Detecting Spam in VoIP Networks

Ram Dantu, Prakash Kolan
Dept. of Computer Science and Engineering
University of North Texas, Denton
{rdantu, prk0002}@cs.unt.edu

Abstract

Voice over IP (VoIP) is a key enabling technology for the migration of circuit-switched PSTN architectures to packet-based networks. The problem of spam in VoIP networks has to be solved in real time compared to e-mail systems. Many of the techniques devised for e-mail spam detection rely upon content analysis and in the case of VoIP it is too late to analyze the media after picking up the receiver. So we need to stop the spam calls before the telephone rings. From our observation, when it comes to receiving or rejecting a voice call people use social meaning of trust and reputation of the calling party. In this paper, we describe a multi-stage spam filter based on trust, and reputation for detecting the spam. In particular we used closed loop feedback between different stages in deciding if the incoming call is a spam or not. For verifying the concepts, we used a laboratory setup of several thousand soft-phones and a commercial grade proxy server. We verified our filtering mechanisms by simulating the spam calls and measured the accuracy of the filter. Results show that multistage feedback loop fares better than any single stage. Also, the larger the network size, the harder to detect a spam call. Further work includes understanding the behavior of different controlling parameters in trust and reputation calculations and deriving meaningful relationships between them.

1. Introduction

Defending the country's telecommunication networks requires cooperation between service providers, equipment vendors, enterprises and the government. Currently VoIP infrastructure is being aggressively deployed in enterprises and residential areas without much security analysis. It is estimated that by 2006 IPPBX deployments will outnumber the traditional PBX deployments. This can be a clear recipe for a possible disaster to critical infrastructure like telecommunications network. There is very little work reported in the literature on how to defend VoIP against attacks like DOS (Denial of Service), session hijacking and termination, monitoring and eavesdropping, service disruption, toll fraud, identity fraud, spamming etc.. Also, the impact of vulnerabilities on a large scale (e.g., several millions of IP phones) VoIP network is not well understood. Hence it is imperative that we investigate the vulnerabilities and threats to residential communities due to the new real-time services like VoIP. All the threats need to be addressed before VoIP services are deployed on a mass scale because the lack of security has the potential of delaying and disrupting next generation voice communications.

The possibility of VoIP network replacing the PSTN network depends on enhancing the existing IP network to carry voice traffic. With the usage of IP network to carry voice traffic, existing problems on the IP network holds for the VoIP network too. One of the major issues that the present day IP networks face is the problem of controlling spam - the unsolicited bulk mail. Spam control has been perceived to be the most important problem of research with present traditional e-mail systems. The problem of spam is increasing day-by-day and recent results indicate that of all the e-mail that is circulating in the internet right now, as high as 80% of that is spam (junk or unsolicited messages). A study[12] by Radicati Group a California based consultancy states that "last year daily global e-mail traffic via the Internet amounted to 56.7 billion messages per day. Of that, the firm says, 25.5 billion messages were spam, or about 45%. Daily traffic is expected to rise above 68 billion messages per day, and more than half of it--52%--will be spam. With this magnitude of junk or spam messages circulating all through the internet every day, the problems like low availability, network congestion etc. would not be a surprise. In VoIP networks, spam refers to the unsolicited voice calls, which end up consuming many resources on the end VoIP phones and intermediate VoIP infrastructure components. With the advent of VoIP and openness of internet, the spamming attacks on the VoIP infrastructure are estimated to take the world to the same position as the traditional e-mail systems with respect to e-mail spam. While there are many techniques that have been designed to avoid e-mail spam, such techniques can be of limited application to avoid the problem of voice spam. The reason lies in the real time application of VoIP. *The problem of spam in VoIP networks has to be solved in real time compared to e-mail systems. Compare receiving an e-mail spam at 2:00 AM that sits in the Inbox until you open it next day morning to receiving a junk voice call at the same time. Moreover, many of the techniques devised for e-mail* *spam detection rely upon content analysis. The same with VoIP calls is already late.*

2. Background

Most of the present day e-mail spam filters employ content filtering as both the signaling and media arrives at the spam filter at the same time. Content filtering is not useful in VoIP spam analysis as media flows in after the two participating entities have agreed upon to start the communication and would be too late to filter the call. This poses a serious challenge of detecting spam in real time with the available signaling messages and a danger of increasing the end-to-end delay on the communication between participating entities during call set up.

