Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4449 - 4455
Received 05 March 2021; Accepted 01 April 2021.

A Hybrid Groupkey Management Service for Static Iot Applications

Antony Taurshia[a,*], G. Jaspher W. Kathrine[a], Shibin David[a], S. Sudhakar Ilango[b]
[a,*]*Department of Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India*
[b]*School of Computer Science and Engineering, VIT- AP University, India*
Corresponding author Email: taurshia@gmail.com

Abstract: Group management is essentially used for broadcasting and multicasting messages encrypted using a common group key so that only an authorized group member can decrypt the messages. Group key management is done using a centralized trusted server, responsible for key distribution and updates. Software Defined Network (SDN) based security controllers are used for group management service to provide horizontal security control. Several security services are provided by cloud service centres. The application specific security solutions are limited. In the proposed work, an SDN embedded fog or edge server is used for group key management service to route keys to the respective devices to secure group communication. A novel hybrid tree-based group key management technique is proposed which uses a combination of Logical Key Hierarchy (LKH) and One-Way Function Trees (OFT) scheme. The performance evaluation shows that the scheme is efficient when compared to the existing key management techniques in terms of member device computation and storage overhead.

Keywords: group key management, Software Defined Networks, Internet of Things, security services

I. INTRODUCTION

Cloud is an efficient platform capable of providing services to IoT applications. Fog and edge are the branches of cloud that collaborate to provide efficient services[1]. Services are provided by the cloud platform with the support of technologies like Virtual Functions (VFs) and Software Defined Network (SDN) [2]. SDNcentered group management is proposed in [3] to enact horizontal end to end security management. Recently the attacks on IoT devices increased greatly. Thousands of IoT devices are captured as bots to perform attacks like DDoS. Mirai, Reaper, and Wirex are the botnet attacks on IoT applications[4]. To enforce access control is one of the ways to prevent the capture of IoT devices and reduce attacks. Since IoT devices are resource-constrained, implementing strong cryptographic primitives live ABE is difficult [5]. Group management is one of the ways to prevent unauthorized access control and enforce secure group-based communication. Group management is of three types, centralized, decentralized, and distributed [6]. In the centralized method, a trusted server is used for key management and group member management. In the decentralized method, all the members participate in key management and group member management. In distributed both a central server and members participate effectively in key management and member management.

In group management, a common group is issued to members to broadcast, multicast messages, and also to secure communication within group members. A non-group member cannot access the messages as the message is encrypted using the group key. Forward secrecy, backward secrecy, and collision resistant are the important security properties a group should follow. Forward secrecy is the case that a leaving member should not be able to derive any future keys. Backward secrecy is a joining member should not derive any previous group keys. Collision resistance is when two or more users collude to derive unauthorized group keys. To ensure these security properties rekeying is done whenever a member joins or leaves. To reduce the communication and computation cost several key management schemes are proposed. Our work uses SDN based security controller as a trusted server to enact centralized group management service to IoT applications. Tree-based key management is one of the efficient centralized key management schemes[6]. Our work proposes a novel hybrid key management scheme combining LKH and OFT schemes. The scheme is collision resistant and also ensures forward and backward secrecy. The performance evaluation shows that thescheme has lesser member device computation overhead and storage overhead compared to the existing schemes. Hence suitable for resource constrained IoT devices

Our work unfolds as follows. Section I reviewing existing centralized tree-based key management schemes. Section II briefs the proposed novel hybrid tree-based key management scheme and how SDN based security controller act as a trusted key management server to IoT applications. Section III gives a security analysis of our work. Section IV gives the performance evaluation of our work with existing works and section V gives the conclusion.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4449 - 4455
Received 05 March 2021; Accepted 01 April 2021.

