Electronic Preprint for Journal of Information Processing Vol.21 No.2

Regular Paper

Trust-based VoIP Spam Detection based on Calling Behaviors and Human Relationships

Noppawat Chaisamran[1,a)] Takeshi Okuda[1,b)] Suguru Yamaguchi[1,c)]

Received: May 5, 2012, Accepted: November 2, 2012

Abstract: Spam over Internet Telephony (SPIT) will become a serious threat in the near future because of the growing number of Voice over IP (VoIP) users, the ease of spam implementation, and the low cost of VoIP service. Due to the real-time processing requirements of voice communication, SPIT is more difficult to filter than email spam. In this paper, we propose a trust-based mechanism that uses the duration of calls and call direction between users to distinguish legitimate callers from spammers. The trust value is adjustable according to the calling behavior. We also propose a trust inference mechanism in order to calculate a trust value for an unknown caller to a callee. Realistic simulation results show that our approaches are effective in discriminating spam calls from legitimate calls.

Keywords: VoIP, SPIT, Trust

1. Introduction

With the growth of broadband connectivity, the use of VoIP is a new trend in voice communication. Like email technology, VoIP systems are also susceptible to abuse by malicious parties who initiate unsolicited advertising calls or SPIT [1]. But SPIT is a much bigger problem than email spam because the callee is directly denied service by the incoming call. For instance, a spam email that arrives at an inbox at 2 a.m. will not disturb the user. However, a ringing phone at 2 a.m. will disturb most users. SPIT is becoming popular because of its cost-effectiveness for spammers. As we know, the traditional telephony call spam already exists in the form of telemarketer calls. But its volume is not as much as email spam because of the cost. However, the cost is dramatically lower when switching to SPIT for many reasons: low call fee, low hardware cost, no boundary of international calls, etc. Also, unlike spam in email systems, the problem of spam in VoIP networks has to be solved in real time. Many techniques devised for email spam filtering rely upon content analysis, but in the case of VoIP analysis after picking up the phone is too late. Any delay due to anti-SPIT processing would degrade the quality of service. This places strong limitations on what operations can actually be performed on incoming voice packets in terms of the speed and complexity of the analysis. Therefore, SPIT prevention is one of the greatest challenges for future large-scale deployments of VoIP telephony.

To be effective, then, a SPIT prevention system has to meet the following basic requirements. First, it must minimize the probability of blocking legitimate calls while maximizing the probability of blocking SPIT calls. Second, it should minimize the additional effort imposed on users. Third, it should be deployed without any significant changes in the existing infrastructure. And finally, it should be as general as possible to allow it to be applied despite barriers such as different cultures and languages. However, it is likely that no prevention system meets all of these general requirements. For example, many methods of identifying human beings, such as the challenge-response schemes, require significant interaction with the caller and are therefore too intrusive for the user. There is insufficient incentive to adopt such measures since they are so inconvenient.

To fulfill the SPIT filtering requirements, we propose a novel mechanism based on call duration and direction to distinguish between legitimate users and spammers. We observed that the call duration of a legitimate user is significantly longer than that of a spammer, and that a spammer will typically receive no calls or few calls from other users. We use this pattern to calculate a trust value for each individual user. Long call duration indicates high trust. One can assign trust values to friends by calling them. A trust value is calculated by comparing one's call duration with the average call duration among friends. The following scenario shows how our trust value can be constructed. Assume that Alice makes a call to Bob and Carol lasting, 5 and 15 minutes respectively. After comparing with the average value, the trust value of Carol will be higher than Bob's because the call duration between Alice (trustor) and Carol (trustee) is longer than with Bob. Unlike other trust-based detection systems, our trust value is automatically assigned to each user and adjusted by human calling behavior. It allows avoiding the biasing problems that occur when one legitimate user incorrectly rates another legitimate user as a spammer.

In case of an unknown caller, trust values inferred from other users in the callee's community are used to calculate a trust value

[1] Nara Institutute of Science and Technology, Ikoma, Nara 630–0192, Japan
[a)] noppawat-c@is.naist.jp
[b)] okuda@is.naist.jp
[c)] suguru@is.naist.jp

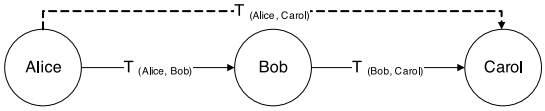


Fig. 1 Trust inference from Carol to Alice via Bob.

for this caller. We use two theories to propagate a trust value among nodes in the VoIP network. First, from the trust definition [2], trust can be inferred from one node to another node via a social relationship. For example, if Alice knows Bob, and Bob knows Carol, then Alice can use the relationship path to infer a trust rating for Carol, as shown in **Fig. 1**. Second, we use the concept of six degrees of separation to limit the relationship links when propagating a trust value. This small world hypothesis refers to the idea that everyone is on average approximately six steps away from any other person on Earth [3]. However, a Microsoft research team came up with the idea of using data from their Messenger client to analyze the characteristics of a social network. They examined a data set of 240 million people with 180 billion different connection pairs [4]. They found the average path length between users to be 6.6, implying that 78% of the world's population can be linked in seven steps or less. So, in this work, we limit the trust inference range to 7 hops from a callee to a caller. Based on this concept, we infer that if there are no relationships between the callee and caller within this range, the call can be classified as an unwanted call. This is different from other trust-based systems because in other trust-based systems a trust value is inferred at most two hops away. This may cause an increase in false alarms.

Our detection system at the VoIP operator calculates a trust value of each caller before establishing a call to a callee. If this value is lower than the predefined SPIT threshold, the call is rejected. This method aims to detect SPIT calls before call establishment while requiring the least possible interaction with the caller and the callee. Moreover, this system does not require changing the existing VoIP protocols. The VoIP operator can apply this system directly to any VoIP technology.

At the same time, there is a trade-off in managing spam callers because sometimes they are beneficial and sometimes detrimental to the revenue of a service provider. From the perspective of a VoIP service provider, spam callers are also a type of customer and sometimes they are even valuable for increasing revenue. For example, some organizations may purchase an advertising service for broadcasting their information to targeted customers. Targeted customers who prefer to subscribe to this content will classify this call as a legitimate call while it is a spam call for another user. If a service provider fails to design ways to manage spam calls that allow for such services, they can cause a loss of revenue when a system blocks all suspect traffic. As far as the author knows, there is no academic research which considers this business aspect.

