4 Technical Means to Combat Spam in the VoIP Service

Spam refers in general to any unsolicited communication. Spam will also become one of the serious problems for multimedia communication in the near future. Spam in multimedia communication is referred to as SIP spam or SPIT (Spam over Internet Telephony), where SIP is used to manage the session between two end users. In this paper, the types of SIP spam are introduced and various pragmatic solutions applicable to combat the SIP spams are described including content filtering, white list, black list, and the reputation system. Finally, the detailed operation and principles for the authenticated identity in SIP header, which is a prerequisite for the solutions above, are also described. The possible solutions to combat the SIP spasm have been listed and the background technology to those solutions, an authenticated identity between the domains, is also introduced.

Heung Youl Youm (PhD)
Professor, Soonchunhyang University, South Korea
Rapporteur, Q.9/SG17, ITU-T
hyyoum@sch.ac.kr

1 Introduction

IP telephony is known as a technology that allows standard telephone voice signals to be compressed into data packets for transmission over the Internet or other IP network. The protocols used in carrying the voice signals over the IP networks are commonly referred to as Voice over IP (VoIP). The spam problem in email and instant messaging (IM) makes the email or the IM users to trust less of these tools and consequently reduce their usage. While the security mechanisms for the IP telephony are being studied, the spam problem in VoIP has not been studied extensively yet. Spam is not a problem only to e-mail, but to multimedia services such as VoIP. To provide satisfactory VoIP services, SIP providers need to deal with unsolicited or spam communications [1]. Though SIP is yet to be the target of this kind of attacks, it seems it is only a matter of time. So, the technical means to combat the VoIP spam becomes indispensable to providing the trusted VoIP service to the user. There are practical approaches to email spam: Content filtering, black list, white list and the reputation system. It seems that most successful anti-spam measure taken to combat email spam is almost useless for the prevention of VoIP spam. The possible solutions to combat the VoIP spam should be developed before the VoIP service proliferates. The main purpose of this paper is to describe the applicable solutions to combat the VoIP spam.

The remaining sections are organised as follows: Section 2 presents an overview of the SIP architecture, SIP session establishment procedure, and various message types which are closely related to SIP spam. Section 3 describes SIP spam types, characteristics of SIP spam, and the possible solutions to deal with. Finally, section 4 concludes the paper.

2 Background

2.1 Overview of IP Telephony

IP telephony allows the traditional voice signal to be transferred over a packet switched network, instead of public switched telephone network [4]. The advantages of IP telephony over traditional telephony are amongst others: lower costs per call (or even free calls in some cases) and lower infrastructure costs. IP networks are considered as best-effort networks, so unfortunately there is no guarantee of constant traffic flow or reliable transport. Therefore the serious problems facing IP telephony include: Quality of Service (QoS) guarantees, latency, and possible data integrity and privacy problems including SIP spam.

IP telephony, sometimes referred to as VoIP, usually comprises a signalling plane and a multimedia plane. Figure 1 illustrates the protocol stacks for VoIP services. The signalling plane is used to transport the necessary signalling information for managing the session between IP telephony devices, while after call setup, the media transport plane is used to transfer the compressed voice data packets between IP telephony components. SIP can be used to transfer the signalling information, while the real voice traffic can be transferred through the RTP (Real-time Transport Protocol) which provides a framework for delivery of audio and video across IP networks with unprecedented quality and reliability.

Before audio can flow from the originator to destination, various protocols must be employed to find the remote device and to negotiate the means by which audio will flow between the two devices [1]. SIP, published as RFC 3261, is a session initiation protocol developed by IETF. SIP uses the Session Description Protocol (SDP) to select the type of media and to negotiate a codec for media transmission [3]. A session description expressed in SDP is a short structured textual description of the name and purpose of the session, and the media, protocols, codec formats, timing and transport information that are required to decide whether a session is likely to be of interest and to know how to start media tools to participate in the session. The Session Description Protocol (SDP) describes multimedia sessions for the purpose of session announcement, session invitation and other forms of multimedia session initiation. After a session has been established with SIP, the actual media transfer is based on the Real-time Transport Protocol (RTP). Security of RTP is not within the scope of this paper.

