Journal of Engineering Sciences

Vol 12, Issue 05, MAY /2021
ISSN NO:0377-9254

Novel Approach on Data Access Control With Fine-Grained Data Protection In Cloud-Assisted IIOT

[1]J.SUDHEER KUMAR,[2] Kolla Vamshi Kumar Reddy, [3]Konatham Sai Yashwanth Reddy, [4]Yalaka Srikanth Reddy

G.Archana Assistant Professor in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana.

Kolla Vamshi Kumar Reddy UG Scholar in in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana.

Konatham Sai Yashwanth Reddy UG Scholar in in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana,

Yalaka Srikanth Reddy UG Scholar in in Department of IT Teegala Krishna Reddy Engineering College,Hyderabad,Telangana,

Abstract—

Industrial Internet of Things (IIoT) has provided a promising opportunity to build digitalized industrial systems. A fundamental technology of IIoT is Radio-Frequency IDentification (RFID) technique, which allows industrial participants to identify items and anchor time series IoT data for them. They can further share the IoT data through the cloud service to enable information exchange and support critical decisions in production operations. Storing IoT data in the cloud, however, requires a data access control mechanism to protect sensitive business issues. Unfortunately, using traditional cryptographic access control schemes for time series IoT data face severe efficiency and key leakage problems. In this paper, we design a secure industrial data access control scheme for cloud-assisted IIoT. Our scheme enables participants to enforce fine-grained access control policies for their IoT data via ciphertext policy-attribute based encryption (CP-ABE) scheme. Our scheme adopts a hybrid cloud infrastructure for participants to outsource expensive CPABE tasks to the cloud service with strong privacy guarantees. Importantly, our scheme guarantees a new privacy notion named item-level data protection for IoT data to prevent key leakage problem. We achieve these goals via several encryption and optimization techniques. Our performance assessments combine system implementation with large-scale emulations and confirm the security and efficiency of our design.

I. INTRODUCTION

Industrial Internet of Things (IIoT) allow industrial system to collect a vast amount of IoT data about all aspects of the production process. A foundational technology for IIoT is the RFID technology, which allows industrial participants to attach RFID tags to items, automatically identify items and anchor time series IoT data for them derived from a spectrum of IoT devices throughout their life cycle. The IoT data can be then shared among the industrial participants to

build new businesses and services. Generally speaking, IIoT facilitates data collection and sharing within the industrial system, which can benefit many applications such as decision making [5], personalized design [6], quality and safety [18], lifecycle management [19], etc.

To facilitate the sharing of IoT data, a cloud service is usually used to enable seamless, efficient, and robust data sharing for IIoT [6], [12]–[14], [17]. By storing IoT data in the cloud, industrial participants can use the cloud service as a central repository to share the IoT data even when they are distributed in different geo-locations. However, the combination of cloud and IIoT raises fundamental privacy issue. IoT data stored in the cloud can be closely related to sensitive business issues such as the design of a new item or customer preference to provide personalized item service. Besides, the cloud should not be trusted in practice, since the data stored in the cloud can be disclosed intentionally by the cloud administrator or unintentionally due to system misconfiguration. Therefore, it is indispensable to enforce data access control on the potentially untrusted cloud.

A straightforward but inefficient approach is to encrypt IoT data using cryptographic primitives and distribute decryption keys only to authorized participants. One of the most expressive of these is ciphertext policy-attribute based encryption (CP-ABE) [38], which is a natural fit for enforcing fine-grained access control. A participant can specify access policies based on logical expressions over attributes for its data encryptions before outsourcing to the cloud service. A key authority assigns each participant a secret key corresponding to the set of attributes that describes the participant's features (e.g., company nationality, business domain, etc). CP-ABE ensures that only participants with attributes satisfying the logical expression can decrypt the data. Unauthorized parties including the cloud cannot learn any nontrivial information from the encrypted data.

