Peer-to-Peer Netw Appl
DOI 10.1007/s12083-009-0064-4

A survey on peer-to-peer SIP based communication systems

Xianghan Zheng · Vladimir Oleshchuk

Received: 1 May 2009 / Accepted: 23 December 2009

Abstract Recently, both academia and industry have initiated research projects directed on integration of P2PSIP paradigm into communication systems. In this paradigm, P2P network stores most of the network information on each participating peer without help of the central servers. The concept of self-configuration, self-establishment greatly improves the robustness of the network system compared with the traditional Client/Server based systems. In this paper, we survey P2PSIP solutions proposed recently both in the academic and industrial research. We consider technical issues that include Chord overlay topology, P2PSIP session initiation (including enrollment and bootstrap, NAT traversal, message routing, P2PSIP interworking, P2PSIP Client, etc), and security issues. Our survey is based on recent research publications.

Keywords Peer-to-Peer (P2P) · Session Initiation Protocol (SIP) · P2PSIP · Chord · Public Key Infrastructure (PKI) · Pre-Shared Key (PSK) · Chord Secure Proxy (CSP)

1 Introduction

Currently Peer-to-Peer (P2P) computing has attracted great attention in both academia and industry. Compare with the traditional server-based system architecture in which most of the functionality is allocated on the server side, P2P-based computing spreads computing task among all participating peers. This might eliminate (at least reduce) the role of server and therefore provide better robustness for the system.

In the communication field, the most well-known application is Skype [1], which offers free Voice-over-IP (VoIP) and Instant messaging service for computer-to-computer communication and charged service for computer-to-PSTN communication. Also, Skype service has been extended to the mobile world. Many mobile devices today (e.g. Nokia N800 [2], SonyEricsson P1 [3]), have been embedded Wi-Fi based Skype application. According to official statistics [1], the number of Skype users has reached 280 million until Feb, 2009, and the number is still growing fast at the speed of 6 million each month.

Session Initiation Protocol (SIP) is designed to create, modify, and terminate sessions with one or more participants. With concrete characteristics (e.g. simplicity, extensibility, flexibility, etc), SIP is chosen by 3rd Generation Partnership Project (3GPP) as the main protocol for the IP multimedia Subsystem (IMS)-based future All-IP network [4]. SIP with respect to future communication systems is regarded as important as HTTP to the Internet.

P2P computing has begun to infiltrate into SIP communication systems. The SIPpeer project at Columbia University [5] and the SOSIMPLE project at William & Mary College [6] are the first attempt in the study of P2PSIP based communication system. In the following years, P2PSIP research has attracted the great attention from both academia and industry. IETF P2PSIP working group defines the concept and motivation behind P2PSIP [7]: *The concept behind P2PSIP is to leverage the distributed nature of P2P to allow for distributed resource discovery in a SIP network, eliminating (at least reducing) the need for centralized servers*.

In the following sections, we give a survey on P2PSIP based communication systems. Our survey is mainly based

X. Zheng (✉) · V. Oleshchuk
University of Agder,
Grimstad, Norway
e-mail: xianghan.zheng@uia.no

V. Oleshchuk
e-mail: vladimir.oleshchuk@uia.no

Published online: 23 January 2010

on a few typical research projects (e.g. SIPPeer [5], P2PP [8], SIPDHT [9], and dSIP [10], etc) and recent research literature. In Section 2 we introduce the P2PSIP requirement; Section 3 specifies Chord-based P2PSIP overlay and the corresponding improvement; P2PSIP session initiation services, including enrollment and bootstrap mechanism, NAT traversal, message routing, P2PSIP interworking, P2PSIP Client, are described in Section 4. Section 5 is the introduction of security problems and the corresponding solutions. Finally, we include conclusions and open issues in Section 6".

2 Requirement statement

In this Section, we briefly describe the requirement to P2PSIP communication system, according to paper [5, 6, 11–13].