There is a lot of literature on spam filtering for the present day e-mail infrastructure. Spam filters have known to use a wide variety of filtering mechanisms like text classification and rule based scoring systems, Bayesian filtering, pattern recognition, identity recognition etc [1][2][3][6][7][8][10]. Cohen[8] recommends spam filtering based on a set of rules for identifying the message body content. Features of the message are identified and scored to compute the total spam score of the e-mail spam message and the messages having a score more than a given threshold is identified to be spam e-mail. Large quantities of spam and legitimate messages are used to determine the appropriate scores for each of the rules in the rule-based scoring systems. Sakkis[7] suggests probabilistic inference for calculating the mutual information index (MI) and a vector of attributes having the highest MI scores is constructed for spam identification. The Memory-based algorithms attempt to classify messages by finding similar previously received messages by storing all training instances in a memory structure, and using them directly for classification. Soonthornphisaj[6] spam filtering technique works by constructing the centroid vector of the e-mail and is classified based on its similarity measured between the centroid vector of spam e-mail class and the legitimate e-mail class. Rigoutsos[10] suggests pattern discovery scheme for identifying unsolicited e-mails by training the system with a large number of spam messages. The system matches the e-mail message with the available patterns; more the patterns are matched more is likelihood that the message is spam. Sahami[1] proposes that incorporating domain specific features in addition to identifying various textual phrases and probabilistically inferring the spam behavior of the constructed message vector leads to a more accurate spam analysis. All the solutions account for some sort of identification and filtering based on message body content. These solutions do not have direct applicability to VoIP systems, as content filtering cannot be achieved before the users communicate.

The standard for VoIP, SIP (Session Initiation Protocol), establishes an open model where users have IP phones linked to the pervasive Internet infrastructure. To realize the objective of receiving a call from a person anywhere in the world, static junk call filtering mechanisms have to be replaced with adaptive learning systems. These systems apart from learning spam behavior have to account for modeling human behavior. For example, whenever a phone rings, we first look into our state of mind (or presence), and see if the call is from a trusted party. If we do not know who the caller is, then we guess the trust and reputation of the calling party. After picking up the telephone, we query and move forward only when we are satisfied with the response. Similarly, our proposed research uses an intelligent call admission control consists of the presence of the called party (e.g., state of mind, location), the rate of incoming calls from a given user (by computing first and second order differentials), trust between calling and called parties (using Bayesian theory), and reputation graphs based on the social network of the calling party. In addition, all the above techniques are combined in deciding whether to accept/reject a call or forward it to voice mail. We propose a *Voice Spam Detector(VSD)* acting as a separate process running along with the domain proxy and processes the incoming call and informs the proxy about the spam nature of the call based on past feedback from the end users in its domain.

3. Methodology

VoIP spam detection process does not pertain to a single technique of detection. The detection needs to be done using various techniques at different stages. At each stage the spam detection process qualified by that stage eliminates most of the spam and any subsequent spam left through or forwarded would be quarantined in the next stage. The techniques employed at each stage would determine the spam behavior of the call and with the available feedback information from the called domain end user, the call is either stopped or forwarded to the user voicemail box. The basic criterion on which the call processing depends is on whether a similar call had been designated as a spam or a valid call before.

3.1 Architecture

The architecture behind the spam detection process would take into account all the user preferences of wanted and unwanted people, his or her presence of mind, the reputation and trust of the calling party. The basic architecture would be as shown in Figure. Each stage represents a technique based on which the call would be quarantined by employing a specific set of mechanisms and user feedback. Each stage of the spam detection process gives feedback about the possibility of the call to be spam and the collective inference of all stages would give the spam nature of the call that can be used for quarantining the call.