II. EXISTINGCENTRALIZED TREE-BASED KEY MANAGEMENT TECHNIQUES

LKH [7] is a stateful centralized tree-based group management technique where the new keys are distributed by encrypting them in old keys. The users form the leaf node of the tree. They have a secret individual key, a set of intermediate keys on the way to the root of the tree, with which the group key can be obtained. This tree-based key management reduces the communication cost to log(dn) where d is the degree of key tree and n is the number of members in the group. A variation of the LKH scheme is the OFT in which the group key is generated by the nodes themselves in the bottom-up approach[8]. Each user holds a secret key and a one-way function of it gives the blind key. The intermediate nodes containing key encryption key KEK are a mix of the child nodes blinded keys. The schemes have reduced rekeying communication cost compared to LKH scheme. Still, the scheme is vulnerable to the collision attack.

A pseudorandom generator based OFT scheme is proposed in[9]. The user is assigned from the leaf node to the rootwith keys generated using a pseudorandom generator. The scheme is more secure than OFT since the keys cannot be derived even if a node is captured. The rekeying cost is reduced compared to the LKH scheme. A lightweight group key management scheme based LKH structure for IoT devices and user groups is proposed in [6]. The device groups are subscribed by user groups to obtain information from subscribed devices. Whenever a device joins a hash of device id and a secret key is used for the key update. When a device leaves the group, the tree structure is changed to update the device IDs to avoid collusion attacks. The proposed scheme is lightweight in terms of computation compared to the traditional LKH scheme. Still, if a device subscribes to the same group, then there is a possibilityof forgery.

Another tree-based group key management technique is proposed in [3]. The LKH key structure is used and the blind key of the adjacent pair node is used for finding intermediate key using the Diffie-Hellman method. Still, the use of the Diffie-Hellman method in highly resource-constrained devices of IoT is not feasible.Two schemes for collision-resistant one way function tree (OFT)are proposed in [10]. Scheme ROFT performs a hash on the old keys on the path to the root of the node that joins and also on the known keys in the opposite side tree. Scheme NOFT uses virtual nodes to compute keys when a node joins. The scheme is collision resistant and computationally efficient than the existing collision-resistant OFT. Still,the computational load on the member device during the join operation is higher than the original OFT. A novel LKH based scheme using a ternary tree is used in [11]. The scheme has minimal storage overhead and uses one-way key derivation for batch-based rekeying to reduce communication overhead. Though tree-based key management is efficient balancing a binary tree causes extra overhead[12].

III. OUR PROPOSED KEY MANAGEMENT AS A SERVICE TO IOT APPLICATIONS

A hybrid novel key management technique that uses a combination of LKH OFT scheme is proposed. The fig. 1 depicts the structure of the key tree. The subtrees generate the KEK as in oft scheme using a one-way function in bottom-up fashion. The group key GK can be decrypted using the KEK in top-down fashion as in LKH scheme.