1. Availability, Stability and Efficiency. This is the basic requirement of P2PSIP communication systems.
2. Transport requirement. The designed peer protocol SHOULD deliver P2PSIP messages reliably and efficiently.
3. DHT requirement. The designed peer protocol SHOULD be extensible to accommodate difference among different overlay technologies (e.g. the existing Pastory, Kademlia, etc [7, 8]), including new algorithms that might appear in the future.
4. Interworking requirement. P2PSIP protocol SHOULD interwork with traditional network (e.g. PSTN, etc) and advanced IP networks (e.g. SIP, IMS, etc).
5. NAT Traversal. Since many devices are behind the protection of NAT, the designed peer protocol SHOULD provide efficient NAT traversal mechanisms.
6. P2PSIP Client. The designed peer protocol SHOULD contain the client protocol to support the legacy devices that participate the P2PSIP overlay but do not make contributions due to lack of the support in DHT algorithm or the limitation of devices capability (e.g. energy, CPU processing power, bandwidth, etc).
7. Security requirement. Security mechanisms should be implemented to protect P2PSIP systems from security breaches, such as malicious or faulty peers.

3 Chord-based P2PSIP overlay

One of the best definition of overlay network is given in [14]: *"An overlay network is virtual network of nodes and logical links that is built on top of an existing network with the purpose to implement a network service that is not available in the existing network"*. In the following subsection, we specify Chord-based P2PSIP overlay, which has been suggested as a mandatory overlay to support P2PSIP communication [5, 6, 11, 13].

3.1 Chord-based overlay

In Chord overlay, peers and resources construct a ring, as shown in Fig. 1. In the ring, peer and resource are represented by integer Node ID/Resource ID. Each peer stores a certain amount of <*id*, *value*> pairs, in which *id* is the peer/resource ID, *value* is the peer address information or the data storage. Peer/resource ID is assigned by consistent hashing [15], e.g. SHA-1 algorithm, etc. For instance, the peer ID can be produced by hashing the IP address of the particular peer; and the resource ID can be generated by hashing the data value. The Resource ID is stored in the first peer, whose ID> = Resource ID (see Fig. 2).

Each peer contains a routing table, called Finger table, for storing the routing information records. The Finger table records log N successors where N is the number of peers in the overlay (see Fig. 3). Suppose the space size of overlay is 2^m, for some integer m and the i-th successor ID of a peer with ID P is:

$$Succid(i) = \left(P + 2^{i-1}\right) \bmod 2^m (0 < i \leq m)$$

Each peer contacts periodically its successors for updating the Finger table. It also contacts the predecessor that is the previous peer in the identifier circle. This is useful when a peer leaves the ring and asks the previous peer to update its Finger table.

Chord routes the message by sending messages to the next successor nearest to the destination identifier. Consider an example, when peer 3 is searching peer 28 (Fig. 4.). The peer 3 would first check its finger table records; choose a successor (peer 22) nearest to the destination, and then send a request to this successor. The peer 22 would also check its

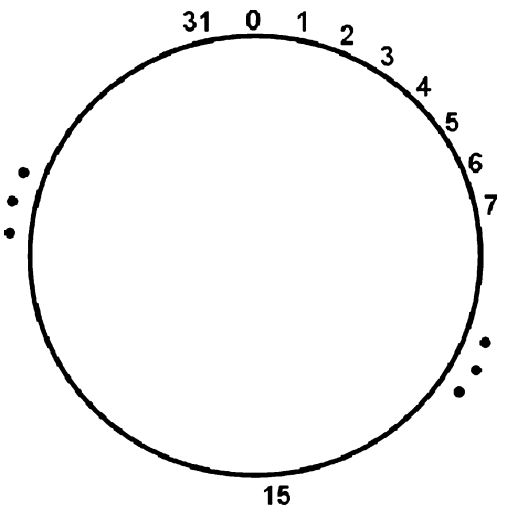

Fig. 1 Chord ring

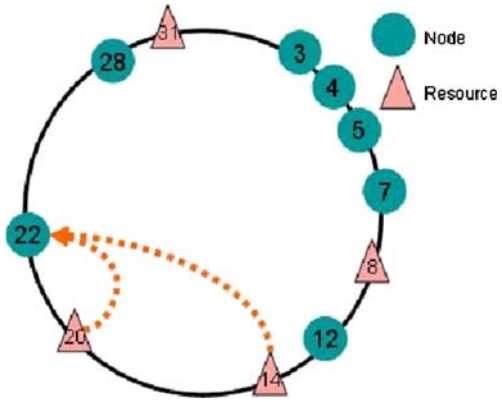

Fig. 2 Chord storage

own finger table and forward the message to its successor (destination peer 28). According to the simulation result in paper [16], the average path length of Chord is 1/2 log*N*, where N is the number of peers in the overlay.