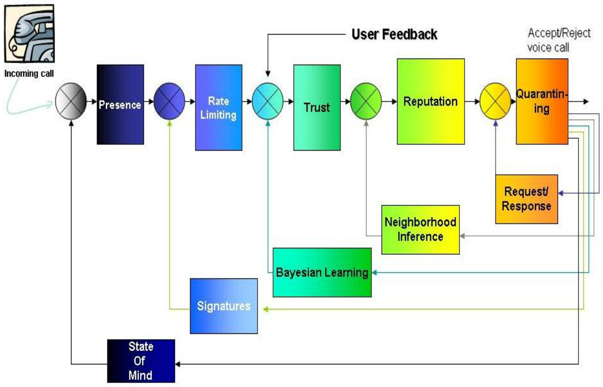

Fig 1: Functional Elements of VSD

3.2 Functional Elements in Voice spam Detection

Presence: Whenever we receive a voice call, we normally pick up the telephone receiver depending on our state of mind. So, the definition of a spam call depends on one's 'state of mind'. Hence the first step in this filtering process is the characterization of spam depending on the state of mind. For example, a state of mind changes depending on a location, do-not-disturb-me mode, follow-me mode, and 911-emergency-mode. One example of assessing the state of mind is to synchronize the system with an individual's calendar. The filtering process that takes place during this stage is based on static/dynamic rules (like firewall rules).

Rate Limiting: Based on known traffic patterns, signatures can be used to detect the rate of incoming calls. For example, velocity and acceleration values (first and second order derivative) of the number of arriving calls from a given user/host/domain can be used as a detection mechanism. That is, when the velocity/acceleration reaches a certain threshold, the drop rate can be updated through feedback control. As expected, the sooner we detect a change in the incoming pattern based on signatures, the faster there will be a reduction in the spread of the spam. Once spamming is identified, PID (Proportional Integral Control) feedback control can be used to reduce the velocity of spreading. This method of detection is useful not only in deterring spamming attacks but also in DOS (denial of service, where large number of messages sent in a short period of time) attacks. All the results have been discussed in[5].

Black and White Lists: Most of the spam detection is done using a set of valid and invalid signatures. These signatures would make the Spam detection server know which calls the server has to forward and which calls the server has to block. This is a direct way of quarantining the calls where the end user would specify a set of entities from which it is always ready to receive calls encoded in white lists and a different set of entities from which it would like to see all calls being blocked that are encoded in Blacklists. The entities might be any of the end user or an end soft-phone or a domain. Depending upon the specification of the end user, the specified calling users would be allowed or denied calling. The lists are customized. i.e. each end user would have the flexibility of specifying its own entries. Entries differ in each of the lists specified by different end users and thus would have no bearing whatsoever of influencing the call forwarding or blocking of other end users. i.e. each end user would be guaranteed of forwarded or denied calls based on its own customized list. The Voice Spam Detector would let forward all the calls from the trusted elements in the white lists and block all the calls from un-trusted elements in the blacklist.

The black and white lists are constructed using user feedback to the VSD. When after forwarding the call, the user responds with a spam feedback message saying that the present call was a spam call to the VSD, the VSD adds the new entry to the black list and any future call with the same parameters is directly blocked at the server and is not forwarded. On the other hand, if the user specifies that the present call is important to it and want to receive any such calls in the future with these parameters, the entry is added to the user white list and any future calls with the same parameters are directly forwarded to the end user.

Bayesian Learning: Learning the behavior of the participating entities would let us make many intelligent decisions regarding the call. The behavior of the participating entities can be learnt during the course of a period of time. The behavior can be estimated by their past history of calling to the called party's domain end users. This process of observing the calling party's behavior over a period of time is termed as Learning. Learning as such represents an abstract modeling of the calling party's past behavior. The observed behavior over the period of time would classify the participating entities as spam producing or valid.