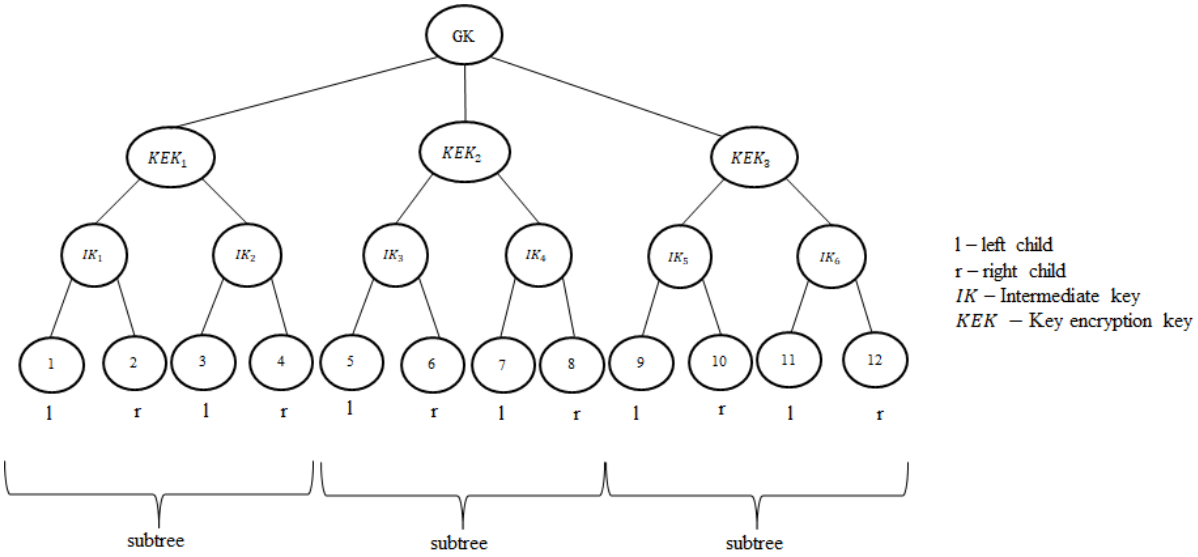

Fig. 1 Proposed group key tree.

All the users share an individual secret key s_i with the key management server. The users form the leaf node. Each subtree contains 4 leaf nodes as in fig. 1. Every node in the key tree is associated with a node secret and

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4449 - 4455
Received 05 March 2021; Accepted 01 April 2021.

blinded node secret. The node secret is used for secret key distribution, whereas the blinded node secret is distributed to adjacent nodes to generate the intermediate keys. The users are issued with anindividual node secretS_i. A one-way function of node secret$f(S_i)$is the blinded node secretBl_i of individual nodes. IK node secretS_{IKi} is calculated as the xor of blinded node secrets of left and right child nodes$Bl_{li} xor Bl_{ri}$. IK node's blinded node secretBl_{IKi} is the one-way function of IK node's secret $f(Bl_{li} xor Bl_{ri})$. KEK node donot have a blinded node secret since it is not distributed to the adjacent subtress. The $KEK$is calculated as$f(Bl_{IKli} xor Bl_{IKri})$. The group key is obtained using the KEKas in LKH scheme.

User join procedure

When a user joins as in fig. 2 the joining user node 10 is issued with new node secret S_{10} encrypted using its individual secret key s_{10}, blinded node secret $Bl_9$of node 9. The node 9 is issued with the blinded node secret$Bl_{10}$of node 10. Both node 9 and 10 calculate the intermediate key IK_5 node secret$S_{IK5} = Bl_9 xor Bl_{10}$and blinded secret$Bl_{IK5} = f(Bl_9 xor Bl_{10})$. The nodes 9 and 10 are issued with the updated blinded node secret of IK6. Updated Bl_{IK6} is updated by performing a one-way function on the existing blinded node secret of IK_5. The nodes 9,10,11,12 calculate the KEK and obtain the group keyGK. (Add about individual secret.)