Chord also defines the advertisement function supporting joining/leaving procedure for peers. The advertisement function would tell the corresponding successor and predecessor to update their finger table.

3.2 Improvement of Chord-based overlay

Many efforts have been done to improve the Chord lookup efficiency. In the following, we mainly describe three types of improvement.

One approach is through revising Chord lookup algorithm. For example, BiChord [17] gets the efficiency improvement by initiating bi-directional lookup request from source peer. EpiChord [18] proposes that source peer initiates queries in parallel to *p* immediate successors and to *p-1* immediate predecessors, where *p* is a system attribute.

Another improvement is using Cache mechanism to reduce the delay, as proposed in paper [19, 20]. Cache

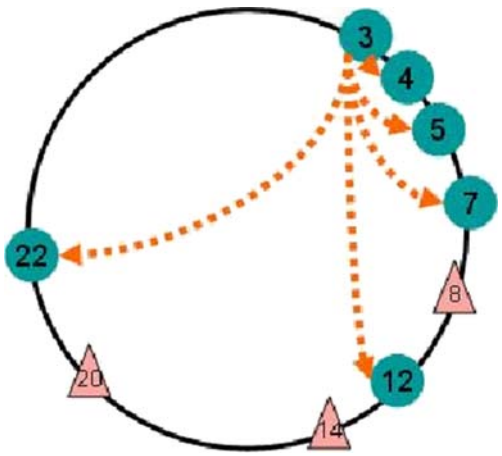

Fig. 3 Direct connection

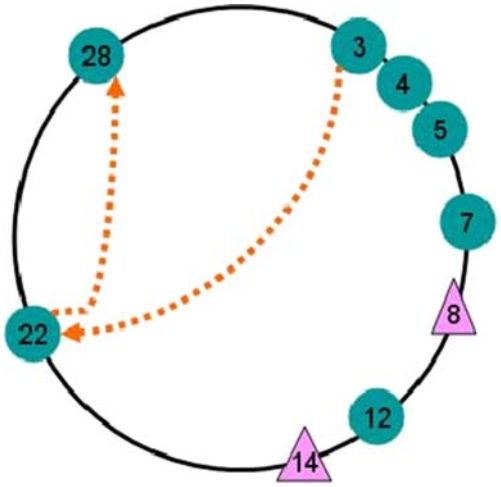

Fig. 4 Message flow

records the communication history, e.g. peer ID, public IP, port, etc. With these information, the session might be established directly. The simulation result in paper [21] shows this approach greatly improves the peer/resource lookup efficiency, especially in the stable overlay where peers do not join/left the overlay in high frequency.

The hierarchical model could be also an enhancement, e.g. two-layer overlay model proposed in [22, 23] and k-layer model proposed in [24, 25]. The idea is to select relatively powerful and stable peers for the upper layer, and relatively weaker and unstable peers for the lower layer. This approach makes the system more scalable, and guarantees the peer/resource lookup in the higher overlay layer.

4 P2PSIP session initiation

4.1 Enrollment and bootstrap

When a new user wants to join the P2PSIP overlay for the first time, enrollment procedure need to be initiated. One possible approach is to use a central enrollment server, from which the new peer learn the particular overlay network, including the overlay algorithm, type of credentials required, address of credential servers, a list of bootstrap peers, etc. These parameters are encapsulated into an XML file and sent from enrollment server to the peer. Based on overlay parameters, the peer chooses the corresponding handling mechanisms. For instance, if the credential is needed for security reason, the peer has to send its public key to a particular credential server for generating user certificate (described in Section 4). Enrollment server also informs the policy of being a peer in the overlay. For example, the peer should have enough CPU processing power, bandwidth and should be able to make contributions (e.g. cooperative routing, storage, etc) to the other peers.

Bootstrap takes place every time when the peer boots. The bootstrap peers are a set of static peers collected by the enrollment server. Their mission is to help the joining peer to find its neighbors (successors) in the overlay. Bootstrap peer is the first contact point for a joining peer.

