IJSEAT International Journal of Science Engineering and Advance Technology

Analyzation Of The Scalability Problems Of Distributed Presence Server Architectures

R.Sangeetha [1], M.M.Balakrishna [2], K.T.V Subbarao [3]
1 2 3 Department of Computer Science And Engineering
1 2 3 Akula Sree Ramulu institute of Engineering and Technology, prathipadu, Tadepalligudem, A.P,India
[1]E-Mail:sangeethabaasha26@gmail.com,[2] balu_522@yahoo.co.in,[3]ogidi@rediffmail.com

Abstract:
A mobile presence service is an important factor of social network services in cloud computing environments. The key function of a mobile presence service is to maintain an up-to-date list of presence information of all mobile users. The presence information includes details about a mobile user's location, availability, activity, device capability and preferences. The service must also connect the user's ID to his/her current presence information as well as regain and subscribe to changes in the presence information of the user's friends. We suggest competent and scalable server architecture, called Presence Cloud which facilitates mobile presence services to hold up large-scale social network applications. When a mobile user joins a network Presence Cloud searches for the presence of his/her friends and notifies them of his/her arrival. Presence Cloud categorizes presence servers into a quorum-based server-to-server architecture for efficient presence searching.

Keywords: Social Networks, Mobile presence services, distributed presence servers, cloud computing.

Introduction:

In social network services each mobile user has a friend list normally called a buddy list which includes the contact information of other users that he/she wants to converse with. The mobile user's status is transmitting automatically to each person on the buddy list whenever he/she transits from one status to the other. For illustration when a mobile user logs into a social network application such as an IM system through his/her mobile device the mobile presence service searches for and informs everyone on the user's buddy list. To exploit a mobile presence service's search speed and reduce the notification time most presence services use server cluster technology. A mobile presence service is an necessary component of a social network application because it upholds each mobile user's presence information such as the current status (online/offline), GPS location and network address and also updates the user's online friends with the information repeatedly. If presence updates occur regularly the huge number of messages distributed by presence servers may guide to a scalability problem in a large-scale mobile presence service.

Related Work:

The arrangement of Presence Cloud is a scalable server-to-server architecture that can be used as a building block for mobile presence services. We describe previous researches on presence services, and survey the presence service of existing systems. Well known commercial IM systems leverage some form of centralized clusters to provide presence services. Jennings III *et al.* presented taxonomy of different kind and functions supported by the three most popular IM systems, AIM, Microsoft MSN and Yahoo! Messenger. The authors also provided an impression of the system architectures and observed that the systems use client-server-based architectures. Skype a popular voice over IP application utilizes the Global Index (GI) technology to supply a presence service for users. GI is a multi-tiered network architecture where each node preserves full knowledge of all available users. Since Skype is not an open protocol it is difficult to determine how GI technology is used exactly.

Existing System:

The previous work presented classification of different features and functions supported by the three most popular IM systems, AIM, Microsoft MSN and Yahoo! Messenger. Skype a popular voice over IP application uses the Global Index (GI) technology to provide a presence service for users. They found that the presence information is one of most messaging traffic in instant messaging systems. It shows that the largest message traffic in existing presence services is buddy NOTIFY messages.

Disadvantages:

Presence Cloud does not address the communication security problem and the presence server authentication problem. There are Security problems such as malicious user attacks and the user privacy.

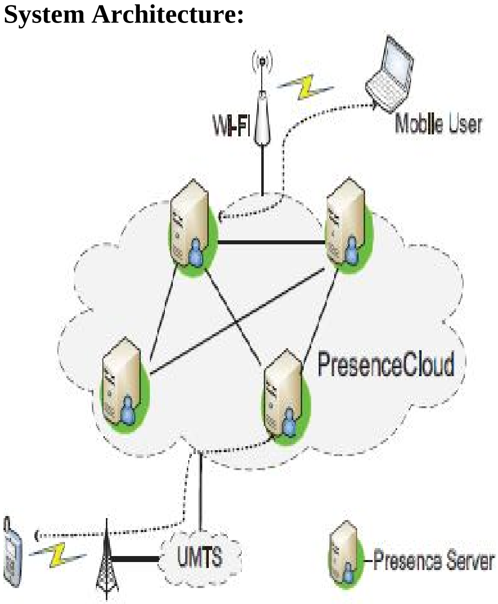
Proposed System:

To maintain presence information clients are organized in a DHT system rather than in a centralized server. The works in study related problems and developing an initial set of guidelines for optimizing inter-domain presence traffic and present a DHT-based presence server architecture. Recently presence services are also integrated into mobile services. However it also suffers scalability problem since it uses a central SIP server to perform presence update of mobile users.