In this work, one can subscribe to an advertisement service while preserving spam prevention for other users. In general trust-based systems, when a user wants to subscribe to an advertisement service, he must keep the advertisement service caller in his buddy list to bypass the detection process. However, this might produce false negatives when the trust value of this node is inferred for another friend. With our trust computation algorithm, on the other hand, we can handle spam detection even in this extreme case because the trust value of an advertiser will be reduced automatically.

The rest of this paper is organized as follows. Section 2 surveys related works that deal with spam prevention based on trust. Sections 3 and 4 describe the trust calculation and the trust inference methodology respectively. An evaluation of the method by realistic simulation and the results are discussed in Section 5. We also discuss some concerns about this method in Section 6. Finally, Section 7 concludes the paper and introduces our future work.

2. Related Works

This section presents the state of the art of techniques to prevent and mitigate spam in VoIP networks.

Blacklisting is an approach whereby the spam filter maintains a list of call numbers that identify spammers. However, collecting these numbers can be tedious and users can still receive unwanted phone calls from numbers not on the list.

Whitelisting is the opposite of blacklisting. It is a list of valid senders that a user is willing to accept calls from. Unlike blacklists, a spammer cannot change identities to get around the white list [5]. However, this approach greatly reduces the usability of VoIP because legitimate callers not in the list cannot contact the user at all.

Quittek *et al.* propose a hidden Turing test technique to identify spammers [6]. It requires a SIP server or a user agent to check the Real-time Transportation Protocol (RTP) before establishing a call session between a caller and a callee.

Vinokurov and MacIntosh propose a VoIP spam detection based on recognizing abnormalities in signaling message statistics [7]. A caller that sends too many call setup requests, while at the same time receiving too many or too few call termination requests in a relatively short time, is assumed to be a spammer. This approach has to maintain the signaling behavior of every caller in the system which needs to be updated for every call that a node makes or receives.

Dantu and Kolan propose a combined filter technique based on trust and reputation for detecting spam [8]. They combined techniques such as rate limiting, Bayesian learning, and the concept of social networks for predicting the nature of a call. During the learning period, human intervention was required to identify unwanted callers. Even though this technique can isolate spammers, it suffers from two major drawbacks. First, trust and reputation in this system were assigned to an entire domain rather than individual users. Thus, if a particular domain has a lot of spammers and few legitimate users, those legitimate users would be penalized along with the rest of the spammers. Second, the system relies exclusively on the users' feedback to report spam domains. If the users do not report spam, those domains can continue sending spam. To solve these problems, our trust value is individually and automatically assigned to each friend in the buddy list based on the direct experience of a user. The manual feedback reports from users are not required.

Balasubramaniyan *et al.* propose to use call duration and social network graphs to establish a measure of reputation for callers, named CallRank [9]. In this filtering scheme, call duration is the main factor deciding the credibility of the caller. It is used along with the Eigentrust algorithm to develop a global view of the reputation of all users who either belong to or interact with a domain. A callee can decide to answer or reject a call based on this mechanism. However, CallRank produces false positives when a new legitimate user joins the VoIP system. Because he has no social network linkage in that system, all his calls will be classified as spam calls. Due to its centralized perspective, there are some problems. First, the users are usually reluctant to give a negative rating because of the other's negative rating. Second, if a user has a bad reputation rating, the system will discard its old identity. The third problem is that users can increase their reputation artificially by creating fake identities and using them to give themselves a high rating. Unlike this work, our technique acts as a decentralized perspective to avoid these problems. Each user is responsible for evaluating the trust of other users based on their direct interactions.

Another study proposed a collaborative reputation-based voice spam filtering framework [10]. This approach used the cumulative online presence duration of a VoIP user as a reputation value. The authors automatically classified calls shorter than 20 seconds as spam calls. However, with this system, spammers can increase their reputation easily by maintaining a connection with the VoIP server. Our trust value, on the other hand, is calculated by using call duration and call direction. Assuming that user A and user B are friends, when user A calls user B, the trust value from user A will be automatically assigned to user B according to the duration of a conversation. In the case of a spam call, a spammer calls user A. The trust value will never be given to a spammer because user A does not call a spammer. If a spammer wants to increase his trust value, he needs to trick user A into calling him back. Therefore, it is difficult to alter the trust value in our proposed technique.

3. Trust

Trust has been traditionally used in solving the authentication problem in ad-hoc, peer-to-peer, and decision support systems. The basic idea is to let parties rate each other and use the aggregated rating about a given party to derive a trust score, which can assist other parties in deciding whether or not to interact with that party in the future. In voice communication, trust represents an abstract modeling of the caller's and the callee's past interactions. In this work we attempt to formalize a structure similar to human intuitive behavior for detecting SPIT based on a trust relationship with the caller and calculate trust automatically from an individual aspect. To begin, we define trust using three properties. First, trust represents a callee's belief in a caller's reliability based on his/her own direct experiences. Second, the trust of a caller can be increased or diminished over a period of time based on the interaction with the callee (trustor). Third, trust can be derived from outgoing calls. This means we assign a trust value to a user when we call him.

In VoIP system, there are many factors that can be used to identify a malicious node, as shown in **Table 1**. However, some of these factors are quite ineffective. A sophisticated spammer can observe a filtering system and then adjust his/her spam behavior in order to break the detection criteria. For example, the error of calling occurs when the destination address does not exist. Because spammers randomly generate the target callee IDs, there is a possibility that some of them do not exist. Then, the call error rate of a spammer will be high. However, a spammer can collect the existing numbers from Yellow Pages or buy them from the black market to reduce this rate. For a call rate and a call interval time, they are the parameters that can be adjusted easily. IP addresses and Uniform Resource Identifiers (URI) of end users are also used to classify a spam call. However, these parameters can be spoofed by a sophisticated spammer [11]. Then, a spammer can subvert many spam detection systems such as blacklists and white lists.

Table 1 Calling characteristics.

