Appears in ICIW'06 © IEEE

Providing Secure Services in Peer-to-Peer Communications Networks with Central Security Servers

Feng Cao
Critical Infrastructure Assurance Group
Cisco Systems, INC
170 West Tasman Drive
San Jose, CA 95134
Email: fcao@cisco.com

David A. Bryan and Bruce B. Lowekamp
Department of Computer Science
College of William and Mary
P.O. Box 8795
Williamsburg, VA 23187
Email: {bryan, lowekamp}@cs.wm.edu

Abstract

IP Telephony based on Peer-to-Peer (P2P) technology has being gaining attentions for its innovative approach to providing VoIP service. At the same time, it has raised many new research topics about how to provide the secure services, such as voice mail and supplementary service, inside new distributed P2P environments. This study proposes a generic architecture for providing secure information in P2P IP Networks. Some secure techniques and components are shown to guarantee privacy of voice mail service. This new architecture is scalable and generic for providing other secure services for IP Telephony in distributed systems.

1. Introduction

Voice over IP (VoIP) has been widely deployed in different segments to replace the traditional circuit switched infrastructure for telephony service.

IP Telephony based on Peer-to-Peer (P2P) technology has been gaining attention in the academic, standards, and commercial communities for its innovative approach to providing VoIP service. P2P Telephony provides telephony services to registered users in a distributed, highly scalable way that requires minimal infrastructure and no changes to the underlying IP infrastructure.

While presenting many opportunities, this technology has raised many research questions including how to secure the P2P environment and how to implement supplementary services such as voice mail inside this new environment.

Several scenarios in P2P communications systems rely on a third party to enable the communications. Using a third party to relay media, for example to enable communications through a NAT, is one such scenario. Many services users have come to expect in traditional VoIP systems, such as voice mail services, will also require third party cooperation in P2P architectures.

The third party peer is dynamically chosen in a P2P environment and cannot be trusted. It is unacceptable for the confidentiality of media traversing a relay or stored voice messages to be compromised by the third party.

In this paper we explore a mechanism for securing voice mail using a sender-oriented encryption mechanism, allowing for encryption for multiple recipients using only one key and one copy of the message, and in the presence of changing user keys. We store the encryption keys and message availability notifications on a centralized login server, but store the encrypted media on the distributed overlay network. This new architecture is scalable and generic for providing other secure services in IP Telephony.

This rest of this paper is organized as follows. In Section 2, basic background, architectures for P2P IP Telephony, the reference architecture used in this paper, and some problems for P2P IP Telephony are presented. We next discuss preventing attacks and alternatives for voice mail in such systems. In Section 3 we present our suggested solution, and discuss implementing it using SIP[1] in Section 4. We conclude and comment on some possible issues and future work in Section 5 and Section 6.

2. P2P IP Telephony Primer

P2P communications networks of many different types have been proposed. All share some common features. As with any P2P network the peers, rather than centralized servers, provide some portion of the functionality of the system. While some approaches proposed recently [2] are fully decentralized (in this case based on the Chord protocol [3]), some models have at least some services provided by a centralized server. One such model, widely in use today, is Skype. [4,5,6] A Skype-like architecture featuring central authentication servers is used in this paper as an example P2P telephony architecture. In this particular P2P overlay network, there are three main architectural components: ordinary nodes, super nodes (SN), and the secure central authentication servers.

2.1. Reference Architecture

In the architecture used in this paper a P2P overlay network is formed using three main architectural components: ordinary nodes, super nodes (SN), and the secure central authentication servers. Other than the secure authentication servers, there are no other centralized servers. The P2P communications application is loaded onto a personal computer or other device belonging to the user. The user contacts the central authentication server at login time, as well as when they wish to store or retrieve voice mail, as discussed in Section 4. All other operations, such as search and message exchange, are performed between the peers.

A P2P node must login to the network using an authentication or login servers for its active presence to place voice calls and send text messages. The user names and passwords are saved in the login servers and used for each login procedure. During the login process, the user's public keys are certified by the login server for use in encrypting text messages or media streams.