Although CP-ABE has been broadly used to design various access control schemes in the setting of untrusted cloud [21]– [25], there are still several key differences that render CPABE inapplicable to cloud-aided IIoT. First, as an expensive cryptographic primitive, CP-ABE is too slow to meet the high throughput requirement of time series IoT data in an industrial context. Second, CP-ABE relies on a key authority to derive all the ABE keys from a master key. However, deploying a key authority in a cloud-aided IIoT system introduces severe

privacy vulnerability. If the key authority is compromised, the IoT data of the whole system will be disclosed.

In this paper, we design a secure industrial data access control scheme for cloud-aided IIoT, which enables participants to enforce fine-grained access policies for their IoT data. Our scheme constructs the cloud service as a hybrid cloud infrastructure, which consists of a private cloud and a public cloud. For efficiency, our scheme enables the participants to delegate expensive CP-ABE tasks to the resource-abundant cloud service, i.e., relying on the public cloud to store encrypt ed IoT data and the private cloud to execute CP-ABE tasks over the data. For security, our scheme protects IoT data with item keys in item-level. As a result, our scheme guarantees a new privacy property named item-level data protection, which ensures that only participants that are involved in the production of an item can access its IoT data even the key authority is compromised.

Our contributions are summarized as follows.

• We devise a set of encryption techniques to ensure itemlevel data protection while enabling the private cloud to execute CP-ABE tasks over IoT data. With these techniques, our scheme enables industrial participants to only execute lightweight symmetric encryption tasks to meet the high throughput requirement of time series IoT data.

• To further improve the performance of our scheme, we devise a set of optimization techniques with new tradeoffs for the private cloud to execute CP-ABE tasks in a scalable way. With these techniques, our scheme enables the private cloud to execute CP-ABE encryption/decryption tasks in batch level and CP-ABE re-encryption tasks regardless of the size of IoT data.

• We implement a prototype of the private cloud as a distributed computing infrastructure, which is customized for CP-ABE tasks. We evaluate several critical performance metrics of our implementation. Comparing with a raw CP-ABE processing engine, our scheme achieves 2 times of speedup ratio. When the optimization techniques are turned on, our scheme further achieves at least two orders of magnitude of speedup ratio.

II. PROBLEM STATEMENT

A straightforward but inefficient approach is to encrypt IoT data using cryptographic primitives and distribute decryption keys only to authorized participants. One of the most expressive of these is ciphertext

policy-attribute based encryption (CP-ABE) [38], which is a natural fit for enforcing fine-grained access control. A participant can specify access policies based on logical expressions over attributes for its data encryptions before outsourcing to the cloud service. A key authority assigns each participant a secret key corresponding to the set of attributes that describes the participant's features (e.g., company nationality, business domain, etc). CP-ABE ensures that only participants with attributes satisfying the logical expression can decrypt the data. Unauthorized parties including the cloud cannot learn any nontrivial information from the encrypted data

III. PROPOSED MODEL

For efficiency, our scheme enables the participants to delegate expensive CP-ABE tasks to the resource-abundant cloud service, i.e., relying on the public cloud to store encrypt ed IoT data and the private cloud to execute CP-ABE tasks over the data. For security, our scheme protects IoT data with item keys in item-level. As a result, our scheme guarantees a new privacy property named item-level data protection, which ensures that only participants that are involved in the production of an item can access its IoT data even the key authority is compromised.

Our contributions are summarized as follows

• We devise a set of encryption techniques to ensure itemlevel data protection while enabling the private cloud to execute CP-ABE tasks over IoT data. With these techniques, our scheme enables industrial participants to only execute lightweight symmetric encryption tasks to meet the high throughput requirement of time series IoT data.

• To further improve the performance of our scheme, we devise a set of optimization techniques with new tradeoffs for the private cloud to execute CP-ABE tasks in a scalable way. With these techniques, our scheme enables the private cloud to execute CP-ABE encryption/decryption tasks in batch level and CP-ABE re-encryption tasks regardless of the size of IoT data.