Factor	Legitimate	SPIT
Call duration	Irregular	Usually very short
Call direction	Bidirectional	Unidirectional
Call error rate	Low	High
Call rate	Low	High
Call interval time	Irregular	Very irregular

Call duration and direction are used to calculate a trust value because they are reliable. Call duration is the important information used to distinguish a spammer because, in general, people do not like to talk with a spam caller for long. Therefore, the call duration of a spam call will be significantly shorter than legitimate call. And most spam calls are unidirectional communication because a spammer calls target users only. There is a rare case that a normal user calls back to a spammer. Therefore, if a caller has a high rate of unidirection call, he might be classified as a spammer. So these two characteristics cannot be altered by a spammer except by using social engineering techniques. The spammer has to trick the user into calling him back and maintaining a long call in order to increase his trust value. In addition, since trust is derived from direct interaction between two people, they can express the extent of trust reliably.

Additionally, using call duration has the following advantages: it is implicit, quantifiable, easily verifiable, and easily understood. In VoIP networks, the VoIP server keeps track of call duration for billing purposes. Our detection system does not require any alteration of the VoIP infrastructure. Moreover, our system provides an automatically assigned trust value. It avoids unfair rating problems.

The cumulative call duration of each friend in the buddy list is calculated at a time t that might be weekly, monthly, or every billing period, depending on the VoIP operator. The raw trust value of a friend i (R_i) can be computed by comparing this cumulative value (C_i) with the average call duration of all friends as shown in Eq. (1) where n is the number of friends in the buddy list.

$$R_i = \frac{C_i}{\sqrt[n]{C_1 \cdot C_2 \cdot \ldots \cdot C_n}} \tag{1}$$

In our system, we use the range of numbers between 0 and 1 to represent the trust value, where 0 indicates a spammer and 1

indicates a normal user. In Eq. (1), assume we have n users in the buddy list. R_i and C_i represent the raw trust value and cumulative call duration to a friend i respectively. If R_i is greater than 1, it is rounded down to 1. This means that we give higher trust to friends with a duration of calls longer than average. In our system, we use the geometric mean as an average function.

In human society, trust depends on past experiences with a person. So, to compute the final trust value, we combine the raw trust value and historical trust, which is the trust value computed at a previous time. Let $T_{i(t)}$ denote the trust value of a friend i at time t. $T_{i(t-1)}$ is a historical trust value. The weighted value (α) is the exponentially weighted sum where $0 < \alpha < 1$. According to human reasoning, we consider past experience before deciding something. Therefore, in this work, we will give a higher weight value for historical trust than for raw trust by defining α less than 0.5. The trust calculation formula is shown in Eq. (2).

$$T_{i(t)} = \alpha R_{i(t)} + (1 - \alpha) T_{i(t-1)} \quad (2)$$

3.1 Distinguishing Legitimate Users and Advertisers in the Buddy List

Equation (2) is a degraded function applied to a user in the buddy list who has not been contacted for a long time. It affects both long-time-no-call friends and subscribed advertisement services. However, our raw trust formula can help us improve trust values for friends by requiring a sufficient level of call duration as shown in **Fig. 2**. In this simulation, we calculate trust monthly (cf. **Table 2**). The cumulative call durations of friends A and B are 100 and 30 minutes respectively every month. We called friend C only once, for 20 minutes, in the last month. Ad represents an advertisement service the user has subscribed to. The result shows that it is not necessary to establish a very long call in order to increase the trust value. The algorithm is sensitive to human behavior: bidirectional communication and normal call duration. The raw trust value is high when the call duration is close to the average call duration among friends. In this case, the average duration of a call at $t = 12$ is 39.15, then $R_C = 0.51$. The raw trust is high enough to make the trust value of user C increase dramatically. At this time, the trust value of friend B is also increased because of high raw trust. Note that we assign the weighted function, α, as 0.2 for all simulations.

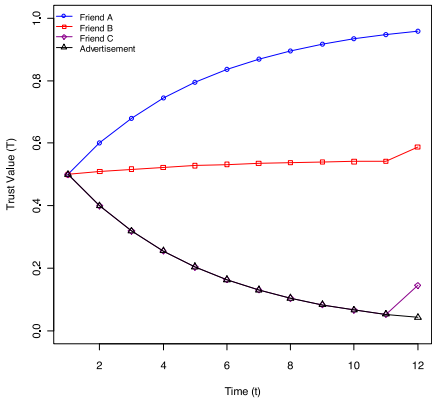


Fig. 2 Trust value of each friend in a buddy list.

Table 2 Call pattern of each user in Fig. 2.

Time	Total Call Duration (min)			
	A	B	C	Ad
1	100	30	0	0
2	100	30	0	0
3	100	30	0	0
:	:	:	:	:
10	100	30	0	0
11	100	30	0	0
12	100	30	20	0

These formulas affect not only subscribed advertisement services but also infected users. For example, a user is infected silently by malware that registers itself as a user's friend in the buddy list. The initial trust value of a friend in the buddy list allows this malicious node to call a user and form a malicious community to subvert the trust-based system. This is called a Sybil attack where a spammer obtains many accounts for corrupting their trust value. However, the trust value of this malicious node has been continuously decreasing because no user has called this node. So creating new accounts will not help at all.

4. Trust Inference

A trust value in Section 3 is only assigned to friends in the buddy list who have direct interaction. In the real world, there is a possibility that a call will come from an unknown person. If we use only a trust value in a callee's buddy list, we will not be able to properly classify a caller who is not on this list. To solve this problem, we apply a situation found in human daily life to our system. Generally, when encountering an unknown person, it is common for people to ask trusted friends for opinions about how much to trust this new person. Therefore, to improve the spam detection scalability, we gather the trust values of an unknown caller from other friends in the network who already know about that person in order to classify a call. Hence, the primary property of trust in this work is transitivity. The trust values of friends in the buddy list will be shared to other nodes through a relationship path in the VoIP network. These inferred trusts will be used when a caller and a callee do not have a direct relationship.

We assume that a VoIP system is represented as a social network. It contains nodes that are user agent clients (UAC). Every node maintains a buddy list. A VoIP social network is constructed by connecting a node to all the nodes in its buddy list. To meet our requirements, the buddy lists are kept in a central database on the provider side. Each buddy list is shared to others within the provider side during the trust computing process. For improving scalability, the buddy lists need to be distributed throughout the global telephony network. There are already some possible techniques available for sharing the buddy lists. For instance, Trust path discovery, the IETF Internet draft, offers address book propagation within SIP messages [12]. In general, though, we may need some agreements between different VoIP providers that allow each other to exchange any information related to the users' buddy lists without privacy concerns. We also assume that any network that interconnects with others should make use of strong SIP identity as described in RFC 4474 [13].