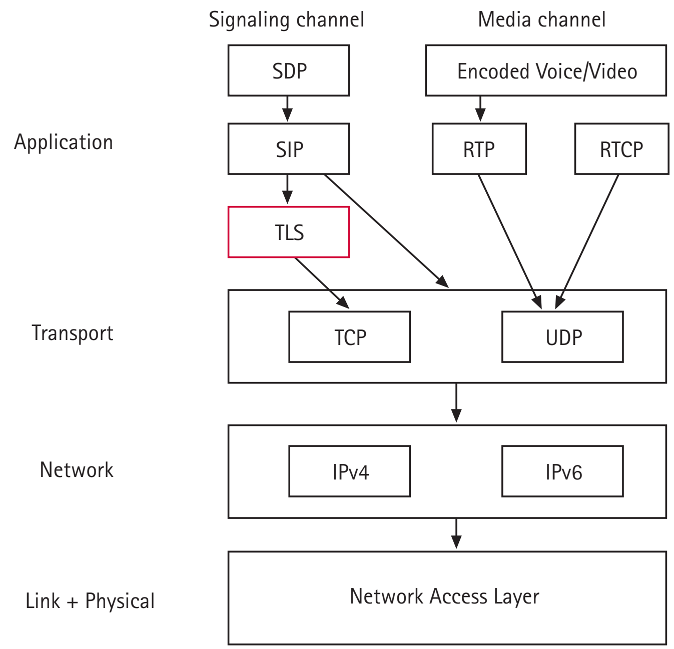

Figure 1: Protocol Stack for VoIP

2.2 SIP Overview

A session can be established between two end-users or more and can involve IP phone calls, conferencing and messaging. The Session Initiation Protocol (SIP) is an application-layer protocol for establishing, modifying, and terminating multimedia sessions with one or more users. In addition, SIP is a text-based protocol, based on an HTTP/SMTP request-response model where SIP addresses users by an email-like address typically containing a username and a host name. SIP is a peer-to-peer protocol where each peer is referred to as a User Agent (UA) where UAs can either act in client or server mode. SIP identity, a type of Uniform Resource Identifier, called a SIP URI, is used for initiating interactive communication sessions between users. The SIP architecture comprises five entities (Figure 2), namely: SIP user agents, Proxy Servers, Redirect Servers, Location Servers, and Registrar Servers.

