International Journal of Recent Trends in Engineering & Research
ISSN (ONLINE) : 2455 – 1457
IMPACT FACTOR : 4.101

EFFICIENT WAY TO STORE AND RETRIEVE DATA IN AN INDUSTRIAL IoT IN CLOUD COMPUTING

A.Ancy Juliet[1](PG Scholar)
Guided by: Ms.S.Abarna[2](Assistant Professor/CSE)
CHANDY COLLEGE OF ENGINEERING, TUTICORIN TAMIL NADU, INDIA

Abstract-With the fast development of Industrial Internet of Things (IIoT), a large amount of data is being generated continuously by different sources. Storing all the raw data in the IoT devices locally is unwise considering that the end de-vices' energy and storage spaces are strictly limited. In addition, the devices are unreliable and vulnerable to many threats be- cause the networks may be deployed in remote and unattended areas. In proposed system we discuss the emerging challenges in the aspects of data processing, secure data storage, efficient data retrieval and dynamic data collection in IoT. Then, we design a flexible and economical framework to solve the problems above by integrating the fog computing and cloud computing. Based on the time latency requirements, the collected data are processed and stored in the cloud server. Specifically, all the raw data are first preprocessed and then the time-sensitive data (e.g., control information) are used and stored locally. The non-time-sensitive data (e.g., monitored data) are transmitted to the cloud server to support data retrieval and mining in the future.

Index Terms: Industrial internet of things, secure data storage and retrieval, fod computing, cloud computing.

INTRODUCTION

The Internet of Things (IoT) era, terabytes of data with different sources and structures are being produced worldwide per day. The generated data of IIoT are of great value and they can be used to run the networks or extract knowledge and rules. How to process, store and manage the data securely and efficiently is a great challenge. Fortunately, fog computing and cloud computing provide us an opportunity to solve these problems properly. Fog is close to the networks, i.e., the sources of the data, and it can access the data in a time- efficient manner. Consequently, the time- limited data should be processed and stored locally to run the network normally. However, storing all the data in the edge servers is unwise considering the low stability and reliability.

Moreover, retrieving and mining the data stored by numerous edge servers in a distributed manner is impractical. Cloud computing is treated as a promising IT infrastructure, which can gather and organize huge IT resources to support on-demand access service in a flexible and economical manner. Pushed by the data storage requirement of IIoT and attracted by these excellent features of cloud computing, an intuitive approach is outsourcing then on-time- sensitive data to the cloud .while guaranteeing both the security and search ability of the data. Note that, though quite a large portion of the data is stored in the cloud, the whole system needs to employ the edge server as a fundamental tool. In fact, cloud computing and edge computing are interdependent with each other and they together form a service continuum between the cloud and the end devices of IIoT.

In this paper, explain about data processing framework for IIoT by integrating the functions of data preprocessing, storage and retrieval based on both the fog computing and cloud computing. Cloud computing is the delivery of computing services—servers, storage, databases, networking, software, analytics, intelligence and more—over the Internet ("the cloud") to offer faster innovation,

flexible resources and economies of scale. It typically pay only for cloud services you use, helping lower your operating costs, run your infrastructure more efficiently and scale as your business needs change. Cloud computing is a big shift from the traditional way businesses think about IT resources. Here are seven common reasons organizations are turning to cloud computing services. Cloud computing eliminates the capital expense of buying hardware and software and setting up and running on-site datacenters—the racks of servers, the round-the-clock electricity for power and cooling, the IT experts for managing the infrastructure. It adds up fast. Most cloud computing services are provided self service and on demand, so even vast amounts of computing resources can be provisioned in minutes, typically with just a few mouse clicks, giving businesses a lot of flexibility and taking the pressure off capacity planning.

The benefits of cloud computing services include the ability to scale elastically. In cloud speak, that means delivering the right amount of IT resources for example, more or less computing power, storage, bandwidth—right when it is needed and from the right geographic location. On-site datacenters typically require a lot of "racking and stacking "hardware set up, software patching and other time- consuming IT management chores. Cloud computing removes the need for many of these tasks, so IT teams can spend time on achieving more important business goals. The biggest cloud computing services run on a worldwide network of secure datacenters, which are regularly upgraded to the latest generation of fast and efficient computing hardware. This offers several benefits over a single corporate datacenter, including reduced network latency for applications and greater.

IoT data storage schemes have been designed based on cloud computing in the literatures. It proposed a data storage framework enabling efficient storage of both structured and unstructured data. The framework combines and extends multiple existing databases such as Hadoop, NoSQL database and relational database to store and manage diverse types of IoT data. The data users can access the stored data through the interfaces provided by the cloud server. However, a disadvantage of this framework is the long latency which is an inherent property of cloud-based data storage schemes.