4.1 Trust Propagation

Trust can be inferred from one person to another. In human reasoning, a person is much more likely to believe his/her friends than a stranger. Likewise, a trusted acquaintance will also trust the beliefs of his/her friends, so it is possible to find a path of friends from trustor to trustee with appropriate discounting [14]. For example, if Alice trusts Bob completely and Bob trusts Carol completely, then Alice may trust Carol, but not necessarily completely. So, we decide to use a multiplicative function for propagating a trust value, as shown in Eq. (3).

$$T_{callee,caller} = \prod_{m \in path}^{caller} T_{m,m+1} \tag{3}$$

According to the seven degrees of separation, any pair of nodes in a random network will be connected by a relatively short chain of random acquaintances. The number of intermediates is finite: here, the trustor and trustee are connected through not more than seven intermediaries. This implies that if such mutual chains of acquaintances are used to determine the initial trust between a pair of entities, then the method will scale up well because these chains are likely to be short. In this work, if a callee cannot evaluate the trust value of a caller within seven hops, this implies that the caller is an unknown person.

4.2 Trust Path Selection

In a real network, there may be many trust paths between a caller and a callee. Computing all trust paths is time consuming and requires significant computing resources that would affect the real-time requirement of voice communication. To avoid these problems, we select only one trust path that produces the highest trust value between a caller and a callee within the limitation of relationship length, as shown in Eq. (4); where m is a node along the trust path within the hop count limitation.

$$T_{callee,caller} = \max \left(\prod_{m \in path}^{caller} T_{m,m+1} \right)_{hop<limit} \tag{4}$$

We apply a landmark-based method for finding the highest trust path between two nodes in the VoIP network. This method uses precomputed information to provide fast estimates of the actual distance in a very short period of time. We select a subset of nodes as *landmarks* in an offline step and then compute distances from the landmarks to every node. To select a good landmark, we choose a vertex that is very central in the graph with many of the shortest paths passing through it. We follow the landmark selection algorithm proposed in Potamias's work [15]. With this method, the cost of the offline computation is $O(md)$, where m and d are the number of edges and landmarks respectively. Then, the online computation is only $O(d)$.

5. Evaluation

The accuracy of the SPIT detection system is the ratio of correctly classified calls. We calculate sensitivity and specificity to measure the performance of the detection system. Sensitivity Eq. (5) is the proportion of correctly detected spam calls to all actual spam calls. With higher sensitivity, fewer actual cases of SPIT go undetected. Specificity Eq. (6) is the proportion of correctly detected legitimate user calls to all actual legitimate calls. Higher specificity indicates that the system detects legitimate calls more accurately.

$$Sensitivity = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{5}$$

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive} \tag{6}$$

This section includes a description of the system design, the simulation information, and 4 experiments performed.

5.1 System Design

The detection module is integrated into the VoIP server at the carrier side to collect the calling statistic and buddy list for each user. The initial trust value ($Known_{init}$) is set to all friends in the buddy list. It can be any value that is greater than the predefined SPIT threshold. A call from friends present in the buddy list is accepted automatically. For an unknown caller who is not in the blacklist, the system will calculate the trust value of this caller inferred from other nodes in the community. If the inferred trust is higher than the SPIT threshold, this call will be connected to the callee. Otherwise, it will be rejected. The callee can decide to save this caller in his buddy list or not. If not, the system will store this contact and its trust value in a hidden buddy list for future use. The hidden buddy list can reduce the computation task when the same caller, who is in neither the buddy list nor the blacklist, contacts the callee the next time. For an unknown caller of whom trust cannot be computed or a new VoIP user, the system will assign an initial trust value ($Unknown_{init}$) and then connect the call to the callee. This can eliminate the barrier for new users who do not have a trust value assigned by other users. The relationship between initial values and the SPIT threshold is $Known_{init} > Unknown_{init} > SPIT_{threshold}$. If a callee finds that a call is spam, he can put this caller number in the blacklist. The trust value of every node in the blacklist is zero and is also used in the trust inference process.

5.2 Simulation Parameters and Datasets

An evaluation of our work in the real world would require call logs from a VoIP system along with actual cases of VoIP spam. Call logs are hard to come by due to privacy concerns and VoIP spam is still not widespread enough. Instead, we simulated the system with a synthetic call workload to evaluate its effectiveness, ensuring that the simulations model real world call characteristics as closely as possible. The main objective of the simulations was to study the performance of the proposed technique in terms of spam detection accuracy.

Many researches [8], [9] simulated their testbed by only randomly choosing call parties. But in our evaluation, we are concerned about the realism of the VoIP user network. So, we used two kinds of datasets in the simulation: directed random graphs and the Epinion social network [16]. A random graph is a graph that is generated by some random process. It is obtained by starting with a set of n vertices and adding edges between them at random. We considered the nodes as users and edges as call directions. We adjust the clustering coefficient of the graphs to eval-

uate our detection accuracy. The latter dataset, Epinions, is the who-trusts-whom online social network of a general consumer review site Epinions.com. Members of the site can decide whether to trust each other. All the trust relationships interact and form the web of trust, which is then combined with review ratings to determine which reviews are to be shown to the user. The details of these datasets are shown in **Table 3**. We referred to the published reports by NTT East Corporation [17], [18] to decide the average call duration and number of calls of legitimate users. The reports show the average VoIP call duration of home users and company users. The average of a home user is 204 seconds and that of a company user is 124 seconds. The average number of calls of a legitimate user is 2 calls/day. Other simulation parameters are shown in **Table 4**.

Next we will describe 3 experiments that show the detection accuracy of the proposed technique and prove that the concept of the seven degrees of separation can be applied to a SPIT detection system. The forth experiment shows the CPU time of our approach when calculating a trust value of a caller using various relationship lengths. Next, we compare our proposed method with the call rate limiting method and prove that the remaining variables in Table 1 are ineffective in detecting a spam call. Finally, we compare our proposed with a fuzzy-based detection method.