For an incoming call, the VSD would examine the participating entities like the call source (end user, host, domain etc.), participating proxies in routing etc with the help of fields like "from", "to", "record route", "via". VSD checks for any spam behavior associated with any of the participating entities by looking up trust information available for those entities. The trust information would be available if any of the entities has a history of calling an end user in the analysis domain. The spam probability of the call(i.e., associated trust level of the call) can be computed using Bayesian inference techniques [1]. The spam probability of an incoming call is P(X | C = spam) for a message X= $\{x_1,x_2,x_3 \ldots x_n\}$ and can be calculated by

$$\frac{P(C = spam)\prod_{i=1..n} P(x_i = spam)}{\sum_{k=spam,valid} P(C = k)\prod_{i=1..n} P(x_i = k)}$$

where each of $x_1..x_n$ represent different identifiers in the header of a signaling message, like "From", "To", "Via" "Record Route", and "Contact Info"). VSD would be filtering out the calls if the spam probability of the call would be greater than the permissible limit or tolerance level. Otherwise, the call is forwarded to the actual recipient of the call and the VSD waits for a feedback from the recipient. All the call processing depends on the end users reaction on the just forwarded call. The recipient responds with a message about the nature of the call. If the recipient responds with a message saying that the present call is a spam call, the VSD logs the call source information for future spam analysis. Future calls with any of the above participating entities would have a high degree of spam probability and more chances of getting stopped at VSD. On the other hand, if the recipient responds with a valid call message, the trust of the participating entities is updated to depict more valid probability i.e. less spam probability. Often, the called party does not know the calling party and hence there is no history of trust for a specific caller. In this context, we can infer the reputation of the calling party by using social networks.

The permissible limit or tolerance level is chosen by giving a preference of valid calls over spam calls i.e. the number of spam messages that can be let in so as to minimize the blocking of valid calls called "False Positives". The main aim of any spam filtering technique should be to minimize the "False Negatives"(spam calls let in as valid) keeping the false positives to zero. This ratio of valid calls to permitted spam calls would give a measure of the permissible limit. And any call that exceeds the permissible limit would be classified as spam and quarantined.

Social Networks and Reputation: Social networks can be used to represent user relationships that can be derived along the paths of the network. These relationships are transitive and transparent [4]. If Alice is related to Bob and Bob is related to Charles, then with a high degree of confidence, an argument can be made that Alice is related to Charles. These social networks can be used to infer the associated relations between the social elements. With respect to a VoIP service user, the user's social network represents the associated and trusted neighbors from which the user is willing to receive calls.

With respect to a proxy, a graph can be generated using the neighboring proxies and their users. Subsequently this graph can be used in deriving the reputation of a calling party. Reputation implies social understanding. Reputation is derived from trusted peers (e.g., nearest proxies reachable in one hop) while the trust is calculated based on the past history. The peer proxies would derive reputation by their trusted peer proxies, and this would continue until the last proxy in the "via" list or the proxy that is reachable from source by one hop is reached. Based on the reputation inference from the peer proxies of the source and the entities in-between, the reputation can be inferred. If R (a,b) gives a's reputation on c, R(a,c) = Θ(R(a,b),R(b,c)) for all (b) in trust-neighbors of (a). We believe that Θ is a Bayesian inference function on the proxies bearing the topology depicted by the graph. For a given call to an end user in the receiving domain, the reputation of the domain from which the call originated is inferred and the spam probability of the call obtained by trust inference is updated based on reputation inference. If the call is let through VSD to the receiving end user, based on the feedback given by the end user to the VSD, the reputation of the call originating domain and all intermediate domains that have routed the call are updated. The update is positive for a valid call and negative for a spam call.