A framework including frontend layer, middle layer and backend layer is proposed to seamlessly integrate IoT data storage schemes to existing enterprise information systems. This method can be easily accepted by the data owners considering that existing information systems are mature. To store a huge amount of heterogeneous data, a hybrid approach is proposed to optimize data storage and retrieval which couples the document and object-oriented strategies. Moreover, some implementation details are also discussed. Designed a polynomial time algorithm for efficiently down loading the packages from the cloud to the IoT devices. This approach can compute the amount of power allocation based on buffer backlog and the state of communication links to improve the overall performance. Except for cloud computing, the fog computing technology is also employed to store and share the data in IoT.

To support the latency sensitive data processing and storage, an efficient data sharing scheme that allows smart devices to share the data with others at the edge of the IoT is proposed. In addition, the data users can search and retrieve interested data by keywords and their secret keys. Simulation result demonstrates that the proposed scheme has the potential to be effectively used in the IoT. However, the size of the network is strictly limited in the scheme and in addition it is impractical to store a large amount of data for further processing and mining considering the efficiency and security problems. The proposed in this paper, some other schemes also attempt to combine the fog computing and cloud computing to improve the quality of service in terms of latency, security and flexibility discussed the advantages of cloud computing and edge computing, respectively. The cloud computing can construct a shared pool of computing and storage resources and the edge computing can process the data in real time.

By combining these two techniques, the proposed framework can obtain the network wide knowledge by exploiting the historical information stored in the cloud center and the knowledge can be used to guide the edge computing to satisfy various performance requirements of heterogeneous IoT networks. An attribute based

encryption scheme is proposed in to make full use of edge servers. The collected data are first encrypted by theedge server before being outsourced to the cloud server. Experimental results illustrate that the edge servers bear a large portion of the workload. However, this scheme doesn't support efficient data search and hence the functionalities are limited took the lessons of designing an operating system from the long history of operation systems and designed a distributed operation system specifically for the IoT, i.e., FogOS, which can manage both the cloud re-sources and fog resources. In addition, FogOS is also a platform of incentivizing and connecting individually owned IoT devices.

EXISTING SYSTEM

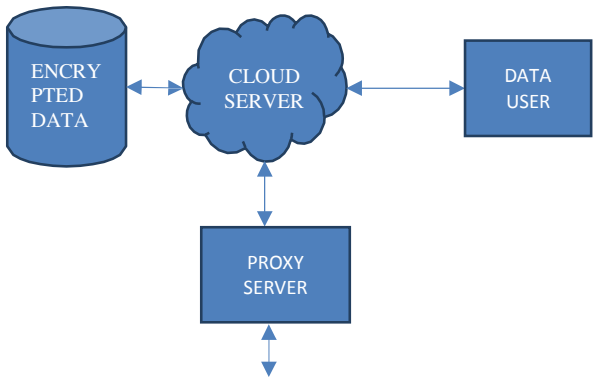
The emerging of IoT and cloud computing, many IoT data storage schemes have been designed based on cloud computing in the literatures. It proposed a data storage framework enabling efficient storage of both structured and unstructured data. The framework combines and extends multiple existing databases such as Hadoop, No SQL database and relational database to store and manage diverse types of IoT data. The data users can access the stored data through the interfaces provided by the cloud server. An attribute-based encryption scheme is proposed to make full use of edge servers. The collected data are first encrypted by the edge server before being outsourced to the cloud server. Experimental results illustrate that the edge servers bear a large portion of the workload. However, this scheme doesn't sup- port efficient data search and hence the functionalities are limited.

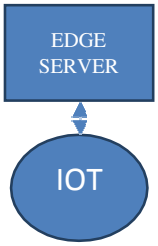
PROPOSED SYSTEM

In proposed system, we design a data processing framework for IIoT by integrating the functions of data preprocessing, storage and retrieval based on both the fog computing and cloud computing. The overall data processing system of IIoT consists of five main entities IIoT, Edge server, Proxy server, Cloud server and Data users. The IIoT continuously collects data from physical environments and then sends the data to the edge server.

The time-sensitive data are first extracted and processed by the edge server and then the data will be dropped if they will not be used in the future. However, some archived data need to be preprocessed and uploaded to the cloud server for storage and retrieval. The proxy server is responsible for improving the quality of the data generated by a set networks and making the data suitable for being stored in the cloud server. Moreover, the data need to be encrypted by the proxy server while maintaining both the security and search ability when an authorized data user wants to obtain some specific historical data, he just needs to build a trapdoor with the help of the proxy server and then send the trapdoor to the cloud server. Based on the trapdoor, the cloud server searches the encrypted index structures by a search engine to get the result and sends the encrypted data to the data user. At last, the data user de-crypts the search result to get the plaintext data.

