Volume 3, Issue 5, May-2016, pp. 361-368 ISSN (O): 2349-7084

International Journal of Computer Engineering In Research Trends

Available online at: www.ijcert.org

A Scalable and Reliable Matching Service for Content-based Publish/Subscribe Systems

[1]Mrs.B.PRATHYUSHA,[2]K.ASHOK KUMAR

[1] Pursuing M.Tech(CSE)from Jagruti Institute of Engineering and Technology

[2] Assistant Professor, Department of Computer Science and Engineering, Jagruti Institute of Engineering and Technology, Telangana State, India.

Abstract: Security is one of the extensive and complicated requirements that need to be provided in order to achieve few issues like confidentiality, integrity and authentication. In a content-based publish/subscribe system, authentication is difficult to achieve since there exists no strong bonding between the end parties. Similarly, Integrity and confidentiality needs arise in published events and subscription conflicts with content-based routing. The basic tool to support confidentiality, integrity is encryption. In this paper, we propose SREM, a scalable and reliable event matching service for content-based pub/sub systems in cloud computing environment. To achieve low routing latency and reliable links among servers, we propose a distributed overlay SkipCloud to organize servers of SREM. Through a hybrid space partitioning technique HPartition, large-scale skewed subscriptions are mapped into multiple subspaces, which ensures high matching throughput and provides multiple candidate servers for each event. Moreover, a series of dynamics maintenance mechanisms are extensively studied.

Keywords— Pairing-based cryptography, Key server, Credential, Publish/Subscribe.

1. INTRODUCTION

Common requirement for any system is security. The need for security must be extremely high. It is one of the major requirements to protect or control any sort of failures. There are number of mechanisms which are available to provide security. In that one of the most important mechanisms is encryption. In cryptography encryption is the process of converting plain text to cipher text which is unreadable from unauthorized users. The cryptography mechanism is required in publish/subscribe system. In publish/subscribe system publisher is one who publishes his content without specifying a particular destination to reach publisher will not program the documents to be delivered to a particular subscriber. Publisher will classify publishing documents based on different criteria and release it and subscriber will show interest on one or more documents and subscribe to that particular one in order to have access over it. This publish/subscribe system is traditionally carried out in broker-less [12] content based routing which forwards or routes the message based on the content of the message instead of clearly routing to an specified destination.

Content based routing applies some set of rules to It's content to find the users who are interested in its content. Its different nature is helpful for huge-level scattered applications and also provides a high range of flexibility and adaptability to change. Authorized publisher have permission to publish events in the network and similarly subscribers who likes the content can gets subscribed to a particular published content and have access over it by which high level access control [7] can be achieved. Here published content should not be exposed to routing infrastructure and subscribers should receive content without leaking subscription identity to the system, which is a highly challenging task which needs to be carried out in content-based pub/sub system. Publisher and subscriber are the two entities and they do not trust each other. Even though authorized publisher publish events, nasty publisher pretend to be the real publisher and may spam the network with fake and duplicate contents similarly subscribers are very much eager to find other users and publishers which are challenging tasks. Finally, Transport Layer Security (TLS) or Secure Socket Layer (SSL) is secure channels for distributing keys from key server to the required. Existing security approach deals with traditional network and security is based on restricted manner which tells about key word matching [8]. Key management was the challenging task in the existing approach, so to overcome all these, we use new approach called pairing-based cryptography mechanism, which helps in mapping between to end parties so called cryptographic groups. Here, Identity Based Encryption Technique (IBE) [9] is used under this mechanism. New approach IBE provide greater concern towards authentication and confidentiality in the network. Our approach permit users to preserve credentials based on their subscriptions. Secret keys provided to the users are labeled with the credentials. In Identity-based encryption (IBE) mechanisms 1) key can be used to decrypt only if there is match between credentials with the content and the key; and 2) to permit subscribers to check the validity of received contents. Moreover, this approach helps in providing fine-grained key management, effective encryption, decryption operations and routing is carried out in the order of subscribed attributes.

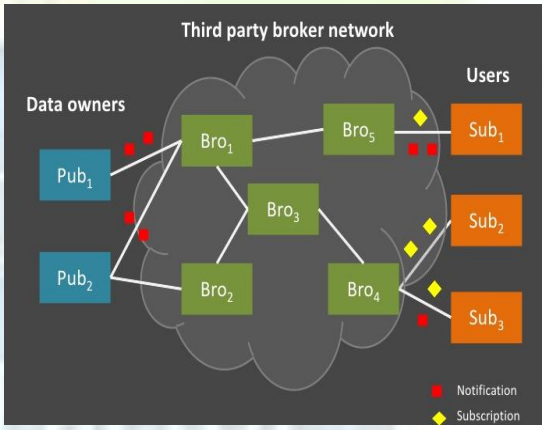

Fig 1. Subscriber/Publisher System

II. RELATED WORK

There are two entities in the System publishers and subscribers. Both the entities are computationally bounded and do not trust each other. Moreover, all the peers (publishers or subscribers) participating in the pub/sub overlay network are honest and do not deviate from the designed protocol. Likewise, authorized publishers only allow valid events in the system. However, malicious publishers may masquerade the authorized publishers and spam the overlay network with fake and duplicate events. We do not intend to solve the digital copyright problem; therefore, authorized subscribers do not reveal the content of

successfully decrypted events to other subscribers.