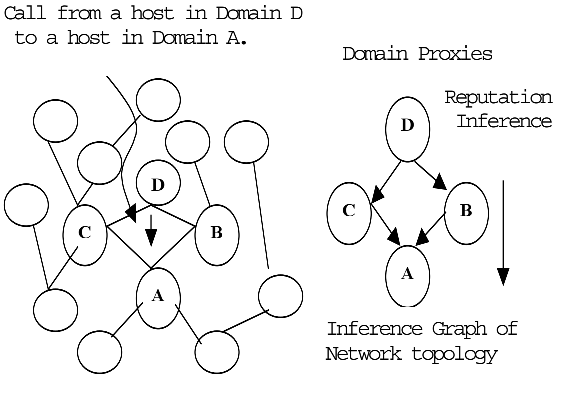

Fig 2: Reputation Inference for a call from Domains D to A

Many a times, trust and reputation are used for representing human belief. Trust represents caller's past behavior while reputation signifies social *status. While trust is a calculated entity, reputation is derived. Reputation is inferred by modeling human behavior.* i.e., in [Fig 2], if there was a spam call from domain D to domain A through domain C, then the reputation of D is decreased and also the reputation of C is decreased for forwarding a spam call. For a second spam call from domain D to domain A, the decrease in reputation is more than compared to the decrease for the first spam call. Also, the decrease in reputation for a spam call is more than the increase in reputation for a valid call. We achieve this by increasing the reputation additively for a valid call and decreasing multiplicatively for a spam call i.e. *an additive increase and multiplicative decrease in reputation*. In this way, using reputation and trust from past history, the calls can be quarantined or classified as spam.

For the given topology graph in [Fig 2], reputation is inferred by using Bayesian networks. For a call form domain D to domain A, the reputation can be inferred by calculating P(A|D) i.e. the posterior probability of A given an event that a call has been generated at D.

$$P(A|D) = P(A,B|D) + P(A,\sim B|D)$$

$$= P(A|B)P(B|D) + P(A|\sim B)P(\sim B|D) \quad \text{Eq 1}$$

where $P(A|B) = P(A,C|B) + P(A, \sim C|B)$

$$=P(A|B,C)P(C) + P(A|B\sim C)P(\sim C) \quad \text{Eq 2}$$

and $P(A|\sim B) = P(A,C|\sim B) + P(A, \sim C|\sim B) \quad \text{Eq 3}$

$$=P(A|\sim B,C)P(C) + P(A|\sim B, \sim C)P(\sim C)$$

$$P(C) = P(C,D) + P(C, \sim D) \quad \text{Eq 4}$$

$$= P(C|D)P(D) + P(C|\sim D)P(\sim D)$$

Solving equations 1-4 gives the updated probability or updated reputation of D. For a given set of initial or prior probabilities to the nodes of the topology graph representing the reputation of those domains, for a spam call from domain D to domain A the Bayesian inference calculations shown above would decrease the reputation for B,C and D proxies, and increase the reputation for a valid call for the same.

4. Experimental Setup and Results

The experimental setup consists of the Voice spam detection server, the end users for whom the VSD is acting as a spam detector and the call generating domains from which calls would be generated to the end users in the receiving domain. The end clients in the calling domain and the called domain are simulated SIP soft clients strictly in compliant with SIP RFC[11]. *All the simulated clients and the Voice Spam Detector are compatible with the real SIP phones and are capable of establishing sessions with them.* The simulated clients on the call-generating end generate calls by using randomly chosen usernames and hosts in the SIP URI of "from" field. The call generation process uses a Bernoulli distribution and the calls are generated with an average rate of 8 calls/minute. Neither the VSD nor the called domain end users have any idea regarding the call generation process. *A button called "SPAM" included in each IP phone in the receiving domain to give feedback to the VSD.*

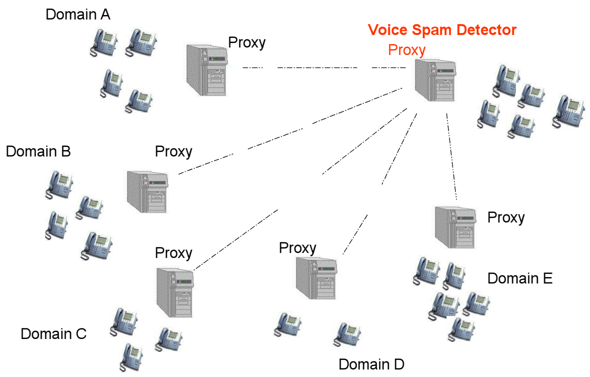

Fig 4: Experimental Setup showing the Calling and Receiving domains.