Super Nodes (SNs) are chosen from among qualified ordinary nodes. A qualified node is typically one with a public IP address and reasonable system resources such as CPU, memory, and network bandwidth. An ordinary peer must connect to a super node for exchanging information and P2P management. Because a SN is chosen from ordinary nodes, SN could be offline without advance notice. This requires all the ordinary nodes connecting to that SN locate another SN and reestablish connections to the new SN.

2.2. Issues for P2P Voice Services

For P2P VoIP to be a viable alternative to client-server based systems, it must support a wide range of services. There are many different services available in traditional IP telephony, including voice mail, advanced telephony services, and emergency telephony services.

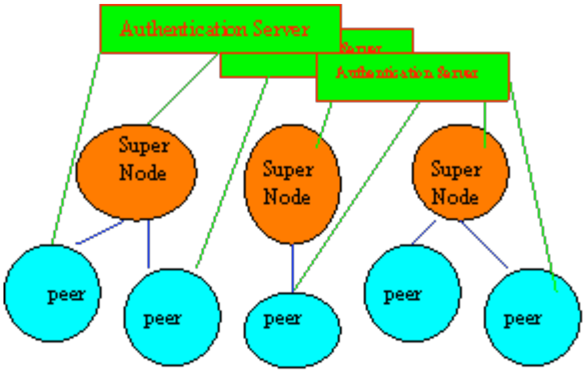

Figure 1. Reference Architecture.

Besides concerns about availability and scalability, there are additional concerns about confidentiality because media may be relayed or stored on third party nodes.

For example, in the traditional voice mail service when a user does not answer calls are redirected to a trusted VM server. This VM server records and stores the messages, and users access the server to retrieve messages. Because VM servers are designated and trusted by the users through initial provisioning and are usually protected by a private key password shared only between the server and the user, the stored voice messages are considered to be secure. Neither the caller nor callees needs to worry about privacy of the voice messages.

In the P2P architecture presented, there is no such central server. If third party nodes are instead used for storage and are dynamically chosen, they cannot be trusted. There are no assurances that the users of the machine where the message is stored will not eavesdrop by reviewing voice messages. For P2P systems to be effectively deployed a level of privacy and reliability similar to client server systems must be achieved.

In a similar way, if the P2P system relies on third party peers for media relay, for example for NAT traversal, these peers can mount man-in-the-middle attacks in which the media being relayed is intercepted.

With such an architecture, rouge third parties can mount attacks such as

- Collecting confidential information about the end users (such as pin numbers personal information, or passwords)
- End users can be redirected to arbitrary targets by the third party for the purpose of DoS attacks or to illegitimate voice mail services.
- Users could be deceived to believe they are communicating with a different user than they

believe they are via interception at a media relay.

2.3. Preventing attacks in P2P IP Telephony

With centralized authentication servers, a number of the attacks discussed above can be prevented. Real-time communication between two users can encrypted using a public key system stored on the authentication servers. This prevents eavesdropping of traffic by intermediate nodes.

Similarly, verification of these public keys can verify that two parties communicating are indeed connected to the appropriate parties, preventing spoofing of identities as well as limiting redirect DoS attacks, as a quick verification of keys will indicate that the party being contacted is not the appropriate user, and the call will be terminated.

Offline storage of information presents a number of problems and is not immediately resolved simply by having a key system managed by centralized servers. In this paper. Voice Mail Service (VMS) is used to demonstrate how to protect the privacy of such services in P2P IP Telephony.

One solution is to add dedicated VMS servers to the architecture. The users rely on these servers for storing voice mail and protecting the media content from being stolen.

There are many drawbacks to this traditional approach. It breaks the basic P2P model by adding additional centralized servers. The cost in terms of hardware and provisioning for adding these servers may not be scalable for a large and growing P2P environment and certainly reduces the advantage of P2P to service providers – namely a reduction in capital and personnel expenditures. Finally, by centralizing these servers, issues such as load balancing, DoS attacks, and capacity problems arise.

Another similar approach is to allow the centralized login servers to act as VM servers. While this reduces the number of servers, most of the issues outlined above are still applicable.