Advantages:

Presence Cloud achieves low search latency and improves the performance of mobile presence services. Presence Cloud achieves major performance gains in terms of the search cost and search satisfaction.

System Architecture:



In the mobile Internet a mobile user can access the Internet and make a data connection to Presence Cloud via 3G or Wi-Fi services. After the mobile user connects and validates him/her to the mobile presence service the mobile user is determinately directed to one of Presence Servers in the Presence Cloud by using the Secure Hash Algorithm such as SHA-1. The mobile user opens a TCP connection to the Presence Server (PS node) for control message transmission mainly for the presence information. After the control channel is recognized the mobile user sends a request to the connected PS node for his/her buddy list searching.

Presence Cloud Server Overlay:

The low-diameter property makes sure that a PS node only needs two hops to reach any other PS nodes. The Presence Cloud server overlay edifice algorithm put in order the PS nodes into a server-to-server overlay which offers a good low-diameter overlay property.

One-Hop Caching Strategy:

In order to become accustomed to changes in the presence of users the caching approach should be asynchronous and not require luxurious mechanisms for distributed agreement. In Presence Cloud each PS node upholds a user list of presence information of the attached users and it is accountable for caching the user list of each node in its PS list. PS nodes only reproduce the user list at most one hop away from itself. The cache is updated when neighbors create connections to it and periodically updated with its neighbors. To progress the competence of the search operation Presence Cloud necessitate a caching approach to reproduce presence information of users.

Directed Buddy Search:

First by systematize PS nodes in a server-to-server overlay network we can therefore use one-hop search precisely for queries and thus decrease the network traffic without major impact on the search results. Second by capitalizing the one-hop caching that preserves the user lists of its neighbors. We get better response time by escalating the chances of finding buddies. The population of mobile users can be regaining by a broadcasting operation in any PS node in the mobile presence service. The buddy list penetrating algorithm of Presence Cloud joined with the two-hop overlay and one-hop caching approach ensures that Presence Cloud can characteristically provide quick responses for a large number of mobile users.

Performance Evolution:

The functioning of the network simulator and the related architectures including a Mesh based, Presence Cloud and Chord-based presence server architecture was written in Java. The packet-level simulator permits us to carry out tests up to

20,000 users and 2,048 PS nodes after which simulation data no longer fit in RAM and makes our experiments difficult. In our experiments the simulator first goes through a warming-up phase to arrive at the network size both PS nodes and users and the simulator starts of the 1,800 seconds test after the measurement approach has stabilized is based on the system size.

Algorithm Used:

Load balancer allocates the work to the clusters of presence server. The several load balancing algorithms for distributing Session Initiation Protocol request to a clusters of presence servers. The load balancer algorithm Transaction Least Work Left is used to allocate work to least values of the servers. It is pool information of the SIP. Two types of state exist in SIP. The first, session state, is created by the INVITE transaction and is destroyed by the BYE transaction. The session-oriented nature of SIP has important implications for load balancing.

Loadbalancer(Req from client,call_id)

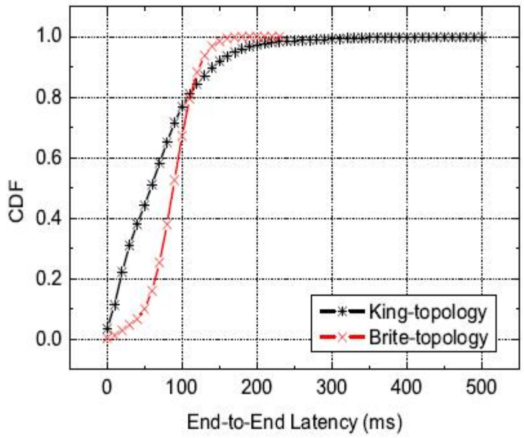
If(Req==INVITE)

- Server S=Execute TLWL Algorithm to select the server
- Forward the request to the server S
- Establish the session between client and server
- Send 200 response to client
- Put entry in the Active table
- Increment the load count of server S
- Update the load in to the table
- Else if(Req==BYE)
- Check if the session is active
- S=Get the server for the session for the call_id
- If(session==Active)
- Terminate the session
- Move the client entry to expired table
- Decrement the load count of the server S
- Update the load in to the table
- Else if(Req= chat or voice chat or upload)
- S=Get the server for the session for the call_id
- If(session==Active)
- Forward the req to server S
- Else
- Throught the exception that INVITE is not provided early
- Else
- Throught the exception that INVITE is not provided early
- END

Algorithm 2

Presence Server(Req from load balancer,call_id)
If(Req==chat)
Get the desination address from the req
Search for destination next hop
Forward the req to the destination next hop
Else
Send error response
END

Experimental Results:



The replicated topology places every PS node at a position on the King-topology or the Brite-topology selected consistently at random. Note that our simulations engage networks of less than 2,048 PS nodes we use a pair wise latency matrix copied from measuring the inter-PS node latencies. And each mobile user also consistently is attached to a random PS node, the broadcast hold-up between mobile user and PS node is randomly assigned in the range [1, 20] (ms). The King-topology is believed as the default IP network topology. Every simulation result is the average 20 runs. The average delay of King-topology is 77.4 milliseconds and 96.2 milliseconds in the Brite topology. Consequently the number of users is set to be 20,000, unless otherwise specified.