• We implement a prototype of the private cloud as a distributed computing infrastructure, which is customized for CP-ABE tasks. We evaluate several critical performance metrics of our implementation. Comparing with a raw CP-ABE processing engine, our scheme achieves 2 times of speedup ratio. When the optimization techniques are turned on, our scheme further

achieves at least two orders of magnitude of speedup ratio.

IV. IMPLEMENTATION

We implement CSP in our scheme as a distributed computing infrastructure to process CP-ABE tasks in a scalable way. Although many distributed computing infrastructures (e.g., mapreduce, hadoop) have been proposed, these infrastructures are not efficient to process CP-ABE tasks as they are designed for general computation tasks. Instead, we tailer and implement an efficient distributed computing infrastructure customized for CP-ABE tasks. We emphasize that our implementation is a primary prototype without advanced system optimization. According to the characteristics and requests of real industrial applications, the system manager can select suitable computing environment (e.g., GPU, multi-core, computing cluster), adjust computation resource in running time, and use optimization techniques (e.g., queuing theory, control-feed back) to meet the application requirement and reduce the energy cost. Our implementation uses C and python. For CP-ABE tasks, we use an implementation of the CP-ABE scheme [56] with elliptic curves from the Stanford Pairing-Based Cryptography library [42] and open SSL [57] for AES implementation. Our implementation uses the standard 80-bits security parameter. In the following, we first describe our implementation of CSP. We then describe the design and implementation of our task schedule framework which is deployed at CSP to schedule ABE-tasks.

At a high level, our CSP implementation provides three function calls for participants to interact with SSP:

• Data-sub: a participant invokes this function to submit a ABE-encrypted IoT record to SSP for share. Upon receiving the request, Data-sub generates and sends a CPABE encryption task to the task schedule framework.

• Data-re: a participant invokes this function to retrieve an IoT record submitted by other participants from SSP. Upon receiving the request, Data-re generates and sends a CP-ABE decryption task to the task schedule framework.

• Policy-up: a participant invokes this function to change the access authority of one of its submitted IoT records. Upon receiving the request, Policy-up generates and sends a CPABE re-encryption task to the task schedule framework. CSP accepts and processes all the CP-ABE tasks submitted by the three function calls. In

processing these tasks, CSP interacts with SSP by submitting/retriving ABE-encrypted IoT records.

Overview: Our CSP implementation consists of a master node and a set of slave nodes to support scalable task processing. To assign CP-ABE tasks to slave nodes, our implementation adopts a two-layer task schedule framework to achieve a balance between load balancing and schedule flexibility. In the first layer, the master node runs a global task schedule module to assign CP-ABE tasks submitted by participants to slave nodes. The global task schedule module analyzes the workloads of the submitted CP-ABE tasks and assigns them to balance the loads of the slave nodes. In the second layer, each slave node runs a local task schedule module to process the assigned CP-ABE tasks. According to different service requests, the local task schedule module adopts specific schedule algorithms (e.g., FCFS, SJF etc) to schedule the processing order of its assigned CP-ABE tasks. The master node needs to transfer encrypted IoT records to slave nodes for them to execute CP-ABE tasks, which incurs extra communication overhead. Since we adopt hybrid encryption strategy, an encrypted IoT record consists of a short key part which is an ABE-encrypted data key, and a long data part which is a symmetrically-encrypted IoT record. We observe that a slave node only needs the key part to process an ABE-encryption/decryption task. Based on this observation, we propose to divide encrypted IoT records into key parts and data parts. The master node caches the data parts in its local storage and only transfers the key parts to slave nodes. This divide-and-cache optimization avoids a large volume of data transfer overhead between slave nodes and the master node.