The simulated end clients on the call generating end randomly generate calls to the VSD and the VSD analyzes the call based on the caller trust and reputation. VSD calculates the spam probability of the call and compares with a predetermined threshold value to infer the spam behavior and block them. The threshold values chosen for each stage of analysis depends upon factors like the learning period, minimization of false alarms (false positives and false negatives) etc. Learning period signifies the minimum number of calls required by the VSD to learn the spam behavior before it starts blocking spam calls.

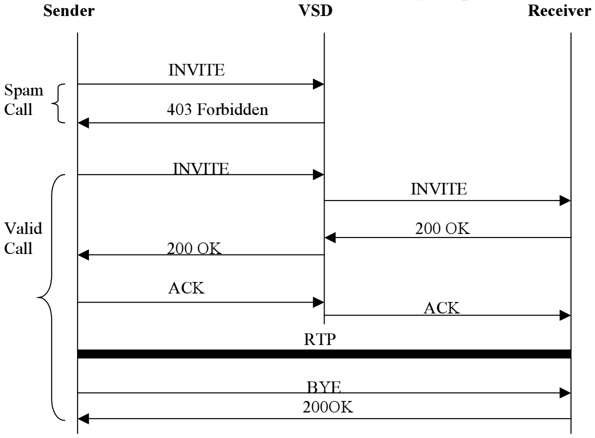

Fig 3: Call Flow for Spam and Valid calls through the VSD

The called domain users are equipped with spam recognition capabilities. We configure the calling domain with randomly chosen set of users, hosts and domains as spammers before the start of the experiment. The call received by the receiving client is analyzed and a feedback is given to the VSD about the nature of call. The Voice Spam Detector learns by observing the calling pattern with respect to called users, hosts and domains and the received feedback.

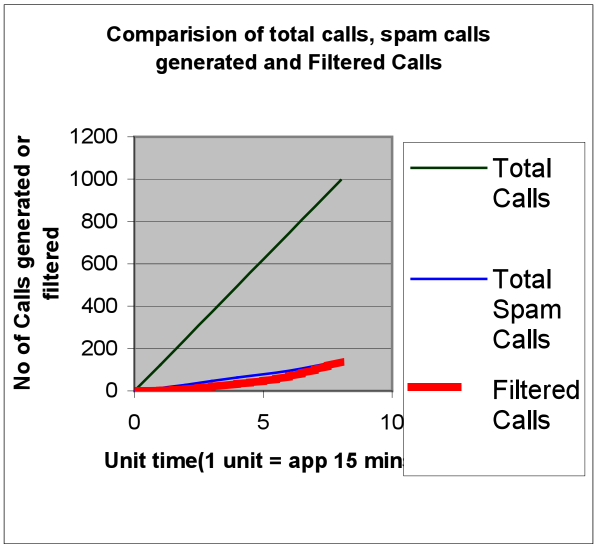

Fig 5: Comparing the actual calls generated, actual spam calls generated and filtered calls.

[Fig 5] represents the comparison between total calls, total spam calls generated and number of spam calls blocked by the VSD. The results are shown for five calling domains with each domain having an average of 100 users and 35 hosts. The number of calls blocked is a result of all the three stages of analysis [See Sec 3.2]. i.e. the black and white listing, trust (past history) and reputation of the calling party.