5.3 Single Spammer with Different Clustering Coefficient Network

In graph theory, the clustering coefficient (CC) quantifies how well connected are the neighbors of a vertex in a graph. Therefore, the different networks have different clustering coefficient values. Then, the objective of this experiment is to evaluate the spam detection efficiency with different network characteristics. We observed the performance with 5 different CC random graphs: 0.1–0.5. **Figure 3** compares the results over the different datasets. From these results, the CC does not affect our spam call classification but does slightly influence the legitimate call classification. The accuracy of legitimate call classification in a high CC network is greater than in a low CC one. In a high CC network, the nodes are close together, so the trust path length between a caller and a callee is shorter than in a low CC network. Since a multiplicative function is used for the trust inference Eq. (4), the trust value of a shorter path will be greater than for the longer path. Therefore, the false positives in a high CC network will be lower. The effectiveness of an anti-SPIT system can also be evaluated by observing how quickly it identifies a spammer and isolates it by preventing it from placing any further calls. **Figure 4** illustrates more details of the sensitivity of two different CCs. From these graphs, the spammer was detected completely after the first few time intervals.

Table 3 Datasets.

Dataset	Nodes	Edges	CC
Random Graph	1,000	Varied	Varied
Epinions	75,879	508,837	0.23

5.4 Multiple Spammers

We used the Epinion dataset as the initial VoIP network. We observed the detection performance with different numbers of spammers, and found the detection time that the system takes to completely classify the spammers. **Figure 5** shows the average sensitivity and specificity of the dataset with 1% to 10% of spammers. The spammers are added in the network at the beginning stage of the simulation. Then, each spammer randomly generates the spam calls to the victims. From these results, the number of spammers did not affect the detection accuracy. According to the sensitivity graphs in **Figs. 6** and **7**, more spammers affected the spam classification only the first few time intervals.

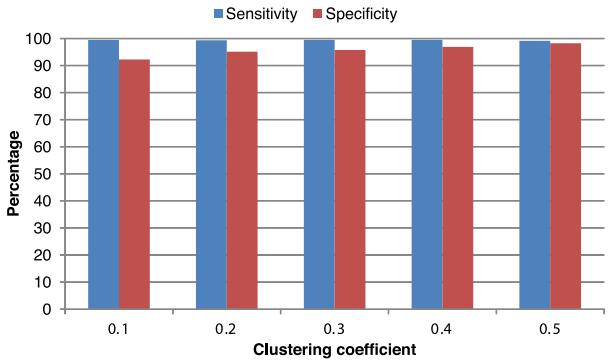


Fig. 3 Average sensitivity and specificity with different clustering coefficients.

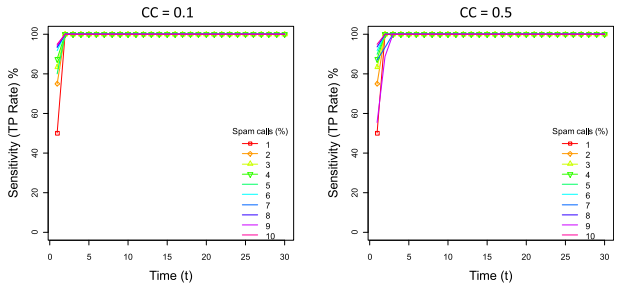


Fig. 4 Sensitivity of each different clustering coefficient graph.

Table 4 Simulation parameters.

Parameter	Value	Description
Normal call duration (sec.)	124–204	Generated by using a normal distribution
Spam call duration (sec.)	<10	Generated by using a normal distribution
No. of normal calls per user	2 calls/day	The IP phone usage statistics from NTT Corporation [17], [18]
Initial trust value of friend	0.5	This value will be added to a friend in a buddy list automatically when a user adds his friend for the first time
Trust of unknown caller	0.4	This value is assigned to an unknown caller in the case of no direct relationship
SPIT threshold	0.25	The optimized value of our proposed technique
No. of advertisement subscribers	Varied	Some normal users are randomly selected and then some spammers are added to their buddy list

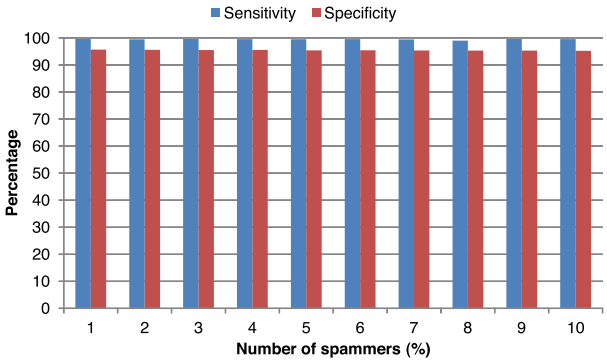


Fig. 5 Average sensitivity and specificity with different number of spammers.

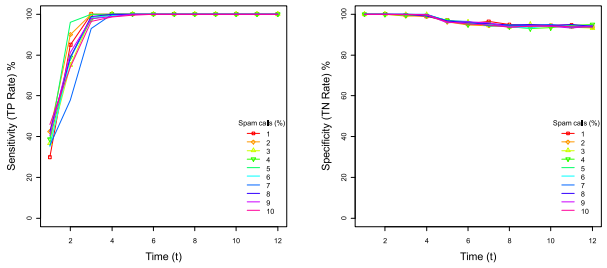


Fig. 6 Sensitivity and specificity with multiple spammers (1%).

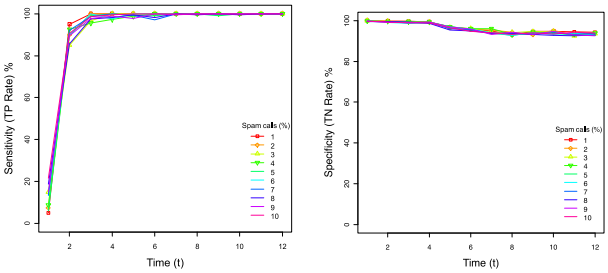


Fig. 7 Sensitivity and specificity with multiple spammers (10%).

Because we accept the first call of an unknown caller, when the number of spammers increased, sensitivity dropped because every first call of the new spammer was a false negative. However, the sensitivity increased significantly over the next few time intervals because some nodes in the network had enough information about the spammers. After this period, the actual trust values of the spammers were inferred accurately.

