World Applied Sciences Journal 29 (Computer Sciences, Engineering and Its Applications): 140-144, 2014
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2014.29.csea.2244

Secure and Efficient Ey Distribution for Scada

K.P. Thooyamani, R. Udayakumar and V. Khanaa

School of Computing Science, Bharath University, Chennai-73, India

Abstract: Supervisory Control and Data Acquisition (SCADA) is a computer based monitoring tool which controls and monitors the process and take supervisory decisions accordingly. These days' modern industrial facilities have command and control systems. So need for SCADA connection to open network increases and thereby increases the issues in security. One critical aspect for SCADA system is that the communication needs to be secured. SCADA consists of three types of communication equipments and data transmitted between these equipments have to be protected by encryption. A secure key management is essential for data encryption. Several key management schemes already exist for static and dynamic groups. Perfect secrecy can only be achieved in multicast groups by ciphering data sent o group with a different key every time a member joins or leaves the group. This paper proposes ternary tree based new technique to establish a contributory secure group key where ternary tree has at most three children per node. This method provides some advantage over the existing binary tree based technique. As a result, the scheme supporting broadcast and multicast efficiently is proposed.

Key words: Key management · Supervisory-control-and-data acquisition (SCADA) systems · Logical key hierarchy · Key establishment protocols

INTRODUCTON

SCADA systems have been commonly used in national infrastructures such as electric grids, water supplies and pipelines. However the SCADA system is vulnerable to variety of attacks. Security of SCADA systems is under research. Most SCADA systems require message broadcasting and secure communications. A SCADA system consists of three types of communication equipment:

- Human-machine interface (*HMI*)
- Master terminal unit *(MTU*) and
- The remote terminal unit (*RTU*).

HMI: Is an interface for the SCADA system operator. The HMI usually supports a graphic interface.

MTU: Provides supervisory control of RTUs. This device is a root node of the SCADA system architecture. The MTUs have reasonable computational resources as a desktop computer.

RTU: Is a device composed of sensors that are used for Data acquisition, a component that carries out communications and a component responsible for executing instructions coming from the MTU. In this paper we proposed a key management scheme for secure SCADA communications.

The reminder of this paper is organized as follows. Section II summarizes related key-management schemes. Then we propose key-management protocol for SCADA in section III. Section IV discusses the implementation results of existing scheme. Finally section V concludes this paper along with future enhancement is specified.

Related Work:

SKE: A cryptographic key management approach, Key Establishment for SCADA (*SKE*) [1] was proposed by Sandia National Laboratories in 2002. In this technique, the communication is divided in two categories.

- Controller-Subordinate communications
- Peer-to-Peer communications

Corresponding Author: R. Udayakumar, School of Computing Science, Bharath University, Chennai-73, India.

The prime communications strategy proposed in SKE is for controller-to-subordinate messages. SKE uses symmetric key techniques for Controller-to subordinate Communication. SKE uses set of keys: Long Term Key (*LTK*) General Seed key (*GSK*), General Key (*GK*), Session Key.

Peer-to-peer communications are used for communication between substations (sub-master stations). The peer-to-peer channels use public key cryptography for key exchange messages. The keys required for peer-to-peer communications are: Cryptographic Authority Public Key (*CAPK*), Public Key Signature Key (*PKSK*), Public Private Key Pair, Common Key and Session Key.

SKE forces communications between RTUs to be performed through a Substation. The substation will act as a controller and will receive an encrypted message, decrypt it and then re-encrypt it for the recipient (Beaver 1. 2002).

Limitation:

- Both symmetric and public key cryptography techniques are used.
- Long term keys are shared between nodes via manual installation. If a Remote Telemetry Unit (RTU) has multiple master stations, its key will need to be installed on each master station. Also, if a master station is compromised, long term keys are also compromised.

SKMA: Key Management Architecture for SCADA systems (*SKMA*) [2] was proposed by Information Security Institute that overcomes the limitations of SKE. SKMA does not differentiate between master controller and peer-to-peer communications in the same way that SKE does. A consistent approach is used for all communications, which simplifies the process. A Key Distribution Center (*KDC*) is used to maintain a long term key for each node. When a new node joins the system, a node-KDC key is manually installed in it. When two nodes need to communicate, a node-node key is created with the help of node-KDC key. This is also a long term key. For communication, a session key is generated using the node-node key.

Limitation: A session key is generated using the node-node key. This technique supports RTU to RTU communication but does not support broadcast. Furthermore, each RTU needs to store two types of long term keys, one for node to KDC communication and another for node to node communication, which increases the storage overheads.

ASKMA: The authors redefined SCADA security equipment and proposed key management scheme suitable for secure SCADA communication. This approach combines LKH and IOLUS framework. In LKH, a logical tree of keys constructed to support broadcast communication multicast communication.

Limitation: This ASKMA is less efficient during multicast communication.