4.2 NAT traversal

Network Address Translators (NAT) provides benefits as well as drawbacks. One main drawback is that NAT is not friendly for connection establishment between two endpoints. In order to solve this problem, STUN[1]/TURN[2]/ICE[3]-based approaches [26, 27] are proposed. STUN approach uses a STUN server in the middle of two endpoints to learn the NAT status (e.g. existence of NAT, NAT type, public endpoint address, port, etc). With these information, two endpoints might be able to establish the session directly. However, STUN approach does not work in symmetric NAT where each connection is mapped to a specific IP address and port. One possible solution is to use a TURN server to relay data traffic during the connection and transmission. ICE combines the usage of STUN and TURN approaches. It firstly selects STUN for handling, while turns to TURN if STUN is not available. Besides, ICE supports the negotiation of session establishment (e.g. latency, jitter measurement, error handling, best route, etc) between two endpoints.

Another approach is based on Universal Plug and Play (UPnP) [28, 29], pushed by Microsoft. In this solution, client queries the NAT via "UPnP asking", asking what mapping it should use if it wants to receive on a certain port. The NAT responds with the IP and port pair that can be reached from public internet. Today, more and more internet gateway vendors offer the support of UPnP protocol, which makes this technology quite promising for P2PSIP communication systems.

4.3 Message routing

P2PSIP message flow in overlay network should comply with a few routing styles, for instance, Iterative, Recursive, and Semi-Recursive [6]. The implementation of these mechanisms in P2PSIP environment should consider different technical environments, such as NAT Traversal (in Section 4.2), security (Section 5), etc.

In Iterative routing, source peer is redirected by each intermediate peer to the destination, as shown in Fig. 5. In Recursive routing, the request is forwarded hop by hop by each intermediate peer until the destination. The response follows the same route back to the source (See Fig. 6). Another routing is Semi-Recursive routing, the compromise of two routing styles above. In this approach, request message is forwarded by intermediate peers hop by hop to the destination, while the response is directly returned (See Fig. 7).

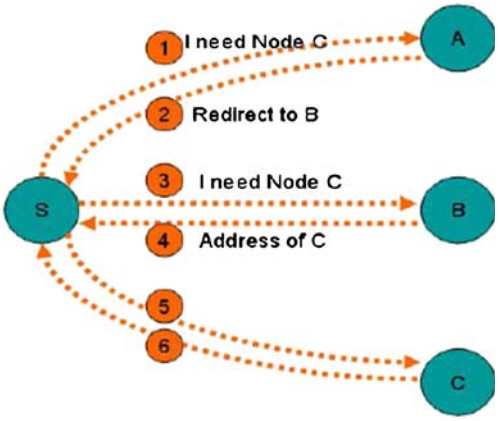

Fig. 5 Iterative routing

In interactive routing, source peer is able to check the validity and correctness of each response. It might be implemented in security sensitive environment. However, this solution does not provide guarantee for NAT traversal when the destination peer is behind NAT protection. Recursive routing has little trouble in NAT traversal; however, it might cause long delay due to lots of message flows. Therefore, this approach is only suggested for high capability overlay (e.g. more CPU processing power, high bandwidth, etc). The third approach has no problem in NAT traversal and system delay. Also, it provides better security than Recursive routing since the response is directly forwarded to the source peer. This approach is capable to be implemented in the environment that needs NAT traversal, lower latency, and better security.

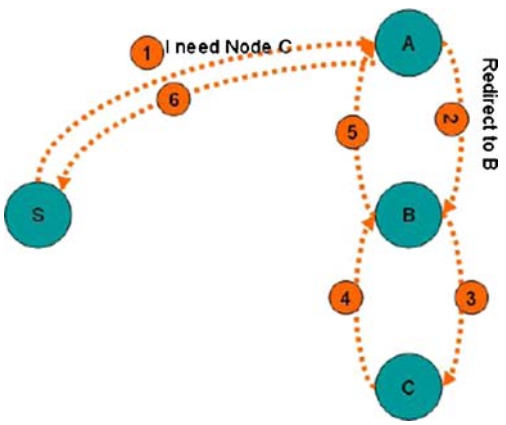

Fig. 6 Recursive routing

[1] STUN: Simple Traversal of User Datagram Protocol through NATs.

[2] TURN: Traversal Using Relay NAT.

[3] ICE: Interactive Connectivity Establishment.