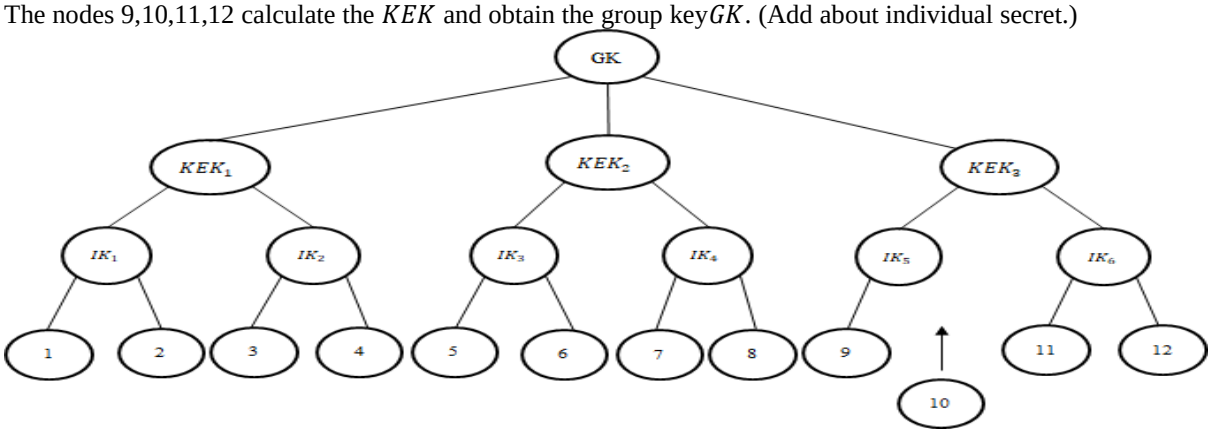

Fig. 2 User 10joins adjacent to leaf node9

$user\ 10 \leftarrow [Bl_9, S_{10}, Bl_{IK6}]s_{10}$
$user\ 9 \leftarrow [Bl_{10}]\ S_9$
$user\ 11{,}12 \leftarrow [Bl_{IK5}]S_{IK6}$
$user\ 1-8 \leftarrow [new\ GK]old\ GK$
$user\ 9-12 \leftarrow [new\ GK]KEK_3$

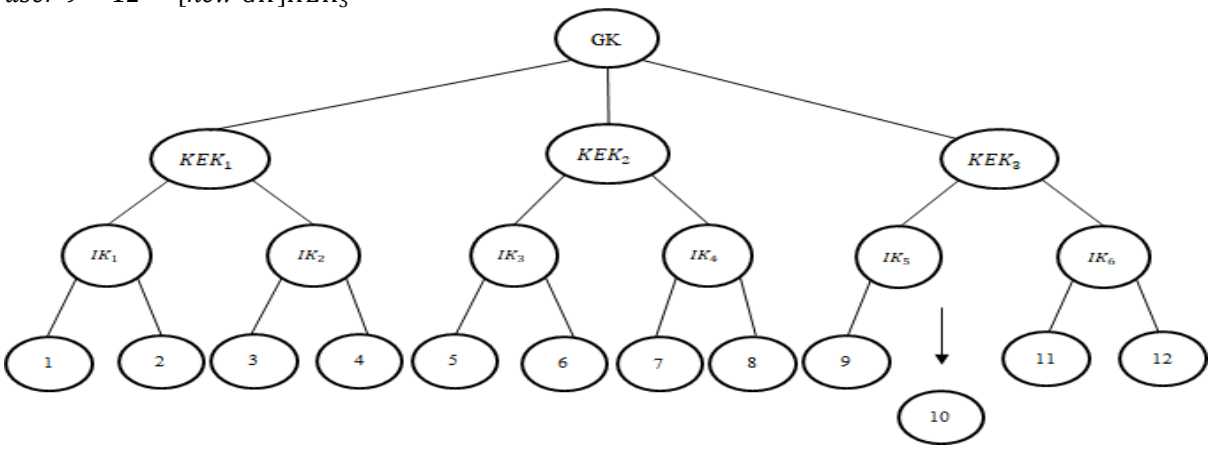

Fig. 3 User 10 leaves adjacent to leaf node 9

User leave procedure

When a user leaves as in Fig.3, the node 9 is issued with new node secret S_9. Hence the blinded node secret is$Bl_9 = f(S_9)$. The blinded intermediate node secret $Bl_{IK5} = Bl_9$. The users 11 and 12 are issued with new blinded intermediate key to calculate the new KEK.

$user\ 9 \leftarrow [S_9]s_9$
$user\ 11{,}12 \leftarrow [Bl_{IK5}]S_{IK6}$
$user\ 1-4 \leftarrow [GK]KEK_1$
$user\ 5-8 \leftarrow [GK]KEK_2$
$user\ 9-12 \leftarrow [GK]KEK_3$

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4449 - 4455
Received 05 March 2021; Accepted 01 April 2021.

