineering Center of Network Monitoring,
Nanjing University of Information Science and Technology, Nanjing 210044, China
e-mail: s_shenjian@126.com

J. Zhang
School of Mathematics and Statistics, Nanjing University of Information Science
and Technology, Nanjing 210044, China

S. Wang
College of Bin Jiang, Nanjing University of Information Science and Technology,
Nanjing 210044, China

Y.-M. Huang et al. (eds.), *Advanced Technologies, Embedded and Multimedia for Human-centric Computing*, Lecture Notes in Electrical Engineering 260,
DOI: 10.1007/978-94-007-7262-5_70, © Springer Science+Business Media Dordrecht 2014

this system can have a widespread negative effect to society. One critical security requirement for SCADA systems is that communication channels need to be secured. Secure keys need to be established before cryptographic techniques can be used to secure communications.

Note that un-encrypted data communication via networks is vulnerable to several types of attacks. Therefore, secure data communication between each device is required to secure the SCADA system. Secure key management is essential for data encryption. In this paper, we focus on the key management scheme for SCADA systems and propose an efficient key management scheme (EKMS) with good security properties. Compared with the previous schemes, the presented key management scheme is more efficient in terms of the communication cost. Our scheme is based on a symmetric balanced incomplete block design (SBIBD), which can provide the authentication service and resist different key attacks. The structure of SBIBD makes the computation of a common conference key for each remote terminal unit (RTU) quite convenient.

The rest of this paper is organized as follows: In the following section, related work is briefly introduced. The proposed key management scheme is described in detail in section Efficient Key Management Scheme for SCADA System. Security analysis and performance analysis of our scheme are presented in section Security Analysis and Performance Analysis. Finally, the conclusions of this paper are covered in section Conclusions.

Related Work

A SCADA system consists of three types of equipment communicating with each other: (1) human–machine interface (HMI) that operators interact with; (2) master terminal unit (MTU) that provides supervisory control of an RTU; and (3) the remote terminal unit (RTU) that interacts with the physical environment. In this paper, the term node will be used to refer to any entity in the system. The structure of SCADA systems is based on master–slave structure, which is shown in Fig. 1. The structure of a SCADA system will normally include one central MTU, which communicates with a hierarchy of other nodes, including Sub-MTU and RTUs. Master stations and sub-master stations, are computers with resources at least as plentiful as a modern desktop computer.

Recently, SKE [1] was proposed by Sandia, where the MTU has to encrypt data with each key of the RTUs individually to broadcast a message. After that, SKMA [2] was proposed, where two types of keys must be managed by an MTU or RTUs. The long-term node-key distribution center (KDC) key is shared between the KDC and a node. The other key is the long-term node–node key shared between two nodes. Later, ASKMA [3] proposed a key-management scheme suitable for secure SCADA communication using a logical key hierarchy to support broadcast communication and multicast communication, but it may be less efficient.

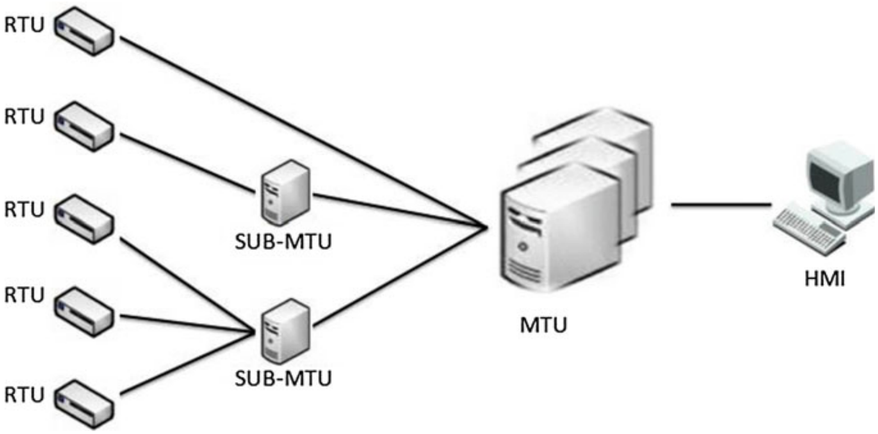


Fig. 1 SCADA system architecture

Due to the constrains of low-rate data transmission and real-time processing in different operational environment, satisfying the security requirements of confidentiality, integrity and availability in a SCADA system is really a challenging issue. In this paper, resort to a symmetric balanced incomplete block design (SBIBD), we design a novel key management scheme for SCADA systems with good security properties and performance.