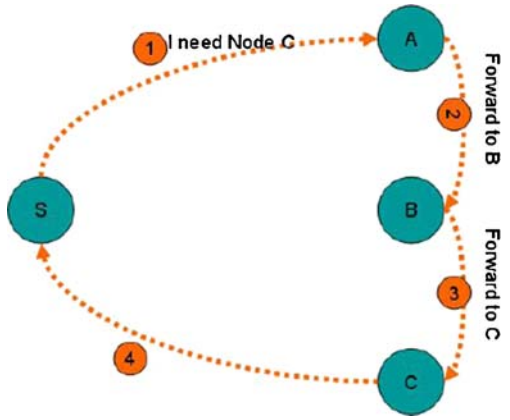

Fig. 7 Semi-recursive routing

4.4 Interworking

Paper [30] proposes an "super peer" based hierarchical architecture for interconnection among different P2PSIP domains/overlays. In order to have inter-domain connectivity, every domain should have at least one "super peer", which is the stable peer owning high system capability (e.g. CPU, bandwidth, etc). These super peers connect each other to form an upper layer overlay, which provide helps when connection is needed between peers among different P2PSIP domain.

P2PSIP should also handle the interconnection with future All-IP networks (e.g. SIP or IP Multimedia Subsystem (IMS) based network). A possible system architecture is suggested in [31], in which a Gateway Application Server (AS) is proposed as the key interworking unit between two different network (See Fig. 8). Gateway AS acts as an ordinary P2PSIP peer in P2PSIP network and an IMS Application Server in IMS network.

4.5 P2PSIP client

A special type of P2PSIP entity, called "P2PSIP client", is now discussed by IETF P2PSIP Working Group. P2PSIP client is the devices that participate in the overlay but do not make contribution (e.g. routing, storage, etc) due to lack of coherent support in the overlay algorithm or limited capability (e.g. energy, bandwidth, CPU processing power, etc). One possible approach [11] is to define a separate client protocol for the association between P2PSIP client and an associated peer in the overlay. The associated peer acts as relay proxy for P2PSIP client to send out P2PSIP messages and receive the corresponding response.

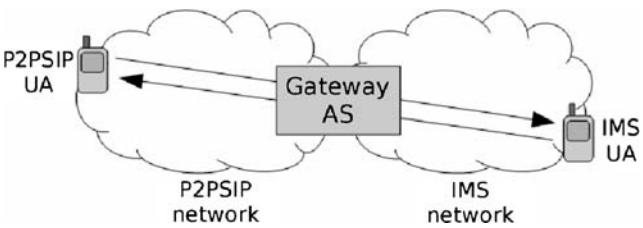

Fig. 8 P2PSIP-IMS interconnecting architecture

5 Security challenges

P2PSIP system security could be divided into three parts: authentication and authorization, trust management, and transport security. Firstly, a peer must be authenticated and authorized. Then, trust management is responsible to build trust relationship among communication parties. Finally, transport security protects the confidentiality, integrity and availability in data transmission. In this section, we identify security problems and review a few proposed solutions.

5.1 Security problems

The decentralization of P2PSIP network comes to the cost of reduced management capability and control that results in security problems. Some possible problems are listed below (according to [12, 32]):

- *Impersonation*: an intermediate malicious peer might misroute, discard and temper the received data traffic.
- *Sybil attack*: an attacker could join the overlay with different identities and control a part of the overlay.
- *Eclipse attack*: intermediate peers can conspire to hijack and dominate the neighbor set of correct peers by controlling the data traffic through routing.
- *Partition attack*: a malicious bootstrap peer might provide wrong information to the joining peer and prevent the normal joining process.
- *Replay attack*: a malicious peer may resend the older message, trying to replace the newer data with the old information.
- *Eavesdropping*: an intermediate peer can passively record activities of the other peers in the network. For example, it can record the activities of the neighbors, when they register, who they are calling, etc.
- *DoS attack*: a malicious peer could launch a DoS attack against one or more peers to consume computing resource.
- *SPAM*: even worse than the email SPAM, P2PSIP-based phone might ring at any time.

5.2 PKI-based certificate

Public Key infrastructure (PKI) based certificate is supposed to be implemented [33]. Certificate proves the existence and legitimacy of the specific peer so that the communication session is trustful and precise. In additional to a few basic elements (e.g. version number,

signature algorithm, digital signature of the issuer, etc), P2PSIP peer certificate might include P2PSIP related information: peer specific ID and one or more user names (e.g. alice@operator.com, etc). Public and private keys are used to handle the task of encryption, decryption, and digital signature.