Conclusion:

Presence Cloud attains low search latency and improves the presentation of mobile presence services. In addition we discussed the scalability problem in server architecture designs and initiated the buddy-list search problem which is a scalability problem in the distributed server architecture of mobile presence services. Through a simple mathematical replica we show that the total number of buddy search messages augments considerably with the user arrival rate and the number of presence servers. The results of simulations demonstrate that Presence Cloud achieves main presentation gains in terms of the search cost and search satisfaction. Overall Presence Cloud is shown to be a scalable mobile presence service in large-scale social network services.

References:

[1] Facebook, http://www.facebook.com.
[2] Twitter, http://twitter.com.
[3] Foursquare http://www.foursquare.com.
[4] Google latitude, http://www.google.com/intl/enus/latitude/intro.html.
[5] Buddycloud, http://buddycloud.com.
[6] Mobile instant messaging, http://en.wikipedia.org/wiki/Mobile instant messaging.
[7] R. B. Jennings, E. M. Nahum, D. P. Olshefski, D. Saha, Z.-Y. Shae, and C. Waters, "A study of internet instant messaging and chat protocols," *IEEE Network*, 2006.
[8]Gobalindex, http://www.skype.com/intl/en-us/support/user-guides/ p2pexplained/.
[9] Z. Xiao, L. Guo, and J. Tracey, "Understanding instant messaging traffic characteristics," *Proc. of IEEE ICDCS*, 2007.
[10] C. Chi, R. Hao, D. Wang, and Z.-Z. Cao, "Ims presence server: Traffic analysis and performance modelling," *Proc. of IEEE ICNP*, 2008.
[11] Instant messaging and presence protocol ietf working group http://www.ietf.org/html.charters/impp-charter.html.
[12] Extensible messaging and presence protocol ietf working group http://www.ietf.org/html.charters/xmpp-charter.html.
[13] Sip for instant messaging and presence leveraging extensions ietf working group. http://www.ietf.org/html.charters/simplecharter.html.
[14] P. Saint-Andre., "Extensible messaging and presence protocol (xmpp): Instant messaging and presence describes instant messaging (im), the most common application of xmpp," *RFC 3921*, 2004.
[15] B. Campbell, J. Rosenberg, H. Schulzrinne, C. Huitema, and D. Gurle, "Session initiation protocol (sip) extension for instant messaging," *RFC 3428*, 2002.
[16] Jabber, http://www.jabber.org/.
[17] Peer-to-peer session initiation protocol ietf working group, http://www.ietf.org/html.charters/p2psip-charter.html.
[18] K. Singh and H. Schulzrinne, "Peer-to-peer internet telephony using sip," *Proc. of ACM NOSSDVA*, 2005.
[19] P. Saint-Andre, "Interdomain presence scaling analysis for the extensible messaging and presence protocol (xmpp)," *RFC Internet Draft*, 2008.
[20] A. Houri, T. Rang, and E. Aoki, "Problem statement for sip/simple," *RFC Internet-Draft*, 2009.

Ms.R.SANGEETHA is a student of Akula sreeramulu College of Engineering & Tech., Tadepalligudam Presently She is pursuing her M.Tech [CSE] from this college and she received her M.C.A., from Dr.Pauls college of engineering, affiliated to Anna University, chennai in the year 2010. Her area of interest includes Data Mining, Computer Networks and Object oriented Programming languages, all current trends and techniques in Computer Science

Mr. *M.M.BALAKRISHNA* ,well known Author and teacher received M.Tech (CSE) and working as Associate Professor and HoD, Department of CSE, Akula Sree Ramulu institute of Engineering and Technology, He is an active member of ISTE. He has 7 years of teaching experience in various engineering colleges. His area of interest includes Computer Networks, Operating systems and other advances in computer Applications.

Prof. K.T.V Subbarao ,well known Author and teacher received M.Tech (CSE) and working as Principal, Akula SreeRamulu institute of Engineering and Technology, He is an active member of ISTE. He has 12 years of teaching experience in various engineering colleges. To his credit couple of publications both national and international conferences/journals. His area of interest includes cryptography and network security, Distributed databases, Operating systems and other advances in computer Applications.