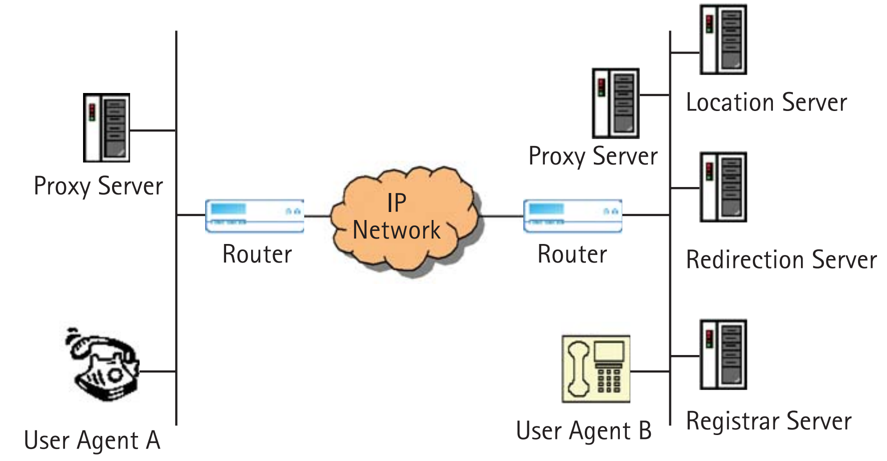

Figure 2: Entities in SIP Architecture

User agents (UAs) designates any terminal (hardware or software) participating in SIP-communications. They are the end devices in a SIP network, called SIP phone. They can generate SIP requests to establish a media session and send and receive media. A user agent can be either a SIP phone or a SIP client software running on a PC. Servers are the intermediate SIP entities in a SIP network that assist the user agents in establishing media sessions and some other functions. SIP servers are classified into three categories: proxy servers, redirect servers and registrars. A proxy contacting more than one user receives requests or responses and forwards them to another server or user agent. A Redirect Server is to provide alternative location information to a UA in response to a request, but doesn't participate in the connection setup. A redirect server, when receiving a message, informs the sender of the message where to send the message, rather than forwarding it. It maps the user address into zero or more addresses and returns those address to the user agent. Registration Servers are used by a SIP device, such as a phone, to register its current location. Users can register their current location with the registrar of their domain to facilitate mobility. Location Server is to store location information in a database. It usually runs on the same physical server as the registration server. A location server is used by a registrar to store the location of users (the binding of a SIP-URI with a current IP-address). Other SIP entities (proxies or redirect servers) can use the location server to look up the current location of SIP users. In fact, a location server is a database in which user information such URLs, IP addresses, scripts and other preferences are stored. A location server may also contain routing information such as locations of proxies, gateways and other location servers.

2.3 SIP Messages

SIP is a client-server protocol similar to HTTP [1]. Signalling in SIP is based on UTF-8 text messages. A message consists of a message header and an optional message body. Messages can be classified into either requests or responses. The original RFC 3261 presents six types of request (also called methods) methods: INVITE, BYE, ACK, OPTIONS, CANCEL, and REGISTER. Table 1 provides a description for each request. Other requests have been added to SIP to provide more functionality (e.g. for event subscription and notification, session transfer, etc.). In short, INVITE method is used to start a call, BYE method is used to end a call, OPTIONS method is used to enable the negotiation of the capability using SIP options negotiation, CANCEL option is used to abort a call setup, and REGISTER option is used to register the current location of user at the registrar.

SIP Request method	Description
INVITE	A session is requested to be setup using a specified media
BYE	A call is terminated by either party
OPTIONS	A Query to a server about its capabilities
CANCEL	Cancels any pending requests. Usually sent to a Proxy Server to cancel searches
REGISTER	Used by client to register a particular address with the SIP server
ACK	Message from client to indicate that a successful response to an INVITE has been received

Table 1: SIP Request Methods

If a SIP entity receives a request, it performs the corresponding action and then sends back a response to the originator of the request. Responses are three-digit status codes (similar as in http/1.1), categorised into six classes. Table 2 lists these classes for response codes. Concrete examples for response codes are '180-ringing', '302-moved temporarily', or '404-not found'.

	Description	Examples
1xx	Informational - Request received, continuing to process request.	180 Ringing 181 Call is Being Forwarded
2xx	Success - Action was successfully received, understood and accepted.	200 OK
3xx	Redirection - Further action needs to be taken in order to complete the request.	300 Multiple Choices 302 Moved Temporarily
4xx	Client Error - Request contains bad syntax or cannot be fulfilled at this server.	401 Unauthorised 408 Request Timeout
5xx	Server Error - Server failed to fulfill an apparently valid request.	503 Service Unavailable 505 Version Not Supported
6xx	Global Failure - Request is invalid at any server.	600 Busy Everywhere 603 Decline

Table 2: SIP Response Codes

2.4 Session Establishment (Call Setup)

Figure 3 illustrates a SIP session setup between two endpoints which belong to a single operator with a proxy server.

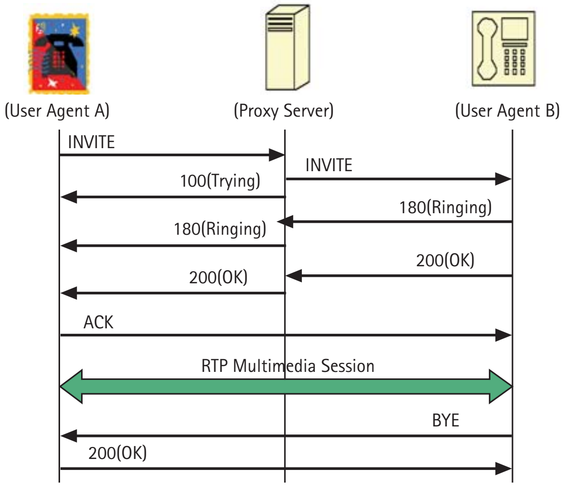

Figure 3: SIP Requests and Responses

The set up and termination of a voice connection between two users is illustrated in Figure 3. It shows the messages (requests and responses) that are being exchanged if user agent A wants to initiate a session with user agent B. Consider the case where two users belong to the same operator. Both user agents here use the same proxy. User agent A starts the request (INVITE), the proxy passes it on to the receiver (user agent B) and sends back a 100 (trying) response message. User agent B responds to the request first with a 180 (ringing) message, and eventually with a 200 (OK) after a user has picked up the phone. Both of these messages are forwarded to user agent A by the proxy. User agent A can now request the start of the media transfer (ACK). Note that the proxy is not needed for this. After a session has been established, user agents can communicate with each other directly. At the end of the conversation, some user agent (here: B) terminates the session by sending a BYE request to its counterpart.