5.3 Pre-shared-key approach

In the closed networks, Pre-Shared Key (PSK) approach might be more convenient. In cryptography, Pre-Shared Key is a symmetric key shared among the peers before establishing secure connections. The secret or key can be a password like "hElLo#QWoRld", a passphrase like "Wo ai ni", or a hexadecimal string like "AUS30209-DOP745".

Using Pre-Shared Key can help to avoid the heavy work of certificate based operations. Also, it is more convenient to configure a PSK in closed environments. However, it is weak to the DoS attack when an attacker initiates a large amount of exchange key requests to consume the computing resources.

5.4 Trust-based security enhancement

A typical example of trust-based security enhancement is to implement reputation system [34]. A reputation system collects, distributes, and aggregates feedback about participants' past behavior. In this approach, the reputation is represented in a discrete trust value (e.g. 1 represents positive reputation and 0 represents negative reputation). The reputation is earned by contribution, for instance, relaying data traffic, acting as the STUN server, etc. A peer with sufficient trust value means it behaves well and is trustworthy, while a malicious peer that experience negative behavior will probably have a low trust value, and thus is not trustworthy.

Another novel approach is based on subjective logic, proposed in [35]. The subjective logic defines the term opinion $\omega = \{t, d, u\}$, in which t, d and u correspond to trust, distrust, and uncertainty respectively. Subjective logic also defines logical operators for combining opinions. For example, the recommendation operator $\otimes$ can be introduced to evaluate the trust model of p which might be a statement like "the message traverse from A to B is unchanged results of measurement", as following:

$$\omega_p^{AB} = \omega_B^A \otimes \omega_p^B = \left\{ t_p^{AB}, d_p^{AB}, u_p^{AB} \right\}$$

Paper [36] implements this approach for the enhancement of P2PSIP security. Suppose that a request goes through the source peer A, intermediate peers B_1, B_2, $B_{n\text{-}1}$, and ended in the destination peer B_n. By applying the rules of subjective logic recommendation, the trustworthiness of this data flow is:

$$\omega_p^{AB_1B_2\ldots B_{n-1}B_n} = \omega_{B_1}^A \otimes \omega_{B_2}^{B_1} \otimes \omega_{B_3}^{B_2} \otimes \ldots \otimes \omega_{B_n}^{B_{n-1}} \otimes \omega_p^{B_n}$$

Therefore, by mathematic calculation, it is possible to evaluate the trust level for each message flow.

5.5 Proxy based security

A possible architecture that provides the secure service for P2PSIP communication systems is proposed in [37, 38]. The proposed architecture contains three main parts: P2PSIP Peer, Resource, Chord Secure Proxy (CSP), as shown in Fig. 9. For locating a peer/resource in the overlay, the source peer first sends the P2PSIP request to a specific CSP (Step 1). The CSP acts as a proxy server to probe the destination peer through multicasting a "Hello" message to its successors through Chord algorithm. When the destination peer receives a "Hello" message, it contacts the CSP to catch the original P2PSIP request (Step 2). After that, the connection between source and destination peers can be established (Step 3).

The connections in the system architecture are SSL/TLS secured. The "Hello" multicast mechanism (in Step 2) firstly makes sure that intermediate peers are incapable to receive original P2PSIP request, and secondly guarantees in some level that destination peer could receive at least one "Hello" message. Therefore, this architecture provides the secure P2PSIP session initiation.

6 Conclusion and open issues

In this paper, we consider the P2PSIP paradigm in communication systems. We make a survey on the current

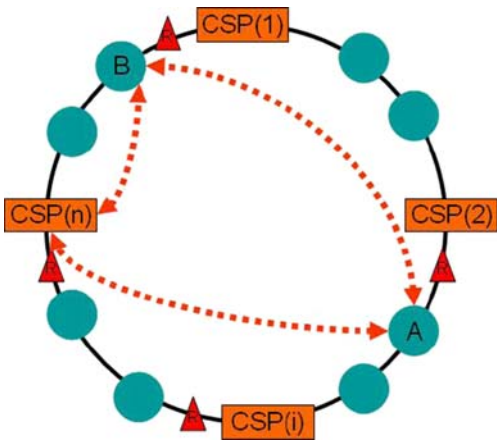

Fig. 9 Architecture overview [37]

approaches implemented/proposed in academia and industry. What we considered in the survey is: Chord-based P2PSIP overlay, P2PSIP session initiation (e.g. enrollment and bootstrap, NAT traversal, message routing, interworking issues, P2PSIP client, etc) and security challenges. However, the research of P2PSIP is still far from mature. A lot of open questions are waiting for the urgent answer.