A. Publisher subscriber technique Publishers and subscribers interact with a key server. They provide credentials to the key server and in turn receive keys which fit the expressed capabilities in the credentials. Subsequently, those keys can be used to encrypt, decrypt, and sign relevant messages in the contentbased pub/sub system, i.e., the credential becomes authorized by the key server. A credential consists of two parts: 1) a binary string which describes the capability of a peer in publishing and receiving events, and 2) a proof of its identity [1].

B. Identity based encryption Identity(ID)-based public key cryptosystem, which enables any pair of users to communicate securely without exchanging public key certificates, without keeping a public key directory, and without using online service of a third party, as long as a trusted key generation center issues a private key to each user when he first joins the network [2].

C. Identity Handling: Identification provides an essential building block for a large number of services and functionalities in distributed Information systems. In its simplest form, identification Is used to uniquely denote computers on the Internet By IP addresses in combination with the Domain Name System (DNS) as a mapping service between symbolic Names and IP addresses. Thus, computers can conveniently Be referred to by their symbolic names, whereas, in The routing process, their IP addresses must be used.[3] Higher-level directories, such as X.500/LDAP, consistently Map properties to objects which are uniquely identified by Their distinguished name (DN), i.e., their position in the X.500 tree [4].

D. Content based publish/subscribe: Content-based networking is a generali- zation of the content based publish/subscribe model. [4] In content-based networking, messages are no longer addressed to the communication end-points . Instead, they are published to a distributed information space and routed by the networking sub -strate to the "interested" communication end-points. In most cases, the same substrate is responsible for realizing naming, binding and the actual content delivery [5].

E. Secure Key Exchange: A key-exchange (KE) protocol is run in a network of interconnected parties where each party can be activated to run an instance of the protocol called a session [6]. Within a session a party can be activated to initiate the session or to respond to an incoming message. As a result of these activations, and according to the specification of the protocol, the party creates and maintains a session state, generates outgoing messages, and eventually completes the session by outputting a session-key and erasing the session state [7].

III.SYSTEM STUDY

EXISTING SYSTEM:

A number of pub/sub services based on the cloud computing environment have been proposed, However, most of them can not completely meet the requirements of both scalability and reliability when matching large-scale live content under highly dynamic environments. This mainly stems from the following facts: Most of them are inappropriate to the matching of live content with high data dimensionality due to the limitation of their

subscription space partitioning techniques, which bring either low matching throughput or high memory overhead. These systems adopt the one-hop lookup technique among servers to reduce routing latency. In spite of its high efficiency, it requires each dispatching server to have the same view of matching servers. Otherwise, the subscriptions or events may be assigned to the wrong matching servers, which bring the availability problem in the face of current joining or crash of matching servers. Matching servers. Otherwise, the subscriptions or events may be assigned to the wrong matching servers, which bring the availability problem in the face of current joining or crash of matching servers.

Disadvantage:

- Lower rate of scalability and reliability of event matching.
- High routing Latency.

PROPOSED SYSTEM:

we propose a scalable and reliable matching service for content-based pub/sub service in cloud computing environments, called SREM. Specifically, we mainly focus on two problems: one is how to organize servers in the cloud computing environment to achieve scalable and reliable routing. The other is how to manage subscriptions and events to achieve parallel matching among these servers. We propose a distributed overlay protocol, called Skip Cloud, to organize servers in the cloud computing environment. Skip Cloud enables subscriptions and events to be forwarded among brokers in a scalable and reliable manner. Also it is easy to implement and maintain.

Advantage:

- High scalability and reliability of event matching.
- Reducing the optimal routing latency.

PROBLEM STATEMENT:

The proposed event matching service can efficiently filter out irrelevant users from big data volume, there are still a number of problems we need to solve. Firstly, we do not provide elastic resource provisioning strategies in this paper to obtain a good performance price ratio.

SCOPE:

Scope is to design and implement the elastic strategies of adjusting the scale of servers based on the churn workloads. Secondly, it does not guarantee that the brokers disseminate large live content with various data sizes to the corresponding subscribers in a real-time manner. For the dissemination of bulk content, the upload capacity becomes the main bottleneck. Based on our proposed event matching service, we will consider utilizing a cloud-assisted technique to realize a general and scalable data dissemination service over live content with various data sizes.