Figure 4 illustrates the session establishment in the case where two different domain proxies are involved. In this example, it is assumed that user agent A and B exist in different domains and have different proxies. First, the user agent B needs to register with its local registrar (1) to be able to receive calls from the any user. The registrar stores the location information at a location server (2). When user agent A wants to call user agent B, it sends an INVITE-request to its local SIP-proxy (3) which passes on the request (possibly after a DNS lookup) to the proxy of user B's domain (4). The proxy in domain B needs to look up the IP-address of user agent B at the location server (5, 6) before it can send the request to user agent B (7). In this example, the response message for user agent A takes the same route back (8, 9, 10), possibly for billing purposes. The remaining steps (11, 12, 13) are performed in a similar manner as shown in Figure 3.

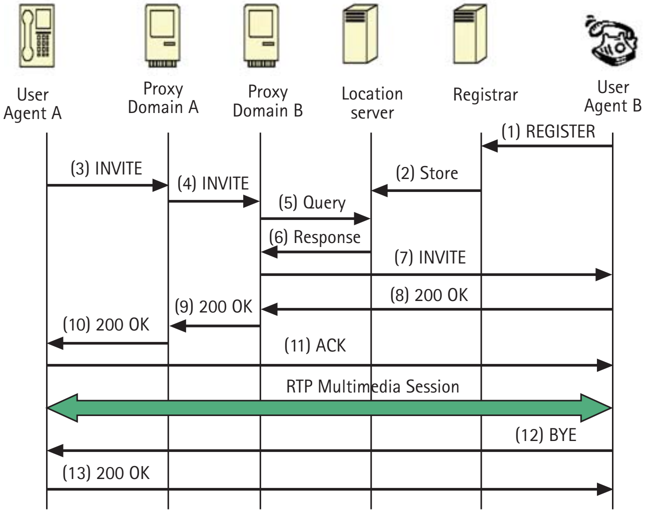

Figure 4: Session establishment of voice connection with two different domains

2.5 The Future of SIP

The SIP standard is accepted more and more by the IT community. The SIP protocol has shown a strong acceptance by the market as some service providers offer already free and non-free SIP-based VoIP and Instant Messaging services. For instance, the new Messenger for Windows launched by Microsoft, includes SIP-based telephony, presence, instant messaging and voice/video communication. SIP is gaining its popularity in the mobile world too. In the 3GPP consortium for 3G mobile networks, SIP has been chosen as a signalling protocol and Nokia has announced SIP support for its Series 60 platform of mobile terminals. Among these SIP providers, one can mention earthlink and iptel.org.

3 VOIP Spam

3.1 VOIP Spam Type

As a new emerging standard, SIP might also be the target of some spam attacks. As a consequence, identifying SIP spam and the mechanisms to combat are a very crucial task before the problem arises. SIP spam can take one of the following types [6].

- Call spam: This is the case of unsolicited messages for establishing voice, video or IM session. If the user should answer, the spammer proceeds to relay his message over the real time media. This type is the usual way used by telemarketers applied to SIP.

- IM spam: This form is similar to email spam, unsolicited IMs whose content contains the message that the spammer is trying to convey. The SIP MESSAGE request will be used here but also some other messages such as INVITE with text or html bodies. IM spam is most naturally sent using the SIP MESSAGE request. However, any other request which causes content to automatically appear on the user's display will also belong to the IM spam.

- Presence spam: This spam is similar to the previous one, i.e. unsolicited presence (subscribe) requests are sent to get on the buddy list of a user and send him IM or initiate other types of communications. This spam uses SUBSCRIBE requests for the presence event package in an attempt to get on the 'buddy list' or 'white list' of a user in order to send them IM or initiate other forms of communications. Unlike IM spam, presence spam does not actually convey content in the messages.