Firstly, although there are a few proposals discussing P2PSIP interworking issue, much more should be done to test and evaluate the availability and efficiency. Besides, the interworking between P2PSIP and conventional PSTN network should be further studied.

Secondly, IETF P2PSIP WG is still discussing the need of client protocol and how efficient it could work in reality. An Internet-browser based approach (described in [39]) might be an alternative solution.

Thirdly, the security mechanisms proposed are not enough. For instance, there is no efficient protection against the security problems such as DoS attack and SPAM (described in Section 5.1).

References

1. *Make the most of Skype - free internet calls and cheap calls*. p. http://www.skype.com/
2. *Nokia - Nokia on the Web*. p. http://www.nokia.com/
3. *Sony Ericsson*. p. http://www.sonyericsson.com/
4. *The SIP Center - A portal for the commercial development of SIP Session Initiation Protocol*. p. http://www.sipcenter.com
5. Kundan S, Henning S (2005) *Peer-to-peer internet telephony using SIP*, in *Proceedings of the international workshop on Network and operating systems support for digital audio and video*. Stevenson, Washington, ACM
6. Bryan DA, Lowerkamp BB,Jennings C (2005) *SOSIMPLE: A Serverless, Standards-based, P2P SIP Communication System* First International Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications (AAA-IDEA'05), pp 42–49.
7. *P2PSIP*. p. http://www.p2p-sip.org
8. *Peer-to-Peer*. p. http://www1.cs.columbia.edu/~salman/peer/
9. *SIPDHT2*. p. http://sipdht.sourceforge.net/sipdht2/index.html
10. *MjSip*. p. http://www.mjsip.org
11. Matuszewski M, Kokkonen E (Jan, 2008) *Mobile P2PSIP - Peer-to-Peer SIP Communication in Mobile Communities*, in *5th IEEE Consumer Communications and Networking Conference, 2008. CCNC 2008*. Las Vegas
12. Seedorf J (2006) Security challenges for P2P-SIP. IEEE Network Special Issue on Securing Voice over IP 20(5):38–45
13. Bryan DA, Lowekamp BB, Zangrilli M (April, 2008) *The Design of a versatile, secure P2PSIP communications architecture for the public internet*, in *IEEE international Parallel and Distributed Processing Symposium*. Lyon, France
14. Naoki W, Masayuki M (2006) *Overlay network symbiosis: evolution and cooperation*, in *Proceedings of the 1st international conference on Bio inspired models of network, information and computing systems*. Cavalese, ACM
15. David K, Eric L, Tom L, Rina P, Matthew L, Daniel L (1997) *Consistent hashing and random trees: distributed caching protocols for relieving hot spots on the World Wide Web*, in *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*. El Paso, ACM
16. Stoica I, Morris R, Liben-Nowell D, Karger DR, Kaashoek MF, Dabek F, Balakrishnan H (2003) Chord: a scalable peer-to-peer lookup protocol for internet applications. IEEE/ACM Trans Netw 11(1):17–32
17. Junjie J, Ruo P, Changyong L, Weinong W (2005) *Bi-Chord: An IMproved Approach for Lookup Routing in Chord*. Lect Notes Comput Sci
18. Ben L, Barbara L, Erik DD (2006) EpiChord: parallelizing the Chord lookup algorithm with reactive routing state management. Comput Commun 29(9):1242–1259
19. Vasilios D, Nicolas L, Oliver H, Andreas M, Ralf S (2006) Cacheing indices for efficient lookup in structured overlay networks. Lect Notes Comput Sci 4118:81–93
20. Bhattacharjee B, Chawathe S, Gopalakrishnan V, Keleher P, Silaghi B (2003) Efficient peer-to-peer searches using result-caching. Lect Notes Comput Sci 2735:225–236
21. Zheng X, Oleshchuk V (June, 2008) *Improving Chord lookup protocol for P2PSIP-based Communication Systems*. 2009 International Conference on New Trends in Information and Service Science (3rd NISS)
22. Zoels S, Despotovic Z, Kellerer W (2008) On hierarchical DHT systems - An analytical approach for optimal designs. Comput Commun 31(3):576–590
23. Joung Y-J, Wang J-C (2007) Chord2: a two-layer Chord for reducing maintenance overhead via heterogeneity. Comput Netw 51(3):712–731
24. Le L, Kuo G-S (June, 2007) Hierarchical and Breathing Peer-to-Peer SIP System, in IEEE International Conference on Communications, 2007. ICC'07
25. Shi J, Wang Y, Gu L, Li L, Lin W, Li Y, Ji Y, Zhang P (Nov. 2007) A Hierarchical Peer-to-Peer SIP System for Heterogeneous Overlays Interworking, in Global Telecommunications Conference (GLOBECOM'07), IEEE
26. Rosenberg J, Weinberger J, Huitema C, Mahy R (March 2003) STUN-Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs). RFC 3289
27. Stukas M, Sicker DC (2004) An evaluation of VoIP traversal of firewalls and NATs within an enterprise environment. Information Systems Frontiers 6(3):219–228
28. *Welcome to the UPnP Forum !* : p. http://www.upnp.org
29. Sterman B, Schwartz D. *NAT Traversal in SIP*. IEC Annual Review of Communications
30. Martinez-Yelmo I, Bikfalvi A, Cuevas R, Guerrero C, Garcia J (2009) H-P2PSIP: interconnection of P2PSIP domain for global multimedia services based on a hierarchical DHT overlay network. Comput Netw 53(4):556–568
31. Hautakorpi J, Salinas A, Harjula E, Ylianttila M (Sept, 2008) *Interconnecting P2PSIP and IMS*, in *Next Generation Mobile Applications, Services and Technologies*. Wales
32. Fessi A, Niedermayer H, Kinkelin H, Carle G (July, 2007) *A cooperative SIP Infrastructure for Highly Reliable Telecommunication Services*. IPTCOMM'07
33. Cao F, Bryan DA, Lowekamp BB (Feb, 2006) *Providing Secure Services in Peer-to-Peer Communications Networks with Central Security Servers*. International Conference on Internet and Web Applications and Services (ICIW)
34. Seedorf J (Oct, 2006) Security challenges for peer-to-peer SIP. IEEE Netw 20(Issue 5)
35. Josang A, Hayward R, Pope S. Trust network analysis with subjective logic, in Proceedings of the 29th Australasian Computer Science Conference, 2006, Australia
36. Zheng X, Oleshchuk V (Nov, 2009) *Trust-based Framework for Security Enhancement of P2PSIP Communication Systems*, in *The*