Software Defined Security Controller for group management in IoT applications

For a centralized group key management scheme, a trusted key management server (KMS) is used for group member management and key distribution. A SDN-based security controller is used for group management in [3]. For the proposed work also an SDN-based Security Controller is used for grouping the devices and for efficient key management. SDN is a logically centralized platform that is capable of virtualizing network functions using software. SDN is used for clustering and cluster head selection in applications like VANETS[13]. In our work SDN is used for grouping the devices. The communication link between non-group devices can be switched off to avoid any unnecessary messages. The devices that form a system or the devices that communicate often can be formed into a group as in fig. 7. For example, a smart home application contains a security surveillance system, home entertainment system, lighting management system, water management system, temperature, and air-conditioning system, etc. The devices responsible for security surveillance form a group and the devices involved in lighting management form a group. With a common group key, they can broadcast, multicast messages efficiently. This reduces the communication overhead as well as preventing the spread of attacks like node capture attacksin case a device in the smart home application is captured.

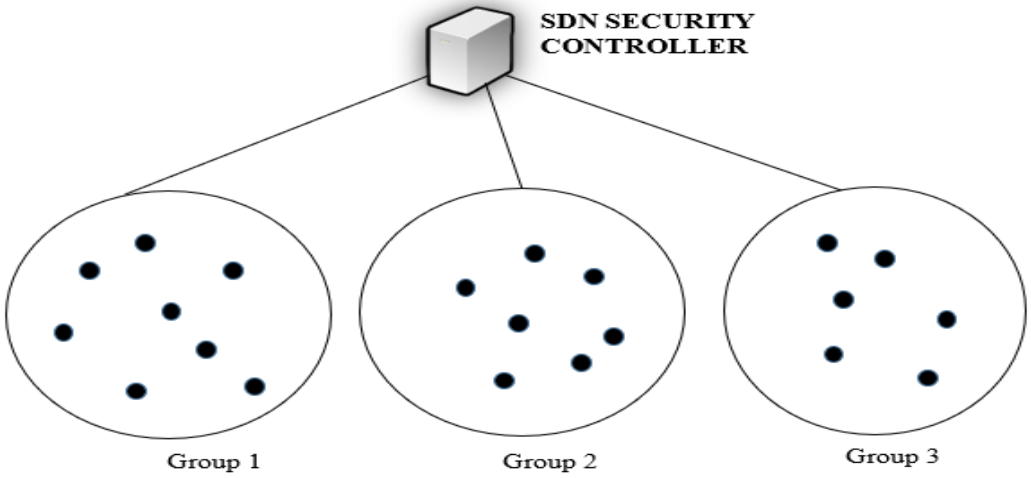

Fig. 4 SDN security controller for group management of IoT applications

The nodes in the group communicate using a common group key. For intergroup communication, a communication request is sent to the security controller. After verifying the node as a valid group member, the request is forwarded to the admin device of the application to grant or deny requests. If the request is granted a session key is forwarded to both devices for one-time communication.