3.2 Characteristic of Spam

Spam could be worse than email and phone telemarketing, it immediately interrupts the user and it can be delivered to any user in any place. VoIP spam differs from e-mail spam in that it is significantly more obtrusive (a phone will actually ring with every spam message, possibly in the middle of the night). Furthermore, E-mails are 'pulled' from a server by the user, while VoIP calls are 'pushed' to the user. Authentication can only provide limited protection against spam. Certificate authorities would need a policy that revokes certificates of servers that are used for spamming, and they would need to do so very quickly.

3.3 Countermeasures to VOIP Spam

There are a lot of solutions to SIP spam; content filtering, white list, black list, consent-based communication, and identity authentication. In general, these solutions are some mechanisms developed to combat email spam and which could be adapted to SIP spam because of some similar natures. Among these solutions, it is worth mentioning content filtering, white lists, black list, consent-based communications, reputation system, and the authenticated identity.

3.3.1 Content Filtering

The most common type of spam protection used in email is based on content filtering. The spam filters analyse the content of the email, and search for clues that the email is spam, such as a specific word or sentence. Bayesian spam filters are in this category. This type of solution may not be an efficient method to combat VoIP spam for two reasons. First, the spam cannot be analyzed by the content filter before the user answers it. Second, it needs the real-time or offline speech or video recognition technology which might be impossible, considering the current state of the technology.

3.3.2 Black List

Black listing is an approach in which the spam filter maintains a list of addresses that identify spammers. These addresses may include either usernames (spammer@domain.com) or entire domains (spammers.com). The black list is unlikely to have effectiveness for SIP spam for two reasons. First, it is easy for the spammer to spoof the SIP address. Second, the spammer can obtain the new SIP address from any public provider. If an authenticated identity is used in the inter-domain communication, it may be difficult for spammer to forge the SIP identity. However, even in the case of using the authenticated identity, the spammer can obtain the new SIP addresses, which might cause black list to be useless.

3.3.3 White Lists

White lists are the opposite of black lists. It is a list of valid senders that a user is willing to accept email from. Unlike black lists, a spammer cannot change identities to obtain the white list. White lists are susceptible to address spoofing, but using strong identity authentication mechanism can prevent that type of problem. As a result, the combination of white lists and strong identity could be a good solution to VoIP spam. However, they do not provide a complete solution, since they would prohibit a user from ever being able to receive the call from someone who was not explicitly put on the white list. As a result, white lists require a solution to the 'introduction problem' - how to meet someone for the first time, and decide whether they should be placed in the white list. In addition to the introduction problem, white lists demand time from the user to manage. In IM systems, white lists have proven exceptionally useful at preventing spam. This is due to the fact that the white list exists naturally in the form of the buddy list. Users don't need to manage this list just for the purposes of spam prevention; it provides general utility, and assists in spam prevention for free. IM systems also have strong identity mechanisms due to their closed nature. The introduction problem in these systems is solved with a consent framework, described next.

3.3.4 Consent-based Communications

A consent-based solution is used in conjunction with white or black lists. That is, if user A is not on user B's white or black list, and user A attempts to communicate with user B, user A's attempt is initially rejected, and they are told that consent is being requested. Next time user B connects, user B is informed that user A had attempted communications. User B can then authorise or reject user A. These kinds of consent-based systems are used widely in presence and IM but not in email. This solution should be combined with a secure authenticated identity mechanism, which is a pre-requisite. Since most of today's IM systems are closed, sender identities can be authenticated. This kind of consent-based communications has been standardised in SIP for presence, which allows a user to find out that someone has subscribed. If they were extended to cover IM and calling, it may not be useful, since instead of being bothered with content, in the form of call spam or IM spam, users are bothered with consent requests.