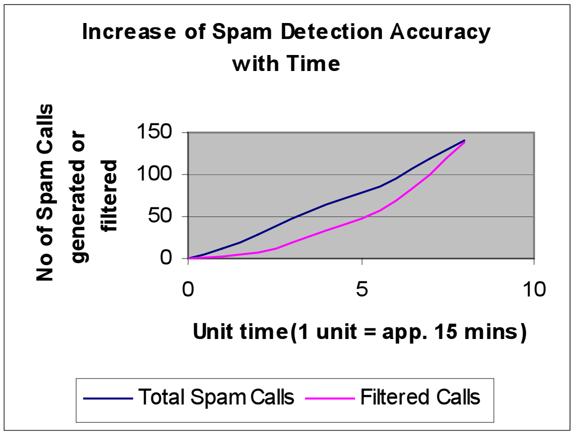

Fig 6: Spam Detection Accuracy increases with time.

[Fig 6] represents the comparison between the spam calls generated and filtered calls by the VSD. Initially VSD has no knowledge of spam generating clients, but learns the spam behavior with time and feedback from the end users. The spam calls detected are equal to the actual spam calls generated after certain learning period with an accuracy of 97.6% and a false positive percentage of 0.4%.

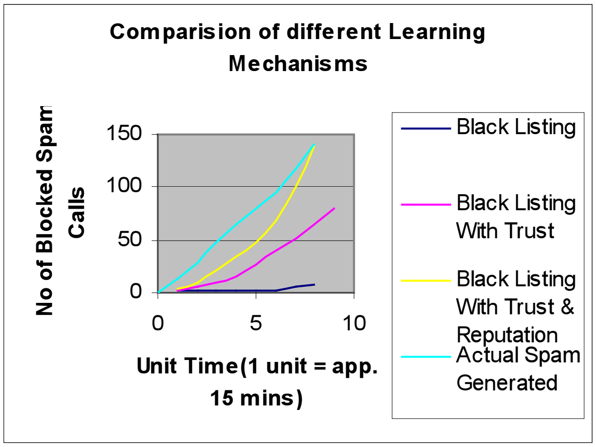

Fig 7: Spam Calls blocked by VSD for different stages of analysis.

[Fig 7] shows the spam calls blocked for the three stages of analysis. The experiments are conducted with a random 100 users and 35 hosts in each of 5 domains on the call generating end. It can be observed that the number of spam calls blocked using blacklisting, trust and reputation is approximately 97.16% compared to 4.25% if only blacklisting is implemented.

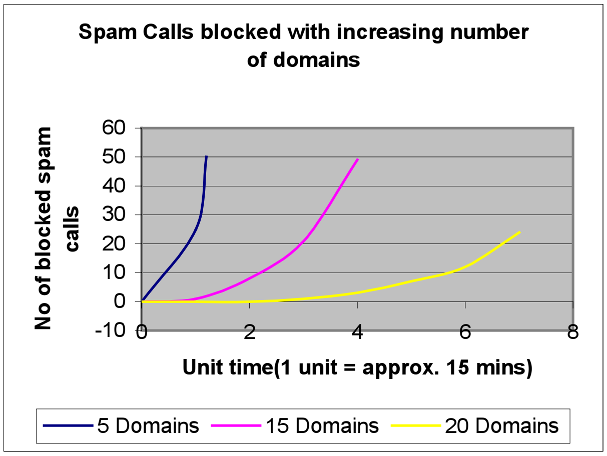

Fig 8: Blocked Spam Calls for increasing scalability on call generation.

[Fig 8] gives the number of spam calls blocked for three different sizes of topology. The time taken by VSD for learning spam behavior from 20 domains is more compared to time taken for 15 domains for the same set of spammers i.e. for the same set of spam users, hosts and domains. However, the VSD would have the near about the same approximate accuracy of spam recognition when the number of spammers increase with increase in the number of call generating users, hosts and domains. For the analysis shown in [Fig 8], the false alarms and the accuracy of VSD is as shown in [Tab 1].

# of Domains	Filter Accuracy %	False Positives %	False Negatives %
5	97.6	0.4	2
15	97.52	0.19	2.3
20	97.595	0.11	2.36

Tab 1: False Alarms and Spam recognition accuracy for the analyzed calls in Fig 8.