Without additional new centralized VM servers, another approach is to use sRTP [7], IPSec [8] or TLS [9] to send the voice message securely to some peer for storage. In this case, the messages are then stored on these peers, but only the transmission, not the storage is secure. The messages can still be intercepted by the storing node, even if protected from relaying nodes.

If the public key infrastructure discussed above is available, one alternative is to use the recipients' public key to encrypt the voice mail by the senders. There are some instances where this approach is undesirable. Voice mail messages may persist for some time before being retrieved, and it is possible that the recipient's key pair may change during this time. Furthermore, if the same voice mail is delivered to multiple recipients, multiple copies with different encryption keys are required by this alternative. In such scenarios, a sender-oriented approach may prove more attractive.

In addition, this mechanism allows for anonymous messaging. While this raises issues about VM SPAM (often referred to as SPIT), it may also prove useful in certain situations.

In the following sections, we propose one approach to address these concerns in an environment with secure central authentication servers. The approach is scalable and generic enough to provide security for a number of IP communications services.

3. A New Generic Architecture

Our goal is to discuss a preliminary architecture for VM in P2P IP Telephony that addresses some of these security concerns.

There are several components required for this architecture, including

- A group of secure authentication servers
- Heuristic algorithm(s) for choosing one or more a storage location
- A sender-oriented message security scheme
- A distributed VM storage and playback application in each peer

With the help of these components, this architecture can provide a more secure VMS in a large P2P environment.

3.1. Secure Group of Authentication Servers

As discussed in Section 2.1, secure authentication servers are available in the system. To increase availability and reliability, these servers are replicated. Figure 2 is an example for illustrating such a group.

The distributed authentication servers store the user's keys and provide login services. The servers form a secure multicast group amongst themselves for handling the tasks of exchanging the key information needed to allow peer logins, and as described below, VM security. This secure key sharing mechanism serves to allow additional authentication servers to be added to the overlay transparently, which would form a bottleneck in a P2P architecture.

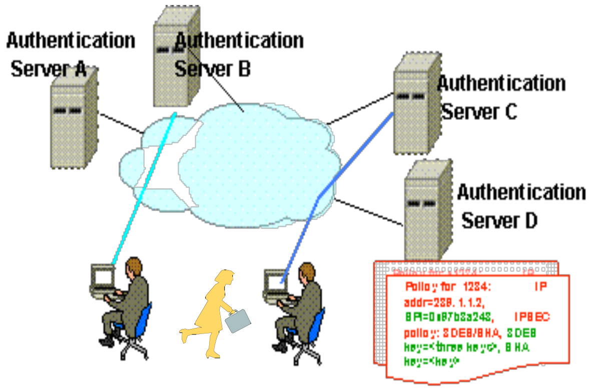

Figure 2: An example of Secure Group of Authentication Servers

3.2. Heuristic algorithms for selecting a storage location

We use Figure 2 as a simple example for discussing heuristic algorithms for obtaining storage locations for VM messages.

Assume Alice is offline and Bob wants to leave a voice mail for her. In a P2P environment, Bob is connected to his SN and uses this node for P2P operations. Bob wants to find one or more storage locations (peers) for saving this message for retrieval when Alice is online again.

There are two options for locating a storage peer. Bob can either ask his SN for a desirable location, or Bob can search the candidates by himself. Either option requires some heuristic algorithms for obtaining the location.

There are many factors which could be used in determining which storage location should be picked, including

- How frequently the peer is available
- Available storage space on the peer
- Frequency of this being selected as storage location
- Last time this node was chosen as a storage location
- A predefined, preferred list of storage locations
- Recommended storage locations from SN or other peers
- Past experience with nodes
- Network conditions for reaching candidate peers

By giving weights to each related factor, a score is calculated for each node based on their abilities and determining their suitability as a storage location. Many possible mechanisms can be used to calculate this score. For example, one possible heuristic algorithm could be defined as follows

$$Score = a \times e \times \frac{c}{\min\{f,1\}}$$

where:

- a: Probability peer is available (0-1)
- e: Past experience with peer. If there is no previous experience with the candidate peer, this value could be .5. Negative experience would reduce this value, and positive experience would increase the value.
- c: Available storage capacity, using a predefined scale. For example, the value could have a minimum of 1 if the server has less than 1.5MB available, and a maximum of 100 if the server has more than 5MB available.
- f: Frequency of being picked as storage location: defined as messages stored per unit time

Based on the scores from a heuristic algorithm such as the one shown above, Bob selects a peer to store his voice mail to Alice. For replication purposes, Bob may select multiple locations.