3.3.5 Reputation System

A reputation system is also used in conjunction with white or black lists. Assume that user A is not on user B's white list, and they attempt to contact user B. If a consent-based system is used, B is prompted to consent to communications from A, a reputation score might be displayed in order to help user B decide whether or not they should accept communications from user A. Traditionally, reputation systems are implemented in highly centralised messaging architectures; the most widespread reputation systems in messaging today have been deployed by monolithic instant messaging providers. Reputation is calculated based on user feedback. For example, a button on the user interface of the messaging client might empower users to inform the system that a particular user is abusive. Reputation systems based on negative reputation scores suffer from many of the same problems as black lists, since effectively the consequence of having a negative reputation is that you are blacklisted. Reputation systems based on positive reputation, where users praise each other for being good, rather than blaming each other for being bad, have some similar drawbacks. Collectives of spammers, or just one spammer who acquires a large number of identities, could praise one another in order to create an artificial positive reputation. Unlike negative reputation systems, however, positive reputation is not circumvented when users require a new identity, since basing authorisation decisions on positive reputation is essentially a form of white list. So, while positive reputation systems are superior to negative reputation systems, they are far from perfect. Intriguingly, though, combining presence-based systems with reputation systems leads to an interesting fusion. The 'buddy-list' concept of presence is, in effect, a white list - and one can therefore probably infer that the users on one's buddy list are people whom you are 'praising'. This eliminates the problem of user inertia in the use of the 'praise' button, and automates the initial establishment of reputation.

3.3.6 Other Sophisticated Solutions to SIP Spam

More sophisticated methods against VoIP-spam include payments-at-risk, memory bound functions and turing tests. Payments-at-risk is a concept that depends on micropayment. Unfortunately, there is no micropayment standard in the Internet today, but a connection to the existing telephone networks could solve this problem. Memory bound functions would consume computing power at the sender's device for each SIP-message being sent, making huge amounts of spam in a short period of time expensive, while not bothering the average user. A turing test is a challenge for a sender intended to distinguish automatically generated messages from human interaction. For example, a caller could be asked to enter the result of a numerical calculation into his phone. Though these methods pose some additional problems, they appear promising to be used against VoIP-spam.

3.3.7 SIP Authenticated Identity

Authentication is only a prerequisite to VoIP-spam because it can only provide assurance about a sender's identity, not about a sender's trustworthiness level [2, 4, 5]. To be effective against spam, authentication should be used in conjunction with an anti-spamming policy which is enforced at all participating parties. The existing mechanisms do not allow an administrative domain to verify the

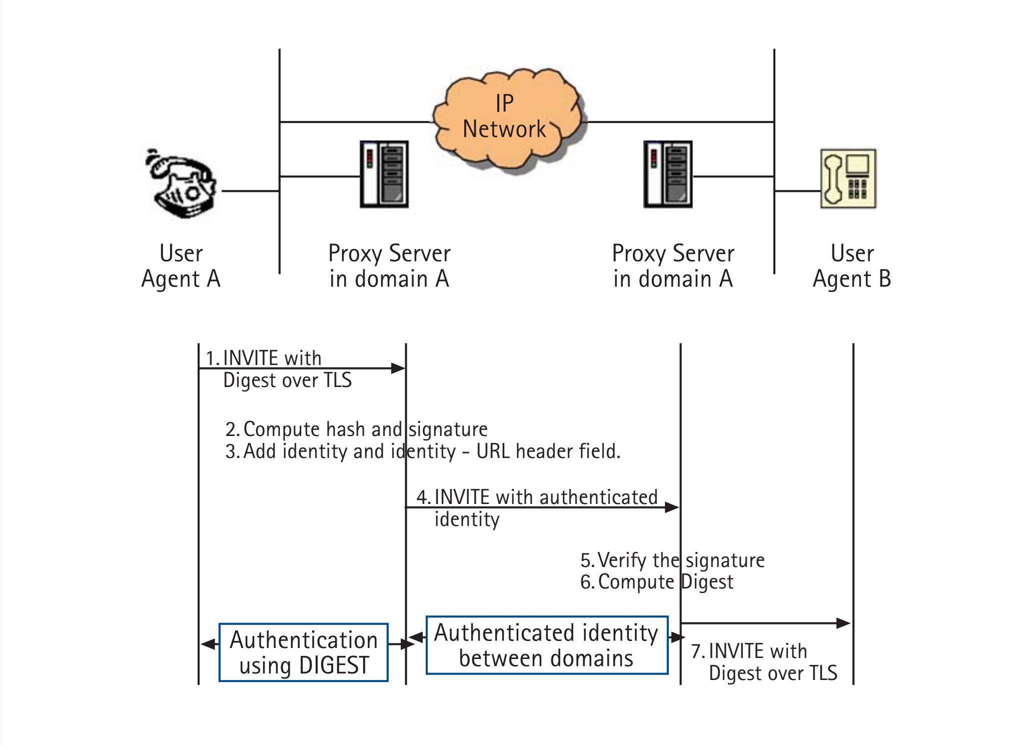
IP Network
User Agent A
Proxy Server in domain A
Proxy Server in domain A
User Agent B
1. INVITE with Digest over TLS
2. Compute hash and signature
3. Add identity and identity - URL header field.
4. INVITE with authenticated identity
5. Verify the signature
6. Compute Digest
Authentication using DIGEST
Authenticated identity between domains
7. INVITE with Digest over TLS