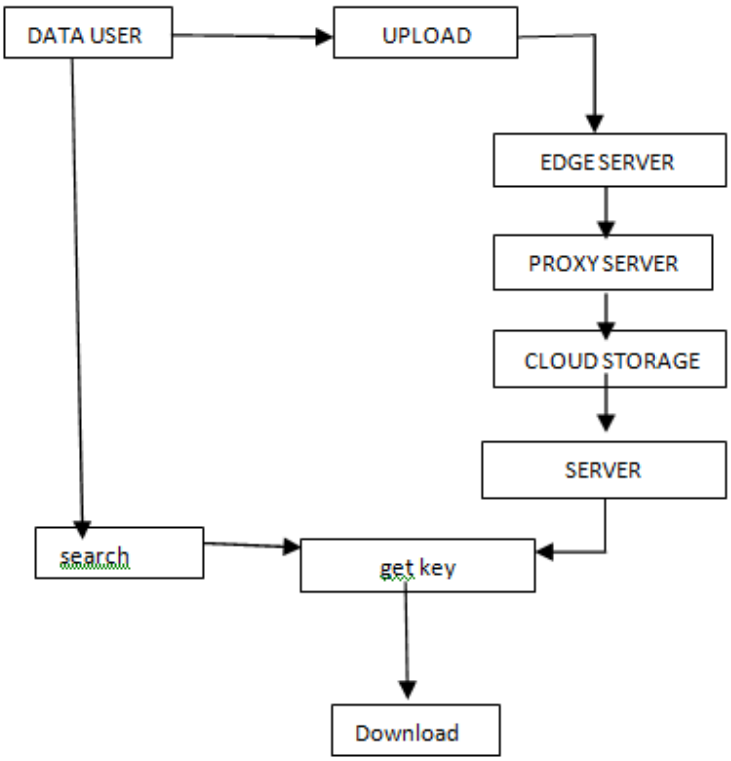
BLOCK DIAGRAM





BLOCK DIAGRAM DESCRIPTION

The IIoT continuously collects data from physical environments and then sends the data to the edge server. The time sensitive data are first extracted and processed by the edge server and then the data will be dropped if they will not be used in the future. However, some archived data need to be preprocessed and uploaded to the cloud server for storage and retrieval. The proxy server is responsible for improving the quality of the data generated by a set networks and making the data suitable for being stored in the cloud server. Moreover, the data need to be encrypted by the proxy server while maintaining both the security and search ability. When an authorized data user wants to obtain some specific historical data, he just needs to build a trapdoor with the help of the proxy server and then send the trapdoor to the cloud server.



DATA FLOW DIAGRAM AND DESCRIPTION

Running a public auditing system consists of two phases, Setup and Audit process in the proposed scheme of our project Setup: The user initializes the public and secret parameters of the system by executing Key Gen, and preprocesses the data file F by using Sig Gen to generate the verification metadata.

MODULES DESCRIPTION

USER AUTHENTICATION

User authentication is a process that allows a device to verify the identity of someone who connects to a network resource. There are many technologies currently available to a network administrator to authenticate users. The process of identifying an individual usually based on a username and password. In security systems, authentication is distinct from authorization, which is the process of giving individuals access to system objects based on their identity. In this module, the participants who entered into the cloud should be registered. In registration process user has to input their details like user name, email address and the password they want to access. They cannot access directly. User has to wait for the access from cloud. That is server must give the permission to access the cloud. Through this process our authentication process is very efficient. When a user wants to log in, then the user first types his/her details.

FILE UPLOAD

Uploading is the transmission of a file from one computer system to another, usually larger computer system. From a network user's point-of-view, to upload a file is to send it to another computer that is set up to receive it. User can upload the file into the cloud through edge server and proxy server. An edge server is any server that resides on the “edge” between two networks, typically between a private network and the internet. Edge servers can serve different purposes depending on the context. A proxy server acts as a gateway between user and the internet. It's an intermediary server separating end users from the websites they browse. Proxy servers provide varying levels of functionality, security, and privacy depending on your use case, needs, or company policy. The time- sensitive data are first extracted and processed by the edge server and then the data will be dropped if they will not be used in the future. Archived data need to be preprocessed and uploaded to the cloud server for storage and retrieval. The proxy server is responsible for improving the quality of the data generated by a set networks and making the data suitable for being stored in the cloud server. Moreover, the data need to be encrypted by the proxy server while maintaining both the security and search ability.