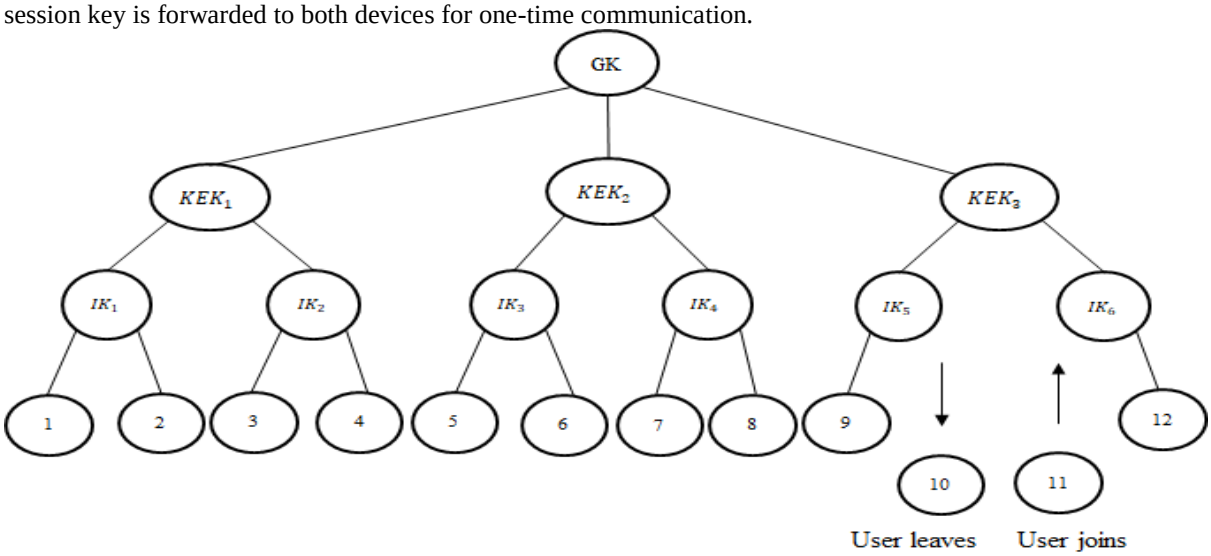

Fig. 5 User 10 and 11 collides

IV. SECURITY ANALYSIS

Condition: The one-way function used should be strong enough that the output can be computed if and only if both the input values are known.

Backward secrecy

Theorem 2. In proposed scheme when a user joins, the joining user donot possess any unauthorized previously used keys to ensure backward secrecy.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4449 - 4455
Received 05 March 2021; Accepted 01 April 2021.

Proof: When user 10 joins as in fig.2 at time t_1, the user is issued with its own node secretS_{10} and adjacent blinded node secret $Bl_9$to compute the intermediate secret $S_{IK5}$and blinded secretBl_{IK5}. The updated intermediate blinded node secretBl_{IK6} is issued to calculate the new KEK. Hence the joining user donot possess any unauthorized previously used keys ensuring backward secrecy.

Forward secrecy

Theorem 1. In proposed scheme when a user leaves, the leaving user do not possess any unauthorized future keys to ensure forward secrecy.

Proof: When user 10 leaves as in fig. 3 at time t_2, the keys user A knows while leaving include node 9 blinded node secretBl_9intermediate node secretS_{IK5}, blinded intermediate node secretBl_{IK5}, blinded intermediate nodesecretBl_{IK6} and KEK. Since the node secret of its adjacent node changes all the keys known to user 10 consequently changes except the blinded intermediate node secretBl_{IK6}. Still 10 cannot generate the KEK without knowing the updated blinded intermediate node secretBl_{IK5}. Hence the user A do not possess any unauthorized future keys ensuring forward secrecy

Collision attack

According to Liu's theorem a one-way function tree is collision resistant if and only if an arbitrary number of users $\{u_1, u_2, \dots u_n\}$cannot collude to compute any node secrets unknown including the group key[14].

Theorem 3: The proposed scheme is collision resilient.

Proof: Collusion attack is feasible only when a set of malicious users collude to obtain any unauthorized node secret thereby gaining the group key for unauthorized time period t. According to theorem 1 the only key known to the leaving user at time t_1 is intermediate blinded node secretBl_{IK6}. To obtain KEK the leaving user colludes with a joining user as in fig. 5 at time t_2. According to theorem 2 the joining user donot possess any previously used unauthorized key. Hence the proposed scheme is collision resilient.

V. PERFORMANCE EVALUATION

The performance of our proposed scheme is compared with an existing tree-based key management schemes proposed in , and.

Computation overhead per member

The computation cost of the proposed scheme is compared with the existing schemes in table 1.C_D denote the computation cost for one key decryption and C_H denote the computation cost for one secure hash function. The proposed work has lesser member computation overhead than the existing ones.

Table 1 Comparison of computation overhead.