Figure 5: Overview of the authenticated identity

identity of the originator of a request. This mechanism recommends practices and conventions for identifying end users in SIP messages, and proposes a way to distribute cryptographically secure authenticated identities.

This authenticated identity provides an informative (non-normative) high-level overview of the mechanisms. Imagine the case where Alice, who has the home proxy of example.com and the address-of-record sip:alice@example.com, wants to communicate with sip:bob@example.org.

Alice generates an INVITE and places her identity in the From header field of the request. She then sends an INVITE over TLS to an authentication service proxy for her domain. The authentication service authenticates Alice (possibly by sending a Digest authentication challenge) and validates that she is authorised to assert the identity which is populated in the From header field. This value may be Alice's AoR, or it may be some other value that the policy of the proxy server permits her to use. It then computes a hash over some particular headers, including the From header field and the bodies in the message. This hash is signed with the certificate for the domain (example.com, in Alice's case) and inserted in a new header field in the SIP message, the 'Identity' header.

The proxy, as the holder of the private key of its domain, is asserting that the originator of this request has been authenticated and that she is authorised to claim the identity (the SIP address-of-record) which appears in the From header field. The proxy also inserts a companion header field, Identity-Info, that tells Bob how to acquire its certificate, if he doesn't already have it.

When Bob's domain receives the request, it verifies the signature provided in the Identity header, and thus can validates that the domain indicated by the host portion of the AoR in the From header field authenticated the user, and permitted them to assert that From header field value. This same validation operation may be performed by Bob's UAS.

This method defines a new role for SIP entities called an authentication service. The authentication service role can be instantiated by a proxy server or a user agent. Any entity that instantiates the authentication service role must possess the private key of a domain certificate, and must be capable of authenticating one or more SIP users that can register in that domain. Commonly, this role will be instantiated by a proxy server, since these entities are more likely to have a static hostname, hold a corresponding certificate, and have access to SIP registrar capabilities that allow them to authenticate users in their domain.

In conclusion, sending domain computes hash over some headers (including From), then signs with certificate for domain, and inserts in new header. Receiving domain may retrieve public key of sending domain via URI given in new Identity-Info header. Receiving domain authenticates call.

3.3.7.1 Authentication Service Behaviour

Entities instantiating the authentication service role performs the following steps, in order, to generate an Identity header for a SIP request:

- Step 1: The authentication service extracts the identity of the sender from the request. The authentication service takes this value from the From header field which is referred to here as the 'identity field'. The authentication service verifies the SIP or SIP URI by comparing it to the domain(s) for which it is responsible. If the authentication service is not responsible for the identity in question, it should process and forward the request normally, but it must not add an Identity header.

- Step 2: The authentication service must determine whether or not the sender of the request is authorised to claim the identity given in the identity field. In order to do so, the authentication service must authenticate the sender of the message. Some possible ways in which this authentication might be performed include: If the authentication service is instantiated by a SIP intermediary (proxy server), it may challenge the request with a 407 response code using the Digest authentication.

- Step 3: The authentication service should ensure that any pre-existing Date header in the request is accurate. Local policy can dictate precisely how accurate the Date must be, a recommended maximum discrepancy of ten minutes will ensure that the request is unlikely to upset any verifiers. If the Date header contains a time different by more than ten minutes from the current time noted by the authentication service, the authentication service should reject the request. Finally, the authentication service must verify that the Date header falls within the validity period of its certificate.