ASKMA+: This approach is more efficient scheme in supporting both broadcast and multicast communication.Ths approach also employs LKH and IOLUS framework. LKH is a binary tree construction. This approach decreases the computational cost for multicast communication. It reduces the number of keys o be stored in remote terminal unit.

Proposed Key Management: With the increasing demand of different secure groupware application like conference calls, distributed computation etc over insecure networks like Internet, it is required to develop an efficient group key agreement protocol for secure communications. Although several tree-based but efficient group key establishment techniques are available, all employ a binary tree for arranging the group members as the leave nodes and use the two-party DH technique in steps to generate the group key among the members.

In this paper, we propose a new technique that uses a ternary tree and GDH (*Group Diffie Hellman*) with three party as the basic operations to establish a contributory group key among the group members. It has certain advantages over the binary tree based approaches as the number of rounds is reduced due to reduction of the height of the ternary tree, the coverage of the broadcast messages is increased as large number of group members appear at the same level of the ternary tree etc.

This section discusses the major group key management operations and evaluation metrics used in proposed technique and then discusses the implementation of major group key management operation.

Group Key Management Operations: The major group key management operations are as follows,

Initialization Operation: This is the initial creation of the group key and organization of the key management infrastructure.

Single Join: This operation brings one new member in to the existing group.

Mass Join: This operation allows many new members to be added to an existing group.

Mass Leave: This operation is used when multiple member are simultaneously removed from the existing group.

Evaluation Metrics: The proposed technique uses following evaluation metrics to compare the proposed technique with other binary tree based existing techniques.

Number of Rounds: This is a generic time unit used to compare the number of steps taken in different operations.

Number of Messages: This is the sum of all unicast and broadcast message. This number is used to determine the total time of communication in an underlying broadcast network.

Initialization Operation: The initialization operation is the group genesis algorithm. Initially each group member is treated as one independent group having own secret key. Hence if there are n members available in a group then initially there are n independent group exists. In this paper we assume that number of group members that participate in communication is always *3h* form, where *h* is any integer number. The steps are as follows.

- In round 1, each set of three members forms a subgroup and uses group key distribution protocol, GDH to generate the new group key shared by all three members of a subgroup. We choose the rightmost member (node) of a subgroup as the sponsor (a group member controls the group operation) of the subgroup. In this way if we treat every new subgroup as a new node, so the new number of nodes are *n/3*.
- In subsequent rounds each set of three new subgroup (*nodes*) uses following stages to compute the new group key of each three new subgroups (*nodes*).

First Stage (Up Flow): The sponsor of first new subgroup, broadcasts a message including the set constructed as discussed in GDH for party, to all the members of second group, including sponsor of second new subgroup. Then sponsor of second new subgroup similar to GDH.2, constructs a set of values and broadcasts to third new subgroup including sponsor. The sponsor of third new subgroup, similar to GDH computes intermediate values and new group key of these three new subgroups from received set of values.

Second Stage (Down Flow): In down flow stage sponsor of third new subgroup, broadcasts a message including computed intermediate values to all members of two other subgroups (nodes), including sponsor of the other two subgroups. From these values all other members computes the new group key shared by all three new subgroup (nodes).We again choose the rightmost member (node) as the sponsor of a group of three new subgroups.

- Repeat the above process until round h =log3n *(h is the height of the ternary key tree*).At this point we have a single group, which covers all the members, each sharing the common group key.

Join Operation: The join operation occurs when a new joining member sends join request to the group. The procedure for the join operation is as follows.

Suppose that the group has n members, the new member, broadcasts a message including its own blinded key (gk mod p) to every member of the group. Simultaneously the sponsor of the original group unicasts the group's blinded key to new member.

Mass Join Operation: The mass join operation occurs when two or more members send join requests to the group sponsor. Suppose N new member send join requests to sponsor of the original group. The proposed work uses mass join merge method as discussed as follows.

- Initialize the N members to form their own independent group by initialization operation.
- Merge this group with already existing group with initial n members.To merge these two groups, sponsor of each groups broadcasts the blinded key (gk mod p) of the group to the all members of other groups.

Mass Leave Operation: The mass leave operation occurs when more members send leave requests to the group sponsor. In mass leave operation the number of leaving group members is very large,it is more efficient to reconstruct the key tree using the initialization operation discussed in subsection A.In this updating key tree become balanced. Let suppose N member wants to leave the group then reconstruct he key tree for n-N members using the initialization operation discussed in subsection A. Here the value n-N must be of 3 form, where h is any integer number.

Implementation and Results: The following are the results of the Logical key hierarchy of Binary Tree Structure.