CLUSTER

Clustering is the process of grouping similar data. Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing. Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns. In the field of biology, it can be used to derive plant and animal taxonomies, categorizgenes with similar functionalities and gain insight into structures inherent to populations.
Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location. Clustering also helps in classifying documents on the web for information discovery. Clustering is also used in outlier detection applications such as detection of credit card fraud. As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster. In this module the uploaded file will be clustered depends on the similar data which is country and other data which is presented in the dataset.

TRAPDOOR GENERATION

Trap doors in a computer system (or cryptosystem or algorithm) is a method of bypassing normal authentication, securing remote access to a computer, obtaining access to plaintext, and so on, while attempting to remain undetected. The trap doors may take the form of an installed program (e.g., Back Orifice) or may subvert the system through a root kit.

KEY GENERATION

Key management refers to management of cryptographic keys in a cryptosystem. This includes dealing with the generation, exchange, storage, use, crypto-shredding (destruction) and replacement of keys. It includes cryptographic protocol design, key servers, user procedures, and other relevant protocols. Key management concerns keys at the user level, either between users or systems. This is in contrast to key scheduling, which typically refers to the internal handling of keys within the operation of a cipher. In this module the cloud will generate a key using AES (Advanced Encryption Standard) algorithm.AES, or Advanced Encryption Standards, is a cryptographic cipher that is responsible for a large amount of the information security. The AES cipher is part of a family known as block ciphers, which are algorithms that encrypt data on a per-block basis.AES is considered in crackable by itself. Even most implementations are considered safe. (no side-channels).OTR protocol uses as symmetric encryption algorithm and the world's best funded intelligence agency NSA wasn't able to decrypt it. The key is used for authentication purpose. The generated key will send to the user to decrypt their data.

DATA SEARCH

Computing services will be able to encrypt documents to keep them safe in the cloud. Data search is the process of searching the data in a cloud for searching the data user must get the key. Getting the key is the only way to get the decrypted file. if not they will only get the encrypted file. Encryption and decryption of the file uses the AES algorithm and the key is also generated by the AES (Advanced Encryption Standard) technique. In this Module user can search for their uploaded file which is stored in the cloud. They will get the decrypted file after the key generated by the AES in the server.

CONCLUSION

In proposed system, a secure, flexible and efficient data storage and retrieval system is designed based on both the fog com-putting and cloud computing techniques. The main challenges in terms of data refinement, data organization, searchable encryption and dynamic data collection are summarized, and corresponding solutions are also provided. Specifically, are trivial features tree is designed to support efficient and accurate data retrieval and an index encryption scheme is proposed to support privacy-preserving data search. A flowchart including data mining and remote control is also presented from a wider view.

REFERENCES

[1] Fu J S, Liu Y, Chao H C, et al. Green alarm systems driven by emergencies in industrial wireless sensor networks[J].IEEE Communications Magazine, 2016, 54(10):16-21.

[2] Li L. Technology designed to combat fakes in the global supply chain[J]. Business Horizons, 2013, 56(2):167-177

[3] Mollah M B, Azad M A K, Vasilakos A. Secure Data Sharing and Searching at the Edge of Cloud-Assisted Internet of Things[J]. IEEE Cloud Computing, 2017, 4(1):34-42.

[4] Qing Li, Ze-yuan Wang, Wei-hua Li, et al. Applications integration in a hybrid cloud computing environment: modeling and platform[J]. Enterprise Information Systems, 2013, 7(3):237-271.

[5] Ren, Kui, Cong Wang, and Qian Wang. "Security challenges for the public cloud." IEEE Internet Computing 16.1 (2012): 69-73.

[6] Shancang Li, LidaXu, Xinheng Wang, et al. Integration of hybrid wireless networks in cloud services oriented enterprise information systems[J]. Enterprise Information Systems, 2012, 6(2):165-187.
[7] Tao F. A methodology towards virtualisation-based high performance simulation platform supporting multidisciplinary design of complex products[J]. Enterprise Information Systems, 2012, 6(3):267-290.

Xu L D. Enterprise Systems: State-of- the-Art and Future Trends [J]. IEEE Transactions on Industrial Informatics, 2011, 7(4):630-640.

Xu L D. Introduction: Systems Science in Industrial Sectors[J]. Systems Research & Behavioral Science, 2013, 30(3):211–213.

Yi S, Hao Z, Qin Z, et al. Fog Computing: Platform and Applications[C]// Third IEEE Workshop on Hot Topics in Web Systems and Technologies. IEEE Computer Society, 2015:73-78.

Zhang, W., Zhang, Z., & Chao, H. C. Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management IEEE Communications Magazine, 2017, 55(12), 60-67.