- Step 4: The authentication service must calculate the identity signature and add an Identity header to the request containing this signature. After the Identity header has been added to the request, the authentication service must also add an Identity-Info header. The Identity-Info header contains a URI from which its certificate can be acquired.

- Finally, the authentication service must forward the message normally.

3.3.7.2 Verifier Behaviour

In order to verify the identity of the sender of a message, an entity acting as a verifier must perform the following steps, in the order specified here.

- Step 1: The verifier must acquire the certificate for the signing domain. Implementations supporting this specification should have some means of retaining domain certificates in order to prevent themselves from needlessly downloading the same certificate every time a request from the same domain is received. Certificates cached in this manner should be indexed by the URI given in the Identity-Info header field value. SIP entities should discover this certificate by dereferencing the Identity-Info header. The client processes this certificate in the usual ways, including checking that it has not expired, that the chain is valid back to a trusted CA, and that it does not appear on revocation lists. Once the certificate is acquired, it must be validated following the procedures in RFC3280.

- Step 2: The verifier must follow the process to determine if the signer is authoritative for the URI in the From header field.

- Step 3: The verifier must verify the signature in the Identity header field, following the procedures for generating the hashed digest-string.

- Step 4: The verifier must validate the Date, Contact and Call-ID; recipients that wish to verify Identity signatures must support all of the operations described there. Furthermore, it must ensure that the value of the Date header falls within the validity period of the certificate whose corresponding private key was used to sign the Identity header.

3.3.8 Possible Solutions to Each SIP Spam

Table 3 illustrates the possible pragmatic solutions to the three SIP spam types, where 'X' indicates 'don't work', '△' indicates 'may be applicable', and 'O' indicates 'works well'. However, if these solutions can be applied to the SIP spam problem, they seem to be insufficient and need to be combined with each other or with some other techniques to provide robustness to SIP spam.

	Call spam	IM spam	Presence spam
Content filtering	X	0	X
Black list	△	△	△
White list (with an authenticated identity)	△	△	0
Content-based communication	X	X	0
Reputation system	0	0	0

Table 3: Applicable solutions versus SIP spam types

4 Conclusions

Spam in VoIP will become serious problems, since it interrupts the user immediately. There are several solutions to combat the spam in VoIP. However, some of the most effective anti-spam techniques used on e-mail spam are hardly useful for combating VoIP-spam: white list, black list, and content filtering. Using white list would limit the set of callers to those on the white list: a VoIP user would not receive calls from someone not on the white list. Black lists are only of limited use because spammers can either forge any new SIP-addresses, or create new SIP-addresses which don't exist on the blacklist. For content filtering, a semantic analysis of real-time audio or video traffic would be necessary, which are not probably feasible Thus, while content-filtering is very effective on e-mail spam, it cannot be used on VoIP-spam. It clearly indicates that using a certain type of solution alone cannot give a complete solution to combat efficiently spam and instead, combining several solutions would be a possible solution to combat VoIP spams. Moreover, the prerequisite to the solutions above is an authenticated identity in the header, that is, an authenticated identity between the domains should be developed to reduce the occurrence of SIP spams, which are originated from the senders who forge the From field in the header.

Acknowledgement: This paper was supported by both the ITRC project, MIC and the Project, ETRI, South Korea.

5 References

[1] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and E. Schooler, 'SIP: Session Initiation Protocol', RFC 3261, June 2002.

[2] Peterson, J., 'Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)', draft-ietf-sip-identity-03 (work in progress), September 2004.

[3] M. Handley and V. Jacobson. SDP: Session Description Protocol. RFC 2327, April 1998.

[4] N.J Croft and M.S Olivier, A Model for Spam Prevention in IP Telephony Networks using Anonymous Verifying Authorities, April 2005.

[5] Katz, Schwartz, Sterman, Tschofenig. 'SPAM for Internet Telephony (SPIT) Prevention using the Security Assertion Markup Language (SAML)'. 2005. Accessed: 10 April 2006. http://www3.ietf.org/proceedings/05nov/IDs/draftschwartz-sipping-spit-saml-00.txt

[6] J. Rosenberg, C. Jennings, J. Peterson, 'The Session Initiation Protocol (SIP) and Spam', draft-rosenberg-sipping-spam-01, October 2004.