Authentication
Keygeneration

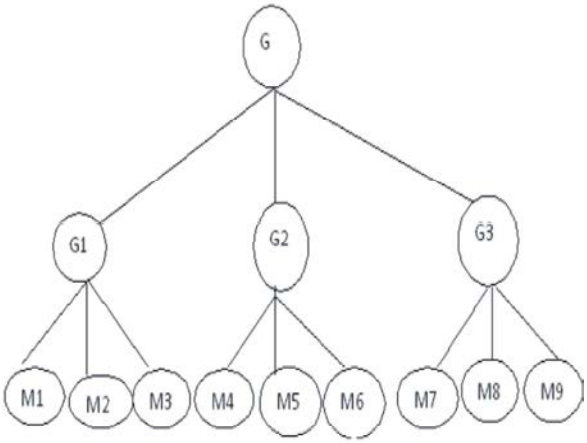

Fig 1: Group Members Using Proposed Technique

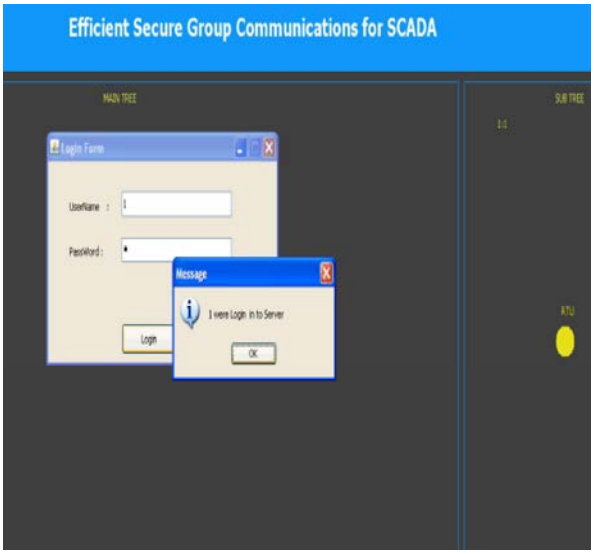

Fig 1: User Authenticaton.

Fig 2: LKH Binary tree constructed for group members who joins and leaves the group

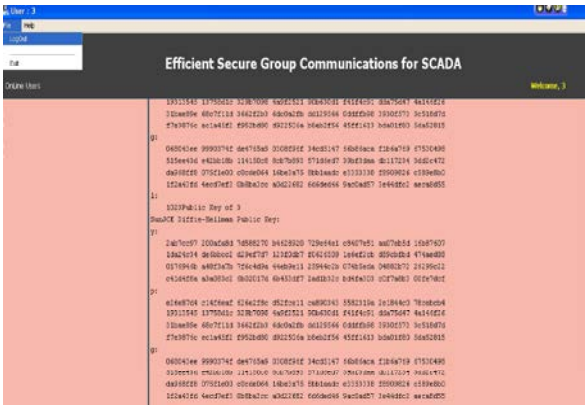

Fig 3: Two party Diffie Hellman technique is used to generatethe group key among members.

Conclusion and Future Enhancements

Conclusion: This paper introduces a new ternary tree based efficient group key agreement protocol for the any group size. The proposed technique uses ternary tree instead of binary tree. In ternary tree based approach group of three members is formed, whereas in binary tree based approach group of two members is formed, so per iteration the number of members covered by subgroup are more in ternary tree based approach. Moreover height of ternary tree is less than height of binary tree, where height of tree is number of iterations required to compute group shared key. If the members of a group uses ternary tree based approach, which uses GDH for two party then cost decreases proportionally with increasing group size as compared to binary tree based approach which uses two party DH.

Future Enhancements: The above project has been simulated for wired environment. The same can be implemented for wireless environment.

REFERENCES

1. Beaver, C.L., D.R. Gallup, W.D. Neu Mann and M.D. Torgerson. 2002. "Key Management for SCADA (SKE)", printed at Sandia Lab.
2. Robert Dawson, 2006. Colin Boyd, Ed Dawson, Juan Manuel, Gonzalez Nieto "SKMA-A Key Management Architecture for SCADA Systems", Fourth Australasian Information Security Workshop (AISW-NetSec 2006).
3. Choi, D., H. Kim, D. Won and S. Kim, 2009. "Advanced key management architecture for secure SCADA communications," IEEE Trans. Power Del., 24(3): 1154-1163.
4. Choi, D., H. Kim, D. Won and S. Kim, 2010. "Efficint Secure Group Communications for SCADA", IEEE Trans. Power Del., 25: 2.
5. Deuk-Whee Kwak, SegJoo Lee, Jong Won Kim and Eunjin Jung, 2006. "An Efficient Lkh Tree Balaning algorithm for Group Key management", IEEE Communication, 10: 3.
6. Ompal, Sharda Saiwan, Peeyush Jain, Zia Saquib, Dhirn, *at el*, 0000. "Cryptographic Key Management fo SCADA System:An Architectural Framework",International Conference on Advances in computing,Control and Telecommunication Technologies.
7. Lihao Xu and Cheng Huang," 2008. Computational-Efficient Multicst Key Distribution", IEEE transaction on Parallel and Distributed Systems, 19: 5.
8. Sachin Tripathi and G.P. Biswas, 0000. "Design of Efficient Ternary-Tree Based Group Key Agreement Protocol for Dynamic Groups".