In structured P2P systems, data is located based on an identifier. The identifier then determines which node the data is stored on. Since Bob wishes to select a specific node, he must query that node for the range of identifiers it is responsible for and random select one to use as the identifier for the VM to be stored. Again, this is repeated if replication is desired for reliability. By storing based on identifier, rather than directly on the particular peer, the message will be moved automatically if the selected peer exits the system normally. Replication prevents against storage peer failure.

3.3. Sender-oriented message security scheme

A sender-oriented multimedia security scheme is shown here to provide efficient voice service in a P2P environment, even in the presence of changing user keys or multiple voice mail recipients.

Bob takes the following steps after using a heuristic to select one or more peers to store the message and generating an appropriate identified in the range the selected peer is responsible for..

0. During the initial login of Bob, Bob will negotiate a shared master key with authentication server to be used for VM in the following steps.

1. Bob generates a new key, called a session key using his shared master key, a random number, an integer timestamp and Alice's username.
2. Bob notifies the authentication server he is sending this new voice message. He provides the server with the recipient's username(s), the newly generated session key and the identifier for the data. Bob sends this notification over an encrypted channel, using the same key set encrypted real time communication.
3. The authentication server saves these parameters and makes note in Alice's record there is a VM waiting.
4. Bob encrypts the voice mail message using the new session key.
5. Bob delivers the encrypted message to the chosen storage location.

Bob can transmit the message to the storage node with no protection on the transmission channel and with no concern the storing peer can interpret the message, since the content is encrypted.

When Alice logs in, she uses the following mechanism to retrieve the message:

1. Alice logs into the P2P overlay. After verifying her login, the authentication server checks if there are any voice messages waiting for Alice.
2. If there are messages waiting, the authentication server passes the session key and the identifier to locate the message to Alice.
3. Alice performs a search for the identifier, then contacts the peer storing the message and retrieves the message. The peer does not need to authenticate Alice because she cannot decrypt the message without the appropriate session key, and the transmission mechanism need not be encrypted during transit since the payload is encrypted.
4. Alice uses the session key received from authentication server to decrypt the voice message and review it.

This approach offers several advantages.

- The allowed recipients are clearly specified by the sender instead of other agents.
- Only the named recipients can obtain the session key to decrypt the messages.
- The voice messages are secured by the sender from the very beginning by applying the session key.
- If multiple recipients are specified, only a single copy of encrypted voice mail is stored in the overlay, reducing the storage requirements for broadcast messages.

Once the voice mail is downloaded into Alice's local machine, local software on the peer can manage her voice mail locally. This VM application, along with the simple file storage functionality needed should be bundled with the P2P application.

4. Implementation based on SIP

Session Initiation Protocol (SIP) can be used as the transport mechanism for a P2P voice service. The architecture presented here can be implemented using SIP with only a few additional headers, as in [2].

Using SIP, and the steps shown above we show how the same architecture works for providing the secured VMS in a P2P SIP environment:

0. The negotiation of a shared master key with the authentication can be done at login time using extensions to the SIP REGISTER message. This masker key will be used for VMS in the following steps.
1. Bob's generation of the session key is internal to his peer and requires no new SIP functionality.
2. Bob's notifies the authentication server he wishes to store a VM, and sends the list of recipients, identifier(s) for the voice message, and session keys using extensions to the SIP UPDATE or PUBLISH mechanisms.
3. The authentication server stores the message internally, requiring no new SIP functionality.
4. Bob encrypts the message. This requires no new SIP functionality.
5. Bob delivers the encrypted payload to the chosen storage location(s). This could be done using the SIP PUBLISH mechanism or using the new IETF MSRP [11] standard.