Efficient Key Management Scheme for SCADA System

In this section, we propose an efficient key management scheme for SCADA system. By our scheme, the communication among Sub-MTUs can be secure and efficient, so can the communication among RTUs. The process of key management among RTUs is described as follows. Note that the process of key management among Sub-MTUs is similar to that of RTUs.

Each RTU registers to the Sub-MTU and gets their private key. After that, every RTU can process the key agreement to compute the common conference key. First of all, the Sub-MTU chooses two prime order group G_1 and G_2 and a modified Weil pairing map $\widehat{e}$ defined in [4]. Next, the Sub-MTU selects two one-way hash functions $H : \{0,1\}^* \to G_1$ and $h : \{0,1\}^* \to Z_q^*$ where H maps its arbitrary length to a nonzero point of G_1 while h maps its input with arbitrary length to a nonzero integer. At last, the Sub-MTU picks a random integer $s \in Z_q^*$ as its private key, computes its public key $P_{pub} = s\mathcal{G}$, and publishes $\left(p, q, G_1, G_2, \mathcal{G}, \widehat{e}, P_{pub}, H, h\right)$, but keeps s secret. Each RTU U_i's identity is $ID_i \in (0,1)^*$. The Sub-MTU computes U_i's public key $Q_i = H(ID_i)$ and then U_i's private key $S_i = sQ_i$ which is issued to U_i via a secure channel.

Fig. 2 (7 × 7) incidence matrix corresponding to the (7, 4, 2)-design

$$\mathcal{L} = (\ell_{ij}) = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

The common conference key among RTUs is calculated by employing SBIBD, where the number of blocks is the same as that of participants. We choose a (7, 4, 2)-design. Let a finite set X $= \{1, 2, 3, 4, 5, 6, 7\}$, then $B_1 = \{1, 2, 4, 7\}$, $B_2 = \{1, 2, 3, 5\}$, $B_3 = \{2, 3, 4, 6\}$, $B_4 = \{3, 4, 5, 7\}$, $B_5 = \{1, 4, 5, 6\}$, $B_6 = \{2, 5, 6, 7\}$, $B_7 = \{1, 3, 6, 7\}$. Accordingly, a (7 × 7) incidence matrix L is depicted in Fig. 2. The rows and columns of the matrix correspond to the blocks and the elements, respectively. The entry l_{ij} in the ith row and the jth column of L is a 1 if the block i contains the element j and is a 0 otherwise.

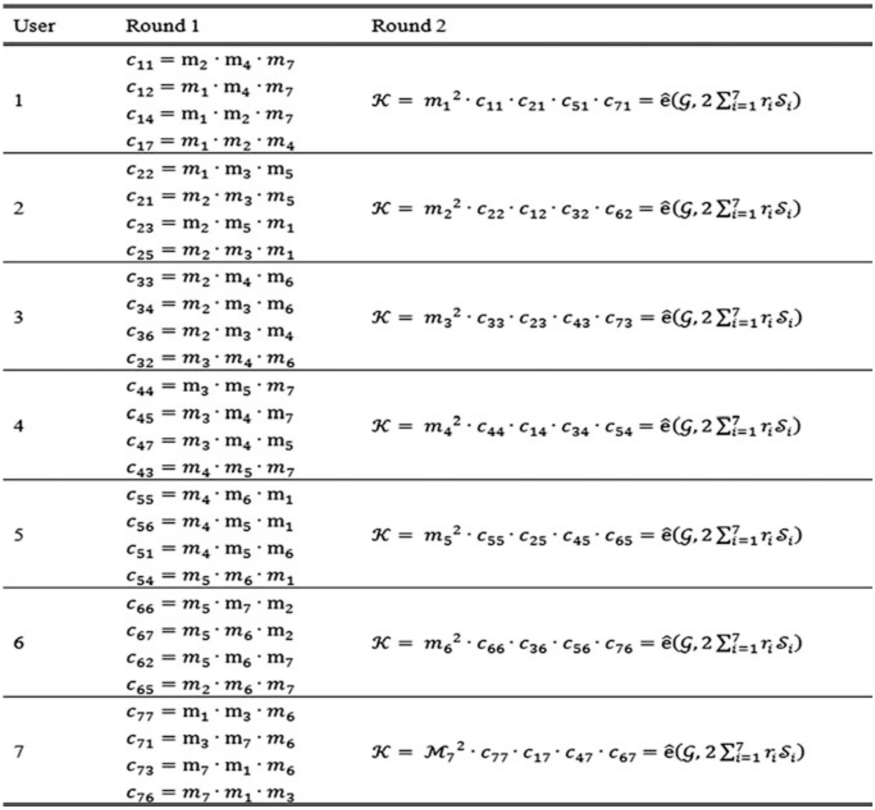
For computing the common conference key among RTUs, two rounds are required in our scheme.