Computation overhead	Group key management techniques					
	LKH	OFT	ROFT	NOFT	Proposed	
					Member of same subtree	Member of other subtrees
User leave	$2log_2n \times C_D$	$C_D + log_2n \times C_H$	$C_D + log_2n \times C_H$	$C_D + log_2n \times C_H$	$C_D + 2 \times C_H$	$1\times C_D$
User Join	$3log_2n \times C_D$	$2C_D + log_2n \times C_H$	$2C_D + 2(log_2n - 1) \times C_H$	$2C_D + 2(log_2n - 1) \times C_H$	$C_D + 2 \times C_H$	$1\times C_D$

Storage overhead per member

The number of keys the user stores in tree-based key management keeps increasing with the increase in the height of the tree. For the proposed work each member device stores their individual node secret S, blinded node secret of adjacent node Bl_a, blinded intermediate node secret Bl_{IK}, and group keyGK. Hence the storage cost is 4 and remains the same with the increase in the group size. Table 2 depicts the member storage overhead of the different schemes where n is the no of users in a group. The comparison of the storage overhead of the proposed scheme with the existing schemes is depicted in fig. 9.

Table 2 Comparison of storage overhead.

Storage overhead	Group key management techniques				
	LKH	OFT	ROFT	NOFT	Proposed
	$log_2n + 1$	$2log_2n + 1$	$3log_2n$	$2log_2n + 2$	4

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4449 - 4455
Received 05 March 2021; Accepted 01 April 2021.

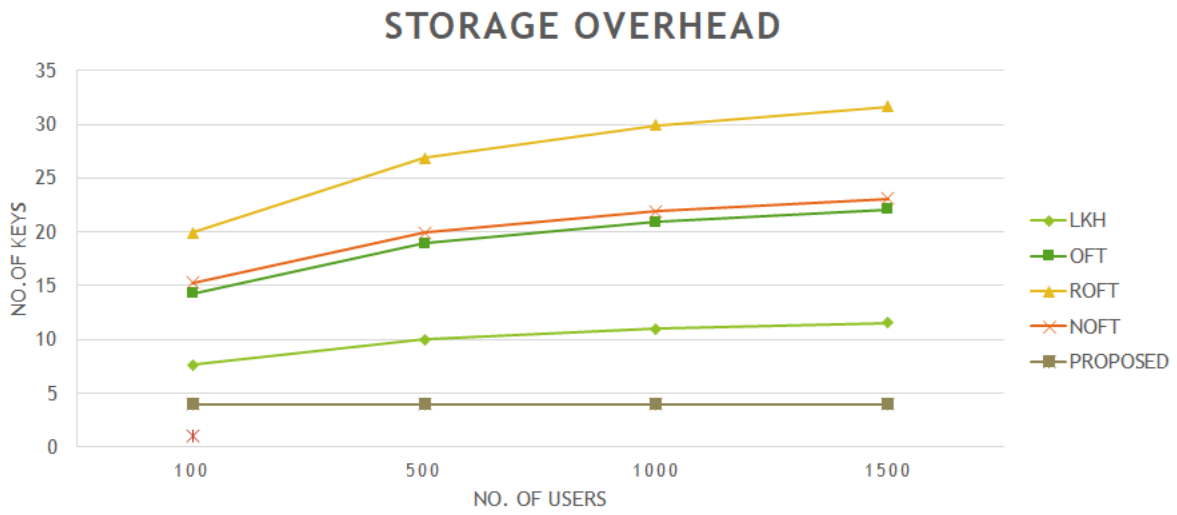

Fig. 6 Comparison of storage overhead of proposed scheme with existing schemes

VI. SIMULATION RESULTS

Group management using SDN is simulated using mininet installed in Oracle VM. The SDN controller along with the switches and hosts are created. The hosts are grouped into three groups using python script. Communication link is set only for the hosts of same group. Hence a host from group 2 cannot ping a host from group 1. Hence a secure group communication is ensured using SDN.