III. SYSTEM ARCHITECTURE

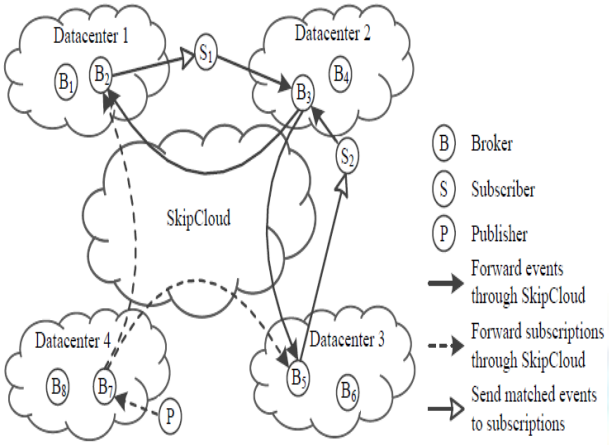

Fig. 2. System Architecture.

MODULE DESCRIPTION:

Number of Modules

After careful analysis the system has been identified to have the following modules:

1. Scalable and Reliable Event Matching.
2. Skip Cloud Performance.
3. Hybrid multidimensional partition Technique.
4. Publisher/Subscriber Module.

1. Scalable And Reliable Event Matching:

All brokers in SREM as the front-end are exposed to the Internet, and any subscriber and publisher can connect to them directly. To achieve reliable connectivity and low routing latency, these brokers are connected through an distributed overlay, called SkipCloud. The entire content space is partitioned into disjoint subspaces, each of which is managed by a number of brokers. Subscriptions and events are dispatched to the subspaces that are overlapping and events falling into the same subspace are matched on the same broker. After the matching process completes, events are broadcasted to the corresponding interested subscribers.

2. SkipCloud Performance:

SkipCloud organizes all brokers into levels of clusters. At the top level, brokers are organized into multiple clusters whose topologies are complete graphs. Each cluster at this level is called top cluster. It contains a leader broker which generates a unique b-ary identifier with length using a hash function cluster are responsible for the same content subspaces, which provides multiple matching candidates for each event. Since brokers in the same top cluster generate frequent communication among themselves, such as updating subscriptions and dispatching events, they are organized into a complete graph to reach each other in one hop. After the top clusters have been well organized, the clusters at the rest levels can be generated level by level.. This identifier is called ClusterID.

3.Hybrid multidimensional partition Technique:

achieve scalable and reliable event matching among multiple servers, we propose a hybrid multi-dimensional space partitioning technique, called HPartition. It allows similar subscriptions to be divided into the same server and provides multiple candidate matching servers for each event. Moreover, it adaptively alleviates hot spots and keeps workload balance among all servers. HPartition divides the entire content space into disjoint subspaces. Subscriptions and events with overlapping subspaces are dispatched and matched on the same top cluster of SkipCloud. To keep workload balance among servers, HPartition divides the hot spots into multiple cold spots in an adaptive manner.

4. Publisher/Subscriber:

Each subscriber establishes affinity with a broker (called home broker), and periodically sends its subscription as a heartbeat message to its home broker. The home broker maintains a timer for its every buffered subscription. If the broker has not received a heartbeat message from a subscriber over Tout time, the subscriber is supposed to be offline. Next, the home broker removes this subscription from its buffer and notifies the brokers containing the failed subscription to remove it.

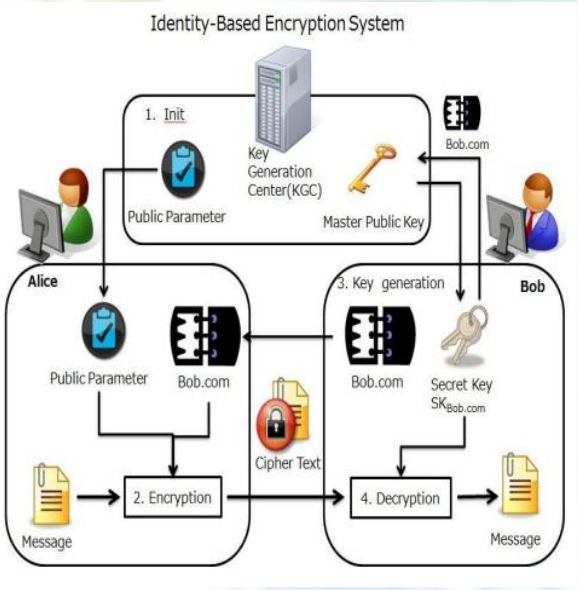

Fig 3. Encryption Mechanism

D. Certificate based encryption

Certificate-based cryptography (CBE) is formal security model,it concerned 2 entities that\'s certifier and a shopper. Definition of CBE somewhat almost like the powerfully key-insulated cryptography and in distinction