1. Each RTU U_i selects a random number r_i as secret key by itself for every session and then calculates $m_i = \widehat{e}(\mathcal{G}, r_i S_i)$. Simultaneously, U_i calculates $T_i = r_i Q_i$. Let $D_i = \{m_i, T_i\}$. RTU i receives message D_j from RTU j in case $l_{ij} = 1$ and $j \neq i$, namely $j \in B_i - \{i\}$. m_i is used for generating conference key while T_i is used for authentication. We now describe the key agreement process from the viewpoint of RTU 1. U_1 receives D_2, D_4, D_7 from U_2, U_4, U_7 and then makes

$$\begin{aligned} c_{11} &= m_2 \cdot m_4 \cdot m_7 = \widehat{e}(\mathcal{G}, r_2 S_2 + r_4 S_4 + r_7 S_7), \\ c_{12} &= m_1 \cdot m_4 \cdot m_7 = \widehat{e}(\mathcal{G}, r_1 S_1 + r_4 S_4 + r_7 S_7), \\ c_{14} &= m_1 \cdot m_2 \cdot m_7 = \widehat{e}(\mathcal{G}, r_1 S_1 + r_2 S_2 + r_7 S_7), \\ c_{17} &= m_1 \cdot m_2 \cdot m_4 = \widehat{e}(\mathcal{G}, r_1 S_1 + r_2 S_2 + r_4 S_4), \\ W_{12} &= T_1 + T_4 + T_7, \\ W_{14} &= T_1 + T_2 + T_7, \\ W_{17} &= T_1 + T_2 + T_4, \end{aligned}$$

where $c_{ij} = \prod_{x \in B_i - \{j\}} m_x$ and $W_{ij} = \sum_{x \in B_i - \{j\} \text{ and } j \neq i} T_x$. In the viewpoint of RTU 1, we have that $c_{1j} = \prod_{x \in B_1 - \{j\}} m_x$ and $W_{1j} = \sum_{x \in B_1 - \{j\} \text{ and } j \neq 1} T_x$. Simultaneously, other RTUs do the same process.

2. Let $E_{ji} = \{c_{ji}, W_{ji}\}$. RTU i receives E_{ji} from RTU j in case $l_{ji} = 1, j \neq i$. Here, similar to that in round 1, c_{ji} is used for generating conference key while W_{ji} is used for authentication. Particularly, U_1 receives E_{j1} from RTU j, if $l_{j1} = 1$,