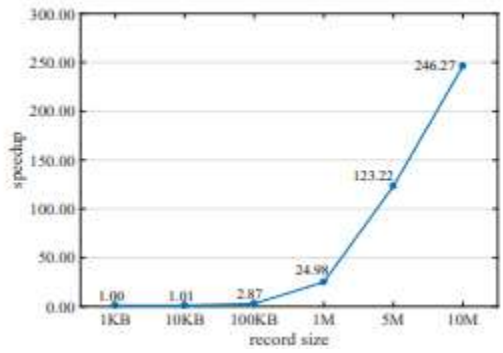
Redis implementation: Our implementation is built on Redis. Redis is a memory-based key-value database and is often used as a structure server to support network communication, persistent storage, and various language APIs. Redis stores data in memory and has extremely fast access speed. Redis also has some features that a database should have, such as replication, adjustable levels of durability, clustering, and high availability. Compared to memcached, Redis provides richer data types to accommodate different scenarios. In combination with the above advantages, we choose to use Redis to implement key data structures such as task queue in our implementation. Figure 6 shows the design details of our Redis based implementation.

The master node maintains two important data structures, caching-queue, and node state-queue. caching-queue is used to store CP-ABE tasks submitted by the participants and task results returned by slave nodes. node state-queue is used to manage information about the slave nodes such as their working queue IDs and current loads. As we have discussed above, the master node also maintains a global task schedule module to assign CP-ABE tasks from caching-queue to working-queues of slave nodes. On the other hand, each slave node maintains a local task schedule module to assign threads for the CP-ABE tasks in its working-queue. Once a task has been processed, the slave node returns the task result to the master node and updates its current load maintained in node state-queue.

V. RESULTS ANALYSIS AND EVALUATION

We now evaluate the empirical performance of our system prototype. We only consider the three types of data operations of our scheme on the cloud and do not consider the issue of performance degradation [49] incurred by other types of tasks. Our evaluation consists of three groups of experiments: (1) basic evaluation to show that CSP affords most of the overhead in CP-ABE tasks; (2) evaluation of our optimizations to demonstrate their efficiency advantages; and (3) evaluation

of our scheme as a whole to show its effectiveness. For a CPABE encryption task, we fix its policy complexity to 10. For a CP-ABE decryption task, we randomly set the size of its MCA set from 1 to 20. We deploy CSP on a high performance workstation in our lab. Since CSP is implemented as a distributed computing infrastructure, we use the workstation to simulate the infrastructure which consists of one master node and three slave nodes. Each slave node is equipped with a 3.30 ghz 4-cores Intel i5-4590 CPU, a 8 GB RAM and an operating system of Ubuntu 16.04. We deploy SSP on tencent cloud host with a single core, a 1 GB RAM and an operating system of Centos6.5. The bandwidth between CSP and SSP is 10 Mbps. Finally, we deploy participant on a personal laptop with a 2.3 ghz 4-core Intel i5-6300hq CPU and a 8 GM RAM



VI. CONCLUSION

We design a secure industrial data access control scheme for cloud-aided IIoT to enforce fine-grained access policies and item-level data protection. Our scheme adopts a hybrid cloud infrastructure and enables participants to delegate expensive access enforcement tasks to a dedicated computing service provider. Our scheme proposes a set of encryption techniques to ensure item-level data protection for item records to prevent key leakage problem. It also proposes several optimization techniques to optimize the performance of the computing service provider while preserving the item-level data protection.

VII. REFERENCES

[1] L. D. Xu, W. He, S. Li, "Internet of Things in Industries: A Survey", in IEEE Transactions on Industrial Informatics, Volume: 10 Issue: 4, Page(s): 2233 - 2243, 2014.

[2] Y. Li, M. Hou, H. Liu, and Y. Liu, "Towards a theoretical framework of strategic decision, supporting capability and information sharing under the context of Internet of Things", in Inf. Technol. Manage, Volume: 13, No: 4, Page(s): 205 C 216, 2012.

[3] L. Tan, and N. Wang, "Future internet: The internet of things", in Proceedings of ICACTE, 2010. [

4] GE, "The rise of industrial big data", 2012.

[5] A. Bougdira, A. Ahaitouf, and I. Akharraz, "Cloud of things-based decision-making process using product's traceability", in Proceedings of IEEE CloudTech, 2016.