From the specificity graphs, although the number of spammers increased, the detection system still accurately detected legitimate calls. The specificity may have dropped for two reasons. First, the trust value of a long-time-no-call friend will be decreased if we have not called him for a long period. So, if the trust value of this friend is referred to other nodes, it may produce a false positive. Moreover, the number of calls per user in this simulation was only two. This means there might be many friends in the buddy list that were not called. The false positives will increase in some periods. Second, the long trust path also affects the trust value of the legitimate users. As stated in Section 4, a multiplicative function is used for propagating a trust value from caller to callee. The length of the propagation path will diminish the trust value. Consequently, this value may be lower than the SPIT threshold, and then the system produces a false positive. The results of this experiment were captured over

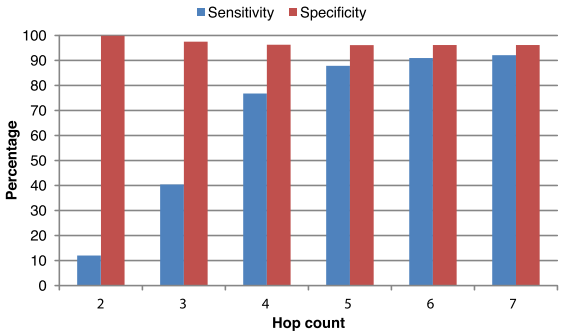


Fig. 8 Average sensitivity and specificity with different relationship length limitations.

only 12 time points; it seems that the specificity is continuously decreasing. However, the actual specificity is not stable, sometimes increasing and sometimes decreasing depending on these two reasons. The average is more than 95%.

5.5 Relationship Length Evaluation

In this experiment, we examine the effects of changing the relationship length limitation on the SPIT detection system. We adjusted the relationship length limitation in our trust inference Eq. (4), from 2 to 7 hops. **Figure 8** illustrates the average sensitivity and specificity of each hop length. The results show that the 7-hops limitation produced the highest accuracy rate for detecting a spam call because, when we consider a longer path, we can get more and more information about a spammer. Thus, there is a higher probability of producing a correct result. In contrast, with a short relationship length limitation, even though the 2 hops limitation produces the most accurate legitimate call classification, the number of false negatives is too high. Referring to the proof of seven degrees of separation [4], the relationship length between a spammer and a normal user should be longer than 7 hops because, in general, a spammer is not a friend or acquaintance of a normal user. Therefore, a 2-hops limitation system is likely to accept many spam calls, raising the false negative score.

5.6 Computation Time Evaluation

Though, filtering at the carrier has to be done in real-time, the pre-computation of the trust values has to be made at regular intervals. For example, the trust value of each friend in the buddy list is updated at weekly, monthly, or at every billing period depending on the VoIP operator as described in Section 3. To evaluate the performance, we show the computation time when a callee needs to calculate a trust value of friends (single hop) and unknown callers who need the inferred trust value from other nodes in the network. This CPU time also includes finding the trust path between two nodes with seven hops limitation. We did this experiment on the Epinion social network dataset. The simulation is run on five machines that have Intel Xeon Quad-core 2.93 GHz (x2) and 24 GB of memory. **Figure 9** shows the average computation time of a caller's trust value with different trust path length. From the result, our system spends a short time to calculate the trust value of both known and unknown callers. The more distant the relationship, the more computation time is needed.

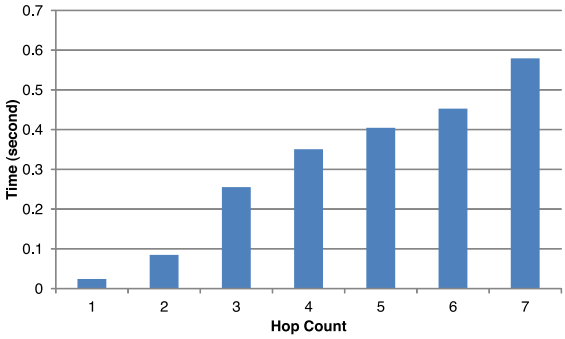


Fig. 9 The average CPU time when calculating a trust value for different path lengths.

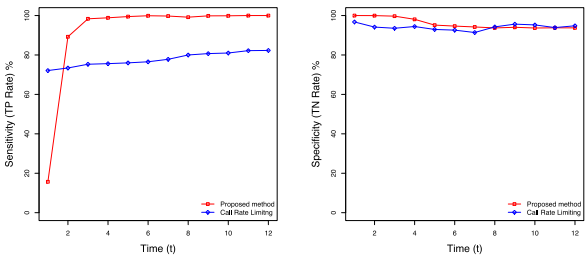


Fig. 10 Sensitivity and specificity of the proposed method and call rate limiting method.

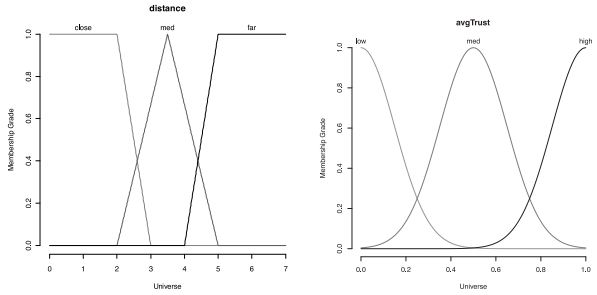


Fig. 11 (left) Distance fuzzy set, (right) average trust fuzzy set.

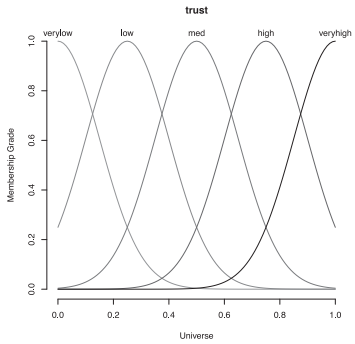


Fig. 12 Output: inferred trust of an unknown caller.

5.7 Call Rate Limiting Comparison

We add more parameters in this simulation. The call duration of SPIT is generated by a Poisson distribution between 10 and 180 seconds. The 10 seconds duration case is when a person accepts a SPIT and hangs up immediately after realizing that it is spam. The 180-second duration is when a telephone answering machine answers a spam call. The call intervals of a spam call range from 3 to 600 seconds. The call destinations of a spam call are random, to both valid and invalid addresses.