E. Advanced Encryption Standard (AES)

AES is regular block cipher that\'s supposed to switch DES because the approved customary for wide selection of application. In AES, Cipher takes a plaintext block size 128 bits or sixteen bytes. during this algorithmic rule key length is sixteen,24 or thirty two bytes. The input to the cryptography and coding algorithmic rule may be a single 128 bits block. AES have classic Feistel Structure, half the information block is employed to switch the opposite half the information block and so the halves are swapped. The structure is sort of easy for each cryptography and coding. The cipher begins with AN AddRoundKey Stage, followed by 9 rounds that every includes all four stages, followed by tenth spherical of 3 stages. Solely the AddRoundKey stages create use of the key. For this reason, the cipher begins and ends with AN AddRoundKey stages. Every stage during

V. CONCLUSION

In this paper, we have presented broker-less approach in content based publish subscribe system for providing authentication and confidentiality. The approach is extremely good for number of subscribers and publishers in the system and the number of keys maintained by them. The keys will be in cipher text format which are labeled with credentials assigned to publishers and subscribers. This paper introduces SREM, a scalable and reliable event matching service for content-based pub/sub systems in cloud computing environment. SREM connects the brokers through a distributed overlay Skip- Cloud, which ensures reliable connectivity among brokers through its multi-level clusters and brings a low routing latency through a prefix routing algorithm. Through a hybrid multi-dimensional space partitioning technique, SREM reaches scalable and balanced clustering of high dimensional

skewed subscriptions, and each event is allowed to be matched on any of its candidate servers. Extensive experiments with real deployment based on a CloudStack testbed are conducted, producing results which demonstrate that SREM is effective and practical, and also presents good workload balance, scalability and reliability under various parameter settings.

REFERENCES

[1] Muhammad Adnan Tariq, Boris Koldehofe, and Kurt Rothermel "Securing broker-less publish/subscribe systems using identity-based encryption"IEEE Transactions On Parallel And Distributed Systems,Vol. 25, No. 2, February 2014.

[2] M. Ion, G. Russello, and B. Crispo, "Supporting Publication and Subscription Confidentiality in Pub/Sub Networks," Proc. Sixth Int'l ICST Conf. Security and Privacy in Comm. Networks (SecureComm), 2010.

[3] L.I.W. Pesonen, D.M. Eyers, and J. Bacon, "Encryption-Enforced Access Control in Dynamic Multi-Domain Publish/Subscribe Networks," Proc. ACM Int'l Conf. Distributed Event-Based Systems (DEBS), 2007.

[4] P. Pietzuch, "Hermes: A Scalable Event-Based Middleware," PhD dissertation, Univ. of Cambridge, Feb. 2004.

[5] M. Srivatsa, L. Liu, and A. Iyengar, "EventGuard: A System Architecture for Securing Publish-Subscribe Networks," ACM Trans. Computer Systems, vol. 29, article 10, 2011.

[6] A. Shikfa, M. O ¨ nen, and R. Molva, "PrivacyPreserving Content-Based Publish/Subscribe Networks," Proc. Emerging Challenges for Security, Privacy and Trust, 2009.

[7] J. Bacon, D.M. Eyers, J. Singh, and P.R. Pietzuch, "Access Control in Publish/Subscribe Systems," Proc. Second ACM Int'l Conf. Distributed Event-Based Systems (DEBS), 2008.

[8] D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public Key Encryption with Keyword Search," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques on Advances in Cryptology (EUROCRYPT), 2004.

[9] D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," Proc. Int'l Cryptology Conf. Advances in Cryptology, 2001.

[10] H. Khurana, "Scalable Security and Accounting Services for Content-Based Publish/Subscribe Systems," Proc. ACM Symp. Applied Computing, 2005.

[11] C. Raiciu and D.S. Rosenblum, "Enabling Confidentiality in Content-Based Publish/Subscribe Infrastructures," Proc. IEEE Second CreatNet Int'l Conf. Security and Privacy in Comm. Networks (SecureComm), 2006.

[12] M.A. Tariq, B. Koldehofe, A. Altaweel, and K. Rothermel, "Providing Basic Security Mechanisms in Broker-Less Publish/ Subscribe Systems," Proc. ACM Fourth Int'l Conf. Distributed Eventt- Based Systems (DEBS), 2010.

ABOUT THE AUTHORS

Mrs.B.PRATHYUSHA is pursuing M.Tech degree in, Computer Science and Engineering from Jagruti Institute of Engineering and

Technology, Telangana State, India.

Mr.K.ASHOK KUMAR, presently working as Assistant Professor in, Department of computer science and engineering, Telangana State,India.