[6] C. Yang, S. Lan, W. Shen, G. Q. Huang, X. Wang, and T. Lin, "Towards product customization and personalization in IoT-enabled cloud manufacturing", in Cluster Computing, Volume 20, Issue 2, pages 1717- 1730, 2017.

[7] K. Challapalli, "The Internet of Things: A time series data challenge", IBM, Online: http: //www.ibmbigdatahub.com/blog/internetthings-timeseries-data-challenge, 2014.

[8] I. Influxdata, "Modern IoT Data Platform", Online: https://www.influxdata.com/customers/ iot-data-platform/, 2017.

[9] "Industry Insiders Report", https://www.renesas.com/us/en/about/edgemagazine/global/13-big-data.html

[10] L. Burkhalter, A. Hithnawi, A. Viand, H. Shafagh, and S. Ratnasamy, "TimeCrypt: Encrypted Data Stream Processing at Scale with Cryptographic Access Control", in Proceedings of NSDI, 2020.

[11] A. Pal and K. Kant, "Smart Sensing, Communication, and Control in Perishable Food Supply Chain", in ACM ToSN, Vol 1, Issue 1, 2020, Pages 1-33.

[12] S. Qi, Y. Zheng, M. Li, Y. Liu, and J. Qiu, "Scalable Industry Data Access Control in RFID-Enabled Supply Chain", in ACM/IEEE ToN, Vol PP, Issue 99, 2016, Pages 1-14.

[13] Y. Zhang, R. H. Deng, D. Zheng, J. Li, P. Wu, and J. Cao, "Efficient and Robust Certificateless Signature for Data Crowdsensing in CloudAssisted Industrial IoT", in IEEE Transactions on Industrial Informatics, Volume: 15, Issue: 9, 2019.

[14] J.S Fu, Y. Liu, H.C. Chao, B. K. Bhargava, and Z.J. Zhang, "Secure Data Storage and Searching for Industrial IoT by Integrating Fog Computing and Cloud Computing", in IEEE Transactions on Industrial Informatics, Volume: 14, Issue: 10, 2018.

[15] X. Wang, Y. Qi, Z. Wang, Y. Chen, and Y. Zhou, "Design and implementation of secpod: A framework for virtualization-based security systems", in IEEE Transactions on Dependable and Secure Computing, Volume: 16, Issue: 1, 2019.

[16] J. Yan, Y. Qi, and Q. Rao, "Detecting malware with an ensemble method based on deep neural network", in Proceedings of Security and Communication Networks, 2018.

[17] M. Ma, D. He, N. Kumar, K. R. Choo, and J. Chen, "Certificateless Searchable Public Key Encryption Scheme for Industrial Internet of Things", in IEEE Transactions on Industrial Informatics, Volume: PP, Issue: 99, DOI: 10.1109/TII.2017.2703922, 2017.

[18] Y. Xu, L. Wang, B. Xu, W. Jiang, C. Deng, F. Ji, and X. Xu, "An information integration and transmission model of multi-source data for product quality and safety", in Information Systems Frontiers, 2016.

[19] H. Cai, L. D. Xu, B. Xu, C. Xie, S. Qin, and L. Jiang, "IoTBased Configurable Information Service Platform for Product Lifecycle Management", in IEEE Transactions on Industrial Informatics, Volume: 10, Issue: 2, Page(s): 1558-1567, 2014.

[20] A. Kyrola, G. Blelloch, and C. Guestrin, "GraphChi: Large-Scale Graph Computation on Just a PC", in Proceedings of OSDI, 2012.

Vol 12, Issue 05, MAY /2021
ISSN NO:0377-9254

[21] X. Xie, H. Ma, J. Li, and X. Chen, "An efficient ciphertext-policy attribute-based access control towards revocation in cloud computing", in Journal of Universal Computer Science, Volume: 19, Issue: 16, Page(s): 2349C2367, 2013.