In the call rate limiting approach, we calculate a SPIT score by using the formulas in [19]. The results of the comparison are shown in **Fig. 10**. The average sensitivity of the call rate limiting method is lower than our proposed method because a spammer can subvert the detection mechanism by imitating a call characteristic of legitimate call behaviors. Even if a spammer makes a long call to a user, this will not affect our system. As mentioned in Section 3, the incoming call duration is not used to calculate a trust value in our system.

5.8 Fuzzy-based Technique Comparison

In this subsection, we compare our proposed method with the Fuzzy-based approach. We use the fuzzy logic to calculate trust value between callees and unknown callers in the VoIP network. The following two fuzzy descriptors is used to establish the methodology for inferring trust: distance and average trust.

(1) Distance: The distance refers to the path distance from a callee to a caller. The farther the distance (the farther relationship), the lower the trust level. **Figure 11** (left) shows the fuzzy graph of the distance. It is divided into close, medium, and far depending on the hop count between a caller and a callee. The criterion for dividing these three levels is based on the study of Lekovec which claims that anyone is connected to another person up to seven intermediates [4].

(2) Average trust: The average trust refers to the average value of trust that a caller receives from other users who are directly connected to themselves in the inference paths. Figure 11 (right) shows the fuzzy graph of the average trust. It is divided into low, medium, and high, depending on the trust value between 0 and 1.

After the value of the fuzzy descriptors are determined for each case of the fuzzy rules below, we can have five results: very low, low, medium, high, and very high, as shown in **Fig. 12**. In this way, we can assign the trust of an unknown caller as one of five levels.

Rules:

- If distance is far and average trust is low, then trust is very low.
- If distance is far, then trust is low.
- If average trust is low, then trust is low.
- If distance is medium or average trust is medium, then trust is medium.
- If distance is close, then trust is high.
- If average trust is high, then trust is high.
- If distance is close and average trust is high, then trust is very high.

Figure 13 shows the performance evaluation results. The fuzzy-based method produced a low sensitivity rate. This means it produced a high rate of false negatives. Based on this result, the detection accuracy of the fuzzy-based method is lower than our proposed method.

6. Discussion

One topic not covered so far in this paper is the resistance of the proposed SPIT detection system to a Sybil attack. In such an attack, a spammer subverts the trust-based system by creating a large number of entities and using them to gain a high trust value.

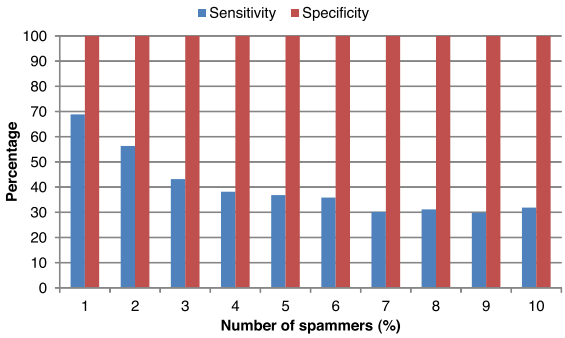


Fig. 13 Sensitivity and specificity of the fuzzy-based approach.

A trust-based system's vulnerability to a Sybil attack depends on how cheaply trust establishment can be generated. However, with our proposed technique, Sybil nodes would be difficult to construct because of the cost of service usage. Trust in our system is calculated based on the call duration of outgoing calls. If a spammer S_1 wanted to maintain a high trust value for a neighbor spammer S_2, S_1 would have to call S_2 frequently. So, if the spammers wanted to construct a large spammer community, they would have to call each other frequently to maintain high trust values. As a result, the spammers would have to pay a lot of money, making it counterproductive for their business. This provides a financial disincentive to spammers.

Another concern is a spoofing of call identity. In the worst case, a sophisticated spammer may be able to break the SIP authentication mechanism and spoof his identity as a legitimate user. In this work, we do not propose any prevention technique against this attack because this is a protocol specification issue. However, spam activity can be noticed easily through a user's bill. The victim should contact an operator to solve the problem.

The one problem with our technique occurs when a user accidentally marks his friend as a spammer and that friend has no social linkage with other users (or he is a newcomer). The system will raise a false alarm when the trust value of this friend is propagated to other users. Therefore, calls from other users to this friend are required in order to increase the trust value.

7. Conclusion

To deal with VoIP advertising calls, we have presented an approach combining trust ratings and the characteristics of callers. A trust value based on call duration is calculated for each friend in the buddy list. This technique provides a simple way to use call duration as an automatically assigned trust value based on human behavior. Due to the reliability of this value, it is difficult for a spammer to subvert the system. The proposed technique is in keeping with real call behavior and human reasoning. The trust values of long-time-no-call friends and spammers are decreased by default. However, the trust value of a legitimate user can be increased with calls lasting long enough. This supports the bidirectional communication characteristic of a legitimate user call. To extend the detection scalability, we further proposed a trust propagation method in case a caller and a callee do not have a direct relationship. Based on realistic simulation results, we found that the proposed technique can detect all SPIT completely after a short learning period while keeping a low false positive rate. We also demonstrated that even when the number of spammers was increased, the accuracy of spam and legitimate call detection were still higher than 98% and 95% respectively. The size and the relationship characteristics among nodes in the network did not affect the detection efficiency. In addition, the computation time of our approach is low. These imply that our proposed technique can be applied to real VoIP networks.

Our proposed detection system meets all the basic requirements introduced in Section 1. The sensitivity and specificity from the experiments show that our system can minimize the probability of blocking legitimate calls and maximize the probability of blocking spam calls. Considering the execution process, the VoIP providers do not need to modify their infrastructures because our system does not change any protocol stacks. Moreover, the proposed system does not require additional effort by a user because the trust value is automatically calculated in the background. Therefore, it is very convenient for all users.

The IP Multimedia Subsystem (IMS) is a new trend for multimedia communication. It merges cellular networks with the Internet and existing circuit switched phone systems. Introduction of the IMS will make VoIP popular and will increase SPIT. Therefore, in future work, we will extend our proposed method to the detection of spam calls in an IMS system.