5. Conclusion

It is estimated that 35 billion spam email messages per day were generated in 2004. These messages are nuisance to the receivers and in addition create low availability and network congestion. VoIP technology is replacing existing PSTN at a rapid pace. The problem of spam in VoIP networks has to be solved in real time compared to e-mail systems. Many of the techniques devised for e-mail spam detection rely upon content analysis and in the case of VoIP it is too late to analyze the media after picking up the receiver. So we need to stop the spam calls before the telephone rings.

In computing, trust has traditionally been a term relating to authentication, security, or a measure of reliability. When it comes to receiving or rejecting a voice call social meaning of trust is applied and in particular reputation of the calling party is analyzed. We developed a five-stage process for identifying if the incoming call is spam or not. These stages include multivariable Bayesian analysis and inferring reputation using Bayesian networks. The results from each stage are fed back for collaboration between different processes. We have verified the results using an experimental setup consists of more than randomly generated calls from several thousand soft clients and a SIP proxy server. This setup includes commercial grade proxy server software as well as soft client. We have added the spam filter software at the proxy server for preventing and detecting the spam. We found that combining black/white lists, trust of the calling party and reputation of the calling party can be used accurately to identify if it is a spam or not. In this analysis we have used a concept where trust can be built up over time but a single spam call can exponentially bring down the trust level. We used this concept and found that the call can be more accurately identified as spam after a period of learning. From our observation of the logs it takes at least 3 spam calls to confirm it is a spam and fourth call can be accurately identified as the spam. Finally we expanded the experiments with large number of domains and verified our filtering mechanism. Further work involves understanding the behavior of different controlling parameters in trust and reputation calculations and deriving meaningful relationships between them. Also, we believe that our multistage filtering architecture can be used in prevented unwanted emails as well as in electronic commerce.

References

1. M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. 1998. A Bayesian Approach to Filtering Junk E-Mail. *Learning for Text Categorization – Papers from the AAAI Workshop*, pages 55–62, Madison Wisconsin. AAAI Technical Report WS-98-05.
2. G. Salton, M.J. McGill. 1983. Introduction to Modern Information Retrieval. McGraw- Hill.
3. T.M. Mitchell, Machine Learning. McGraw-Hill, 1997.
4. J. Golbeck, J. Hendler, ”Reputation Network Analysis for Email Filtering”, IEEE conference on Email and Anti Spam, August 2004.
5. R. Dantu, J. Cangussu, A. Yelimeli, “Dynamic Control of Worm Propagation”, IEEE International Conference on Information Technology ITCC April 04
6. N. Soonthornphisaj, K. Chaikulseriwat, P Tang-On, “Anti-Spam Filtering: A Centroid Based Classification Approach”, IEEE proceedings ICSP 02
7. G. Sakkis, I. Androutsopoulos, G. Paliouras, V. Karkaletsis, C.D. Spyropoulos, P. Stamatopoulos, “A memory based approach to anti-spam filtering for mailing lists”, Information Retrieval 2003.
8. W.W. Cohen, “Learning Rules that Classify e-mail”, In Proceedings of the AAAI Spring Symposium on Machine Learning in Information Access, 1996.
9. P.O. Boykin, V. Roychowdhury, ”Personal email networks: an effective anti-spam tool”. Preprint, http://www.arxiv.org/abs/cond-mat/0402143, (2004).
10. I Rigoutsos, T. Huynh, “Chung-Kwei: A Pattern Discovery based System for the Automatic Identification of Unsolicited E-mail messages”, Proceedings of the first conference on E-mail and Anti-Spam, 2004.
11. J. Rosenberg, H Shulzrinne, G Camerillo, A Johnston, J Peterson, R Sparks, M. Handley, E. Schooler, ”Session Initiation Protocol”, RFC 3261, June 2002.
12. http://www.forbes.com/technology/2004/02/27/cx_ah_0227tentech.html