4th International Conference for Internet Technology and Secured Transactions (ICITST-2009). London
37. Zheng X, Oleshchuk V (Oct, 2009) *A Secure Architecture for P2PSIP-based Communication Systems*, in *2nd International Conference on Security of Information and Networks (SIN 2009)*. North Cyprus
38. Zheng X, Oleshchuk V (Nov, 2008) *Providing Privacy Service for P2PSIP based Communication Systems*, in *Norsk informasjons-sikkerhetkonferanse (NISK)*. Kristiansand, Norway
39. Zheng X, Oleshchuk V, Jiao H (Dec, 2007) *A System Architecture for SIP/IMS-based Multimedia Services* in *International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE)*

Xianghan Zheng is PHD student of Computer Science at University of Agder, Norway. He received his BSc in Computer Science (2005) from Wuhan University of Technology, China and MSc in Distributed System from University of Agder (2007). From 2005 to 2007 he was a research assistant at Agder Mobility Lab (AML) and Ericsson Germany. His current research interests include networking and communication security with special focus on P2PSIP communication systems.

Vladimir Oleshchuk is Professor of Computer Science at University of Agder, Norway. He received his MSc in Applied Mathematics (1981) and PhD in Computer Science (1988) from Kiev Taras Shevchenko University, Ukraine. From 1981 to 1985 he was a software development engineer at Glushkov Institute of Cybernetics, Kiev, Ukraine. From 1987 to 1991 he was Assistant Professor and then Associate Professor at Kiev Taras Shevchenko University. He has been working at University of Agder since 1992. His current research interests include formal methods and information security and privacy with special focus on applications in telecommunication area.