References

[1] Hansen, M., Hansen, M., Moller, J., Rohwer, T., Tolkmit, C. and Waack, H.: Developing a Legally Compliant Reachability Management System as a Countermeasure against SPIT, *3rd Annual VoIP Security Workshop* (2006).

[2] Golbeck, J. and Hendler, J.: Inferring binary trust relationships in Web-based social networks, *ACM Trans. Internet Technol.*, Vol.6, No.4, pp.497–529 (online), DOI: 10.1145/1183463.1183470 (2006).

[3] Watts, D.J.: *Small worlds: The dynamics of networks between order and randomness*, Princeton University Press, Princeton, NJ (1999).

[4] Leskovec, J. and Horvitz, E.: Planetary-scale views on a large instant-messaging network, *Proc. 17th International Conference on World Wide Web*, *WWW '08*, New York, NY, USA, ACM, pp.915–924 (online), DOI: 10.1145/1367497.1367620 (2008).

[5] Rosenberg, J. and Jennings, C.: RFC 5039: The Session Initiation Protocol (SIP) and Spam (2008).

[6] Kusumoto, T., Chen, E. and Itoh, M.: Using Call Patterns to Detect Unwanted Communication Callers, *IEEE/IPSJ International Symposium on Applications and the Internet*, pp.64–70 (online), DOI: 10.1109/SAINT.2009.19 (2009).

[7] Vinokurov, D. and Maclntosh, R.W.: U.S. Patent 7,307,997 - Detection and mitigation of unwanted bulk calls (spam) in VoIP networks (2007).

[8] Dantu, R. and Kolan, P.: Detecting spam in VoIP networks, *Proc. Steps to Reducing Unwanted Traffic on the Internet on Steps to Reducing Unwanted Traffic on the Internet Workshop*, *SRUTI'05*, Berkeley, CA, USA, USENIX Association, pp.31–37 (online), available from ⟨http://dl.acm.org/citation.cfm?id=1251282.1251287⟩ (2005).

[9] Balasubramaniyan, V.A., Ahamad, M. and Park, H.: CallRank: Combating SPIT Using Call Duration, Social Networks and Global Reputation, *Proc. Fourth Conference on Email and Anti-Spam*, *CEAS'07* (2007).

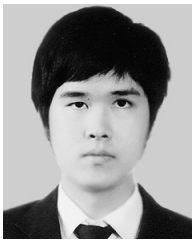
[10] Zhang, R. and Gurtov, A.: Collaborative Reputation-based Voice Spam Filtering, *Proc. 2009 20th International Workshop on Database and Expert Systems Application*, *DEXA'09*, Washington, DC, USA, IEEE Computer Society, pp.33–37 (online), DOI: 10.1109/DEXA.2009.95 (2009).

[11] Cao, F. and Malik, S.: Vulnerability Analysis and Best Practices for Adopting IP Telephony in Critical Infrastructure Sectors, *IEEE Communications Magazine*, Vol.44, No.4, pp.138–145 (online), DOI: 10.1109/MCOM.2006.1632661 (2006).

[12] Ono, K. and Schulzrinne, H.: IETF Internet-Draft: Trust Path Discovery (2006).

[13] Peterson, J. and Jennings, C.: RFC 4474: Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)

(2006).
[14] Guha, R., Kumar, R., Raghavan, P. and Tomkins, A.: Propagation of trust and distrust, *Proc. 13th International Conference on World Wide Web*, *WWW'04*, New York, NY, USA, ACM, pp.403–412 (online), DOI: 10.1145/988672.988727 (2004).
[15] Potamias, M., Bonchi, F., Castillo, C. and Gionis, A.: Fast shortest path distance estimation in large networks, *Proc. 18th ACM Conference on Information and Knowledge Management*, *CIKM '09*, New York, NY, USA, ACM, pp.867–876 (online), DOI: 10.1145/1645953.1646063 (2009).
[16] Richardson, M., Agrawal, R. and Domingos, P.: Trust Management for the Semantic Web, *Proc. 2nd International Semantic Web Conference*, *ISWC '03*, Berlin, Germany, Springer, pp.351–368 (2003).
[17] NTT: IP Phone Usage Status and Network Information 2009, NTT East Corporation (online), available from ⟨http://www.ntt-east.co.jp/info-st/network/traffic_h21/index.html⟩ (accessed 2011-12-22).
[18] NTT: NTT Telecommunication Services 2009, NTT East Corporation (online), available from ⟨http://www.ntt-east.co.jp/info-st/subs/ekimu/h21/index.html⟩ (accessed 2011-12-22).
[19] Kim, H.J., Kim, M.J., Kim, Y. and Jeong, H.C.: DEVS-Based modeling of VoIP spam callers behavior for SPIT level calculation, *Simulation Modelling Practice and Theory*, Vol.17, No.4, pp.569–584 (online), DOI: 10.1016/j.simpat.2008.09.008 (2009).



Noppawat Chaisamran received a B.Sc. degree in ICT from Mahidol University, Thailand, and a M.E. degree in Information Science from Nara Institute of Science and Technology (NAIST), Japan. He is currently a Ph.D. student in the Graduate School of Information Science, NAIST. His research interests are in the area of IP telephony and telecommunication security.

Takeshi Okuda received his M.E. and D.E. degrees in information science from Osaka University, Japan in 1998 and 2011 respectively. He is currently an assistant professor in the Graduate School of Information Science, Nara Institute of Science and Technology, Japan. His research interests include virtual machines, virtual networks and their security. He is a member of IEEE.

Suguru Yamaguchi received his M.E. and D.E. degrees in computer science from Osaka University, Japan, in 1988 and 1991, respectively. He has been working as a university faculty, since 1990, in Osaka University and Nara Institute of Science and Technology, Japan. Since 2000, he has been a Professor of the Graduate School of Information Science, NAIST. From 2004 to 2010, he was appointed as an Adviser on Information Security, in the Cabinet Secretariat of the Government of Japan. He also has been working aggressively for making and running JPCERT/CC since 1996, and APCERT since 2003. Currently he is working also as a steering committee member of FIRST for 2011/2012. His research interests include technologies for information sharing, multimedia communication over high-speed communication channels, network security and network management for the Internet.