User	Round 1	Round 2
1	$c_{11} = \mathrm{m}_2 \cdot \mathrm{m}_4 \cdot m_7$ $c_{12} = m_1 \cdot \mathrm{m}_4 \cdot m_7$ $c_{14} = \mathrm{m}_1 \cdot \mathrm{m}_2 \cdot m_7$ $c_{17} = m_1 \cdot m_2 \cdot m_4$	$\mathcal{K} = m_1{}^2 \cdot c_{11} \cdot c_{21} \cdot c_{51} \cdot c_{71} = \hat{e}(\mathcal{G}, 2\sum_{i=1}^{7} r_i \mathcal{S}_i)$
2	$c_{22} = m_1 \cdot \mathrm{m}_3 \cdot \mathrm{m}_5$ $c_{21} = m_2 \cdot m_3 \cdot m_5$ $c_{23} = \mathrm{m}_2 \cdot \mathrm{m}_5 \cdot m_1$ $c_{25} = m_2 \cdot m_3 \cdot m_1$	$\mathcal{K} = m_2{}^2 \cdot c_{22} \cdot c_{12} \cdot c_{32} \cdot c_{62} = \hat{e}(\mathcal{G}, 2\sum_{i=1}^{7} r_i \mathcal{S}_i)$
3	$c_{33} = m_2 \cdot \mathrm{m}_4 \cdot \mathrm{m}_6$ $c_{34} = m_2 \cdot \mathrm{m}_3 \cdot \mathrm{m}_6$ $c_{36} = m_2 \cdot \mathrm{m}_3 \cdot \mathrm{m}_4$ $c_{32} = m_3 \cdot m_4 \cdot m_6$	$\mathcal{K} = m_3{}^2 \cdot c_{33} \cdot c_{23} \cdot c_{43} \cdot c_{73} = \hat{e}(\mathcal{G}, 2\sum_{i=1}^{7} r_i \mathcal{S}_i)$
4	$c_{44} = \mathrm{m}_3 \cdot \mathrm{m}_5 \cdot m_7$ $c_{45} = m_3 \cdot \mathrm{m}_4 \cdot \mathrm{m}_7$ $c_{47} = m_3 \cdot \mathrm{m}_4 \cdot \mathrm{m}_5$ $c_{43} = m_4 \cdot m_5 \cdot m_7$	$\mathcal{K} = m_4{}^2 \cdot c_{44} \cdot c_{14} \cdot c_{34} \cdot c_{54} = \hat{e}(\mathcal{G}, 2\sum_{i=1}^{7} r_i \mathcal{S}_i)$
5	$c_{55} = m_4 \cdot \mathrm{m}_6 \cdot \mathrm{m}_1$ $c_{56} = m_4 \cdot \mathrm{m}_5 \cdot \mathrm{m}_1$ $c_{51} = m_4 \cdot \mathrm{m}_5 \cdot \mathrm{m}_6$ $c_{54} = m_5 \cdot m_6 \cdot m_1$	$\mathcal{K} = m_5{}^2 \cdot c_{55} \cdot c_{25} \cdot c_{45} \cdot c_{65} = \hat{e}(\mathcal{G}, 2\sum_{i=1}^{7} r_i \mathcal{S}_i)$
6	$c_{66} = m_5 \cdot \mathrm{m}_7 \cdot \mathrm{m}_2$ $c_{67} = m_5 \cdot m_6 \cdot \mathrm{m}_2$ $c_{62} = m_5 \cdot \mathrm{m}_6 \cdot \mathrm{m}_7$ $c_{65} = m_2 \cdot m_6 \cdot m_7$	$\mathcal{K} = m_6{}^2 \cdot c_{66} \cdot c_{36} \cdot c_{56} \cdot c_{76} = \hat{e}(\mathcal{G}, 2\sum_{i=1}^{7} r_i \mathcal{S}_i)$
7	$c_{77} = \mathrm{m}_1 \cdot \mathrm{m}_3 \cdot m_6$ $c_{71} = \mathrm{m}_3 \cdot \mathrm{m}_7 \cdot m_6$ $c_{73} = \mathrm{m}_7 \cdot \mathrm{m}_1 \cdot m_6$ $c_{76} = m_7 \cdot m_1 \cdot m_3$	$\mathcal{K} = \mathcal{M}_7{}^2 \cdot c_{77} \cdot c_{17} \cdot c_{47} \cdot c_{67} = \hat{e}(\mathcal{G}, 2\sum_{i=1}^{7} r_i \mathcal{S}_i)$

Fig. 3 Generating a common key

$j \neq 1$. Therefore, U_1 receives E_{21}, E_{51}, E_{71} from U_2, U_5, U_7 and derives c_{21}, c_{51}, c_{71}. Then the common conference key K is calculated as $K = m_1 \times c_{11} \times c_{21} \times c_{51} \times c_{71} = \widehat{e}\left(\mathcal{G}, 2\sum_{i=1}^{7} r_i S_i\right)$, where $c_{21} = \widehat{e}(\mathcal{G}, r_2 S_2 + r_3 S_3 + r_5 S_5)$, $c_{51} = \widehat{e}(\mathcal{G}, r_4 S_4 + r_5 S_5 + r_6 S_6)$, and $c_{71} = \widehat{e}(\mathcal{G}, r_3 S_3 + r_7 S_7 + r_6 S_6)$.

Then, following our scheme, the process for calculating the common conference key among all the RTUs is shown in Fig. 3.

In our scheme, we take advantage of RTUs' identity information for authentication.

1. Let $D_i = \{m_i, T_i\}$, RTU i receives D_j from RTU j in case $l_{ij} = 1$ and $j \neq i$. We now describe the authentication process from the viewpoint of RTU 1. U_1 receives D_2, D_4, D_7 from U_2, U_4, U_7 and makes

$$\begin{aligned}\widehat{e}\left(P_{pub}, T_2\right) &= \widehat{e}(s\mathcal{G}, r_2 Q_2) = \widehat{e}(\mathcal{G}, r_2 s Q_2) = m_2,\\ \widehat{e}\left(P_{pub}, T_4\right) &= \widehat{e}(s\mathcal{G}, r_4 Q_4) = \widehat{e}(\mathcal{G}, r_4 s Q_4) = m_4,\\ \widehat{e}\left(P_{pub}, T_7\right) &= \widehat{e}(s\mathcal{G}, r_7 Q_7) = \widehat{e}(\mathcal{G}, r_7 s Q_7) = m_7,\end{aligned}$$

Hence, U_1 can authenticate the entity of U_2, U_4, U_7 only if $\widehat{e}\left(P_{pub}, T_2\right) = m_2$, $\widehat{e}\left(P_{pub}, T_4\right) = m_4$, and $\widehat{e}\left(P_{pub}, T_7\right) = m_7$, respectively. Generally speaking, if $\widehat{e}\left(P_{pub}, T_i\right) = m_i$, then U_j can authenticate counterpart's entity.