Similarly, SIP could be used in the retrieval process as follows:

1. Alice logs into the system using SIP REGISTER.
2. If there is a voice mail for Alice, this fact, the identifier to locate the message, and session

key can be passed to Alice using a SIP mechanism such as NOTIFY.

3. Alice searches for and retrieves the message. This file retrieval mechanism could be a special type of media session, using a SIP INVITE, or might employ MSRP.
4. Alice decrypts and plays back the message, requiring no new SIP functionality.

5. Research Issues and Future Work

In this architecture, there is a strong assumption that the central servers be secure. Like any system with centralized authentication servers, if these servers are compromised or a DoS attack is mounted on them, the system can be compromised. Future work includes providing security in the absence or failure of these servers, but the fully distributed identity problem has proven a difficult problem, so compromise solutions are desirable. Exploring solutions incorporating web of trust models may also be an interesting area of work.

Choosing appropriate heuristics for selecting storage peers, as is determining how to best perform replication, is an open question. While we have presented a possible heuristic, further study is needed to determine what heuristics would work best in a real environment. This may include dynamic mechanisms for distributing lists of “preferred” storage peers.

Another concern is developing a policy for when a storage node deletes a message. Alternatives involve allowing the retriever to ask the storage node to delete the message, but this could be an attack, because the storage node does not have the key and cannot confirm the identity of the recipient. Additionally, broadcast messages must be available to multiple recipients.

Finally, exploring issues such as SPIT prevention are very important. We expect many other issues and approaches to arise as this research continues.

6. Conclusion

IP Telephony based on Peer-to-Peer (P2P) technology has been gaining attentions for its innovative approach to providing VoIP service. At the same time, it has raised many new research topics, particularly around the area of security.

We have explored some alternatives for storing VM securely, and presented a solution for VM storage in a specific and widely deployed P2P architecture that we feel is extensible to other services in such a system. The new architecture proposed in this paper is scalable and generic for providing other secure services in P2P-based IP Telephony.

References

[1] Rosenberg, J et al., RFC 3261 - The Session Initiation Protocol (SIP), June 2002

[2] D.A. Bryan, B.B. Lowekamp, and C. Jennings, SOSIMPLE: A Serverless, Standards-based, P2P SIP Communication System, Proceedings of the 2005 International Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications (AAA-IDEA 2005), Jun. 2005

[3] Stoica, M., D. Karger, M. F. Kaashoek, H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In Proc. ACM SIGCOMM (San Diego, 2001).

[4] Baset S.A. and H. Schulzrinne, An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol, Technical Report of Department of Computer Science, Columbia University.

[5] IRT lab. http://www.cs.columbia.edu/IRT

[6] Skype. http://www.skype.com

[7] M. Baugher et al., RFC 3711 - The Secure Real-time Transport Protocol (SRTP), March 2004

[8] IPSec working group, http://www.ietf.org/html.charters/OLD/ipsec-charter.html

[9] T. Dierks and E. Rescorla, RFC 2246 bis 13 - The TLS Protocol Version 1.1

[10] B. Campbell, R, Mahy and C. Jennings, draft-ietf-simple-message-sessions-09 The Message Session Relay Protocol (work in progress)

[11] Larson, J. et. al., Defending VoIP Networks from DdoS attacks, Globecom 2004 VoIP Security Workshop

[12] Cao, F. and S. Malik, “Security Analysis and Solutions for Deploying IP Telephony into the Critical Infrastructure”, Proceedings of IEEE/Create-Net Workshop on Security and QoS in Communication Networks (Secqos’2005), September 2005, Athens, Greece

[13] Cao, F. et. al., Call Filtering and Tracking in IP Telephony, Proceeds of IASTED International Conference on Internet and Multimedia Systems and Applications (IMSA 2003)

[14] Rosenberg J., J. Weinberger, C. Huitema, and R. Mahy. STUN: Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs). RFC 3489, IETF.

[15] Ethereal. http://www.ethereal.com

[16] Net Peeker. http://www.net-peeker.com