VII. CONCLUSION

Security is provided as a service by cloud centers. In our work key management is provided as a service using an SDN based security controller to secure inter-group and intra-group communication in IoT applications. LKH is an efficient tree-based key management scheme for group communication. The OFT is a tree based key management scheme which has lower communication cost compared to the LKH scheme. The scheme uses a bottom-up approach where the ancestor keys are a function of the sibling node's keys. But the scheme is prone to collision attack, where two or more users collide to derive unauthorized keys. Many collision-resistant OFT schemes are proposed, but with increased overhead. Our work proposes a novel tree-based group key management scheme which uses a combination of LKH and OFT scheme. Security analysis shows that our scheme is collision-resistant. Performance evaluation shows that our scheme is more efficient in terms of each member device computation and storage overhead than the existing key management schemes, hence suitable for resource constrained IoT devices. Since the proposed work has higher communication overhead, the scheme is suitable for static IoT applications like smart home where the group member change is non-dynamic.

REFERENCES

1. Hu, P., Dhelim, S., Ning, H., Qiu, T.: Survey on fog computing: architecture, key technologies, applications and open issues. J. Netw. Comput. Appl. 98, 27–42 (2017). https://doi.org/10.1016/j.jnca.2017.09.002.
2. Joshi, K.D., Kataoka, K.: pSMART: A lightweight, privacy-aware service function chain orchestration in multi-domain NFV/SDN. Comput. Networks. 178, 107295 (2020). https://doi.org/10.1016/j.comnet.2020.107295.
3. Festijo, E., Jung, Y., Peradilla, M.: Software-defined security controller-based group management and end-to-end security management. J. Ambient Intell. Humaniz. Comput. 10, 1–18 (2018). https://doi.org/10.1007/s12652-018-0678-6.
4. Vishwakarma, R., Jain, A.K.: A survey of DDoS attacking techniques and defence mechanisms in the IoT network. Telecommun. Syst. 73, 3–25 (2020). https://doi.org/10.1007/s11235-019-00599-z.
5. Ambrosin, M., Anzanpour, A., Conti, M., Dargahi, T., Moosavi, S.R., Rahmani, A.M., Liljeberg, P.: On the Feasibility of Attribute-Based Encryption on Internet of Things Devices. IEEE Micro. 36, 25–35 (2016). https://doi.org/10.1109/MM.2016.101.
6. Kung, Y.H., Hsiao, H.C.: GroupIt: Lightweight Group Key Management for Dynamic IoT Environments. IEEE Internet Things J. 5, 5155–5165 (2018). https://doi.org/10.1109/JIOT.2018.2840321.
7. Hashimoto-Hill, S. et al: 済無No Title No Title. J. Chem. Inf. Model. 53, 1689–1699 (2011).
8. Rafaeli, S., Mathy, L., Hutchison, D.: An efficient one-way function tree implementation for group key

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4449 - 4455
Received 05 March 2021; Accepted 01 April 2021.

management. 1–24 (2001).
9. R. Canetti, J. Garay, G. Itkis, D. Micciancio, M.N. and B.P.: Multicast security: a taxonomy and some efficient constructions. In: IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320). pp. 708–716 (1999). https://doi.org/10.1109/INFCOM.1999.751457.
10. Sun, Y., Chen, M., Bacchus, A., Lin, X.: Towards collusion-attack-resilient group key management using one-way function tree. Comput. Networks. 104, 16–26 (2016). https://doi.org/10.1016/j.comnet.2016.04.014.
11. Pon Senniah, J., Ram Prasad, A. V.: Efficient data sensing with group key management for intelligent automation system by one-way key derivation in wireless networks. J. Ambient Intell. Humaniz. Comput. 1–8 (2020). https://doi.org/10.1007/s12652-020-01862-x.
12. Kwak, D., Lee, S., Kim, J., Jung, E.: An Efficient Key Tree Management Algorithm for LKH Group Key Management. 703–712 (2006).
13. Duan, X., Wang, X., Liu, Y., Zheng, K.: SDN enabled dual cluster head selection and adaptive clustering in 5G-VANET. In: IEEE Vehicular Technology Conference (2016). https://doi.org/10.1109/VTCFall.2016.7881214.
14. Liu, J., Yang, B.: Collusion-resistant multicast key distribution based on homomorphic one-way function trees. IEEE Trans. Inf. Forensics Secur. 6, 980–991 (2011). https://doi.org/10.1109/TIFS.2011.2144584.