2. Let $E_{ji} = \{c_{ji}, W_{ji}\}$ and $W_{ji} = \sum_{x \in B_j - \{i\} \,\text{and}\, j \neq i} T_x$. RTU i receives E_{ji} from RTU j in case $l_{ji} = 1$, $j \neq i$. Particularly, in the viewpoint of RTU 1, $W_{j1} = \sum_{x \in B_j - \{1\} \,\text{and}\, j \neq 1} T_x$ and $E_{j1} = \{c_{j1}, W_{j1}\}$. U_1 receives E_{21}, E_{51}, E_{71} from U_2, U_5, U_7, then derives W_{21}, W_{51}, W_{71} and calculates

$$\begin{aligned}\widehat{e}(P_{pub}, W_{21}) &= \widehat{e}(s\mathcal{G}, T_2 + T_3 + T_5) = m_2 \cdot m_3 \cdot m_5 = c_{21},\\ \widehat{e}(P_{pub}, W_{51}) &= \widehat{e}(s\mathcal{G}, T_4 + T_5 + T_6) = m_4 \cdot m_5 \cdot m_6 = c_{51},\\ \widehat{e}(P_{pub}, W_{71}) &= \widehat{e}(s\mathcal{G}, T_3 + T_7 + T_6) = m_3 \cdot m_7 \cdot m_6 = c_{71},\end{aligned}$$

Therefore, the RTU of U_2, U_5, U_7 can pass the authentication by U_1 only if $\widehat{e}(P_{pub}, W_{21}) = c_{21}$, $\widehat{e}(P_{pub}, W_{51}) = c_{51}$, $\widehat{e}(P_{pub}, W_{71}) = c_{71}$, respectively. Broadly speaking, if $\widehat{e}(P_{pub}, W_{ji}) = c_{ji}$, then U_i can authenticate counterpart's entity.

Security Analysis and Performance Analysis

A passive adversary tries to learn information about the conference key by eavesdropping on the broadcast channel. We show that an eavesdropper cannot get any information about the secret key r_i of U_i due to Weil Diffie-Hellman (WDH) problem [5] in $(G_1, G_2, \widehat{e})$ and discrete algorithm problem (DLP) in elliptic curves. In active attack, an adversary not only just records the data, but also can alter, inject, intercept and replay messages. Our protocol can be able to provide the authentication service by sending a special message T_i and W_{ji} in first round and second round, respectively. Our scheme has the security properties of known session key security, perfect forward secrecy, key-compromise impersonation resistance and no key control.

The communication cost of previous schemes are all $O(n^2)$, while the communication cost of our scheme is only $O(n\sqrt{n})$even though the communication round is 2.

Conclusions

SCADA system is a significantly important system that plays a very important role in national infrastructure, such as electric grids and water supplies. However, SCADA system is becoming increasingly vulnerable to adversarial manipulation due to the extreme operational environment. In this paper, we present a novel key management scheme for SCADA systems with good performance and security properties. We believe that our scheme must be promising in the secure communication in SCADA system in the future.

Acknowledgments This work was supported by the research fund from Nanjing University of Information Science and Technology under Grant No. S8113003001, National Science Foundation of China under Grant No. 61272421.

References

1. Beaver C, Gallup D, Neumann W, Torgerson M (2002) Key Management for SCADA. Available: http://www.sandia.org/ scada/documnets/013252.pdf
2. Colin RD, Boyd C, Manuel J, Nieto G (2006) SKMAA key management architecture for SCADA systems. In: 4th Australasian information security workshop, pp 138–192
3. Choi D, Kim H, Won D, Kim S (2009) Advanced key management architecture for secure SCADA communications. IEEE Trans Power Del 24(3):1154–1163
4. Boneh D, Franklin M (2001) Identity-based encryption from weil pairing. In: Advances in Cryptology-CRYPTO01, Lecture Notes in Computer Science, vol 2139, pp 213–229
5. Kim Y, Perrig A, Tsudik G (2004) Group key agreement efficient in communication. IEEE Trans Comput 53(7):905–921